

# Guía de Operaciones de OmniCRM

**OmniCRM** es la solución integral BSS/OSS de Omnitouch para proveedores de servicios móviles y de línea fija. Una plataforma completa que maneja todo, desde la incorporación de clientes hasta la facturación, aprovisionamiento y soporte, todo en un sistema integrado.

---

## Comenzando

### Para el Personal de Atención al Cliente

#### Tus primeros pasos:

1. [Aprender la interfaz](#) - Familiarízate con la navegación y búsqueda
2. [Crear un cliente](#) - Creación de clientes paso a paso
3. [Agregar un servicio](#) - Aprovisiona tu primer servicio
4. [Procesar un pago](#) - Maneja los pagos de los clientes
5. [Recargar servicios](#) - Agrega crédito a las cuentas de los clientes

#### Tareas diarias:

- [Gestión de servicios](#) - Administra los servicios de los clientes
- [Ver uso](#) - Verifica saldos y uso
- [Modificar servicios](#) - Cambia configuraciones de servicios
- [Generar facturas](#) - Crea y envía facturas

### Para Administradores del Sistema

#### Configuración y configuración:

1. [Entender la arquitectura](#) - Visión general del sistema
2. [Configurar el sistema](#) - Configuraciones del sistema
3. [Crear productos](#) - Construye tu catálogo
4. [Escribir playbooks](#) - Automatiza el aprovisionamiento
5. [Configurar usuarios](#) - Crea cuentas y asigna roles

#### Temas avanzados:

- [Gestión de inventario](#) - Administra activos
- [Personalización](#) - Adapta a tus necesidades
- [Integración de API](#) - Conecta sistemas externos
- [Configuración de seguridad](#) - Configura 2FA y permisos

## Para Clientes

### Uso del Portal de Autocuidado:

- [Acceder a tu cuenta](#) - Inicia sesión y navega
  - [Ver servicios](#) - Consulta tus servicios activos
  - [Verificar uso](#) - Monitorea datos y saldos
  - [Pagar facturas](#) - Realiza pagos en línea
  - [Recargar servicios](#) - Agrega crédito al instante
- 

## Referencia Rápida de Tareas

Quiero...	Documentación
Crear un nuevo cliente	<a href="#">Crear Cliente</a>
Agregar un servicio a un cliente	<a href="#">Aregar Servicio</a>
Ver uso de servicio y saldos	<a href="#">Uso de Servicio</a>
Procesar un pago	<a href="#">Procesar Pago</a>
Generar una factura	<a href="#">Gestión de Facturas</a>
Recargar un servicio	<a href="#">Recarga &amp; Recarga</a>
Gestionar inventario	<a href="#">Sistema de Inventario</a>
Crear un producto	<a href="#">Ciclo de Vida del Producto</a>
Escribir un playbook de aprovisionamiento	<a href="#">Playbooks de Ansible</a>
Configurar cuentas de usuario	<a href="#">RBAC</a>
Habilitar 2FA	<a href="#">Autenticación de Dos Factores</a>
Buscar cualquier cosa	<a href="#">Búsqueda Global</a>
Entender el sistema	<a href="#">Arquitectura del Sistema</a>
Usar la API	<a href="#">Documentación de API</a>

---

## Biblioteca de Documentación

### Lectura Esencial

- [Arquitectura del Sistema](#) - Visión técnica completa con diagramas
- [Guía del Ciclo de Vida del Producto](#) - Gestión de productos de principio a fin
- [Guía de Playbooks de Ansible](#) - Domina la automatización de aprovisionamiento

### Gestión de Clientes

- [Clientes, Contactos & Sitios](#) - Modelo de datos y relaciones
- [Creación de Clientes](#) - Guía paso a paso
- [Etiquetas de Clientes](#) - Organiza con etiquetas
- [Atributos de Clientes](#) - Metadatos personalizados

- [Registro de Actividades](#) - Rastrea todos los cambios

## Operaciones de Servicio

- [Gestión de Servicios](#) - Visión general
- [Aregar Servicios](#) - Flujo de trabajo de aprovisionamiento
- [Asignación de Planes](#) - Asignación de planes
- [Modificación de Servicios](#) - Realizando cambios
- [Uso de Servicios & Saldos](#) - Monitorea el uso
- [Recarga & Recarga](#) - Agrega créditos

## Facturación & Finanzas

- [Visión General de Facturación](#) - Guía completa de facturación
- [Métodos de Pago](#) - Gestión de métodos de pago
- [Transacciones](#) - Cargos y créditos
- [Procesamiento de Pagos](#) - Flujos de trabajo de pagos
- [Gestión de Facturas](#) - Genera y gestiona facturas

## Administración

- [Configuración del Sistema](#) - Configura OmniCRM
- [Personalización](#) - Adapta a tu negocio
- [Gestión de Inventario](#) - Seguimiento de activos
- [Claves de API](#) - Gestión de acceso a la API

## Seguridad & Acceso

- [Flujos de Autenticación](#) - Cómo funciona la autenticación
- [Autenticación de Dos Factores](#) - Configura 2FA
- [Control de Acceso Basado en Roles](#) - Usuarios, roles y permisos

## Integraciones

- [Integración de Stripe](#) - Procesamiento de pagos
- [Integración de Mailjet](#) - Servicio de correo electrónico
- [Documentación de API](#) - Referencia de API REST

## Orientado al Cliente

- [Portal de Autocuidado](#) - Guía del portal para clientes
- [Atención al Cliente](#) - Funciones de autoservicio
- [Sistema de Difusión Celular](#) - Alertas de emergencia

## Referencia

- [Glosario](#) - Términos y definiciones

- [Registro de Cambios](#) - Historial de versiones
- 

## ¿Qué Hace Especial a OmniCRM?

OmniCRM reúne todas las herramientas que necesitas para operar un negocio de telecomunicaciones moderno:

### Aprovisionamiento de Servicios Automatizado

Olvídate de la configuración manual: OmniCRM utiliza [automatización de Ansible](#) para aprovisionar servicios en segundos. Ya sea que estés activando una tarjeta SIM, configurando equipos de cliente o configurando un paquete complejo, el sistema lo maneja automáticamente.

#### Lo que hace:

- Aprovisiona servicios con un solo clic
- Configura equipos de red automáticamente
- Crea cuentas de facturación en tiempo real
- Envía correos electrónicos de bienvenida y notificaciones SMS
- Revierte automáticamente si algo falla

[Aprende más sobre Aprovisionamiento](#) | [Ver Guía de Playbooks de Ansible](#)

### Facturación y Pagos Inteligentes

La integración incorporada con [CGRateS](#) proporciona calificación y cobro en tiempo real, mientras que la [integración de Stripe](#) maneja el procesamiento de pagos sin problemas.

#### Lo que hace:

- Seguimiento y calificación de uso en tiempo real
- Generación automática de facturas
- Procesamiento de tarjetas de crédito a través de Stripe
- Modelos de facturación prepago y postpago
- Precios flexibles y códigos promocionales
- Cálculo y reporte de impuestos

[Explora las Funciones de Facturación](#) | [Procesamiento de Pagos](#) | [Gestión de Facturas](#)

### Gestión Completa de Clientes

Administra clientes, contactos, múltiples sitios y servicios con un modelo de relación integral.

## **Lo que hace:**

- Rastrea clientes, contactos y ubicaciones de servicio
- Administra múltiples servicios por cliente
- Almacena atributos y metadatos personalizados
- Etiqueta a los clientes para organización
- Historial de actividades completo y registros de auditoría
- Integración de Google Maps para geocodificación de sitios

[\*\*Guía de Gestión de Clientes\*\*](#) | [\*\*Crear un Cliente\*\*](#) | [\*\*Etiquetas de Clientes\*\*](#)

## **Gestión de Inventario**

Rastrea y gestiona todos tus activos físicos y virtuales, desde tarjetas SIM hasta enruteadores y bloques de direcciones IP.

## **Lo que hace:**

- Rastrea tarjetas SIM, equipos y accesorios
- Administra números de teléfono y bloques de direcciones IP
- Asignación automatizada durante el aprovisionamiento
- Capacidades de importación y exportación por lotes
- Almacenamiento de configuraciones de equipos
- Plantillas de inventario personalizadas

[\*\*Documentación del Sistema de Inventario\*\*](#)

## **Portal de Autocuidado para Clientes**

Empodera a tus clientes con un [\*\*portal de autoservicio\*\*](#) donde pueden gestionar sus propios servicios.

## **Lo que los clientes pueden hacer:**

- Ver servicios y uso
- Verificar saldos y asignaciones de datos
- Pagar facturas en línea
- Descargar facturas como PDFs
- Actualizar información de contacto
- [\*\*Recargar sus servicios\*\*](#) al instante

[\*\*Guía del Portal de Autocuidado\*\*](#) | [\*\*Funciones de Atención al Cliente\*\*](#)

## **Seguridad Empresarial**

Construido con la seguridad en su núcleo, con autenticación y autorización completas.

## Características de seguridad:

- [Autenticación basada en JWT](#)
- [Autenticación de Dos Factores \(2FA\)](#) con TOTP y códigos de respaldo
- [Control de Acceso Basado en Roles \(RBAC\)](#) con permisos granulares
- Verificación de correo electrónico para cambios de cuenta
- Registro de auditoría completo a través del registro de actividades
- Gestión de sesiones y controles de tiempo de espera

[Guía de Autenticación](#) | [Documentación de RBAC](#) | [Configuración de 2FA](#)

## Integraciones Potentes

OmniCRM se integra sin problemas con servicios líderes en la industria:

- **CGRateS** - Motor de facturación y calificación en tiempo real para cobros de grado telecomunicaciones
- **Stripe** - Procesamiento de pagos seguro y almacenamiento de tarjetas
- **Mailjet** - Entrega de correo electrónico profesional con plantillas
- **Google Maps** - Validación de direcciones y geocodificación para ubicaciones precisas de sitios
- **API RESTful** - Integra con tus sistemas existentes

[Documentación de API](#) | [Integración de Stripe](#) | [Integración de Mailjet](#)

## Catálogo de Productos Flexible

Crea cualquier tipo de oferta de producto, desde servicios simples independientes hasta paquetes complejos con múltiples componentes.

### Tipos de productos soportados:

- **Independientes** - Servicios únicos (planes móviles, paquetes de internet)
- **Paquetes** - Ofertas combinadas (internet + TV + teléfono)
- **Complementos** - Servicios suplementarios (recargas de datos, llamadas internacionales)
- **Promociones** - Ofertas especiales y descuentos

Cada producto puede tener su propia automatización de aprovisionamiento, reglas de precios y lógica de negocio.

[Guía del Ciclo de Vida del Producto](#) | [Conceptos de Productos y Servicios](#)

## Sistema de Difusión de Emergencia

Para operadores móviles, OmniCRM incluye un [Sistema de Difusión Celular](#) para alertas de seguridad pública.

## Lo que hace:

- Envía alertas de emergencia a áreas geográficas
- Soporte de mensajes en múltiples idiomas
- Segmentación por operador de red móvil
- Cumplimiento con estándares de alertas gubernamentales

## [Documentación de Difusión Celular](#)

## Búsqueda y Navegación Potentes

Encuentra cualquier cosa al instante con [\*\*búsqueda global\*\*](#) a través de clientes, servicios, facturas y más.

### Características de búsqueda:

- Busca en todas las entidades
- Filtra y ordena resultados
- Atajos de navegación rápida
- Sugerencias inteligentes

## [Guía de Navegación | Búsqueda Global](#)

---

## Arquitectura del Sistema

### [Ver Documentación Completa de Arquitectura](#)

---

## Conceptos Clave

### Entendiendo Productos vs Servicios

Este es uno de los conceptos más importantes en OmniCRM:

- **Producto** = Una plantilla u oferta en tu catálogo (por ejemplo, "Plan Móvil Ilimitado")
- **Servicio** = Una instancia activa de un producto para un cliente específico (por ejemplo, "Plan Móvil Ilimitado de John Smith")

Cuando aprovisionas un producto, el sistema utiliza [\*\*automatización de Ansible\*\*](#) para crear el o los servicios reales. Un producto puede crear múltiples servicios, ningún servicio (solo configuración) o modificar servicios existentes.

## [Aprende más sobre Productos y Servicios](#)

## La Magia del Aprovisionamiento

La salsa secreta de OmniCRM es su [\*\*sistema de aprovisionamiento automatizado\*\*](#):

El playbook maneja todo, desde la creación de cuentas en CGRateS hasta la configuración de enrutadores y el envío de correos electrónicos de bienvenida. Y si algo falla, se revierte automáticamente.

[\*\*Profundiza: Playbooks de Ansible | Sistema de Aprovisionamiento\*\*](#)

## Facturación Hecha Simple

Cada servicio genera [\*\*transacciones\*\*](#) (cargos o créditos). Estas se agrupan en [\*\*facturas\*\*](#) y se envían a los clientes. Con la integración de Stripe, los pagos pueden procesarse automáticamente.

[\*\*Visión General de Facturación | Procesamiento de Pagos\*\*](#)

## Jerarquía de Clientes

```
Cliente (Compañía ABC)
└── Contactos (John Smith - Facturación, Jane Doe - Técnico)
└── Sitios (Oficina Principal, Oficina Sucursal)
└── Servicios
    ├── Internet - Oficina Principal
    ├── Plan Móvil - John Smith
    └── VoIP - Oficina Sucursal
```

Todo está organizado jerárquicamente, lo que facilita la gestión de clientes con múltiples ubicaciones y servicios.

[\*\*Clientes, Contactos & Sitios\*\*](#)

## ¿Por Qué Elegir OmniCRM?

### Plataforma Todo en Uno

Todo lo que necesitas en un solo sistema integrado: no es necesario manejar múltiples herramientas o proveedores.

### Automatización Primero

El aprovisionamiento impulsado por Ansible significa que los servicios se implementan en segundos, no en horas. Reduce errores, ahorra tiempo, aumenta la satisfacción del cliente.

## Flexible y Personalizable

Desde definiciones de productos hasta playbooks de aprovisionamiento y atributos personalizados: adapta OmniCRM para que coincida con tus procesos comerciales exactos.

## Diseñado para Telecomunicaciones

Diseñado específicamente para proveedores de servicios con características como integración de CGRateS, gestión de inventario y calificación en tiempo real.

## Seguridad Empresarial

La autenticación JWT, 2FA, RBAC y el registro de auditoría completo garantizan que tus datos estén seguros y cumplan con las normativas.

## Impulsado por API

Una API REST completa significa que puedes integrar OmniCRM con cualquier sistema existente o construir herramientas personalizadas.

---

## Obtener Soporte

### Recursos de Documentación

- Comienza con [Arquitectura del Sistema](#) para una visión técnica general
- Consulta las [guías de inicio](#) para la incorporación específica de roles
- Revisa la [Referencia Rápida de Tareas](#) para operaciones comunes
- Consulta el [Glosario](#) para terminología

### ¿Necesitas Más Ayuda?

Explora la biblioteca de documentación completa arriba o utiliza la [búsqueda global](#) para encontrar temas específicos.

---

*OmniCRM - BSS/OSS Completo para Proveedores de Servicios Modernos*

*Última Actualización: 2025-12-23*



# Autenticación de Dos Factores (2FA)

## Seguridad Mejorada con Contraseñas de Un Solo Uso Basadas en Tiempo

OmniCRM admite **autenticación de dos factores (2FA)** utilizando contraseñas de un solo uso basadas en tiempo (TOTP). Esto añade una capa adicional de seguridad al requerir que los usuarios proporcionen tanto su contraseña como un código sensible al tiempo de una aplicación de autenticación.

Véase también: RBAC <rbac> para permisos de gestión de 2FA, Flujos de Autenticación <authentication\_flows> para detalles del proceso de inicio de sesión.

## Propósito

2FA proporciona:

1. **Seguridad Mejorada** — Protege las cuentas incluso si las contraseñas son comprometidas.
2. **Cumplimiento** — Cumple con los requisitos de seguridad para industrias reguladas.
3. **Elección del Usuario** — Opcional para los usuarios, puede ser impuesto por rol o globalmente.
4. **Estándar de la Industria** — Utiliza el protocolo TOTP compatible con Google Authenticator, Authy, Microsoft Authenticator y otras aplicaciones estándar.

## Cómo Funciona 2FA

Cuando 2FA está habilitado para un usuario:

1. **Configuración** — El usuario escanea un código QR con su aplicación de autenticación durante la inscripción.
2. **Inicio de Sesión** — Después de ingresar el nombre de usuario/contraseña, el usuario proporciona el código de 6 dígitos de su aplicación.
3. **Verificación** — El sistema valida que el código basado en tiempo coincida con el valor esperado.
4. **Acceso** — El usuario obtiene acceso solo después de que ambos factores sean verificados.

# Habilitando 2FA

## Para Usuarios Individuales

Los usuarios pueden habilitar 2FA para su propia cuenta:

1. Navegar a **Configuración del Usuario** o **Perfil**
2. Seleccionar **Habilitar Autenticación de Dos Factores**
3. Escanear el código QR con una aplicación de autenticación
4. Ingresar el código de verificación para confirmar la configuración
5. Guardar los códigos de respaldo en un lugar seguro

Configuración de 2FA

## Respaldo y Recuperación

### Códigos de Respaldo

Al habilitar 2FA, los usuarios reciben códigos de respaldo que se pueden usar si su dispositivo no está disponible:

- Cada código es de un solo uso
- Almacenar los códigos de forma segura fuera de línea
- Generar nuevos códigos si todos han sido utilizados

## Restablecimiento de Administrador

Si un usuario pierde el acceso a su autenticador y todos los códigos de respaldo están agotados, un administrador con acceso a la base de datos puede restablecer manualmente el 2FA del usuario borrando el campo `totp_secret` en la base de datos. El usuario puede luego volver a habilitar 2FA.

## Integración de API

### Habilitar 2FA para un usuario

POST /2fa/enable/user/{user\_id}

```
{  
  "password": "current_password"  
}
```

La respuesta incluye la URI de aprovisionamiento (para el código QR) y los códigos de respaldo.

## Verificar la configuración de 2FA

POST /2fa/verify-setup/user/{user\_id}

```
{  
  "code": "123456"  
}
```

## Verificar 2FA durante el inicio de sesión

POST /2fa/verify/user/{user\_id}

```
{  
  "code": "123456"  
}
```

Devuelve el token de acceso, el token de actualización y los datos del usuario tras una verificación exitosa.

Verificación de Inicio de Sesión 2FA

## Regenerar códigos de respaldo

POST /2fa/backup-codes/regenerate/user/{user\_id}

Requiere autenticación. Devuelve un nuevo conjunto de códigos de respaldo.

## Mejores Prácticas

- **Códigos de respaldo primero.** Siempre guarda los códigos de respaldo antes de completar la configuración de 2FA.
- **Educar a los usuarios.** Proporcionar instrucciones claras para la configuración y recuperación.
- **Proceso de restablecimiento seguro.** Verificar la identidad del usuario antes de restablecer manualmente 2FA en la base de datos.

## Preguntas Frecuentes

**¿Qué aplicaciones de autenticación son compatibles?** Cualquier aplicación compatible con TOTP (Google Authenticator, Authy, Microsoft Authenticator, 1Password, etc.).

**¿Qué pasa si pierdo mi teléfono?** Usa un código de respaldo o contacta a un administrador para restablecer 2FA.

**¿Puedo usar SMS en lugar de una aplicación?** Actualmente, solo se admiten aplicaciones de autenticación TOTP.

**¿Es 2FA obligatorio?** Depende de la política de tu organización. 2FA es típicamente obligatorio para roles de personal administrativo y de soporte, pero es opcional para usuarios clientes. El sistema no impone 2FA para cuentas de clientes (los usuarios del rol de cliente no ven indicaciones de inscripción a 2FA).

**¿Cuánto tiempo son válidos los códigos TOTP?** Los códigos se actualizan cada 30 segundos y tienen una pequeña ventana de tiempo para la validación (generalmente acepta códigos de la ventana de 30 segundos actual más las ventanas anteriores/siguientes para tolerancia de desviación del reloj).



# Gestión de Claves API

La interfaz de Gestión de Claves API proporciona una **interfaz web** para crear, monitorear y gestionar claves API utilizadas para el acceso programático a la API de OmniCRM.

## Nota

Para conceptos generales de autenticación API y ejemplos de uso, consulta [concepts\\_api](#).

## Descripción General

Las claves API permiten una **autenticación segura y de larga duración** para:

- Integraciones de servidor a servidor
- Scripts de automatización
- Aplicaciones de terceros
- Tareas programadas y trabajos cron
- Sistemas de monitoreo externos

A diferencia de los tokens JWT (que expiran después de minutos/horas), las claves API permanecen válidas hasta que se revoquen manualmente o hasta su fecha de expiración.

## Accediendo a la Gestión de Claves API

Navega a:

O directamente:

**Permiso Requerido:** MANAGE\_API\_KEYS (rol de administrador)

## Vista de Lista de Claves API

La página principal muestra todas las claves API en un formato de tabla:

### Columnas:

- **Nombre** - Etiqueta descriptiva para la clave API (por ejemplo, "Sistema de Provisionamiento", "Herramienta de Monitoreo")
- **Creado Por** - Nombre de usuario de la persona que creó la clave
- **Clave API** - La cadena de clave real (parcialmente enmascarada por

- seguridad)
- **Estado** - Activa, Expirada o Revocada
  - **Fecha de Creación** - Cuándo se generó la clave
  - **Fecha de Expiración** - Cuándo la clave expirará automáticamente
  - **Acciones** - Botones Editar, Eliminar, Regenerar

### Ejemplo de Visualización:

## Widgets del Tablero

En la parte superior de la página, se muestran estadísticas resumidas:

- **Total de Claves API** - Conteo de todas las claves API (activas e inactivas)
- **Claves Activas** - Claves actualmente válidas
- **Expirando Pronto** - Claves que expiran en los próximos 30 días
- **Claves Expiradas** - Claves que han pasado su fecha de expiración

## Creando una Clave API

### Paso 1: Haz clic en "Agregar Clave API"

Haz clic en el botón **+** **Agregar** en la parte superior derecha de la lista de Claves API.

### Paso 2: Completa los Detalles

Aparece un formulario modal solicitando:

Nombre: \_\_\_\_\_  
(por ejemplo, "Sistema de Provisionamiento")

Descripción: \_\_\_\_\_  
(Opcional - propósito de esta clave)

Fecha de Expiración: [Selector de Fecha]  
(Opcional - dejar en blanco para sin expiración)

Permisos:  Ver Clientes  Crear Clientes  Ver Servicios  Crear Servicios  Provisionamiento  Ver Inventario  Administrador (todos los permisos)

[Cancelar] [Generar Clave]

### Guías para los Campos:

**Nombre** (requerido)

- Identificador corto y descriptivo

- Ejemplos: "Sistema de Provisionamiento", "Integración de Facturación", "Monitoreo"
- Utilizado en registros de auditoría y mostrado en la lista

### **Descripción** (opcional)

- Explicación más detallada
- Ejemplos: "Usado por el servidor de provisionamiento de Ansible", "Sincronización de facturación de terceros"
- Ayuda a futuros administradores a entender el propósito de la clave

### **Fecha de Expiración** (opcional)

- Si está en blanco: La clave nunca expira (no recomendado)
- Si se establece: La clave se vuelve automáticamente inválida después de esta fecha
- Recomendado: Establecer expiración por seguridad (90 días a 1 año)

### **Permisos**

- Selecciona permisos específicos o marca "Administrador" para acceso completo
- Sigue el mismo modelo de permisos basado en roles que las cuentas de usuario
- **Mejor Práctica:** Asignar los permisos mínimos necesarios

### **Paso 3: Generar y Copiar la Clave**

Después de hacer clic en "**Generar Clave**", el sistema muestra la clave API recién creada:

⚠ ¡Copia esta clave ahora - no se mostrará de nuevo!

sk\_live\_a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6q7r8s9t0

[Copiar al Portapapeles]

[Cerrar]

Advertencia

### **¡Guarda la clave API de inmediato!**

Una vez que cierres este diálogo, la clave completa no podrá ser recuperada nuevamente. Solo verás una versión enmascarada (sk\_live\_...XYZ) en la vista de lista.

Si pierdes la clave, debes **regenerarla**, lo que invalida la clave antigua y puede romper integraciones existentes.

## Paso 4: Configura Tu Aplicación

Usa la clave API en las solicitudes de tu aplicación:

```
curl -X GET "https://yourcrm.com/crm/customers" \
-H "X-API-KEY: sk_live_a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6q7r8s9t0"
```

O en variables de entorno:

```
export CRM_API_KEY="sk_live_a1b2c3d4e5f6g7h8i9j0k1l2m3n4o5p6q7r8s9t0"
```

## Gestionando Claves Existentes

### Visualizando Detalles de la Clave

Haz clic en cualquier nombre de clave API para ver los detalles completos:

- Nombre completo de la clave y descripción
- Marca de tiempo de creación
- Nombre de usuario del creador
- Permisos asociados
- Estadísticas de uso (si están implementadas)
- Registros de acceso recientes

### Regenerando una Clave API

Si una clave API está comprometida o perdida, regénérala:

1. Haz clic en el **: (tres puntos)** en la columna de Acciones
2. Selecciona **"Regenerar Clave"**
3. Confirma la acción

Advertencia

#### **Regenerar invalida la clave antigua inmediatamente.**

Cualquier aplicación que use la clave antigua dejará de funcionar. Actualiza todas las integraciones con la nueva clave antes de regenerar.

#### **Qué Ocurre:**

- La clave antigua es revocada
- Se genera una nueva clave con los mismos permisos
- Se muestra la nueva clave (cópiala inmediatamente)
- Nombre, descripción y permisos permanecen sin cambios

## Revocando (Eliminando) una Clave API

Para eliminar permanentemente una clave API:

1. Haz clic en el **: (tres puntos)** en la columna de Acciones
2. Selecciona **"Eliminar"**
3. Confirma la eliminación

### Qué Ocurre:

- La clave es revocada inmediatamente
- Todas las solicitudes que usen esta clave devuelven 401 Unauthorized
- La clave es eliminada de la base de datos
- **No se puede deshacer** - la clave no puede ser recuperada

### Cuándo Revocar:

- La integración ya no es necesaria
- La clave ha sido comprometida
- El sistema que usa la clave ha sido desmantelado
- Reemplazando con una nueva clave con diferentes permisos

## Editando Detalles de la Clave API

Para modificar los detalles de una clave API:

1. Haz clic en el **: (tres puntos)** en la columna de Acciones
2. Selecciona **"Editar"**
3. Actualiza el nombre, la descripción, la expiración o los permisos
4. Haz clic en **"Guardar Cambios"**

### Campos Editables:

- Nombre
- Descripción
- Fecha de expiración
- Permisos

### No Editables:

- El valor de la clave en sí (usa Regenerar para cambiar)
- Fecha de creación
- Usuario que creó

## Estado de la Clave API

Las claves API pueden tener varios estados:

## **Activa**

- La clave es válida y puede ser utilizada
- Dentro de la fecha de expiración (o sin expiración establecida)
- No revocada manualmente
- Se muestra con una insignia verde

## **Expirando Pronto**

- Activa pero expirará en los próximos 30 días
- Se muestra con una insignia naranja/de advertencia
- Considera rotar antes de la expiración

## **Expirada**

- Pasada la fecha de expiración
- Ya no acepta autenticación
- Se muestra con una insignia roja
- Puede ser eliminada o la expiración extendida

## **Revocada**

- Eliminada/deshabilitada manualmente
- Permanentemente inválida
- Ya no se muestra en la lista activa

# **Filtrado y Búsqueda**

La lista de Claves API soporta:

### **Buscar:**

Buscar por nombre, descripción o clave parcial:

### **Filtrar por Estado:**

Desplegable de filtro para mostrar:

- Todas las Claves
- Solo Activas
- Expirando Pronto (próximos 30 días)
- Expiradas

### **Ordenar:**

Haz clic en los encabezados de columna para ordenar por:

- Nombre

- Fecha de Creación
- Fecha de Expiración
- Creado Por

## Mejores Prácticas de Seguridad

### Generación de Claves API

- **Longitud:** Las claves deben tener al menos 32 caracteres (el sistema hace cumplir esto)
- **Aleatoriedad:** Generadas utilizando generadores de números aleatorios criptográficamente seguros
- **Formato:** Típicamente prefijadas (por ejemplo, `sk_live_`) para identificación

### Almacenamiento de Claves API

#### En el CRM:

- Las claves son hasheadas antes de su almacenamiento (como contraseñas)
- La clave completa solo se muestra una vez durante la creación
- La base de datos almacena el hash para verificación
- Incluso los administradores no pueden recuperar la clave completa más tarde

#### En Tu Aplicación:

- Almacena en variables de entorno, no en el código
- Usa sistemas de gestión de secretos (AWS Secrets Manager, HashiCorp Vault)
- Nunca comités claves en control de versiones
- Rota las claves periódicamente (cada 90-365 días)

### Gestión de Permisos

- **Principio de Mínimos Privilegios** - Solo otorga permisos necesarios
- Evita crear claves de administrador a menos que sea absolutamente necesario
- Usa claves separadas para diferentes sistemas/purposes
- Revisa los permisos regularmente

### Monitoreo y Auditoría

- Monitorea el uso de claves API a través de registros de actividad
- Configura alertas para patrones de acceso inusuales
- Revisa las marcas de tiempo de "último uso" regularmente
- Elimina claves no utilizadas

## Rotación de Claves

Establece una política de rotación de claves:

1. **Crea una nueva clave** con los mismos permisos
2. **Actualiza las aplicaciones** para usar la nueva clave
3. **Monitorea** para asegurar que la clave antigua ya no se use
4. **Revoca la clave antigua** después de un período de gracia

## Solución de Problemas

### "401 Unauthorized" al usar la clave API

- **Causa:** Clave inválida, expirada o incorrecta
- **Solución:**
  - Verifica que la clave esté copiada correctamente (sin espacios extra)
  - Revisa el estado de la clave (Activa vs. Expirada)
  - Confirma que la clave tenga los permisos requeridos
  - Asegúrate de usar el encabezado X-API-KEY (no Authorization)

### "Clave API no encontrada" después de la creación

- **Causa:** La clave puede haber sido creada pero no almacenada correctamente
- **Solución:**
  - Revisa la lista de claves API para la nueva entrada
  - Si falta, crea una nueva clave
  - Informa el problema al administrador

### Clave API expirando pronto

- **Causa:** Fecha de expiración aproximándose (dentro de 30 días)
- **Solución:**
  - Crea una nueva clave con expiración extendida
  - Actualiza las aplicaciones para usar la nueva clave
  - Revoca la clave antigua después de la migración

### No se puede eliminar la clave API

- **Causa:** Puede estar protegida o en uso
- **Solución:**
  - Asegúrate de tener permisos de administrador
  - Verifica si la clave está bloqueada/protegida
  - Contacta al administrador si el problema persiste

# Puntos Finales de API (para Gestión Programática)

Las claves API también se pueden gestionar a través de la API (requiere permisos de administrador):

## Listar Claves API

```
GET /crm/api-keys
Authorization: Bearer <admin-token>
```

## Crear Clave API

```
POST /crm/api-keys
Authorization: Bearer <admin-token>
Content-Type: application/json

{
  "name": "Nueva Integración",
  "description": "Sincronización de facturación de terceros",
  "expiry_date": "2026-01-10",
  "permissions": ["view_customer", "view_service"]
}
```

## Respuesta:

```
{
  "api_key_id": 123,
  "name": "Nueva Integración",
  "api_key": "sk_live_a1b2c3d4e5f6g7h8i9j0",
  "status": "active",
  "created": "2025-01-10T10:00:00Z",
  "expiry_date": "2026-01-10T23:59:59Z"
}
```

## Revocar Clave API

```
DELETE /crm/api-keys/{api_key_id}
Authorization: Bearer <admin-token>
```

## Actualizar Clave API

```
PATCH /crm/api-keys/{api_key_id}
Authorization: Bearer <admin-token>
Content-Type: application/json

{
  "name": "Nombre Actualizado",
```

```
        "expiry_date": "2026-12-31"  
    }
```

## Casos de Uso Comunes

### Caso de Uso 1: Integración del Sistema de Provisionamiento

Crea una clave API para tu servidor de provisionamiento de Ansible:

1. Navega a Claves API → Agregar
2. Nombre: "Servidor de Provisionamiento de Ansible"
3. Descripción: "Usado por la automatización de provisionamiento"
4. Permisos: Provisionamiento, Ver/Criar Servicios, Ver/Actualizar inventario
5. Expiración: 1 año
6. Copia la clave y agrégala a `crm_config.yaml` de Ansible

### Caso de Uso 2: Integración de Facturación de Terceros

Crea una clave de solo lectura para la exportación de facturación:

1. Nombre: "Sincronización de Facturación - QuickBooks"
2. Permisos: Ver Clientes, Ver Transacciones, Ver Facturas
3. Expiración: 90 días (rotar trimestralmente)
4. Usar en script de exportación programada

### Caso de Uso 3: Monitoreo y Alertas

Crea una clave para la recolección de métricas de Prometheus/Grafana:

1. Nombre: "Monitoreo - Grafana"
2. Permisos: Ver Servicios, Ver Provisionamiento
3. Expiración: Nunca (el monitoreo necesita acceso continuo)
4. Configura en la fuente de datos de Grafana

### Caso de Uso 4: API del Portal del Cliente

Crea una clave para el portal de autoservicio del cliente:

1. Nombre: "Backend del Portal del Cliente"
2. Permisos: Ver Propio Cliente, Ver Propios Servicios, Crear Pagos
3. Expiración: 180 días
4. Usar en llamadas a la API del backend del portal

## Documentación Relacionada

- `concepts_api` - Conceptos y ejemplos de autenticación API
- `rbac` - Control de acceso basado en roles y permisos
- `2fa` - Autenticación de dos factores para mayor seguridad



# Atributos del Cliente

Los Atributos del Cliente son pares clave-valor flexibles que se pueden adjuntar a cualquier registro de cliente para almacenar metadatos personalizados, configuraciones, o información específica del negocio que no encaja en los campos estándar del cliente.

Para la categorización visual de clientes y enlaces clicables, consulte `Customer Tags <administration_tags>`. Para información básica del cliente, vea `Customers, Contacts, Sites & Services <basics_customers>`.

A diferencia de los campos de base de datos fijos, los atributos permiten extender dinámicamente los registros de clientes sin modificar el esquema de la base de datos. Esto los hace ideales para almacenar datos específicos de implementación, parámetros de integración o banderas de lógica comercial personalizada.

## Propósito y Casos de Uso

Los casos de uso comunes para los Atributos del Cliente incluyen:

### 1. Datos de Integración

Almacenar identificadores de sistemas externos o claves API específicas para este cliente:

- `external_crm_id = "SF-12345"` (ID de cliente de Salesforce)
- `legacy_system_id = "OLD-CRM-789"` (Referencia de migración)
- `hubspot_contact_id = "12345678"` (Integración de HubSpot)

### 2. Lógica Comercial Personalizada

Almacenar banderas o configuraciones que controlan el comportamiento específico del cliente:

- `billing_method = "quarterly"` (Anular la facturación mensual predeterminada)
- `auto_provision = "true"` (Habilitar la provisión automática de servicios)
- `support_tier = "premium"` (Nivel de soporte personalizado)
- `credit_limit = "10000"` (Límite de crédito específico del cliente)

### 3. Datos de Cumplimiento y Regulatorios

Rastrear metadatos relacionados con el cumplimiento:

- `gdpr_consent_date = "2025-01-01"` (Consentimiento para el procesamiento de datos)
- `tax_exempt = "true"` (Estado de exención fiscal)
- `regulatory_entity = "FCC-123456"` (Identificador regulatorio)

## 4. Metadatos Operativos

Almacenar información operativa:

- `preferred_contact_method = "email"` (Preferencia de comunicación)
- `account_manager = "<john.smith@company.com>"` (Gerente de cuenta asignado)
- `onboarding_date = "2025-01-15"` (Seguimiento del ciclo de vida del cliente)
- `churn_risk_score = "0.23"` (Análisis predictivo)

## 5. Parámetros de Provisión

Almacenar configuración específica de provisión:

- `radius_username_format = "email"` (Formato RADIUS personalizado)
- `vlan_id = "100"` (Configuración de red)
- `ipv6_enabled = "true"` (Banderas de características)

# Atributos vs. Campos Estándar

### Usar Atributos Cuando:

- Los datos son específicos de la implementación o varían según la instalación
- Los requisitos cambian con frecuencia
- Almacenar metadatos específicos de integración
- Prototipar nuevas características antes de agregar campos a la base de datos
- Los datos no necesitan consultas complejas o uniones

### Usar Campos Estándar Cuando:

- Los datos son fundamentales para el modelo de cliente (nombre, correo electrónico, dirección)
- Se requiere búsqueda, filtrado o informes frecuentes
- Los datos tienen restricciones de integridad referencial
- El rendimiento es crítico para consultas a gran escala

# Gestión de Atributos a través de la Interfaz de Usuario

## Visualización de Atributos del Cliente

Para ver los atributos de un cliente:

1. Navegue a la página de resumen del cliente
2. Haga clic en la pestaña **Attributes**
3. Verá una tabla de todos los atributos para este cliente, mostrando:
  - Nombre del Atributo (clave)
  - Valor del Atributo
  - Fecha de Creación
  - Fecha de Última Modificación

## Creación de un Nuevo Atributo

Para crear un nuevo atributo para un cliente:

1. Navegue a la página de resumen del cliente
2. Haga clic en la pestaña **Attributes**
3. Haga clic en el botón **Add Attribute**
4. Complete los campos requeridos:
  - **Nombre del Atributo** (requerido): La clave/nombre para este atributo (por ejemplo, `external_crm_id`)
  - **Valor del Atributo** (requerido): El valor a almacenar (por ejemplo, `SF-12345`)
5. Haga clic en **Create Attribute**

## Convenciones de Nomenclatura:

- Use minúsculas con guiones bajos: `external_system_id` ✓
- Evite espacios: `external system id` ✗
- Mantenga los nombres descriptivos pero concisos
- Use una nomenclatura consistente entre clientes para los mismos tipos de atributos

## Edición de un Atributo

Para editar un atributo existente:

1. Navegue a la página de resumen del cliente
2. Haga clic en la pestaña **Attributes**
3. Encuentre el atributo que desea editar en la tabla
4. Haga clic en el botón **Edit** (lápiz)
5. Modifique el nombre o el valor del atributo
6. Haga clic en **Update Attribute**

## Nota

Cambiar el nombre de un atributo crea un nuevo par clave-valor. Asegúrese de que esto no rompa las integraciones que dependen del nombre original del atributo.

## Eliminación de un Atributo

Para eliminar un atributo:

1. Navegue a la página de resumen del cliente
2. Haga clic en la pestaña **Attributes**
3. Encuentre el atributo que desea eliminar en la tabla
4. Haga clic en el botón **Delete** (papelera)
5. Confirme la eliminación en la ventana emergente

## Advertencia

Eliminar atributos utilizados por integraciones, flujos de trabajo de provisión o lógica de facturación puede causar fallos. Verifique las dependencias antes de la eliminación.

## Referencia de Campo de Atributo

### Integración API

Los atributos se pueden gestionar programáticamente a través de la API:

#### Crear o Actualizar un Atributo

**Endpoint:** PUT /crm/attribute/

**Permiso Requerido:** create\_customer\_attribute

#### Cuerpo de la Solicitud:

```
{  
  "customer_id": 123,  
  "attribute_name": "external_crm_id",  
  "attribute_value": "SF-12345"  
}
```

#### Respuesta:

```
{  
  "attribute_id": 456,  
  "customer_id": 123,
```

```
        "attribute_name": "external_crm_id",
        "attribute_value": "SF-12345",
        "created": "2025-01-04 10:30:00",
        "last_modified": "2025-01-04 10:30:00"
    }
```

## Actualizar un Atributo Existente

**Endpoint:** PATCH /crm/attribute/attribute\_id/{attribute\_id}

**Permiso Requerido:** update\_customer\_attribute

**Cuerpo de la Solicitud:**

```
{
    "attribute_value": "SF-54321"
}
```

## Obtener Atributo por ID

**Endpoint:** GET /crm/attribute/attribute\_id/{attribute\_id}

**Permiso Requerido:** view\_customer\_attribute

**Respuesta:**

```
{
    "attribute_id": 456,
    "customer_id": 123,
    "attribute_name": "external_crm_id",
    "attribute_value": "SF-12345",
    "created": "2025-01-04 10:30:00",
    "last_modified": "2025-01-04 10:30:00"
}
```

## Obtener Todos los Atributos por ID de Cliente

**Endpoint:** GET /crm/attribute/customer\_id/{customer\_id}

**Permiso Requerido:** view\_customer\_attribute

**Respuesta:**

```
[
    {
        "attribute_id": 456,
        "customer_id": 123,
        "attribute_name": "external_crm_id",
```

```
        "attribute_value": "SF-12345",
        "created": "2025-01-04 10:30:00",
        "last_modified": "2025-01-04 10:30:00"
    },
    {
        "attribute_id": 457,
        "customer_id": 123,
        "attribute_name": "billing_method",
        "attribute_value": "quarterly",
        "created": "2025-01-04 10:35:00",
        "last_modified": "2025-01-04 10:35:00"
    }
]
```

## Eliminar un Atributo

**Endpoint:** DELETE /crm/attribute/attribute\_id/{attribute\_id}

**Permiso Requerido:** delete\_customer\_attribute

**Respuesta:**

```
{
    "result": "success"
}
```

## Operaciones de Atributo en Masa

### Gestión de Múltiples Atributos

Para establecer múltiples atributos para un cliente a la vez (por ejemplo, durante la incorporación o sincronización de integración):

```
import requests

customer_id = 123
attributes = [
    {"attribute_name": "external_crm_id", "attribute_value": "SF-12345"},
    {"attribute_name": "billing_method", "attribute_value": "quarterly"},
    {"attribute_name": "support_tier", "attribute_value": "premium"}
]

for attr in attributes:
    attr["customer_id"] = customer_id
    requests.put(
        "https://api.example.com/crm/attribute/" ,
```

```
        json=attr,
        headers={"Authorization": "Bearer YOUR_TOKEN"}
    )
```

## Consultar Clientes por Atributo

Aunque los atributos no tienen endpoints de búsqueda integrados, puede filtrar clientes por atributo utilizando la API de búsqueda de clientes con filtrado personalizado:

```
# Obtener todos los clientes, luego filtrar por atributo en el código
# de la aplicación
customers = requests.get("https://api.example.com/crm/
customer/").json()

for customer in customers:
    attributes = requests.get(
        f"https://api.example.com/crm/attribute/
customer_id/{customer['customer_id']}"
    ).json()

    # Encontrar clientes con un atributo específico
    for attr in attributes:
        if attr['attribute_name'] == 'support_tier' and
attr['attribute_value'] == 'premium':
            print(f"Cliente premium: {customer['customer_name']}")
```

Nota

Para consultas frecuentes basadas en atributos, considere agregar campos de base de datos indexados o implementar un endpoint de búsqueda dedicado.

## Mejores Prácticas

### 1. Convenciones de Nomenclatura

- Use snake\_case: `external_system_id` ✓
- Sea descriptivo: `billing_method` ✓ vs `method` ✗
- Evite palabras clave reservadas o caracteres especiales
- Documente los significados de los atributos en su guía de implementación

### 2. Tipos de Datos

- Los atributos almacenan valores como cadenas (máx. 150 caracteres)
- Para booleanos, use "true"/"false" (minúsculas)
- Para fechas, use el formato ISO 8601: "2025-01-04 10:30:00"
- Para grandes datos JSON, considere campos de base de datos dedicados en su lugar

### 3. Validación

- Valide los valores de los atributos en el código de la aplicación antes de guardar
- Use formatos de valor consistentes entre clientes
- Documente los valores esperados para cada nombre de atributo

### 4. Documentación

- Mantenga un registro de nombres y propósitos de atributos
- Documente qué sistemas/integraciones dependen de atributos específicos
- Incluya ejemplos de valores válidos

### 5. Migración y Limpieza

- Audite regularmente los atributos no utilizados
- Elimine atributos obsoletos después de migraciones del sistema
- Versione los nombres de atributos al cambiar esquemas (por ejemplo, `api_key_v2`)

## Ejemplos de Flujos de Trabajo

### Integración de Incorporación

Al migrar clientes desde un sistema heredado:

```
# Almacenar referencia del sistema heredado para depuración
PUT /crm/attribute/
{
  "customer_id": 123,
  "attribute_name": "legacy_crm_id",
  "attribute_value": "OLD-12345"
}

# Rastrear fecha de migración
PUT /crm/attribute/
{
  "customer_id": 123,
  "attribute_name": "migrated_date",
  "attribute_value": "2025-01-04"
}
```

### Reglas de Facturación Personalizadas

Anular el ciclo de facturación predeterminado para un cliente específico:

```
# Establecer facturación trimestral
PUT /crm/attribute/
```

```

{
  "customer_id": 123,
  "attribute_name": "billing_cycle",
  "attribute_value": "quarterly"
}

# Luego en el código de facturación, verifique el atributo antes de procesar
attributes = GET /crm/attribute/customer_id/123
billing_cycle = next(
  (a['attribute_value'] for a in attributes if a['attribute_name'] ==
== 'billing_cycle'),
  'monthly' # predeterminado
)

```

## Banderas de Características

Habilitar características beta para clientes específicos:

```

# Habilitar provisión IPv6
PUT /crm/attribute/
{
  "customer_id": 123,
  "attribute_name": "feature_ipv6_enabled",
  "attribute_value": "true"
}

```

## Permisos

Las operaciones de atributos requieren los siguientes permisos:

- `view_customer_attribute` - Ver atributos
- `create_customer_attribute` - Crear nuevos atributos
- `update_customer_attribute` - Modificar atributos existentes
- `delete_customer_attribute` - Eliminar atributos

Consulte `rbac` para la configuración de control de acceso basado en roles.

## Solución de Problemas

### Atributo No Aparece en la UI

- Verifique que el atributo fue creado (verifique la respuesta de la API)
- Actualice la página para recargar los datos del cliente
- Verifique que el usuario tenga el permiso `view_customer_attribute`

### No Se Puede Actualizar el Atributo

- Asegúrese de tener el permiso `update_customer_attribute`
- Verifique que el `attribute_id` sea correcto
- Verifique que el atributo pertenezca al cliente especificado

## **Integración Fallando Después de la Eliminación del Atributo**

- Restaure el atributo con el valor anterior
- Actualice el código de integración para manejar atributos faltantes de manera adecuada
- Audite las dependencias de atributos antes de la eliminación

## **Valor del Atributo Truncado**

- Los valores de los atributos tienen un límite de 150 caracteres
- Para datos más largos, divídalos en múltiples atributos o use el campo de notas del cliente
- Considere almacenar grandes datos en campos de base de datos dedicados



# Configuración del Sistema

OmniCRM utiliza dos sistemas de configuración principales: **crm\_config.yaml** para configuraciones de API de backend y **variables de entorno** para la interfaz de usuario de React. Esta guía cubre todas las opciones de configuración y cómo modificarlas.

## Resumen de Archivos de Configuración

### Configuración de API de Backend:

- **Archivo:** OmniCRM-API/crm\_config.yaml
- **Formato:** YAML
- **Requiere:** Reinicio de API después de los cambios
- **Usado para:** Base de datos, integraciones, seguridad, aprovisionamiento

### Configuración de UI de Frontend:

- **Archivo:** OmniCRM-UI/.env
- **Formato:** Variables de entorno
- **Requiere:** Reconstrucción de UI después de los cambios
- **Usado para:** Marca, características, servicios externos

## Configuración de Backend (crm\_config.yaml)

El archivo `crm_config.yaml` contiene todas las configuraciones del sistema de backend.

### Configuración de Base de Datos

```
database:  
  username: omnitouch  
  password: omnitouch2024  
  server: localhost
```

#### Campos:

- `username` - Nombre de usuario de la base de datos MySQL
- `password` - Contraseña de la base de datos MySQL
- `server` - Nombre de host o IP del servidor de base de datos (predeterminado: localhost)

#### Conexión a la Base de Datos:

- El nombre de la base de datos está codificado como `omnicrm`
- Puerto predeterminado: 3306 (predeterminado de MySQL)
- Cadena de conexión: `mysql+pymysql://username:password@server/omnicrm`

**Nota de Seguridad:** Nunca comprometas este archivo con credenciales reales en control de versiones. Usa configuraciones específicas del entorno o gestión de secretos.

## Tipos de Servicio

```
service_types:
  - omnicharge
  - mobile
  - internet
  - iptv
  - voip
```

**Propósito:** Define los valores válidos de tipo de servicio para el campo `service_type`.

### Tipos Predeterminados:

- `mobile` - Servicios móviles/celulares
- `internet` - Internet fijo (fibra, DSL, inalámbrico)
- `iptv` - Servicios de televisión
- `voip` - Servicios de voz sobre IP
- `omnicharge` - Servicios de facturación/cobro

Agrega tipos de servicio personalizados aquí para tus casos de uso específicos.

## Configuración de HSS (Servidor de Suscriptores en Casa)

```
hss:
  hss_peers:
    - 'http://10.179.2.140:8080'
  apn_list: "1,2,3,4,5,6"
```

### Campos:

- `hss_peers` - Lista de URLs de servidores HSS para la gestión de suscriptores
- `apn_list` - Lista separada por comas de identificadores de APN (Nombre del Punto de Acceso)

**Usado para:** Aprovisionamiento de redes móviles y autenticación de suscriptores.

# Configuración de Correo Electrónico de Mailjet

```
mailjet:
  api_key: your_mailjet_api_key
  api_secret: your_mailjet_api_secret

  api.crmCommunicationCustomerWelcome:
    from_email: "support@yourcompany.com"
    from_name: "Soporte de Tu Empresa"
    template_id: 5977509
    subject: "Bienvenido a Tu Empresa"

  api.crmCommunicationCustomerInvoice:
    from_email: "billing@yourcompany.com"
    from_name: "Facturación de Tu Empresa"
    template_id: 6759851
    subject: "Tu Factura - "
```

## Tipos de Correo Electrónico Configurados:

- api.crmCommunicationCustomerWelcome - Correo electrónico de bienvenida para nuevos clientes
- api.crmCommunicationCustomerInvoice - Entrega de factura
- api.crmCommunicationCustomerInvoiceReminder - Recordatorios de factura vencida
- api.crmCommunicationUserWelcome - Bienvenida a nuevos usuarios del personal
- api.crmCommunicationUserPasswordReset - Solicitudes de restablecimiento de contraseña
- api.crmCommunicationUserPasswordResetSuccess - Restablecimiento de contraseña exitoso
- api.crmCommunicationUserPasswordChange - Notificaciones de cambio de contraseña
- api.crmCommunicationEmailVerification - Verificación de dirección de correo electrónico
- api.crmCommunicationsBalanceExpired - Notificaciones de expiración de servicio
- api.crmCommunicationsBalanceLow - Alertas de saldo bajo

## IDs de Plantilla:

Obtén de tu cuenta de Mailjet después de crear plantillas de correo electrónico. Consulta `integrations_mailjet` para más detalles.

# Configuración de Aprovisionamiento

```
provisioning:
  failure_list: ['admin@yourcompany.com', 'ops@yourcompany.com']
```

## Campos:

- `failure_list` - Direcciones de correo electrónico a notificar cuando el aprovisionamiento falla

Cuando los playbooks de Ansible fallan durante el aprovisionamiento, el sistema envía detalles de error a estas direcciones.

## Configuración de Factura

```
invoice:  
  template_filename: 'your_invoice_template.html'
```

## Campos:

- `template_filename` - Archivo de plantilla HTML para la generación de facturas

El archivo de plantilla debe existir en el directorio `OmniCRM-API/templates/`.

## URL Base de CRM

```
crm:  
  base_url: 'http://localhost:5000'
```

**Propósito:** Usado por los playbooks de Ansible para hacer llamadas a la API de vuelta al CRM.

## Importante:

- No incluyas una barra diagonal al final
- Usa una URL accesible públicamente si los playbooks se ejecutan en diferentes servidores
- Actualiza al desplegar en producción (por ejemplo, `https://api.tucrm.com`)

## Configuración de OCS (Sistema de Carga en Línea)

```
ocs:  
  ocsApi: 'http://10.179.2.142:8080/api'  
  ocsTenant: 'mnc380.mcc313.3gppnetwork.org'  
  cgbrates: 'localhost:2080'
```

## Campos:

- `ocsApi` - URL del punto final de la API REST de OCS
- `ocsTenant` - Identificador de inquilino para implementaciones de OCS multi-inquilino

- cgrates - Punto final JSON-RPC de CGRateS (host:puerto)

**Usado para:** Carga en tiempo real, gestión de saldo, seguimiento de uso.

## Configuración de SMSC (Centro de SMS)

```
smsc:
  source_msisdn: 'TuEmpresa'
  smsc_url: 'http://10.179.2.216/SMSc/'
  api_key: 'your_smse_api_key'
```

**Campos:**

- source\_msisdn - ID del remitente para SMS salientes (nombre de la empresa o código corto)
- smsc\_url - URL de la API del Centro de SMS
- api\_key - Clave de autenticación para la API de SMSC

**Usado para:** Envío de notificaciones SMS (alertas de saldo, OTP, etc.)

## Configuración de Difusión Celular

```
cbc_url: 'http://10.179.1.113:8080'
```

**Propósito:** Punto final de la API del Centro de Difusión Celular (CBC) para alertas de emergencia.

Consulta features\_cell\_broadcast para detalles de uso.

## Clave Secreta JWT

```
jwt_secret:
'2b93110f723db60172c8e9a1eaa80027a9a9c3f05b44e50dc3fcf38dba68d87e'
```

**Propósito:** Clave secreta para firmar tokens de autenticación JWT.

**Seguridad:**

- Generar usando: openssl rand -hex 32
- **Nunca compartir públicamente**
- Cambiar esto invalida todas las sesiones de usuario existentes
- Usa secretos diferentes para dev/staging/producción

## Configuración de Pagos de Stripe

```
stripe:
  secret_key: 'sk_test_...'
  publishable_key: 'pk_test_...'
```

```
currency: 'usd'  
statement_descriptor_suffix: 'TUEMPRESA'
```

## Campos:

- `secret_key` - Clave API secreta de Stripe (comienza con `sk_`)
- `publishable_key` - Clave publicable de Stripe (comienza con `pk_`)
- `currency` - Código de moneda ISO 4217 (usd, gbp, aud, eur, etc.)
- `statement_descriptor_suffix` - Texto que aparece en los estados de cuenta de tarjetas de crédito de los clientes

## Tipos de Clave:

- Claves de prueba: `sk_test_...` y `pk_test_...` (para desarrollo)
- Claves en vivo: `sk_live_...` y `pk_live_...` (para producción)

Consulta `integrations_stripe` para detalles de configuración.

## Claves API

```
api_keys:  
"your-secure-api-key-minimum-32-chars":  
  roles: ["admin"]  
  ips: ["127.0.0.1", "::1"]  
"another-api-key-for-specific-service":  
  roles: ["customer_service_agent_1"]  
  ips: ["10.0.1.50"]
```

## Estructura:

- **Clave (cadena):** La clave API real (mínimo 32 caracteres)
- **roles (lista):** Nombres de roles a los que esta clave tiene acceso
- **ips (lista, opcional):** Direcciones IP permitidas para usar esta clave

## Generando Claves API:

```
openssl rand -base64 48
```

## Roles:

- `admin` - Acceso completo a todos los puntos finales
- Roles personalizados definidos en el sistema RBAC

Consulta `administration_api_keys` y `concepts_api` para más detalles.

## Lista Blanca de IP

```
ip_whitelist:
```

```
"10.179.2.142":  
  roles: [ "admin" ]  
"192.168.1.100":  
  roles: [ "provisioning" ]
```

**Propósito:** Permitir que direcciones IP específicas accedan a la API sin autenticación.

**Estructura:**

- **Dirección IP (cadena):** Dirección IPv4 a incluir en la lista blanca
- **roles (lista):** Roles asignados a solicitudes desde esta IP

**Advertencia de Seguridad:**

- Solo usar para redes internas de confianza
- No debe usar IPs de localhost (127.0.0.1, ::1)
- Usar claves API en su lugar para acceso externo
- Considerar reglas de firewall como protección adicional

## Configuración de Frontend (.env)

La interfaz de usuario de React se configura a través de variables de entorno en OmniCRM-UI/.env.

### Configuración de Marca

```
REACT_APP_COMPANY_NAME="TuEmpresa"  
REACT_APP_PORTAL_NAME="TuPortal"  
REACT_APP_SELF_CARE_NAME="TuCuidado"  
REACT_APP_COMPANY_TAGLINE="Tu Lema de Empresa"
```

**Campos:**

- REACT\_APP\_COMPANY\_NAME - Nombre de la empresa (aparece en encabezados, correos, marca)
- REACT\_APP\_PORTAL\_NAME - Nombre del portal de administración (títulos de página, navegación)
- REACT\_APP\_SELF\_CARE\_NAME - Nombre del portal del cliente
- REACT\_APP\_COMPANY\_TAGLINE - Lema de marketing (aparece en la página de inicio de sesión)

**Ejemplo:**

### Configuración Regional

```
REACT_APP_DEFAULT_LOCATION="Londres, Reino Unido"  
REACT_APP_DEFAULT_COUNTRY="Reino Unido"
```

```
REACT_APP_DEFAULT_LANGUAGE="en"  
REACT_APP_LOCALE="en-GB"
```

### Campos:

- REACT\_APP\_DEFAULT\_LOCATION - Ubicación predeterminada para mapas y direcciones
- REACT\_APP\_DEFAULT\_COUNTRY - País predeterminado para formularios
- REACT\_APP\_DEFAULT\_LANGUAGE - Idioma de la interfaz de usuario (ar, ch, en, fr, gr, it, ru, sp)
- REACT\_APP\_LOCALE - Configuración regional para formato de fecha/número (en-GB, en-US, etc.)

### Idiomas Soportados:

- ar - Árabe
- ch - Chino
- en - Inglés (predeterminado)
- fr - Francés
- gr - Griego
- it - Italiano
- ru - Ruso
- sp - Español

## Configuración de Moneda

```
REACT_APP_CURRENCY_CODE="USD"  
REACT_APP_CURRENCY_SYMBOL="$"
```

### Campos:

- REACT\_APP\_CURRENCY\_CODE - Código de moneda ISO 4217
- REACT\_APP\_CURRENCY\_SYMBOL - Símbolo a mostrar (£, \$, €, etc.)

### Monedas Comunes:

- USD - \$ (Dólar estadounidense)
- GBP - £ (Libra esterlina)
- EUR - € (Euro)
- AUD - \$ (Dólar australiano)
- CAD - \$ (Dólar canadiense)

**Nota:** Debe coincidir con `stripe.currency` en `crm_config.yaml`.

## Configuración de Tema de Color

```
REACT_APP_PRIMARY_COLOR=#405189  
REACT_APP_SECONDARY_COLOR=#2bFFcf
```

REACT\_APP\_TERTIARY\_COLOR=#1a9fbf

## Colores Disponibles:

- REACT\_APP\_PRIMARY\_COLOR - Color principal de la marca (botones, enlaces, resaltados)
- REACT\_APP\_SECONDARY\_COLOR - Color de acento
- REACT\_APP\_TERTIARY\_COLOR - Acento adicional
- REACT\_APP\_SUCCESS\_COLOR - Mensajes de éxito (#28a745)
- REACT\_APP\_INFO\_COLOR - Mensajes de información (#17a2b8)
- REACT\_APP\_WARNING\_COLOR - Advertencias (#ffc107)
- REACT\_APP\_DANGER\_COLOR - Errores (#dc3545)
- REACT\_APP\_LIGHT\_COLOR - Fondos claros (#f8f9fa)
- REACT\_APP\_DARK\_COLOR - Texto oscuro (#343a40)
- REACT\_APP\_PRIMARY\_DARK\_COLOR - Variante oscura del color primario (para modo oscuro/estados de hover)

**Formato:** Códigos de color hexadecimal (#RRGGBB)

## Configuración de Fuente

REACT\_APP\_FONT\_FAMILY=Quicksand

**Propósito:** Establece la familia de fuentes principal para toda la interfaz de usuario.

**Importante:** Todas las fuentes están **auto-alojadas localmente** dentro de la aplicación OmniCRM-UI. Esto significa:

- **Sin carga de fuentes externas** - Las fuentes están empaquetadas con la aplicación
- **Compatible con jardín amurallado** - No se requiere acceso a Internet para que las fuentes funcionen
- **Operación sin conexión** - Funcionalidad completa en entornos de red restringidos o aislados
- **Privacidad** - Sin solicitudes externas a Google Fonts, Adobe Fonts, o otros CDNs
- **Rendimiento** - Carga más rápida sin dependencias externas
- **Seguridad** - Sin seguimiento de terceros ni filtración de datos a través de solicitudes de fuentes

## Opciones Disponibles:

### Fuentes Sans-Serif:

- Inter
- Roboto
- Open Sans
- Lato

- Quicksand (predeterminado)
- Poppins
- Nunito
- Montserrat
- Work Sans
- Source Sans Pro
- Raleway
- Ubuntu
- Josefin Sans
- HKGrotesk

### **Fuentes Serif:**

- Merriweather
- Lora
- Playfair Display

### **Fuentes del Sistema:**

- Sistema - Usa fuentes nativas del dispositivo para mejor rendimiento y menor tamaño de paquete

**Predeterminado:** Quicksand

### **Agregar Fuentes Personalizadas**

**¡Sí, puedes agregar fuentes adicionales!** Todas las fuentes se almacenan localmente en la aplicación.

#### **Para agregar una nueva fuente personalizada:**

1. **Agregar archivos de fuente** a OmniCRM-UI/src/assets/fonts/tu-nombre-de-fuente/
  - Usa formato WOFF2 para mejor compresión y soporte de navegador
  - Incluye múltiples pesos (300, 400, 500, 600, 700) para un renderizado adecuado
  - Nombra los archivos: tu-nombre-de-fuente-300.woff2, tu-nombre-de-fuente-400.woff2, etc.
2. **Definir reglas @font-face** en OmniCRM-UI/src/assets/scss/fonts/\_google-fonts.scss

```
//  
// Tu Fuente Personalizada - Descripción  
//  
@font-face {
```

```
font-family: 'Tu Nombre de Fuente';
font-style: normal;
font-weight: 400;
font-display: swap;
src: url("../fonts/tu-nombre-de-fuente/tu-nombre-de-fuente-400.woff2") format('woff2');
}

@font-face {
  font-family: 'Tu Nombre de Fuente';
  font-style: normal;
  font-weight: 700;
  font-display: swap;
  src: url("../fonts/tu-nombre-de-fuente/tu-nombre-de-fuente-700.woff2") format('woff2');
}
```

### 3. Establecer en el archivo .env:

```
REACT_APP_FONT_FAMILY=Tu Nombre de Fuente
```

#### Guías de Peso de Fuente:

- 300 - Ligera (opcional, para encabezados sutiles)
- 400 - Regular (requerido, texto predeterminado)
- 500 - Media (opcional, énfasis)
- 600 - Semi-Negrita (opcional, subtítulos)
- 700 - Negrita (requerido, encabezados y texto fuerte)

**Nota:** Todas las fuentes permanecen auto-alojadas y funcionan sin conexión. No se requiere CDN externo ni conexión a Internet.

#### Servicios Externos

```
REACT_APP_GOOGLE_API_KEY=tu_clave_api_de_google_maps
REACT_APP_STRIPE_PUBLISHABLE_KEY=pk_test_...
```

#### Campos:

- REACT\_APP\_GOOGLE\_API\_KEY - Clave API de Google Maps (para mapas, geolocalización)
- REACT\_APP\_STRIPE\_PUBLISHABLE\_KEY - Clave publicable de Stripe (para pagos)

#### Debe Coincidir:

REACT\_APP\_STRIPE\_PUBLISHABLE\_KEY debe coincidir con stripe.publishable\_key en crm\_config.yaml.

## Enlaces Rápidos de la Aplicación Web

```
REACT_APP_WEB_APP_1_NAME="GitHub"  
REACT_APP_WEB_APP_1_URL="https://github.com"  
REACT_APP_WEB_APP_1_ICON_PATH="resources/webapp_icons/github.png"
```

**Propósito:** Configura hasta 6 enlaces de acceso rápido a aplicaciones web en la navegación de la interfaz de usuario.

### Patrón:

- REACT\_APP\_WEB\_APP\_N\_NAME - Nombre a mostrar
- REACT\_APP\_WEB\_APP\_N\_URL - URL de destino
- REACT\_APP\_WEB\_APP\_N\_ICON\_PATH - Ruta del archivo de icono (relativa a public/)

**Ejemplos de Iconos:** GitHub, Xero, Monday.com, Gmail, MailJet, Slack

## Integración de Grafana

```
REACT_APP_GRAFANA_URLS=http://grafana1.local/d/  
abc,http://grafana2.local/d/xyz  
REACT_APP_GRAFANA_LABELS=Monitoreo de Red,Salud del Servicio  
REACT_APP_GRAFANA_API_KEY=tu_clave_api_de_grafana
```

### Campos:

- REACT\_APP\_GRAFANA\_URLS - Lista separada por comas de URLs de paneles de Grafana
- REACT\_APP\_GRAFANA\_LABELS - Lista separada por comas de nombres de paneles
- REACT\_APP\_GRAFANA\_API\_KEY - Clave API de Grafana para incrustar

**Uso:** Incrusta paneles de Grafana en la página de panel de OmniCRM.

## URLs de Soporte

```
REACT_APP_FAQS_URL=https://support.tuempresa.com/faqs  
REACT_APP_SUPPORT_URL=https://support.tuempresa.com
```

**Propósito:** Enlaces a recursos de soporte externos que se muestran en la interfaz de usuario.

## Inicios de Sesión Sociales

```
REACT_APP_ALLOW_SOCIAL_LOGINS=yes
```

### Opciones:

- yes - Habilitar botones de inicio de sesión social (Google, Facebook, etc.)
- no - Deshabilitar inicios de sesión sociales

**Nota:** Los proveedores de inicio de sesión social deben configurarse por separado.

## Configuración de Recarga y Recarga

```
REACT_APP_TOPUP_PRICE_PER_DAY=10
```

**Propósito:** Establece el precio por día para servicios de recarga en el portal de auto-cuidado.

### Campos:

- REACT\_APP\_TOPUP\_PRICE\_PER\_DAY - Precio diario para servicios de recarga (valor numérico)

**Ejemplo:** Si se establece en 10 y la moneda es USD, los clientes pagan \$10 por día de servicio.

**Nota:** Este valor debe coincidir con la configuración de precios del backend. Consulta `features_topup_recharge` para detalles completos de configuración.

## Aplicando Cambios de Configuración

### Backend (crm\_config.yaml)

1. Edita OmniCRM-API/crm\_config.yaml
2. Guarda los cambios
3. Reinicia el servicio de API:

```
cd OmniCRM-API
sudo systemctl restart omnicrm-api
# o
./restart_api.sh
```

**Los cambios entran en efecto inmediatamente** después del reinicio.

### Frontend (.env)

1. Edita OmniCRM-UI/.env
2. Guarda los cambios
3. Reconstruye la UI:

```
cd OmniCRM-UI
npm run build
```

#### 4. Reinicia el servicio de UI o el servidor web

### Modo de Desarrollo:

Durante el desarrollo con `npm start`, reinicia el servidor de desarrollo para aplicar los cambios.

## Mejores Prácticas de Configuración

### Seguridad

- **Nunca comprometas secretos** - Usa `.gitignore` para archivos de configuración con credenciales
- **Usa configuraciones específicas del entorno** - Configuraciones separadas para dev/staging/producción
- **Rota secretos regularmente** - Actualiza secretos JWT, claves API periódicamente
- **Limita permisos de claves API** - Asigna roles mínimos necesarios
- **Usa la lista blanca de IP con moderación** - Prefiere claves API para mejor seguridad

### Mantenimiento

- **Documenta cambios** - Mantén un registro de cambios de modificaciones de configuración
- **Haz copias de seguridad de configuraciones** - Almacena copias antes de cambios importantes
- **Prueba en staging** - Verifica cambios de configuración antes de la producción despliegue
- **Control de versiones** - Rastrear plantillas de configuración (sin secretos) en git

### Rendimiento

- **Usa base de datos local** - Evita base de datos remota para mejor rendimiento
- **Configura caché** - Habilita caché de OCS si está disponible
- **Optimiza Grafana** - Limita el número de paneles incrustados

### Marca

- **Coincide colores** - Asegúrate de que los colores de la UI complementen tu logo
- **Prueba contraste** - Verifica la legibilidad del texto en fondos coloreados
- **Pruebas móviles** - Verifica la marca en dispositivos móviles
- **Colocación del logo** - Coloca los logos de la empresa en OmniCRM-UI/`public/resources/`

# Solución de Problemas

## Cambios no aplicados

- **Causa:** Servicio no reiniciado o UI no reconstruida
- **Solución:** Reinicia los servicios de API/UI después de cambios de configuración

## Errores de sintaxis YAML

- **Causa:** Formato YAML inválido (sangrías, comillas, etc.)
- **Solución:** Valida YAML en línea o usa `yamllint crm_config.yaml`

## Conexión a la base de datos fallida

- **Causa:** Credenciales incorrectas o servidor inalcanzable
- **Solución:** Verifica que la base de datos esté en funcionamiento, las credenciales sean correctas

## Pagos de Stripe no funcionan

- **Causa:** Claves desajustadas entre backend y frontend
- **Solución:** Asegúrate de que `publishable_key` coincida en ambos archivos

## Correos electrónicos no enviados

- **Causa:** Credenciales de Mailjet inválidas o IDs de plantilla
- **Solución:** Verifica la clave/secreto de la API de Mailjet, verifica que existan los IDs de plantilla

## Documentación Relacionada

- `administration_api_keys` - Gestión de claves API
- `integrations_stripe` - Configuración de pagos de Stripe
- `integrations_mailjet` - Integración de correo electrónico
- `concepts_api` - Autenticación de API
- `rbac` - Control de acceso basado en roles



# Personalización y Configuración

Esta guía explica cómo personalizar y configurar OmniCRM para que coincida con la identidad de su marca, los requisitos operativos y las necesidades de integración.

## Variables de Entorno (.env)

OmniCRM utiliza variables de entorno para configurar tanto la API del backend como la interfaz de usuario del frontend. Hay dos archivos .env separados que controlan diferentes aspectos del sistema.

### Configuración de la API del Backend (.env)

Ubicado en la raíz del directorio de OmniCRM, este archivo configura las conexiones a la base de datos y la integración de CGRates.

**Ubicación:** /OmniCRM/.env

#### Configuración de la Base de Datos:

```
# Base de Datos MySQL/MariaDB (Datos del CRM)
MYSQL_ROOT_PASSWORD=your_secure_password
MYSQL_USER=omnitouch
MYSQL_PASSWORD=your_database_password
MYSQL_DATABASE=crm
DB_SERVER=db

# Base de Datos PostgreSQL (Datos de CGRates)
POSTGRES_USER=cgrates
POSTGRES_PASSWORD=cgrates_password
POSTGRES_DB=cgrates_db
```

#### Configuración de CGRates:

```
# Credenciales de la API de CGRates
CGRATES_API_USER=admin
CGRATES_API_PASS=secret
CGRATES_DB_USER=cgrates
CGRATES_DB_PASS=cgrates_password
CGRATES_DB_NAME=cgrates_db
CGRATES_DB_PORT=5432
```

#### Consideraciones de Seguridad:

- **Nunca comprometa archivos .env en el control de versiones** - Use .env.example como plantilla
- **Utilice contraseñas fuertes** - Mínimo 16 caracteres con mezcla de mayúsculas, números y símbolos
- **Rote las credenciales regularmente** - Especialmente para implementaciones en producción
- **Restringa el acceso a la base de datos** - Use listas blancas de IP y reglas de firewall

## Configuración de la Interfaz de Usuario del Frontend (.env)

Ubicado en el directorio OmniCRM-UI, este archivo controla la marca, la apariencia, las integraciones y las banderas de características.

**Ubicación:** /OmniCRM/OmniCRM-UI/.env

### Claves de API e Integración:

```
# API de Google Maps (para autocompletado de direcciones y
geocodificación)
REACT_APP_GOOGLE_API_KEY=your_google_api_key

# Pasarela de Pago Stripe
REACT_APP_STRIPE_PUBLISHABLE_KEY=pk_test_xxxxx

# Desactivar el lanzamiento automático del navegador al iniciar npm
BROWSER=none
```

### Información de Marca y Empresa:

```
# Marca de la Empresa
REACT_APP_COMPANY_NAME="ShellFone"
REACT_APP_PORTAL_NAME="ShellManager"
REACT_APP_SELF_CARE_NAME="ShellCare"
REACT_APP_COMPANY_TAGLINE="Phones with Shells"
```

Estos valores aparecen a lo largo de la interfaz de usuario:

- COMPANY\_NAME - Mostrado en títulos de página, correos electrónicos y comunicaciones con clientes
- PORTAL\_NAME - Nombre del portal para administradores/personal (por ejemplo, "ShellManager")
- SELF\_CARE\_NAME - Nombre del portal de autoservicio para clientes (por ejemplo, "ShellCare")
- COMPANY\_TAGLINE - Aparece en pantallas de inicio de sesión y materiales de marketing

### Localización y Configuración Regional:

```
# Idioma y Localidad
# Idiomas soportados: ar, ch, en, fr, gr, it, ru, sp
REACT_APP_DEFAULT_LANGUAGE=en
REACT_APP_LOCALE="en-GB"

# Ubicación Predeterminada (para autocompletado de direcciones)
REACT_APP_DEFAULT_LOCATION="Sydney, Australia"
REACT_APP_DEFAULT_COUNTRY="Australia"

# Configuración de Moneda
REACT_APP_CURRENCY_CODE="GBP"
REACT_APP_CURRENCY_SYMBOL="£"
```

## Personalización del Esquema de Color:

```
# Color Primario (color principal de la marca)
REACT_APP_PRIMARY_COLOR=#405189

# Opciones de Color Adicionales (ejemplos comentados)
# REACT_APP_SECONDARY_COLOR=#2bFFcf
# REACT_APP_TERTIARY_COLOR=#1a9fbf
# REACT_APP_SUCCESS_COLOR=#28a745
# REACT_APP_INFO_COLOR=#17a2b8
# REACT_APP_WARNING_COLOR=#ffc107
# REACT_APP_DANGER_COLOR=#dc3545
```

El color primario se aplica a:

- Encabezados de navegación
- Botones de acción
- Enlaces y resaltados
- Estados activos
- Elementos de marca

## Integraciones de Aplicaciones Web:

Configure hasta 6 aplicaciones web de acceso rápido que aparecen en el panel de administración:

```
# Aplicación Web 1: GitHub
REACT_APP_WEB_APP_1_NAME="GitHub"
REACT_APP_WEB_APP_1_URL="https://github.com"
REACT_APP_WEB_APP_1_ICON_PATH="resources/webapp_icons/github.png"

# Aplicación Web 2: Xero
REACT_APP_WEB_APP_2_NAME="Xero"
REACT_APP_WEB_APP_2_URL="https://go.xero.com/"
REACT_APP_WEB_APP_2_ICON_PATH="resources/webapp_icons/xero.png"
```

```
# Aplicación Web 3-6: Integraciones adicionales
# (Configurar de manera similar con NAME, URL e ICON_PATH)
```

## Monitoreo y Análisis:

```
# Integración del Panel de Grafana
REACT_APP_GRAFANA_URLS=url1,url2,url3
REACT_APP_GRAFANA_LABELS=label1,label2,label3
REACT_APP_GRAFANA_API_KEY=your-api-key
```

## Banderas de Características:

```
# URLs de Soporte y Documentación
REACT_APP_FAQS_URL=https://docs.yourcompany.com/faqs
REACT_APP_SUPPORT_URL=https://support.yourcompany.com

# Inicio de sesión social (Google, Facebook, etc.)
REACT_APP_ALLOW_SOCIAL_LOGINS=yes
```

# Personalización del Logo e Imagen de Presentación

OmniCRM le permite reemplazar las imágenes de marca predeterminadas con el logo y las pantallas de presentación de su empresa sin modificar el código.

## Archivos de Logo

Los logos se almacenan en `/OmniCRM-UI/src/assets/images/omnitouch/` y utilizan un sistema de respaldo:

**Logos Predeterminados** (siempre presentes):

- `DefaultLogoDark.png` - Logo de tema oscuro (utilizado en fondos claros)
- `DefaultLogoLight.png` - Logo de tema claro (utilizado en fondos oscuros)

**Logos Personalizados** (opcional, tienen prioridad cuando están presentes):

- `logoSm.png` - Logo pequeño para la barra lateral colapsada (recomendado: 100x100px)
- `logoDark.png` - Logo oscuro de tamaño completo para encabezados (recomendado: 200x50px)
- `logoLight.png` - Logo claro de tamaño completo para pantallas de autenticación (recomendado: 200x50px)

## Cómo Funciona el Respaldo de Logos:

El sistema intenta cargar primero los logos personalizados. Si un archivo de logo

personalizado no existe, se recurre al predeterminado:

```
// Desde Header.js
const tryImport = (filename) => {
  try {
    return require(`../assets/images/omnitouch/${filename}`);
  } catch (err) {
    return null; // Recurre al predeterminado
  }
};

const userLogoSm = tryImport("logoSm.png");
const userLogoDark = tryImport("logoDark.png");
const userLogoLight = tryImport("logoLight.png");
```

## Dónde Aparecen los Logos:

- **logoSm.png** - Barra lateral colapsada, navegación móvil, visualizaciones de encabezado pequeño
- **logoDark.png** - Barra de encabezado principal (modo claro), encabezado del panel de administración
- **logoLight.png** - Pantallas de inicio de sesión/registro, fondos oscuros, carrusel de autenticación

## Reemplazando Logos:

### 1. Cree Sus Archivos de Logo:

- Use formato PNG para soporte de transparencia
- Coincida con las dimensiones recomendadas arriba
- Asegúrese de que los logos sean claros tanto en resoluciones regulares como en retina

### 2. Agregue a OmniCRM:

```
# Copie sus archivos de logo al directorio de imágenes de
omnitouch
cp /path/to/your/logoSm.png OmniCRM-UI/src/assets/images/
omnitouch/
cp /path/to/your/logoDark.png OmniCRM-UI/src/assets/images/
omnitouch/
cp /path/to/your/logoLight.png OmniCRM-UI/src/assets/images/
omnitouch/
```

### 3. Reconstruya la Interfaz de Usuario:

```
cd OmniCRM-UI
npm run build
```

#### 4. Verifique los Cambios:

- Verifique el encabezado en modo claro (debería mostrar logoDark.png)
- Verifique el encabezado en modo oscuro (debería mostrar logoLight.png)
- Verifique la barra lateral colapsada (debería mostrar logoSm.png)
- Verifique la pantalla de inicio de sesión (debería mostrar logoLight.png)

#### Mejores Prácticas para el Diseño del Logo:

- **Contraste** - Asegúrese de que los logos sean visibles tanto en fondos claros como oscuros
- **Simplicidad** - Los logos deben ser reconocibles en tamaños pequeños
- **Formato** - Use PNG con fondos transparentes
- **Retina** - Proporcione resolución 2x para pantallas de alta DPI
- **Consistencia** - Use los mismos colores de marca en todas las variantes de logo

### Pantallas de Presentación y Fondos de Autenticación

Las pantallas de autenticación (inicio de sesión, registro, restablecimiento de contraseña) utilizan un fondo de carrusel con imágenes personalizables.

**Ubicación:** /OmniCRM-UI/src/pages/AuthenticationInner/authCarousel.js

#### Configuración Predeterminada:

```
import logoLight from "../../assets/images/logo-light.png";  
  
// Logo mostrado en pantallas de autenticación  
<img src={logoLight} alt="" height="18" />
```

#### Personalizando Pantallas de Autenticación:

1. Reemplace logo-light.png en /OmniCRM-UI/src/assets/images/
2. Agregue CSS de fondo personalizado a la clase .auth-one-bg
3. Modifique las citas del carrusel en authCarousel.js para que coincidan con la voz de su marca

#### Ejemplo de Personalización:

```
/* Agregue a su CSS personalizado */  
.auth-one-bg {  
  background-image: url('/assets/images/custom-auth-bg.jpg');  
  background-size: cover;  
  background-position: center;  
}
```

# Archivo de Configuración del CRM (crm\_config.yaml)

El archivo `crm_config.yaml` es la configuración central para la API de OmniCRM, controlando integraciones, aprovisionamiento, plantillas de correo electrónico y servicios externos.

**Ubicación:** /OmniCRM/OmniCRM-API/crm\_config.yaml

## Configuración de la Base de Datos

```
database:
  username: omnitouch
  password: omnitouch2024
  server: localhost
```

**Nota:** Esto debe coincidir con sus credenciales de base de datos en `.env`. En implementaciones en contenedores, el servidor es típicamente `db` (nombre del servicio de Docker).

## Tipos de Servicio

Defina tipos de servicio válidos para su implementación:

```
service_types:
  - omnicharge
  - mobile
  - fixed
  - fixed-voice
  - hotspot
  - dongle
```

Estos se utilizan en todo el sistema para:

- Categorización de productos
- Filtrado de complementos (los complementos coinciden con los tipos de servicio)
- Flujos de trabajo de aprovisionamiento
- Informes y análisis

## Integración HSS (Servidor de Suscriptores en Casa)

Para operadores de red móvil con integración HSS:

```
hss:
  hss_peers:
    - 'http://10.179.2.140:8080'
```

```
apn_list: "1,2,3,4,5,6"
```

## Configuración:

- hss\_peers - Lista de puntos finales HSS para el aprovisionamiento de suscriptores
- apn\_list - Lista separada por comas de IDs de APN disponibles para el aprovisionamiento

## Configuración de Plantillas de Correo Electrónico de Mailjet

OmniCRM utiliza Mailjet para correos electrónicos transaccionales. Cada tipo de correo electrónico tiene su propia configuración de plantilla:

```
mailjet:  
  api_key: your_mailjet_api_key  
  api_secret: your_mailjet_api_secret  
  
  # Correo Electrónico de Bienvenida al Cliente  
  api_crmCommunicationCustomerWelcome:  
    from_email: "support@yourcompany.com"  
    from_name: "Soporte de Su Empresa"  
    template_id: 5977509  
    subject: "Bienvenido a YourCompany"  
  
  # Correo Electrónico de Factura del Cliente  
  api_crmCommunicationCustomerInvoice:  
    from_email: "billing@yourcompany.com"  
    from_name: "Facturación de Su Empresa"  
    template_id: 6759851  
    subject: "Su Factura - "  
  
  # Recordatorio de Factura  
  api_crmCommunicationCustomerInvoiceReminder:  
    from_email: "billing@yourcompany.com"  
    from_name: "Facturación de Su Empresa"  
    template_id: 5977570  
    subject: "Recordatorio de Pago de Factura"  
  
  # Correo Electrónico de Bienvenida al Usuario (Personal/Admin)  
  api_crmCommunicationUserWelcome:  
    from_email: "admin@yourcompany.com"  
    from_name: "Admin de Su Empresa"  
    template_id: 6118112  
    subject: "Bienvenido al Equipo"  
  
  # Solicitud de Restablecimiento de Contraseña  
  api_crmCommunicationUserPasswordReset:
```

```

from_email: "security@yourcompany.com"
from_name: "Seguridad de Su Empresa"
template_id: 6735666
subject: "Solicitud de Restablecimiento de Contraseña"

# Confirmación de Éxito en el Restablecimiento de Contraseña
api_crmCommunicationUserPasswordResetSuccess:
  from_email: "security@yourcompany.com"
  from_name: "Seguridad de Su Empresa"
  template_id: 6118378
  subject: "Restablecimiento de Contraseña Exitoso"

# Notificación de Cambio de Contraseña
api_crmCommunicationUserPasswordChange:
  from_email: "security@yourcompany.com"
  from_name: "Seguridad de Su Empresa"
  template_id: 6118423
  subject: "Contraseña Cambiada"

# Verificación de Correo Electrónico
api_crmCommunicationEmailVerification:
  from_email: "verify@yourcompany.com"
  from_name: "Verificación de Su Empresa"
  template_id: 6267350
  subject: "Verifique Su Dirección de Correo Electrónico"

# Notificación de Saldo Expirado
api_crmCommunicationsBalanceExpired:
  from_email: "alerts@yourcompany.com"
  from_name: "Alertas de Su Empresa"
  template_id: 7238252
  subject: "Saldo de Servicio Expirado"

# Advertencia de Bajo Saldo
api_crmCommunicationsBalanceLow:
  from_email: "alerts@yourcompany.com"
  from_name: "Alertas de Su Empresa"
  template_id: 7238263
  subject: "Advertencia de Bajo Saldo"

```

## Creando Plantillas de Mailjet:

1. Inicie sesión en el panel de Mailjet (<<https://app.mailjet.com>>)
2. Navegue a **Transaccional → Plantillas**
3. Cree una nueva plantilla o clone una existente
4. Anote el **ID de Plantilla** (valor numérico)
5. Agregue variables de plantilla que coincidan con la estructura de datos de OmniCRM
6. Actualice `crm_config.yaml` con el ID de plantilla

## Variables de Plantilla Disponibles:

Cada tipo de correo electrónico recibe variables específicas. Ejemplos comunes:

- {{customer\_name}} - Nombre del cliente o usuario
- {{service\_name}} - Nombre del servicio o producto
- {{invoice\_id}} - Número de factura
- {{invoice\_amount}} - Monto total de la factura
- {{due\_date}} - Fecha de vencimiento del pago
- {{reset\_link}} - URL de restablecimiento de contraseña
- {{verification\_link}} - URL de verificación de correo electrónico
- {{balance}} - Saldo actual de la cuenta
- {{expiry\_date}} - Fecha de expiración de saldo o servicio

## Configuración de Aprovisionamiento

```
provisioning:  
  failure_list: ['admin@yourcompany.com', 'ops@yourcompany.com']
```

### Propósito:

- failure\_list - Direcciones de correo electrónico notificadas cuando falla el aprovisionamiento de Ansible
- Las notificaciones incluyen el nombre del playbook, detalles del error e información del cliente
- Permite al equipo de operaciones responder rápidamente a problemas de aprovisionamiento

## Configuración de Factura

```
invoice:  
  template_filename: 'yourcompany_invoice_template.html'
```

### Propósito:

Especifica qué plantilla HTML de Jinja2 usar para la generación de facturas PDF.

**Ubicación de la Plantilla:** /OmniCRM-API/invoice\_templates/

Consulte la sección **Generación de PDF de Factura** a continuación para obtener detalles sobre cómo crear plantillas personalizadas.

## URL Base del CRM

```
crm:  
  base_url: 'http://localhost:5000'
```

### Propósito:

- Utilizado por los playbooks de Ansible para hacer llamadas a la API
- Utilizado en plantillas de correo electrónico para generar enlaces al CRM
- Debe ser la URL accesible públicamente de su API (no nombres de contenedor internos)

### **Ejemplos:**

- Desarrollo: `http://localhost:5000`
- Producción: `https://api.yourcompany.com`
- Docker: `http://omnicrm-api:5000` (comunicación interna entre contenedores)

## **Configuración de OCS y CGRates**

```
ocs:
  ocsApi: 'http://10.179.2.142:8080/api'
  ocsTenant: 'mnc380.mcc313.3gppnetwork.org'
  cgrates: 'localhost:2080'
```

### **Configuración:**

- `ocsApi` - Punto final de la API de OCS para la gestión de suscriptores
- `ocsTenant` - Identificador de inquilino para implementaciones de OCS multi-inquilino
- `cgrates` - Punto final de la API JSON-RPC de CGRates (host:port)

## **Configuración de SMSC (Puerta de Enlace SMS)**

```
smsc:
  source_msisdn: 'YourCompany'
  smsc_url: 'http://10.179.2.216/SMSc/'
  api_key: 'your_smse_api_key'
```

### **Propósito:**

- Enviar notificaciones SMS a clientes (saldo bajo, alertas de servicio, códigos 2FA)
- `source_msisdn` - ID del remitente mostrado a los destinatarios (alfanumérico o número de teléfono)
- `smse_url` - Punto final de la API de la puerta de enlace SMSC
- `api_key` - Autenticación para la API de SMSC

## **Clave Secreta JWT**

```
jwt_secret:
  '2b93110f723db60172c8e9a1eaa80027a9a9c3f05b44e50dc3fcf38dba68d87e'
```

### **Seguridad:**

- Utilizada para firmar y verificar tokens de autenticación
- **DEBE ser cambiada del valor predeterminado** en producción
- Genere una cadena aleatoria segura (mínimo 64 caracteres)
- Nunca comparta ni comprometa en el control de versiones

## Generando una Nueva Clave Secreta JWT:

```
# Generar una clave aleatoria segura criptográficamente
python3 -c "import secrets; print(secrets.token_hex(32))"
```

## Configuración de Pago de Stripe

```
stripe:
  secret_key: 'sk_live_xxxxxxxxxxx'
  publishable_key: 'pk_live_xxxxxxxxxxx'
  currency: 'aud'
  statement_descriptor_suffix: 'YOURCOMPANY'
```

### Configuración:

- `secret_key` - Clave API secreta de Stripe (lado del servidor, mantenga confidencial)
- `publishable_key` - Clave publicable de Stripe (lado del cliente, seguro de exponer)
- `currency` - Código de moneda ISO 4217 (aud, usd, gbp, eur, etc.)
- `statement_descriptor_suffix` - Aparece en los estados de cuenta de tarjetas de crédito de los clientes

### Uso del Descriptor de Estado:

- Mostrado en los estados de cuenta bancarios de los clientes como "YOURCOMPANY"
- Máximo 22 caracteres
- Ayuda a los clientes a identificar cargos
- También se utiliza en los nombres de archivo PDF de facturas (por ejemplo, YOURCOMPANY\_12345.pdf)

## Claves de API y Listas Blancas de IP

Defina claves de API con acceso basado en roles y restricciones de IP:

```
api_keys:
  "YOUR_API_KEY_1":
    roles: ["admin"]
    ips: ["127.0.0.1", "::1"]
  "YOUR_API_KEY_2":
    roles: ["customer_service_agent_1"]
    ips: ["127.0.0.1", "::1", "10.0.1.0/24"]
```

```
# Lista Blanca de IP (independiente, sin clave de API)
ip_whitelist:
  "10.179.2.142":
    roles: ["admin"]
```

## Propósito:

- Permitir que sistemas externos se autentiquen a través de la clave de API
- Restringir el acceso por dirección IP
- Otorgar roles específicos a los consumidores de API
- Útil para integraciones (sistemas de facturación, monitoreo, automatización)

## Mejores Prácticas de Seguridad:

- Use claves de API largas y aleatorias (mínimo 32 caracteres)
- Restringa las IPs solo a fuentes conocidas
- Otorgue los roles mínimos necesarios
- Rote las claves de API regularmente
- Monitoree el uso de claves de API en los registros

# Generación de PDF de Factura

OmniCRM genera facturas PDF profesionales utilizando plantillas HTML de Jinja2 y renderizado PDF de WeasyPrint.

## Cómo Funciona la Generación de PDF

1. **Selección de Plantilla:**
  - El nombre del archivo de plantilla se especifica en `crm_config.yaml` bajo `invoice.template_filename`
  - La plantilla se carga desde el directorio `/OmniCRM-API/invoice_templates/`
2. **Preparación de Datos:**
  - Los datos de la factura (ID, fechas, montos, estado) se obtienen de la base de datos
  - La información del cliente (nombre, dirección) se recupera
  - Se compila la lista de transacciones (todos los cargos/créditos en la factura)
3. **Renderizado de Plantilla:**
  - Jinja2 renderiza la plantilla HTML con los datos de la factura
  - Variables como `{{ invoice_number }}`, `{{ total_amount }}`, etc. son reemplazadas
  - El HTML renderizado se guarda en `invoice_templates/rendered/` para depuración
4. **Generación de PDF:**
  - WeasyPrint convierte el HTML renderizado a PDF

- El PDF admite estilos CSS, imágenes, saltos de página, encabezados/pies de página
- Los datos binarios del PDF se generan en memoria

#### 5. Caché:

- El PDF se codifica en Base64 y se almacena en la tabla `Invoice_PDF_Cache`
- Se calcula un hash SHA256 para verificación de integridad
- Solicitudes posteriores devuelven el PDF en caché (entrega instantánea)

#### 6. Invalidación de Caché:

- La caché se invalida cuando la factura se modifica, anula o reembolsa
- Los cambios en la plantilla no invalidan automáticamente las cachés existentes

## Estructura de Plantilla de Factura

Las plantillas de factura son archivos HTML de Jinja2 con variables y lógica incrustadas.

**Ubicación de la Plantilla:** `/OmniCRM-API/invoice_templates/yourcompany_invoice_template.html`

### Variables Disponibles:

```
{
  'invoice_number': 12345,
  'date': '2025-01-04',
  'client': {
    'name': 'John Smith',
    'address': {
      'address_line_1': '123 Main St',
      'city': 'Sydney',
      'state': 'NSW',
      'zip_code': '2000',
      'country': 'Australia'
    }
  },
  'transaction_list': [
    [
      {
        'transaction_id': 1,
        'title': 'Servicio Móvil - Cuota Mensual',
        'retail_cost': 30.00,
        'wholesale_cost': 10.00,
        'created': '2025-01-01'
      },
      {
        'transaction_id': 2,
        'title': 'Llamadas ilimitadas',
        'retail_cost': 20.00,
        'wholesale_cost': 10.00,
        'created': '2025-01-01'
      }
    ]
  ]
}
```

```

        'transaction_id': 2,
        'title': 'Complemento de Datos - 5GB',
        'retail_cost': 15.00,
        'wholesale_cost': 5.00,
        'created': '2025-01-15'
    }
]
],
'total_amount': 45.00,
'due_date': '2025-01-31',
'start_date': '2025-01-01',
'end_date': '2025-01-31',
'paid': False,
'veoid': False
}

```

## Fragmento de Plantilla de Ejemplo:

```

<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8">
    <title>Factura {{ invoice_number }}</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 40px;
        }
        .header {
            text-align: center;
            margin-bottom: 30px;
        }
        .invoice-details {
            margin-bottom: 20px;
        }
        table {
            width: 100%;
            border-collapse: collapse;
        }
        th, td {
            border: 1px solid #ddd;
            padding: 12px;
            text-align: left;
        }
        th {
            background-color: #405189;
            color: white;
        }
        .total {

```

```

        text-align: right;
        font-size: 18px;
        font-weight: bold;
        margin-top: 20px;
    }

```

```
</style>
```

```
</head>
```

```
<body>
```

```
    <div class="header">
```

```
        
```

```
        <h1>FACTURA</h1>
```

```
</div>
```

```
<div class="invoice-details">
```

```
    <p><strong>Número de Factura:</strong> {{ invoice_number
}}</p>
```

```
    <p><strong>Fecha:</strong> {{ date }}</p>
```

```
    <p><strong>Fecha de Vencimiento:</strong> {{ due_date }}</p>
```

```
    <p><strong>Periodo de Facturación:</strong> {{ start_date }} a {{ end_date }}</p>
```

```
</div>
```

```
<div class="customer-details">
```

```
    <h3>Facturar a:</h3>
```

```
    <p>{{ client.name }}</p>
```

```
    <p>{{ client.address.address_line_1 }}</p>
```

```
    <p>{{ client.address.city }}, {{ client.address.state }} {{ client.address.zip_code }}</p>
```

```
    <p>{{ client.address.country }}</p>
```

```
</div>
```

```
<table>
```

```
    <thead>
```

```
        <tr>
```

```
            <th>Descripción</th>
```

```
            <th>Fecha</th>
```

```
            <th>Monto</th>
```

```
        </tr>
```

```
    </thead>
```

```
    <tbody>
```

```
        {% for transaction in transaction_list[0] %}
```

```
        <tr>
```

```
            <td>{{ transaction.title }}</td>
```

```
            <td>{{ transaction.created }}</td>
```

```
            <td>${{ "%2f" | format(transaction.retail_cost) }}
```

```
></td>
```

```
</tr>
```

```

        {% endfor %}
    </tbody>
</table>

<div class="total">
    <p>Total a Pagar: ${{ "%.2f" | format(total_amount) }}</p>
</div>

{% if paid %}
<div style="text-align: center; color: green; font-weight: bold;">
    PAGADO
</div>
{% endif %}

{% if void %}
<div style="text-align: center; color: red; font-weight: bold;">
    ANULADO
</div>
{% endif %}
</body>
</html>

```

## Mejores Prácticas para la Plantilla:

- **Use rutas absolutas para imágenes** - file:///absolute/path/to/image.png
- **CSS en línea** - WeasyPrint no carga hojas de estilo externas de manera confiable
- **Pruebe con datos de muestra** - Use invoice\_templates/rendered/ para inspeccionar HTML
- **Saltos de página** - Use <div style="page-break-after: always;"></div> para facturas de varias páginas
- **Encabezados y pies de página** - Use reglas CSS @page para elementos repetidos
- **Formato de moneda** - Use filtros de Jinja2: {{ "%.2f" | format(amount) }}

## Creando una Plantilla de Factura Personalizada

### 1. Copie la Plantilla de Ejemplo:

```

cd /OmniCRM/OmniCRM-API/invoice_templates
cp norfone_invoice_template.html
yourcompany_invoice_template.html

```

### 2. Edite la Plantilla:

- Reemplace el nombre de la empresa, logo, información de contacto
- Ajuste el estilo (colores, fuentes, diseño) para que coincida con la marca
- Agregue o elimine secciones según sea necesario (desglose de impuestos, instrucciones de pago, etc.)

### 3. Actualice la Configuración:

Edite `crm_config.yaml`:

```
invoice:
  template_filename: 'yourcompany_invoice_template.html'
```

### 4. Pruebe la Generación de Facturas:

- Cree una factura de prueba en el CRM
- Descargue el PDF y verifique el formato
- Verifique `invoice_templates/rendered/{invoice_id}.html` para depuración

### 5. Invalidar Cachés Antiguos (si es necesario):

Si ha cambiado la plantilla y desea regenerar facturas existentes:

```
-- Limpiar todos los PDFs en caché (forzar regeneración)
DELETE FROM Invoice_PDF_Cache;
```

## Sistema de Caché de PDF

Para mejorar el rendimiento, OmniCRM almacena en caché los PDFs generados:

### Comportamiento de Caché:

- **Primera Solicitud** - Se genera el PDF, se almacena en caché y se devuelve
- **Solicitudes Posteriores** - Se devuelve el PDF en caché de inmediato (sin regeneración)
- **Invalidación de Caché** - Ocurre cuando la factura se modifica, anula o reembolsa
- **Limpieza de Caché** - Las cachés antiguas se purgan automáticamente después de 30 días de inactividad

### Almacenamiento de Caché:

- PDF codificado en Base64 almacenado en la tabla `Invoice_PDF_Cache`
- Hash SHA256 del contenido para verificación de integridad
- Incluye nombre de archivo, marca de tiempo de creación, marca de tiempo de último acceso

### Gestión Manual de Caché:

```

# En la API de OmniCRM o shell de Python
from services.invoice_service import cleanup_old_pdf_cache,
invalidate_invoice_cache
from utils.db_helpers import get_db_session

session = get_db_session()

# Limpiar cachés más antiguas de 30 días
result = cleanup_old_pdf_cache(session, days_old=30)
print(result) # {'status': 'success', 'deleted_count': 15}

# Iniciar caché de factura específica
invalidate_invoice_cache(session, invoice_id='12345')

```

## Puntos Finales de API:

Generar/descargar PDF de factura:

GET /invoice/pdf/{invoice\_id}

**Respuesta:** Descarga del archivo PDF con nombre de archivo del descriptor de estado de Stripe

## Encabezados de Caché:

- Primera solicitud: Respuesta más lenta (tiempo de generación)
- Solicitudes en caché: Respuesta instantánea
- El acierto/fallo de caché es transparente para el usuario

## Solución de Problemas

### La Generación de PDF Falla:

- Verifique que WeasyPrint esté instalado: `pip install weasyprint`
- Verifique que el nombre de la plantilla coincida con `crm_config.yaml`
- Verifique `invoice_templates/rendered/` para errores de renderizado HTML
- Revise los registros de API para errores de plantilla Jinja2

### Las Imágenes No Aparecen en el PDF:

- Use rutas de archivos absolutas: `file:///full/path/to/image.png`
- Asegúrese de que los archivos de imagen existan y sean legibles
- Verifique el formato de imagen (PNG y JPEG funcionan mejor)
- Verifique que las rutas de imagen no contengan caracteres especiales

### Problemas de Estilo:

- Inline todo el CSS (hojas de estilo externas no son compatibles)

- Evite características CSS complejas (flexbox, grid pueden no renderizarse correctamente)
- Pruebe primero con diseños simples, agregue complejidad gradualmente
- Use tablas para el diseño en lugar de divs cuando sea posible

### **La Caché No se Invalida:**

- Verifique que `invalidate_invoice_cache()` se llame cuando se modifique la factura
- Verifique que las actualizaciones de transacciones desencadenen la invalidación de caché
- Elimine manualmente de la tabla `Invoice_PDF_Cache` si es necesario

## **Lista de Verificación de Configuración**

Utilice esta lista de verificación al implementar OmniCRM:

### **Configuración del Backend**

- [ ] Copie `.env.example` a `.env`
- [ ] Establezca contraseñas de base de datos fuertes
- [ ] Configure las credenciales de CGRates
- [ ] Actualice `crm_config.yaml` con sus configuraciones:
  - [ ] Conexión a la base de datos
  - [ ] Tipos de servicio
  - [ ] Claves de API de Mailjet y IDs de plantilla
  - [ ] Correos electrónicos de notificación de fallos de aprovisionamiento
  - [ ] Nombre de archivo de plantilla de factura
  - [ ] URL base del CRM (accesible públicamente)
  - [ ] Puntos finales de OCS/CGRates
  - [ ] Configuración de SMSC
  - [ ] Genere una nueva clave secreta JWT
  - [ ] Claves de Stripe (en vivo, no de prueba)
  - [ ] Claves de API y listas blancas de IP

### **Configuración del Frontend**

- [ ] Copie `OmniCRM-UI/.env.example` a `OmniCRM-UI/.env`
- [ ] Establezca la clave de API de Google Maps
- [ ] Establezca la clave publicable de Stripe
- [ ] Actualice la marca de la empresa:
  - [ ] Nombre de la empresa
  - [ ] Nombre del portal
  - [ ] Nombre de autoservicio
  - [ ] Lema de la empresa
- [ ] Configure la localización:
  - [ ] Idioma predeterminado

- [ ] Localidad
- [ ] Ubicación y país predeterminados
- [ ] Código de moneda y símbolo
- [ ] Establezca el color primario de la marca
- [ ] Configure integraciones de aplicaciones web (opcional)
- [ ] Agregue URLs de soporte y preguntas frecuentes (opcional)

## Activos de Marca

- [ ] Cree archivos de logo (logoSm.png, logoDark.png, logoLight.png)
- [ ] Cargue logos en OmniCRM-UI/src/assets/images/omnitouch/
- [ ] Cree HTML de plantilla de factura personalizada
- [ ] Cargue la plantilla de factura en OmniCRM-API/invoice\_templates/
- [ ] Actualice `crm_config.yaml` con el nombre de archivo de la plantilla de factura
- [ ] Pruebe la generación de PDF de factura
- [ ] Reconstruya la interfaz de usuario: `npm run build`

## Seguridad

- [ ] Cambie todas las contraseñas predeterminadas
- [ ] Genere un secreto JWT único
- [ ] Utilice claves de Stripe en producción (no claves de prueba)
- [ ] Rote las claves de API de Mailjet
- [ ] Habilite reglas de firewall
- [ ] Configure listas blancas de IP para el acceso a la API
- [ ] Configure certificados SSL/TLS
- [ ] Habilite HTTPS para todos los puntos finales
- [ ] Revise la configuración de CORS
- [ ] Implemente limitación de tasa
- [ ] Configure procedimientos de respaldo y recuperación

## Pruebas

- [ ] Pruebe el flujo de registro de clientes
- [ ] Pruebe el aprovisionamiento de servicios de extremo a extremo
- [ ] Verifique que las notificaciones por correo electrónico se envíen correctamente
- [ ] Pruebe la generación de facturas y la descarga de PDF
- [ ] Verifique el procesamiento de pagos (Stripe)
- [ ] Verifique la autenticación de usuarios y 2FA
- [ ] Pruebe la suplantación y el registro de auditoría
- [ ] Verifique que los datos de uso se sincronicen desde OCS
- [ ] Pruebe la creación y renovación de ActionPlan
- [ ] Confirme que la asignación de inventario funcione correctamente

## Implementación

- [ ] Construya imágenes de Docker o implemente en servidores
- [ ] Inicie contenedores de base de datos (MySQL, PostgreSQL)
- [ ] Inicie CGRates
- [ ] Inicie la API de OmniCRM
- [ ] Inicie la interfaz de usuario de OmniCRM
- [ ] Configure el proxy inverso (nginx, traefik)
- [ ] Configure monitoreo (Grafana, Prometheus)
- [ ] Configure agregación de registros
- [ ] Configure respaldos automáticos
- [ ] Documente la arquitectura de implementación
- [ ] Capacite al personal en el uso del sistema

## Documentación Relacionada

- RBAC y Gestión de Usuarios </rbac>
- Productos y Servicios </concepts\_products\_and\_services>
- Aprovisionamiento de Ansible </concepts\_ansible>
- Gestión de inventario </administration\_inventory>
- Facturas de Clientes </payments\_invoices>
- Autenticación de Dos Factores </2fa>
- Atención al Cliente y Suplantación </customer\_care>



# Resumen del Inventario en OmniCRM

El sistema de **Inventario** en OmniCRM está diseñado para gestionar y rastrear tanto artículos físicos como virtuales utilizados por operadores de red y clientes.

Esto significa que podemos rastrear todo tipo de artículos, como módems, números de teléfono, bloques de direcciones IP o incluso hardware físico como ONTs GPON o CPEs de Wireless Fijo.

Véase también: [Atributos del Cliente <administration\\_attributes>](#) para almacenar metadatos personalizados, y [Etiquetas del Cliente <administration\\_tags>](#) para la categorización visual.

Para apoyar a los clientes con una huella de red fija, el sistema de inventario también puede rastrear hogares pasados para un servicio determinado, lo que permite a los operadores realizar la calificación de servicio de forma remota, y para aquellos que operan una red inalámbrica fija, podemos rastrear los CPEs desplegados en el campo con sus ubicaciones.

## Nota

Los artículos de inventario están vinculados a productos durante la provisión a través del campo `inventory_items_list`. Para una guía completa sobre cómo se integra el inventario con la provisión de productos, incluida la interfaz de usuario del selector de inventario y la integración del libro de jugadas de Ansible, consulte [Guía Completa del Ciclo de Vida del Producto - Requisitos de Inventario <guide\\_product\\_lifecycle>](#).

## Propósito

El Inventario de OmniCRM cumple varios propósitos clave:

1. **Provisión de Servicios:** Cuando un cliente se inscribe en un servicio, puede ser necesario asignar artículos como módems, tarjetas SIM o números de teléfono. El sistema de inventario rastrea estos artículos y los asocia con los clientes.
2. **Gestión de Stock:** Para el stock físico, como hardware u otro equipo, el inventario ayuda a los operadores a mantener visibilidad sobre lo que está disponible, dónde se almacena y qué se ha asignado o vendido a los clientes.
3. **Asignación a Clientes:** El sistema permite que los artículos se asignen a los clientes, ya sea para su uso en un servicio (por ejemplo, asignar un

- módem a la cuenta de internet de un cliente) o para venta directa.
4. **Calificación de Servicio / Huella de Red:** Al almacenar información sobre la huella de red, como cada hogar pasado para un servicio GPON, o cada CPE de Wireless Fijo desplegado, permite al personal realizar la calificación de servicio de forma remota y ver si hay una huella de red en un área específica.

## **Ejemplo del Ciclo de Vida del Inventario**

Para ilustrar cómo funciona el sistema de Inventario, considere los siguientes ejemplos.

### **Ejemplo de Tarjeta SIM**

Se ordena un lote de 1000 tarjetas SIM de Omnitouch. Primero, se crea una Plantilla de Inventario para tarjetas SIM (si no existe ya) y las SIMs ordenadas se cargan en el Inventario en el estado En Tránsito.

Una vez que se reciben las SIMs, se marcan como En Stock, y pueden ser asignadas a diferentes tiendas minoristas, con la ubicación del artículo de inventario actualizada para reflejar en qué tienda minorista se encuentra cada tarjeta SIM. Esto es útil para rastrear los niveles de stock en cada tienda y asegurar que cada tienda tenga suficiente stock para satisfacer la demanda de los clientes.

Cuando un cliente se inscribe en un servicio móvil en la tienda, se asigna una tarjeta SIM al cliente y el estado se cambia a Asignado. El artículo de inventario de la tarjeta SIM se asigna entonces al cliente, y el estado se actualiza a En Uso.

Si el cliente cancela el servicio o el servicio queda inactivo, la tarjeta SIM se marca como Desactivada.

### **Ejemplo de Hogares Pasados GPON**

Para una red GPON, el sistema de inventario puede rastrear cada hogar pasado para un servicio determinado.

Cuando se construye una nueva área, cada dirección pasada puede ser añadida al inventario.

Esto permite a los operadores ver qué hogares están pasados para un servicio determinado y cuáles hogares aún no están pasados.

Cuando un cliente se inscribe en un servicio, OmniCRM puede ejecutar automáticamente una calificación de servicio contra la dirección del cliente, para ver si la dirección está en el inventario de hogares pasados y qué servicios pueden ofrecerse.

# Plantillas de Inventario

La función de **Plantilla de Inventario** permite la creación de cualquier número de tipos de artículos con campos predefinidos. Estas plantillas actúan como planos que definen las características esenciales de diferentes artículos, tales como:

- **Módems** con una dirección MAC (itemtext1) y un número de serie (itemtext2).
- **Hogares Pasados** para un servicio determinado, con una ubicación y estado (por ejemplo, pasado o no pasado).
- **Números de Teléfono** con un número principal (itemtext1) y una ubicación geográfica (itemtext2).
- **Recursos Virtuales** como bloques de direcciones IP, con identificadores mapeados a través de la plantilla.

Cada plantilla de inventario define hasta 20 campos de texto personalizables (itemtext1 a itemtext20) con etiquetas correspondientes (itemtext1\_label a itemtext20\_label) que describen lo que representa cada campo. Por ejemplo, una plantilla de Módem podría establecer itemtext1\_label como "Dirección MAC" y itemtext2\_label como "Número de Serie".

Los operadores pueden personalizar los campos para cada tipo de artículo utilizando **Plantillas de Inventario**. Estas plantillas permiten que los artículos sean categorizados y gestionados de manera estructurada, asegurando consistencia en cómo se rastrean los artículos.

## Vinculación a Productos:

Los nombres de las plantillas de inventario se refieren en las definiciones de productos a través del campo inventory\_items\_list. Al provisionar un producto, el sistema muestra un selector de inventario que solo muestra artículos que coinciden con los tipos de plantilla requeridos.

**Ejemplo:** Un producto con inventory\_items\_list: "[ 'Tarjeta SIM', 'Número Móvil' ]" requiere que existan dos plantillas de inventario nombradas exactamente "Tarjeta SIM" y "Número Móvil". Los nombres de las plantillas son sensibles a mayúsculas y minúsculas.

Para obtener detalles completos sobre cómo las plantillas de inventario se conectan a la provisión de productos, consulte Ciclo de Vida del Producto - Requisitos de Inventario [<guide\\_product\\_lifecycle>](#).

Visualización de plantillas en OmniCRM

## Creando Plantillas de Inventario a través de la Interfaz de Usuario

Para crear una nueva plantilla de inventario:

1. Navegue a **Inventario** → **Plantillas** desde el menú principal
2. Haga clic en el botón **Agregar Plantilla**
3. Complete los campos requeridos:

#### **Información Básica:**

- **Ícono** (opcional): Nombre de clase del ícono para identificación visual (por ejemplo, fa-solid fa-sim-card)
- **Artículo** (requerido): El nombre de la plantilla (debe coincidir exactamente con lo que se usa en inventory\_items\_list para productos)

#### **Información de Costo (Requerido):**

- **Costo Mayorista** (requerido): Su costo para comprar o provisionar este tipo de artículo
- **Costo Minorista** (requerido): Precio minorista estándar si se vende por separado a los clientes

#### Nota

Los costos mayorista y minorista establecidos aquí sirven como valores predeterminados al crear nuevos artículos de inventario a partir de esta plantilla. Los artículos de inventario individuales pueden tener diferentes costos si es necesario.

#### **Etiquetas de Campo:**

- **Etiqueta de Texto del Artículo 1** (requerido): Etiqueta para el primer campo personalizable (por defecto "Número de Modelo")
  - Ejemplos comunes: "ICCID" para tarjetas SIM, "Dirección MAC" para módems, "Número de Teléfono" para números
- **Etiqueta de Texto del Artículo 2** (requerido): Etiqueta para el segundo campo personalizable (por defecto "Número de Serie")
  - Ejemplos comunes: "IMSI" para tarjetas SIM, "Número de Serie" para hardware, "Región Geográfica" para números
- **Etiquetas de Texto del Artículo 3-20** (opcional): Etiquetas de campo adicionales según sea necesario
  - Haga clic en **Agregar Campo** para añadir más campos personalizados
  - Solo añada campos que realmente utilizará para este tipo de artículo

#### **Configuraciones de Visibilidad:**

- **Permitir Dropdown Personal:** Permitir que el personal seleccione este tipo de inventario en listas desplegables

- **Permitir Dropdown Cliente:** Permitir que los clientes vean/ seleccionen este tipo de inventario (portal del cliente)

4. Haga clic en **Guardar** para crear la plantilla

## **Editando Plantillas de Inventario**

Para editar una plantilla existente:

1. Navegue a **Inventario → Plantillas**
2. Encuentre la plantilla en la lista
3. Haga clic en el botón **Editar**
4. Modifique los campos según sea necesario
5. Haga clic en **Guardar**

### Advertencia

Cambiar las etiquetas de campo (por ejemplo, `itemtext1_label`) solo afecta a los nuevos artículos creados después del cambio. Los artículos de inventario existentes mantienen sus datos pero se mostrarán con los nuevos nombres de etiqueta.

### Precaución

Los nombres de las plantillas referenciados en los campos `inventory_items_list` de los productos son sensibles a mayúsculas y minúsculas. Renombrar una plantilla romperá el vínculo con los productos que usen el antiguo nombre.

## **Ejemplos Comunes de Plantillas**

### **Plantilla de Tarjeta SIM:**

- Artículo: "Tarjeta SIM"
- Costo Mayorista: 2.50
- Costo Minorista: 10.00
- Etiqueta de Texto del Artículo 1: "ICCID"
- Etiqueta de Texto del Artículo 2: "IMSI"
- Etiqueta de Texto del Artículo 3: "Tipo de SIM" (Físico/eSIM)

### **Plantilla de Número Móvil:**

- Artículo: "Número Móvil"
- Costo Mayorista: 1.00
- Costo Minorista: 0.00
- Etiqueta de Texto del Artículo 1: "Número de Teléfono"
- Etiqueta de Texto del Artículo 2: "Región Geográfica"
- Etiqueta de Texto del Artículo 3: "Tipo de Número" (Móvil/Fijo)

## Plantilla de CPE de Wireless Fijo:

- Artículo: "CPE de Wireless Fijo"
- Costo Mayorista: 250.00
- Costo Minorista: 450.00
- Etiqueta de Texto del Artículo 1: "Dirección MAC"
- Etiqueta de Texto del Artículo 2: "Número de Serie"
- Etiqueta de Texto del Artículo 3: "Versión de Firmware"
- Etiqueta de Texto del Artículo 4: "Fabricante"
- Etiqueta de Texto del Artículo 5: "Modelo"

## Plantilla de ONT GPON:

- Artículo: "ONT GPON"
- Costo Mayorista: 45.00
- Costo Minorista: 0.00 (incluido con el servicio)
- Etiqueta de Texto del Artículo 1: "Número de Serie"
- Etiqueta de Texto del Artículo 2: "Dirección MAC"
- Etiqueta de Texto del Artículo 3: "Ubicación PON"
- Etiqueta de Texto del Artículo 4: "Modelo"

Creando un Artículo a partir de una Plantilla en OmniCRM

## Creando y Gestiónando Artículos de Inventario

Una vez que se define una **Plantilla de Inventario**, se pueden crear artículos de **Inventario** individuales. Cada artículo de inventario representa una instancia específica de un tipo de artículo (por ejemplo, un módem o número de teléfono específico) que puede ser:

- **Asignado a Clientes:** Los artículos están vinculados a clientes para la provisión de servicios (por ejemplo, asignar hardware para una conexión a internet).
- **Rastreado para Stock:** Los operadores pueden monitorear el inventario disponible, como artículos no vendidos o no asignados.
- **Vendido o Desactivado:** Una vez vendido, los artículos se marcan con marcas de tiempo relevantes (por ejemplo, `sold_date`) y ya no pueden considerarse stock disponible.

Vista de Lista de Inventario Editar Artículo de Inventario

A través de este sistema, OmniCRM facilita la gestión eficiente del stock, ayuda a asignar recursos a los clientes y proporciona visibilidad detallada sobre el estado y la historia de cada artículo.

**Los Servicios** pueden estar vinculados a un artículo de **Inventario** dado, lo que permite un fácil seguimiento de qué artículos están asociados con qué clientes o servicios.

Una vez que un artículo de inventario ha sido asignado a un cliente, los juegos de Ansible pueden actualizar el estado y la historia del artículo para reflejar la asignación. Esto asegura que los operadores tengan un registro preciso de qué artículos están en uso y cuáles están disponibles para asignación, así como saber qué cliente está utilizando qué artículo.

Podemos ver los artículos asignados a un cliente desde la página de perfil del cliente en la pestaña **Inventario**.

#### Vista de Inventario del Cliente

Para los artículos de Inventario vinculados a un **Servicio**, podemos ver eso editando el servicio, para ver los artículos de Inventario vinculados.

Visualización de Artículos de Inventario vinculados a un Servicio en OmniCRM

## Campos de Artículos de Inventario

Cada artículo de inventario contiene información completa organizada en varias categorías:

### Información Básica del Artículo

- **inventory\_id** - Identificador único para el artículo de inventario (generado automáticamente)
- **item** - Tipo de artículo (coincide con el nombre de la Plantilla de Inventario, por ejemplo, "Tarjeta SIM", "Módem", "Número de Teléfono")
- **inventory\_template\_id** - Enlace a la Plantilla de Inventario que define este tipo de artículo
- **customer\_id** - Si está asignado a un cliente, el ID del cliente (nulo)
- **service\_id** - Si está vinculado a un servicio específico, el ID del servicio (nulo)
- **item\_location** - Ubicación física o lógica del artículo:
  - Para artículos físicos: edificio, almacén, ubicación en estante, nombre de la tienda, etc.
  - Para artículos virtuales: ubicación geográfica, ubicación del bloque IP, región del rango de números, etc.
- **item\_state** - Estado actual del artículo de inventario (valores enumerados):
  - **Nuevo** - Artículo completamente nuevo, sin usar
  - **Usado** - Anteriormente usado pero funcional
  - **Uso Interno** - Asignado para pruebas internas o uso del personal
  - **Asignado** - Actualmente asignado a un cliente o servicio
  - **Dañado** - No funcional, requiere reparación o eliminación
  - **Fuera de Servicio** - Temporalmente no disponible
  - **Perdido** - El artículo no puede ser localizado
  - **Robado** - El artículo fue robado

### Campos Personalizables del Artículo (de la Plantilla)

El sistema de inventario admite hasta 20 campos de texto personalizables cuyo significado está definido por la Plantilla de Inventario:

- **itemtext1** - Primer campo personalizable (requerido, etiqueta definida por `itemtext1_label` de la plantilla)
  - Ejemplo: Para módems, podría ser "Dirección MAC"
  - Ejemplo: Para tarjetas SIM, podría ser "ICCID"
  - Ejemplo: Para números de teléfono, podría ser "Número de Teléfono"
- **itemtext2 a itemtext20** - Campos personalizables adicionales (opcional, etiquetas definidas por la plantilla)
  - Ejemplo: itemtext2 para módems podría ser "Número de Serie"
  - Ejemplo: itemtext2 para tarjetas SIM podría ser "IMSI"
  - Ejemplo: itemtext3 para módems podría ser "Versión de Firmware"

Cada Plantilla de Inventario define cuáles de estos campos se utilizan y qué representan a través de los campos de etiqueta correspondientes (`itemtext1_label`, `itemtext2_label`, etc.).

## Información de Costo

- **wholesale\_cost** - Su costo para comprar/provisionar este artículo (float)
- **retail\_cost** - Precio cobrado al cliente si se vende por separado (float)
- **sold\_date** - Marca de tiempo cuando el artículo fue vendido o asignado al cliente

## Dirección Física (para Equipos de Red y Sitios)

Utilizada para rastrear ubicaciones de despliegue físico, particularmente para equipos de red fijos (CPEs, ONTs, módems) o hogares pasados:

- **address\_line\_1** - Dirección de la calle, número del edificio, número de unidad
- **address\_line\_2** - Información adicional de la dirección (suite, apartamento, piso)
- **city** - Ciudad o pueblo
- **state** - Estado, provincia o región
- **zip\_code** - Código postal
- **country** - Nombre del país

## Ubicación Geográfica (Autocompletada desde la Interfaz Web)

Al crear artículos de inventario a través de la Interfaz Web con autocompletado de direcciones, estos campos se completan automáticamente:

- **google\_maps\_place\_id** - ID de lugar de Google Maps para la dirección
- **plus\_code** - Código Plus de Google Maps (Código de Ubicación Abierto) para ubicación precisa
- **latitude** - Latitud geográfica (grados decimales)
- **longitude** - Longitud geográfica (grados decimales)

Estos campos permiten:

- Mapeo de ubicaciones de inventario en una vista de mapa
- Cálculos de proximidad para calificación de servicio
- Análisis de cobertura para planificación de red
- Optimización de rutas para el despacho de técnicos de campo

## Gestión de Dispositivos y URLs de Acceso

El campo `management_url` proporciona acceso rápido a interfaces de dispositivos y URLs de provisión:

- **management\_url** - URL de acceso para el artículo de inventario
  - **Equipos de Red:** URL de interfaz web (por ejemplo, `https://192.168.1.1` para enrutadores, conmutadores, ONTs, CPEs)
  - **Perfiles eSIM:** dirección LPA (Asistente de Perfil Local) para activación de eSIM (por ejemplo, `LPA:1$mdp.example.com$ACTIVATION-CODE-HERE`)
  - **Otros Casos de Uso:** Cualquier URL que necesite ser fácilmente accesible a través de un dispositivo móvil

## Generación de Código QR

Al ver artículos de inventario con un `management_url`, el sistema genera automáticamente un **código QR escaneable**:

- **Vista de detalle del artículo de inventario:** código QR de 128x128 mostrado junto a la URL
- **Tabla de inventario de servicio:** código QR de 64x64 mostrado para artículos asignados
- **Formato:** Tanto el código QR como el hipervínculo clicable se muestran juntos

## Casos de Uso Comunes

- **Técnicos de Red:** Escanear el código QR para acceder a la interfaz de gestión del dispositivo sin escribir direcciones IP
- **Activación de eSIM:** Los clientes escanean el código QR desde el CRM para instalar el perfil eSIM en su dispositivo
- **Autoservicio del Cliente:** Proporcionar acceso fácil a la configuración del dispositivo o portales para clientes
- **management\_username** - Nombre de usuario administrador para acceso al dispositivo
- **management\_password** - Contraseña de administrador para acceso al dispositivo (encriptada en reposo)

## Gestión de Configuración

Para dispositivos con archivos de configuración:

- **config\_content** - Contenido completo del archivo de configuración (almacenado como texto)
  - Útil para respaldo, versionado y recuperación ante desastres
  - Puede almacenar configuraciones de enrutadores, configuraciones de comutadores, configuraciones de CPE, etc.
- **config\_file\_path** - Ruta al archivo de configuración externo si se almacena por separado
  - Alternativa a almacenar la configuración completa en la base de datos
  - Ruta a un recurso compartido de red, repositorio de control de versiones o sistema de gestión de configuraciones

## Notas y Metadatos

- **inventory\_notes** - Notas en formato libre sobre el artículo de inventario
  - Notas de instalación
  - Historial de mantenimiento
  - Peculiaridades o requisitos de manejo especiales
  - Información del proveedor
  - Detalles de garantía
- **created** - Marca de tiempo cuando se creó el artículo de inventario en el sistema (establecido automáticamente)
- **last\_modified** - Marca de tiempo de la última actualización del artículo de inventario (actualizado automáticamente)

## Ejemplos de Uso de Campos

### Ejemplo 1: Tarjeta SIM Móvil

```
{  
  "inventory_id": 1001,  
  "item": "Tarjeta SIM",  
  "inventory_template_id": 5,  
  "itemtext1": "8961234567890123456",  
  "itemtext2": "310120123456789",  
  "itemtext3": "Físico",  
  "item_location": "Almacén A, Estante 3",  
  "item_state": "Asignado",  
  "customer_id": 456,  
  "service_id": 789,  
  "wholesale_cost": 2.50,  
  "retail_cost": 10.00,  
  "sold_date": "2025-01-15T10:30:00Z",  
  "inventory_notes": "Activado el 2025-01-15"}
```

```
}
```

## Ejemplo 2: Perfil eSIM Móvil

```
{
  "inventory_id": 1002,
  "item": "eSIM",
  "inventory_template_id": 6,
  "itemtext1": "8961234567890123457",
  "itemtext2": "310120123456790",
  "itemtext3": "eSIM",
  "item_location": "Inventario Virtual",
  "item_state": "Asignado",
  "customer_id": 457,
  "service_id": 790,
  "management_url": "LPA:1$smdp.example.com$ACTIVATION-CODE-ABC123XYZ",
  "wholesale_cost": 0.00,
  "retail_cost": 0.00,
  "sold_date": "2025-01-16T14:20:00Z",
  "inventory_notes": "Perfil eSIM listo para activación"
}
```

Al ver este artículo de inventario eSIM, la interfaz de usuario muestra un código QR que contiene la dirección LPA. Los clientes escanean este código QR con su dispositivo móvil para instalar el perfil eSIM.

## Ejemplo 3: Equipos de Cliente (CPE) - Wireless Fijo

```
{
  "inventory_id": 2001,
  "item": "CPE de Wireless Fijo",
  "inventory_template_id": 10,
  "itemtext1": "AA:BB:CC:DD:EE:FF",
  "itemtext2": "FW2024-12345",
  "itemtext3": "v2.4.1",
  "itemtext4": "Ubiquiti",
  "itemtext5": "LiteBeam AC Gen2",
  "item_location": "Sitio del Cliente",
  "item_state": "Asignado",
  "customer_id": 789,
  "service_id": 1234,
  "address_line_1": "123 Calle Principal",
  "address_line_2": "Apto 4B",
  "city": "Sídney",
  "state": "NSW",
  "zip_code": "2000",
  "country": "Australia",
  "latitude": "-33.8688",
```

```

"longitude": "151.2093",
"management_url": "https://192.168.100.1",
"management_username": "admin",
"management_password": "encrypted_password_here",
"config_file_path": "/configs/cpe/fw2024-12345.conf",
"inventory_notes": "Instalado el 2025-01-10. El cliente informa
excelente señal. Apuntando: Azimut 45°, Elevación 15°"
}

```

#### Ejemplo 4: ONT GPON con Dirección Completa

```

{
  "inventory_id": 3001,
  "item": "ONT GPON",
  "inventory_template_id": 15,
  "itemtext1": "ALCL12345678",
  "itemtext2": "AA:BB:CC:DD:EE:FF",
  "itemtext3": "OLT-1, PON 3, ONT 42",
  "itemtext4": "Nokia G-010G-A",
  "item_location": "Locales del Cliente",
  "item_state": "Asignado",
  "customer_id": 321,
  "service_id": 654,
  "address_line_1": "456 Fiber Lane",
  "city": "Melbourne",
  "state": "VIC",
  "zip_code": "3000",
  "country": "Australia",
  "google_maps_place_id": "ChIJ1234567890",
  "plus_code": "4RRH+2C Melbourne VIC",
  "latitude": "-37.8136",
  "longitude": "144.9631",
  "management_url": "https://192.168.1.1",
  "management_username": "admin",
  "config_content": "# Configuración de ONT\\nwlan-ssid: HomeNetwork\\
nwlan-password: encrypted...",
  "wholesale_cost": 45.00,
  "retail_cost": 0.00,
  "inventory_notes": "Provisionado el 2025-01-20. Potencia óptica:
-22dBm"
}

```

#### Nota

Al ver artículos de inventario con un `management_url` (como los Ejemplos 2, 3 y 4 arriba), la interfaz de usuario muestra automáticamente:

- Un código QR escaneable que contiene la URL o dirección LPA
- Un hipervínculo clicable (para URLs web)

## Casos de Uso:

- **Activación de eSIM** (Ejemplo 2): Los clientes escanean el código QR para instalar el perfil eSIM en su dispositivo
- **Acceso a Equipos de Red** (Ejemplos 3 y 4): Los técnicos escanean para acceder a las interfaces de gestión de dispositivos sin escribir manualmente las direcciones IP

## Ejemplo 5: Número de Teléfono (Inventario Virtual)

```
{  
  "inventory_id": 4001,  
  "item": "Número de Teléfono",  
  "inventory_template_id": 20,  
  "itemtext1": "+61412345678",  
  "itemtext2": "Melbourne",  
  "itemtext3": "Móvil",  
  "item_location": "Australia - VIC",  
  "item_state": "Asignado",  
  "customer_id": 555,  
  "service_id": 888,  
  "wholesale_cost": 1.00,  
  "retail_cost": 0.00,  
  "inventory_notes": "Portado de Telstra el 2025-01-05"  
}
```

## Estados de Artículos de Inventario Explicados

El campo `item_state` rastrea el ciclo de vida de los artículos de inventario:

- **Nuevo → Asignado** - Flujo normal al provisionar a un cliente
- **Asignado → Usado** - Después de la desactivación del servicio, el artículo puede ser reutilizado
- **Nuevo → Uso Interno** - Asignado para pruebas, demostraciones o uso del personal
- **Asignado → Dañado** - El artículo falló, requiere RMA o eliminación
- **Cualquier Estado → Perdido** - El artículo no puede ser localizado (activa auditoría)
- **Cualquier Estado → Robado** - El artículo fue robado (activa informe de seguridad)
- **Dañado/Usado → Nuevo** - Después de reacondicionamiento o reparación

Filtrar y buscar inventario por estado permite a los operadores:

- Rastrear stock disponible (Artículos Nuevos)
- Identificar artículos asignados a clientes (Asignados)
- Encontrar artículos disponibles para reutilización (Usados)
- Monitorear problemas de equipos (Dañados, Fuera de Servicio)

- Auditar artículos faltantes (Perdidos, Robados)



# Etiquetas de Clientes

Las etiquetas son enlaces codificados por colores que se pueden agregar a un cliente para ayudar a categorizarlos, por ejemplo, un cliente podría tener una etiqueta para "Ticket de Soporte Abierto" o "Factura Muy Atrasada" o "Desagradable".

Para almacenar metadatos estructurados y datos personalizados de clave-valor, consulte [Atributos del Cliente <administration\\_attributes>](#).

Las etiquetas se muestran como píldoras en la página de perfil del cliente, y el color de la píldora es personalizable junto con el enlace.

Un caso de uso común es etiquetar a los clientes que tienen un ticket de soporte abierto, para que el equipo de soporte pueda saltar fácilmente al ticket abierto desde la página de perfil del cliente.

Las etiquetas se pueden crear en el sistema por un administrador a través de la interfaz de usuario o por sistemas de terceros a través de la API y pueden tener fechas de inicio y fin, por lo que se pueden eliminar automáticamente después de un cierto período.

Etiquetas

## Gestión de Etiquetas a través de la UI

### Visualización de Etiquetas de Clientes

Para ver las etiquetas de un cliente:

1. Navegue a la página de resumen del cliente
2. Haga clic en la pestaña **Etiquetas**
3. Verá una lista de todas las etiquetas activas para el cliente, mostrando:
  - Vista previa de la etiqueta con el color configurado
  - Texto de la etiqueta
  - Fecha activa (cuando la etiqueta se vuelve visible)
  - Fecha de desactivación (cuando la etiqueta se ocultará)
  - Enlace (si está configurado)

### Creación de una Nueva Etiqueta

Para crear una nueva etiqueta para un cliente:

1. Navegue a la página de resumen del cliente

2. Haga clic en la pestaña **Etiquetas**
3. Haga clic en el botón **Agregar Etiqueta**
4. Complete los campos requeridos:
  - **Texto de la Etiqueta** (requerido): El texto que se mostrará en la etiqueta
  - **Color de la Etiqueta** (requerido): Elija un color utilizando el selector de color o ingrese un código hexadecimal
  - **Enlace de la Etiqueta** (opcional): URL que se abrirá cuando se haga clic en la etiqueta
  - **Fecha Activa** (requerido): Fecha en la que la etiqueta debe comenzar a mostrarse
  - **Fecha de Desactivación** (requerido): Fecha en la que la etiqueta debe dejar de mostrarse (por defecto es 2099-01-01)
5. Previsualice su etiqueta en la sección de vista previa
6. Haga clic en **Crear Etiqueta**

Formulario de Creación de Etiqueta de Cliente  
Visualización de Etiquetas de Cliente

## Edición de una Etiqueta

Para editar una etiqueta existente:

1. Navegue a la página de resumen del cliente
2. Haga clic en la pestaña **Etiquetas**
3. Encuentre la etiqueta que desea editar en la lista
4. Haga clic en el botón **Editar** (lápiz)
5. Modifique los campos según sea necesario
6. Haga clic en **Actualizar Etiqueta**

## Eliminación de una Etiqueta

Para eliminar una etiqueta:

1. Navegue a la página de resumen del cliente
2. Haga clic en la pestaña **Etiquetas**
3. Encuentre la etiqueta que desea eliminar en la lista
4. Haga clic en el botón **Eliminar** (papelera)
5. Confirme la eliminación en la ventana emergente

## Referencia de Campo de Etiqueta

## Integración de API

Las etiquetas también se pueden gestionar programáticamente a través de la API:

**Crear una Etiqueta:**

```
PUT /crm/tag/
{
  "tag_text": "Cliente VIP",
  "tag_hex_color": "FFD700",
  "tag_link": "https://example.com/vip",
  "tag_active_date": "2025-01-01 00:00:00",
  "tag_deactivate_date": "2099-12-31 23:59:59",
  "customer_id": 12
}
```

### **Actualizar una Etiqueta:**

```
PATCH /crm/tag/tag_id/{tag_id}
{
  "tag_text": "Texto de Etiqueta Actualizado",
  "tag_hex_color": "FF0000"
}
```

### **Obtener Etiquetas por Cliente:**

```
GET /crm/tag/customer_id/{customer_id}
```

### **Eliminar una Etiqueta:**

```
DELETE /crm/tag/tag_id/{tag_id}
```



# Arquitectura del Sistema OmniCRM

Este documento proporciona una visión general de la arquitectura del sistema OmniCRM, incluyendo las relaciones entre componentes y el flujo de datos.

## Visión General del Sistema

OmniCRM es una plataforma integral BSS/OSS que integra varios componentes clave para proporcionar una gestión completa de servicios para proveedores de telecomunicaciones.

## Componentes Clave

### 1. Aplicaciones Frontend

#### Interfaz Web OmniCRM

- Aplicación de una sola página en React
- Interfaz para el personal para gestión de clientes, aprovisionamiento de servicios, facturación
- Actualizaciones de estado de aprovisionamiento en tiempo real
- Control de acceso basado en roles

#### Portal de Autocuidado

- Portal orientado al cliente
- Gestión de servicios y seguimiento de uso
- Visualización y pago de facturas
- Base de código compartida con la interfaz del personal, diferentes vistas

### 2. Capa API

#### API OmniCRM (Flask/Python)

- API RESTful para todas las operaciones
- Documentación OpenAPI/Swagger
- Autenticación basada en JWT
- Limitación de tasa y caché
- Soporte para WebSocket para actualizaciones en tiempo real

#### Rutas Clave de la API:

- `/crm/customer/*` - Gestión de clientes

- /crm/service/\* - Operaciones de servicio
- /crm/product/\* - Catálogo de productos
- /crm/provision/\* - Operaciones de aprovisionamiento
- /crm/transaction/\* - Transacciones de facturación
- /crm/invoice/\* - Gestión de facturas

### 3. Sistema de Aprovisionamiento

### 4. Facturación y Tarificación

#### Integración OCS/CGRateS

- Tarificación y cargos en tiempo real
- Gestión de cuentas
- Seguimiento de saldos (monetario, datos, voz, SMS)
- Planes de acción para cargos recurrentes
- Notificaciones basadas en umbrales

#### Flujo de Trabajo de Facturación:

### 5. Modelo de Datos

OmniCRM utiliza una base de datos relacional con los siguientes modelos centrales. Para diagramas visuales de relaciones de entidades, consulte [Clientes, Contactos, Sitios y Servicios](#).

#### Cliente y Entidades Relacionadas

**Cliente** - Entidad central que representa una empresa o individuo

Campo	Tipo	Descripción
customer_id	Entero (PK)	Identificador único
customer_name	Cadena	Nombre de la empresa o individuo
customer_account_type	Enum	'Individual' o 'Negocio'
customer_status	Enum	'Abierto', 'Cerrado', 'Suspendido', 'Archivado'
customer_payment_type	Cadena	'prepagado' o 'postpagado'
customer_enabled	Booleano	¿Está activa la cuenta?
tax_identifier	Cadena	Número de IVA/GST
contract_start_date	Fecha/Hora	Inicio del contrato
contract_end_date	Fecha/Hora	Fin del contrato

**Contacto** - Personas asociadas a un cliente

Campo	Tipo	Descripción
contact_id	Entero (PK)	Identificador único
customer_id	Entero (FK)	Cliente padre

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
contact_firstname	Cadena	Nombre
contact_lastname	Cadena	Apellido
contact_email	Cadena	Dirección de correo electrónico
contact_phone	Cadena	Número de teléfono
contact_types	Cadena	'Primario', 'Facturación', 'Técnico'

## **Sitio** - Ubicaciones físicas de entrega de servicios

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
site_id	Entero (PK)	Identificador único
customer_id	Entero (FK)	Cliente padre
site_name	Cadena	Nombre de la ubicación
address_line_1	Cadena	Dirección de la calle
city, state, zip_code	Cadena	Detalles de la ubicación
latitude, longitude	Flotante	Coordenadas GPS
google_maps_place_id	Cadena	Referencia de Google Maps
plus_code	Cadena	Código de Ubicación Abierto

## **Modelos de Servicio y Producto**

### **Servicio** - Instancias de servicio activas

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
service_id	Entero (PK)	Identificador único
customer_id	Entero (FK)	Cliente padre
product_id	Entero (FK)	Plantilla de producto
site_id	Entero (FK)	Ubicación del servicio
service_name	Cadena	Nombre para mostrar
service_uuid	Cadena	Identificador del sistema de facturación
service_status	Enum	Estado actual
service_billed	Booleano	Generar transacciones
wholesale_cost	Flotante	Costo para el proveedor
retail_cost	Flotante	Precio para el cliente
bundled_parent	Entero (FK)	Servicio padre para paquetes

### **Producto** - Plantillas de ofertas de servicio

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
product_id	Entero (PK)	Identificador único
product_name	Cadena	Nombre para mostrar
product_slug	Cadena	Nombre amigable con URL
category	Enum	'independiente', 'paquete', 'complemento',

Campo	Tipo	Descripción
		'promoción'
provisioning_play	Cadena	Nombre del playbook de Ansible
provisioning_json_vars	JSON	Variables del playbook
inventory_items_list	Cadena	Inventario requerido
retail_cost	Flotante	Precio mensual
retail_setup_cost	Flotante	Cargo único
enabled	Booleano	Disponible para la venta

## Modelos de Facturación

**Transacción** - Cargos/créditos individuales

Campo	Tipo	Descripción
transaction_id	Entero (PK)	Identificador único
customer_id	Entero (FK)	Cliente padre
invoice_id	Entero (FK)	Factura padre (opcional)
service_id	Entero (FK)	Servicio relacionado
title	Cadena	Descripción de la transacción
retail_cost	Flotante	Monto
tax_percentage	Flotante	Tasa de impuesto
tax_amount	Flotante	Impuesto calculado
void	Booleano	Transacción cancelada

**Factura** - Transacciones agrupadas para facturación

Campo	Tipo	Descripción
invoice_id	Entero (PK)	Identificador único
customer_id	Entero (FK)	Cliente padre
paid	Booleano	Pago recibido
void	Booleano	Factura cancelada
payment_reference	Cadena	ID de transacción de Stripe
start_date, end_date	Fecha	Período de facturación
due_date	Fecha	Fecha límite de pago
retail_cost	Flotante	Monto total

## Modelos de inventario

**Inventario** - Activos físicos y virtuales

Campo	Tipo	Descripción
inventory_id	Entero (PK)	Identificador único
customer_id	Entero (FK)	Cliente asignado
service_id	Entero (FK)	Servicio vinculado

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
inventory_template_id	Entero (FK)	Plantilla de tipo de artículo
item	Cadena	Tipo de artículo (Tarjeta SIM, Router, etc.)
item_state	Enum	'Nuevo', 'Asignado', 'Usado', etc.
itemtext1-20	Cadena	Campos flexibles
management_url	Cadena	URL de administración del equipo
config_content	Texto	Archivo de configuración

**Inventory\_Template** - Define la estructura del artículo de inventario

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
inventory_template_id	Entero (PK)	Identificador único
item	Cadena	Nombre de la plantilla
itemtext1_label	Cadena	Etiqueta para el campo itemtext1
itemtext2_label	Cadena	Etiqueta para el campo itemtext2

## Modelos de Aprovisionamiento

**Aprovisionar** - Seguimiento de trabajos de aprovisionamiento

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
provision_id	Entero (PK)	Identificador único
product_id	Entero (FK)	Producto que se está aprovisionando
customer_id	Entero (FK)	Cliente objetivo
service_id	Entero (FK)	Servicio creado/modificado
provisioning_play	Cadena	Nombre del playbook de Ansible
provisioning_status	Entero	0=Éxito, 1=Ejecutándose, 2=Fallido

**Provision\_Event** - Pasos individuales de aprovisionamiento

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
provision_event_id	Entero (PK)	Identificador único
provision_id	Entero (FK)	Trabajo de aprovisionamiento padre
event_name	Cadena	Nombre de la tarea
event_number	Entero	Número de secuencia
provisioning_status	Entero	0=Éxito, 1=Ejecutándose, 2=Fallido
provisioning_result_json	JSON	Salida completa de la tarea

## Modelos de Usuario y Seguridad

**Usuario** - Cuentas de usuario

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	Entero (PK)	Identificador único

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
username	Cadena	Nombre de usuario para iniciar sesión
email	Cadena	Dirección de correo electrónico
email_verified	Booleano	Correo confirmado
is_2fa_enabled	Booleano	Autenticación de dos factores habilitada
totp_secret	Cadena	Clave secreta TOTP

### **Rol** - Roles de usuario

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	Entero (PK)	Identificador único
name	Cadena	Nombre del rol
description	Cadena	Descripción del rol

### **Permiso** - Permisos granulares

<b>Campo</b>	<b>Tipo</b>	<b>Descripción</b>
id	Entero (PK)	Identificador único
name	Cadena	Nombre del permiso (recurso.acción)
description	Cadena	Descripción del permiso

### **Relaciones:**

- Los usuarios tienen muchos roles (muchos a muchos)
- Los roles tienen muchos permisos (muchos a muchos)
- Los usuarios pueden vincularse a un contacto (para acceso al portal del cliente)

## **Puntos de Integración**

### **Pasarela de Pago Stripe**

- Métodos de pago tokenizados
- Almacenamiento de tarjetas conforme a PCI
- Pago automático de facturas
- Procesamiento de reembolsos
- Notificaciones de tarjetas que están por expirar

### **Servicio de Correo Electrónico Mailjet**

- Correos electrónicos transaccionales (facturas, bienvenida, notificaciones)
- Sincronización de contactos
- Gestión de plantillas
- Seguimiento de entregas

## Google Maps

- Autocompletar direcciones
- Geocodificación y geocodificación inversa
- Generación de Plus Code
- Mapeo de ubicación de sitios

## OCS/CGRateS

- Aprovisionamiento de cuentas
- Tarificación en tiempo real
- Gestión de saldos
- Procesamiento de CDR
- Planes de acción y horarios

## Arquitectura de Seguridad

### Características de Seguridad:

- Autenticación basada en JWT
- Autenticación de dos factores (TOTP)
- Control de acceso basado en roles (RBAC)
- Acceso a recursos basado en permisos
- Gestión de sesiones con Redis
- Hashing de contraseñas (bcrypt)
- Verificación de correo electrónico
- Registro de auditoría a través del Registro de Actividades

## Arquitectura de Despliegue

### Configuración de Producción Recomendada:

## Stack Tecnológico

### Backend:

- Python 3.x
- Framework web Flask
- SQLAlchemy ORM
- Migraciones de Alembic
- Ansible para automatización

### Frontend:

- React
- Gestión de estado

- React Router
- Axios para llamadas a la API

#### **Base de Datos:**

- Base de datos relacional (almacenamiento de datos primario)
- Redis (caché y sesiones)

#### **Servicios Externos:**

- CGRateS (facturación/tarificación)
- Stripe (pagos)
- Mailjet (correo electrónico)
- Google Maps (geocodificación)

## **Consideraciones de Escalabilidad**

#### **Escalado Horizontal:**

- Diseño de API sin estado permite múltiples instancias
- El balanceador de carga distribuye solicitudes
- Redis para estado de sesión compartido

#### **Escalado de Base de Datos:**

- Réplicas de lectura para consultas de informes
- Agrupamiento de conexiones
- Optimización de consultas e indexación

#### **Aprovisionamiento a Escala:**

- Procesamiento de trabajos en segundo plano
- Múltiples trabajadores de Ansible
- Gestión de cola de trabajos
- Lógica de reintento para aprovisionamientos fallidos

## **Monitoreo y Observabilidad**

#### **Métricas Clave:**

- Tiempos de respuesta de la API
- Tasas de éxito/fallo de aprovisionamiento
- Uso del grupo de conexiones de la base de datos
- Tasas de aciertos de caché
- Sesiones activas

#### **Registro:**

- Registros de aplicación (Flask)
- Registros de aprovisionamiento (salida de Ansible)
- Registro de actividades (ruta de auditoría)
- Seguimiento de errores y alertas

## Documentación Relacionada

- [Playbooks de Ansible](#) - Automatización de aprovisionamiento
- [Sistema de Aprovisionamiento](#) - Detalles del flujo de trabajo
- [Productos y Servicios](#) - Arquitectura de productos
- [Documentación de API](#) - Referencia de API
- [RBAC](#) - Seguridad y permisos



# Flujos de Autenticación y Controles de Administrador

OmniCRM proporciona características de autenticación completas que incluyen inicio de sesión, autenticación de dos factores (2FA), gestión de contraseñas y controles de administrador para gestionar la seguridad del usuario. Esta guía se centra en los flujos de trabajo de la interfaz de usuario tanto para usuarios finales como para administradores.

Véase también: [Self-Care Portal <self\\_care\\_portal>](#) para el inicio de sesión del cliente y acceso al portal, RBAC [<rbac>](#) para permisos del personal.

## Descripción General

La autenticación de OmniCRM incluye:

- **Inicio de Sesión con Correo Electrónico/Contraseña** - Autenticación estándar basada en credenciales
- **Autenticación de Dos Factores (2FA)** - Segundo factor basado en TOTP opcional
- **Recordarme** - Sesión extendida de hasta 30 días
- **Restablecimiento de Contraseña** - Recuperación de contraseña de autoservicio a través de correo electrónico
- **Controles de Administrador** - Herramientas administrativas para restablecer 2FA y contraseñas
- **Inicios de Sesión Sociales** - Integración opcional con Google, Apple, Facebook (si está habilitada)
- **Navegación Basada en Roles** - Enrutamiento automático basado en roles de usuario

## Flujo de Inicio de Sesión

La página de inicio de sesión es el punto de entrada para todos los usuarios (personal y clientes).

### Inicio de Sesión Estándar

Pantalla de Inicio de Sesión

#### Proceso de Inicio de Sesión:

1. Ingrese **dirección de correo electrónico** (correo electrónico del personal o cliente)
2. Ingrese **contraseña**
3. Opcional: Marque "**Recordarme por 30 días**" para una sesión extendida
4. Haga clic en "**Iniciar sesión**"

#### Qué Sigue Despues:

- **Sin 2FA:** Usuario inicia sesión inmediatamente, navegado según el rol:
  - **Clientes** → portal de autoservicio (/self-care)
  - **Personal/Administradores** → panel de clientes (/customers)
  - **Modo CBC** → interfaz de difusión celular (/create-cell-broadcast)
- **Con 2FA Habilitado:** Redirigido a la pantalla de verificación de 2FA

#### Función Recordarme:

Cuando está habilitada, la sesión persiste durante **30 días** en lugar de expirar cuando se cierra el navegador. Utiliza cookies seguras solo HTTP.

#### **Mostrar/Ocultar Contraseña:**

Haga clic en el **ícono de ojo (◊)** para alternar la visibilidad de la contraseña.

### **Inicio de Sesión con 2FA**

Si el usuario tiene 2FA habilitado, después de ingresar el correo electrónico/contraseña, aparece la pantalla de desafío de 2FA:

Pantalla de Autenticación de Dos Factores

#### **Usando el Código del Autenticador:**

1. Abra la aplicación de autenticador (Google Authenticator, Authy, etc.)
2. Encuentre la entrada de OmniCRM
3. Ingrese el código de 6 dígitos
4. El código se envía automáticamente cuando se ingresan los 6 dígitos
5. Si es válido, el usuario inicia sesión y se navega al panel correspondiente

#### **Usando el Código de Recuperación:**

Si la aplicación de autenticador no está disponible:

1. Haga clic en la pestaña "**Código de Recuperación**"
2. Ingrese uno de sus códigos de respaldo guardados (por ejemplo, 3fa5b9c2)
3. Haga clic en "**Verificar**"
4. El código se consume (solo se puede usar una vez)

#### **Cancelar:**

Haga clic en "**Cancelar**" para volver a la página de inicio de sesión.

### **Inicios de Sesión Sociales (Opcional)**

Si está habilitado (REACT\_APP\_ALLOW\_SOCIAL\_LOGINS=yes), aparecen botones de inicio de sesión social:

[◊ Iniciar sesión con Google] [● Iniciar sesión con Apple] [◊ Iniciar sesión con Facebook]

Haga clic en cualquier botón para autenticarse a través de ese proveedor. Actualmente muestra un mensaje de "próximamente" (implementación de inicio de sesión social en progreso).

#### **Enlace de Olvidé mi Contraseña**

Haga clic en el enlace "**¿Olvidó su contraseña?**" para iniciar el flujo de restablecimiento de contraseña.

### **Configuración de Autenticación de Dos Factores (2FA)**

Los usuarios pueden habilitar 2FA para una mayor seguridad de la cuenta. 2FA utiliza TOTP (Contraseña de un Solo Uso Basada en el Tiempo) compatible con aplicaciones de autenticador estándar.

## Acceso a la Configuración de 2FA

Desde el perfil o la configuración del usuario:

### Nota para Clientes:

Los usuarios con rol de cliente no ven los mensajes de 2FA. 2FA generalmente se requiere solo para el personal y los usuarios administrativos.

### Paso 1: Confirmar Contraseña

Contraseña Actual

[Cancelar] [Continuar]

Ingrese su contraseña actual para continuar. Esto verifica su identidad antes de habilitar 2FA.

### Paso 2: Escanear Código QR

[Cancelar] [Confirmar]

#### Instrucciones de Configuración:

- 1. Descargar la Aplicación de Autenticador** (si no tiene una):
  - iOS: Apple App Store → "Google Authenticator"
  - Android: Google Play → "Google Authenticator"
  - Alternativas: Authy, Microsoft Authenticator, 1Password
- 2. Escanear Código QR:**
  - Abra la aplicación de autenticador
  - Toque "+" o "Aregar cuenta"
  - Elija "Escanear código QR"
  - Apunte la cámara al código QR en la pantalla
  - La aplicación agrega la entrada "OmniCRM" con el código de 6 dígitos
- 3. Guardar Códigos de Respaldo:**
  - **CRÍTICO:** Anote o copie estos 8 códigos
  - Almacene en un lugar seguro (gestor de contraseñas, caja fuerte, etc.)
  - Cada código es de un solo uso
  - Se utiliza si pierde el acceso a la aplicación de autenticador
  - Haga clic en "**Copiar Códigos**" para copiar todos los códigos al portapapeles
- 4. Verificar Configuración:**
  - Ingrese el código actual de 6 dígitos de la aplicación de autenticador
  - Haga clic en "**Confirmar**"
  - Si es válido, 2FA ahora está habilitado

### Paso 3: 2FA Habilitado

Aparece un mensaje de éxito:

De ahora en adelante, el inicio de sesión requiere tanto la contraseña como el código de 2FA.

Interfaz de Configuración de 2FA

## Flujo de Restablecimiento de Contraseña (Autoservicio)

Los usuarios que olvidan su contraseña pueden restablecerla a través de correo electrónico.

## **Paso 1: Solicitar Enlace de Restablecimiento**

Desde la página de inicio de sesión, haga clic en "**¿Olvidó su contraseña?**"

Solicitud de Restablecimiento de Contraseña

1. Ingrese **dirección de correo electrónico**
2. Haga clic en "**Enviar Enlace de Restablecimiento**"

### **Qué Sucede:**

- El sistema verifica si el correo electrónico existe en la base de datos
- Si se encuentra, envía un correo electrónico de restablecimiento de contraseña a través de Mailjet
- El correo electrónico contiene un enlace de restablecimiento limitado en el tiempo (típicamente 1 hora de expiración)
- Aparece un mensaje de éxito: "Las instrucciones de restablecimiento se han enviado a su correo electrónico"

### **Si No Se Encuentra el Correo Electrónico:**

Mensaje de error: "No se encontró ninguna cuenta con esa dirección de correo electrónico"

## **Paso 2: Revisar Correo Electrónico**

El usuario recibe un correo electrónico con un asunto como:

Hola [Nombre],

Ha solicitado un restablecimiento de contraseña para su cuenta de OmniCRM.

Haga clic en el enlace a continuación para restablecer su contraseña:  
<<https://yourcompany.com/reset-password/abc123token456>>

Este enlace expira en 1 hora.

Si no solicitó esto, ignore este correo electrónico.

Haga clic en el enlace de restablecimiento para continuar.

## **Paso 3: Establecer Nueva Contraseña**

El enlace de restablecimiento abre la página de creación de contraseña:

◊ (ícono de candado)

Contraseña

Confirmar Contraseña

[Restablecer Contraseña]

1. Ingrese **nueva contraseña**
2. Vuelva a ingresarla en el campo **Confirmar Contraseña**
3. Haga clic en "**Restablecer Contraseña**"

### **Requisitos de Contraseña:**

- Longitud mínima (típicamente 8+ caracteres)
- Las contraseñas deben coincidir

#### Éxito:

- Mensaje de éxito: "La contraseña se ha restablecido con éxito"
- Redirigido automáticamente a la página de inicio de sesión
- El usuario ahora puede iniciar sesión con la nueva contraseña

#### Token Expirado/Inválido:

Si el enlace de restablecimiento ha expirado o es inválido:

[Solicitar Nuevo Enlace de Restablecimiento]

## Controles de Administrador para la Gestión de Usuarios

Los administradores con permisos apropiados pueden gestionar la configuración de autenticación de usuarios desde la interfaz de Gestión de Usuarios.

### Acceso a la Gestión de Usuarios

Muestra una tabla de todos los usuarios con botones de acción.

Nombre	Correo Electrónico	Teléfono	Acciones
John Smith	< <a href="mailto:john@example.com">john@example.com</a> >	+44 123...	    
Jane Doe	< <a href="mailto:jane@example.com">jane@example.com</a> >	+44 456...	    
Bob Wilson	< <a href="mailto:bob@example.com">bob@example.com</a> >	+44 789...	    

     **Iconos de Acción:**

-  **Editar** - Modificar detalles del usuario, roles, permisos
-  **Eliminar** - Eliminar cuenta de usuario
-  **Restablecer Contraseña** - Generar contraseña temporal
-  **Restablecer 2FA** - Deshabilitar 2FA para el usuario (solo se muestra si 2FA está habilitado)
-  **Enviar Correo Electrónico de Bienvenida** - Reenviar correo electrónico de bienvenida (solo se muestra si el usuario nunca ha iniciado sesión)

### Administrador: Restablecer Contraseña de Usuario

Cuando el usuario olvida la contraseña y el administrador necesita ayudar:

#### Paso 1: Hacer clic en el Icono de Restablecer Contraseña (◊)

Aparece un modal de confirmación:

¿Está seguro de que desea restablecer la contraseña de:

Usuario: John Smith (<[john@example.com](mailto:john@example.com)>)

Se generará y mostrará una contraseña temporal. El usuario debe cambiar esta contraseña en el próximo inicio de sesión.

[Cancelar] [Restablecer Contraseña]

#### Paso 2: Confirmar Restablecimiento

Haga clic en "**Restablecer Contraseña**". El sistema genera una contraseña temporal segura.

### **Paso 3: Contraseña Temporal Mostrada**

Contraseña temporal para John Smith:

[ Copiar Contraseña]

 **IMPORTANTE:** • Envíe esta contraseña al usuario a través de un canal seguro • No la envíe por correo electrónico o mensajería no segura • El usuario deberá cambiar la contraseña en el próximo inicio de sesión

[Cerrar]

#### **Acción del Administrador:**

- Copiar la contraseña temporal
- Llamar al usuario o comunicarse por un método seguro
- Proporcionar la contraseña temporal verbalmente
- Indicar al usuario que inicie sesión y cambie la contraseña

#### **Experiencia del Usuario:**

Cuando el usuario inicia sesión con la contraseña temporal:

1. El inicio de sesión tiene éxito
2. Redirigido inmediatamente a la pantalla "Cambiar Contraseña"
3. Debe establecer una nueva contraseña antes de acceder al sistema
4. No puede omitir el cambio de contraseña

### **Administrador: Restablecer 2FA de Usuario**

Cuando el usuario pierde el acceso a la aplicación de autenticador y todos los códigos de respaldo:

#### **Paso 1: Hacer clic en el Icono de Restablecer 2FA ()**

Solo aparece para usuarios con 2FA actualmente habilitado.

Aparece un modal de confirmación:

Confirmación de Restablecimiento de 2FA de Usuario por Administrador

#### **Paso 2: Confirmar Restablecimiento**

Haga clic en "**Restablecer 2FA**"

#### **Paso 3: Confirmación**

Mensaje de éxito:

John Smith ahora puede iniciar sesión solo con su contraseña. Puede volver a habilitar 2FA desde su configuración de usuario.

#### **Experiencia del Usuario:**

- El usuario ahora puede iniciar sesión solo con la contraseña (no se requiere código 2FA)
- El icono de escudo 2FA () desaparece de la fila del usuario en la tabla de administración

- El usuario puede volver a habilitar 2FA voluntariamente desde su configuración

### **Nota de Seguridad Importante:**

Antes de restablecer 2FA, el administrador debe:

1. Verificar la identidad del usuario a través de medios alternativos:
  - Verificación de identificación gubernamental
  - Preguntas de seguridad
  - Verificación de transacciones recientes
  - Verificación en persona (si aplica)
2. Documentar el restablecimiento en las notas del cliente
3. Informar al usuario que vuelve a habilitar 2FA después de recuperar el acceso

### **Administrador: Enviar Correo Electrónico de Bienvenida**

Para usuarios que no han recibido o han perdido su correo electrónico de bienvenida:

#### **Cuando Está Disponible:**

El icono de avión de papel (✉) solo aparece para usuarios que **nunca han iniciado sesión** (login\_count = 0).

#### **Haga clic en el Icono de Enviar Correo Electrónico de Bienvenida (✉)**

Enviar correo electrónico de bienvenida a:

Usuario: Bob Wilson (<[bob@example.com](mailto:bob@example.com)>)

El correo electrónico incluirá:

- Mensaje de bienvenida
- Instrucciones de inicio de sesión
- Enlace para establecer la contraseña inicial (si aplica)
- Información de contacto de soporte

[Cancelar] [Enviar Correo Electrónico]

#### **Haga clic en "Enviar Correo Electrónico"**

Mensaje de éxito:

#### **Correo Electrónico Enviado a través de Mailjet:**

Utiliza la plantilla: `api.crmCommunicationUserWelcome`

### **Administrador: Editar Usuario**

Haga clic en **Icono de Editar (✎)** para modificar los detalles del usuario:

Nombre

Apellido

Correo Electrónico

Número de Teléfono

Roles  admin  customer\_service\_agent\_1  customer

[Cancelar] [Guardar Cambios]

## Campos Editables:

- Nombre, correo electrónico, teléfono
- **Roles** - Asignar/quitar roles (afecta permisos)
- Estado activo/inactivo

## Administrador: Eliminar Usuario

Haga clic en **Icono de Eliminar (⊖)** para eliminar al usuario:

¿Está seguro de que desea eliminar:

Usuario: John Smith (<[john@example.com](mailto:john@example.com)>)

⚠ ADVERTENCIA: Esta acción no se puede deshacer.

Esto eliminará permanentemente:

- Cuenta de usuario y credenciales
- Configuraciones de 2FA
- Historial de sesiones

Los datos y transacciones del cliente NO se eliminarán.

[Cancelar] [Eliminar Usuario]

Haga clic en "**Eliminar Usuario**" para confirmar.

Mensaje de éxito:

## Mejores Prácticas

### Para Usuarios Finales

#### Seguridad en el Inicio de Sesión:

- Use contraseñas fuertes y únicas
- Habilite "Recordarme" solo en dispositivos personales
- Siempre cierre sesión en computadoras compartidas
- Habilite 2FA para mayor seguridad

#### Gestión de 2FA:

- Guarde los códigos de respaldo inmediatamente después de habilitar 2FA
- Almacene los códigos en un gestor de contraseñas o en un lugar seguro
- Pruebe un código de respaldo para asegurarse de que funcione
- Regenerar códigos de respaldo si usa varios
- Contacte al administrador si pierde tanto el autenticador como los códigos de respaldo

#### Gestión de Contraseñas:

- Use un gestor de contraseñas para generar y almacenar contraseñas
- Nunca comparta contraseñas por correo electrónico o mensajería
- Cambie la contraseña si sospecha de un compromiso
- Use una contraseña única para OmniCRM (no reutilice contraseñas)

### Para Administradores

#### Gestión de Seguridad de Usuarios:

- Verifique la identidad del usuario antes de restablecer 2FA o contraseñas
- Nunca envíe contraseñas temporales por correo electrónico
- Documente todos los restablecimientos de seguridad en las notas del usuario
- Anime al personal a habilitar 2FA
- Monitoree patrones de inicio de sesión inusuales

### **Restablecimientos de Contraseña:**

- Comuníquese las contraseñas temporales solo por teléfono o en persona
- Genere contraseñas temporales fuertes (el sistema lo hace automáticamente)
- Asegúrese de que el usuario cambie la contraseña en el primer inicio de sesión
- No restablezca contraseñas innecesariamente - use el flujo de restablecimiento por correo electrónico cuando sea posible

### **Restablecimientos de 2FA:**

- Trate los restablecimientos de 2FA como acciones de alta seguridad
- Verifique la identidad a través de múltiples canales antes de restablecer
- Documente el motivo del restablecimiento
- Anime al usuario a volver a habilitar 2FA inmediatamente después de recuperar el acceso
- Considere requerir 2FA para todos los usuarios administrativos

### **Gestión de Usuarios:**

- Revise regularmente la lista de usuarios en busca de cuentas inactivas
- Elimine usuarios que hayan dejado la organización
- Asegúrese de que las asignaciones de roles sean apropiadas
- Monitoree a los usuarios que nunca han iniciado sesión
- Audite los permisos de usuario trimestralmente

## **Solución de Problemas**

### **Error "Correo electrónico o contraseña inválidos"**

- **Causa:** Credenciales incorrectas
- **Solución:**
  - Verifique que la dirección de correo electrónico sea correcta
  - Asegúrese de que el bloqueo de mayúsculas esté desactivado
  - Intente restablecer la contraseña si la ha olvidado
  - Contacte al administrador si la cuenta está bloqueada

### **Código 2FA no aceptado**

- **Causa:** Problema de sincronización de tiempo o código incorrecto
- **Solución:**
  - Asegúrese de que la hora del dispositivo sea correcta (Configuración → Fecha y Hora → Automático)
  - Espere a que el código se actualice (los códigos cambian cada 30 segundos)
  - Pruebe el siguiente código que aparece
  - Use un código de respaldo si el autenticador no funciona
  - Contacte al administrador para restablecer 2FA si todo lo demás falla

### **"Recordarme" no funciona**

- **Causa:** Cookies deshabilitadas o borradas
- **Solución:**
  - Habilite las cookies en la configuración del navegador

- No borre las cookies al cerrar el navegador
- Desactive las extensiones de privacidad para el dominio de OmniCRM
- Pruebe con otro navegador

## Correo electrónico de restablecimiento de contraseña no recibido

- **Causa:** Correo electrónico no enviado, filtro de spam o correo electrónico incorrecto
- **Solución:**
  - Verifique la carpeta de spam/basura
  - Verifique que la dirección de correo electrónico sea correcta
  - Espere de 5 a 10 minutos (la entrega del correo electrónico puede retrasarse)
  - Verifique que la integración de Mailjet esté funcionando (administrador)
  - Contacte al administrador para un restablecimiento manual de contraseña

## El enlace de restablecimiento de contraseña ha expirado

- **Causa:** Token expirado (típicamente 1 hora)
- **Solución:**
  - Solicite un nuevo restablecimiento de contraseña
  - Revise el correo electrónico y haga clic en el enlace rápidamente
  - Contacte al administrador si hay problemas repetidos

## No se puede habilitar 2FA (contraseña incorrecta)

- **Causa:** Contraseña actual ingresada incorrectamente
- **Solución:**
  - Verifique la contraseña actual
  - Restablezca la contraseña primero si no está seguro
  - Contacte al administrador para obtener ayuda

## Perdió la aplicación de autenticador y los códigos de respaldo

- **Causa:** Teléfono perdido/restablecido, códigos de respaldo no guardados
- **Solución:**
  - Contacte al administrador de inmediato
  - El administrador verificará la identidad y restablecerá 2FA
  - Inicie sesión solo con la contraseña después del restablecimiento
  - Vuelva a habilitar 2FA y GUARDE los códigos de respaldo esta vez

## Error del Administrador: "No se pudo restablecer 2FA"

- **Causa:** Permisos insuficientes
- **Solución:**
  - Asegúrese de tener el rol de administrador
  - Verifique los permisos de la API
  - Contacte al administrador del sistema

## Error del Administrador: No se generó la contraseña temporal

- **Causa:** Error de API o problema de permisos
- **Solución:**
  - Actualice la página y vuelva a intentarlo
  - Verifique los permisos de administrador
  - Verifique los registros de la API en busca de errores
  - Asegúrese de que la base de datos sea accesible

# Consideraciones de Seguridad

## Gestión de Sesiones:

- Las sesiones expiran después de un período de inactividad
- "Recordarme" extiende la sesión a 30 días
- Las sesiones se almacenan como cookies solo HTTP (no accesibles para JavaScript)
- La bandera segura asegura que las cookies solo se envíen a través de HTTPS

## Seguridad de Contraseñas:

- Las contraseñas se hash utilizando algoritmos estándar de la industria
- Las contraseñas en texto plano nunca se almacenan
- Las contraseñas temporales expiran automáticamente después del primer uso
- Se rastrean los intentos de inicio de sesión fallidos ( posible limitación de tasa)

## Seguridad de 2FA:

- Los secretos de TOTP están encriptados en la base de datos
- Los códigos QR se generan del lado del cliente cuando es posible
- Los códigos de respaldo se hash antes de su almacenamiento
- Cada código de respaldo es de un solo uso

## Acciones del Administrador:

- Los restablecimientos de 2FA se registran en el registro de actividad
- Los restablecimientos de contraseñas crean un rastro de auditoría
- Las acciones del administrador requieren permisos de rol apropiados
- Las direcciones IP se registran para eventos de seguridad

# Documentación Relacionada

- [2fa](#) - Referencia detallada de la API de 2FA (enfocada en la API)
- [rbac](#) - Control de acceso basado en roles y permisos
- [administration\\_configuration](#) - Configuración de correo electrónico de Mailjet para el restablecimiento de contraseña
- [integrations\\_mailjet](#) - Configuración de plantilla de correo electrónico
- [customer\\_care](#) - Portal de Autoservicio para clientes

Verificación de Inicio de Sesión 2FA



# Crear un Cliente

0fT52ZvoZBE



# Clientes, Contactos, Sitios y Servicios

Tenemos un modelo simple de un **Cliente** bajo el cual puede haber múltiples **Contactos** y múltiples **Sitios, Servicios**, etc.

Un **Cliente** es una empresa o individuo que tiene una relación con nosotros, a quien enviamos una factura.

Un **Contacto** es una persona que trabaja con el cliente; para un individuo, probablemente sea lo mismo que el cliente, una sola persona, pero podemos tener miembros de la familia u otros contactos, y cada contacto tiene un tipo, por ejemplo, un contacto de facturación, un contacto técnico, etc., lo que influye en cómo manejamos el contacto.

Un **Sitio** es una ubicación física donde entregamos servicios; podría ser un hogar, oficina u otra ubicación. Esto nos permite tener múltiples sitios para un solo cliente, por ejemplo, un cliente con múltiples oficinas, y saber qué servicios están asociados con qué sitio.

Un **Servicio** es algo por lo que facturamos a un cliente; podría ser un servicio de internet en casa, servicio móvil, o incluso servicios abstractos como arrendar una subred o proporcionar electricidad medida a un rack. Cada servicio está vinculado a un cliente y un sitio, y puede tener múltiples cargos asociados.

Los clientes también tienen un Registro de Actividad `<csa_activity_log>`, que es un registro de todos los cambios realizados, Etiquetas `<administration_tags>`, Atributos `<administration_attributes>` para almacenar metadatos personalizados, Artículos de Inventario `<administration_inventory>` e información financiera como Transacciones `<payments_transaction>`, Facturas `<payments_invoices>` y Métodos de Pago `<basics_payment>`.

Una vez que hemos creado un cliente, podemos agregar un servicio `<csa_add_service>` a ese cliente, que es por lo que les facturamos.

Para información sobre cómo crear un cliente, vea [Crear un Cliente](#).

## Descripción General del Modelo de Datos

OmniCRM utiliza un modelo de datos relacional organizado en torno a clientes y sus servicios. El modelo se desglosa en secciones enfocadas a continuación.

### Relaciones Centrales del Cliente

El cliente es la entidad central, con contactos, sitios y servicios relacionados.

#### Puntos Clave:

- Un cliente puede tener múltiples contactos (facturación, técnico, etc.)
- Un cliente puede tener múltiples sitios (oficinas sucursales, ubicaciones)
- Los servicios se entregan a los sitios
- Los contactos pueden tener acceso al portal a través de cuentas de usuario vinculadas

### Datos de Facturación y Financieros

Las transacciones y facturas rastrean toda la actividad financiera.

## **Puntos Clave:**

- Las transacciones pueden ser independientes o agrupadas en facturas
- Los servicios generan automáticamente transacciones mensuales
- Las tarjetas de Stripe están tokenizadas y almacenadas de forma segura
- Las facturas agrupan múltiples transacciones para la facturación

## **Producto y Aprovisionamiento**

Los productos definen las ofertas de servicios; el aprovisionamiento crea servicios reales.

## **Puntos Clave:**

- Los productos son plantillas; los servicios son instancias activas
- El aprovisionamiento crea o modifica servicios a través de playbooks de Ansible
- Cada trabajo de aprovisionamiento tiene múltiples eventos para rastrear el progreso
- Un aprovisionamiento de producto puede crear múltiples servicios (paquetes)

## **Sistema de Inventory**

Rastrea activos físicos y virtuales asignados a clientes.

## **Puntos Clave:**

- Las plantillas de inventario definen la estructura (campos) para cada tipo de artículo
- Campos flexibles itemtext1-20 se adaptan a diferentes tipos de inventario
- Los artículos pueden ser asignados a clientes y vinculados a servicios
- Almacena configuraciones de equipos y credenciales de gestión

## **Usuario y Seguridad**

Cuentas de usuario con control de acceso basado en roles.

## **Puntos Clave:**

- Los usuarios pueden tener múltiples roles; los roles tienen múltiples permisos
- La autenticación de dos factores (2FA) es opcional por usuario
- Los usuarios del personal son independientes; los usuarios clientes se vinculan a contactos
- Permisos granulares controlan el acceso a recursos

## **Lista de Clientes**

La lista de clientes proporciona una tabla buscable y filtrable de todos los clientes en el sistema.

Vista de la Lista de Clientes

### **Características:**

- **Buscar** - Filtrar clientes por nombre o ID
- **Acciones Masivas** - Seleccionar múltiples clientes para operaciones en lote
- **Paginación** - Navegar a través de grandes listas de clientes
- **Acciones Rápidas** - Ver o eliminar clientes directamente desde la lista

## **Cliente - Detalle**

El objeto cliente en sí no contiene mucha información, solo es un nombre y una referencia a los

contactos y sitios.

Su implementación específica puede incluir campos adicionales o personalizaciones, pero el objeto cliente básico es muy simple.

En la página de resumen también hay un gráfico que muestra el Ingreso Promedio por Usuario (ARPU) para el cliente, que es el ingreso total dividido por el número de servicios, y una comparación de cómo este cliente se compara con el ARPU promedio para todos los clientes en el sistema.

Las opciones de Estado del Cliente están adaptadas según las necesidades específicas de su negocio, pero típicamente incluyen opciones como Activo, Inactivo, Pendiente, etc., con diferentes reglas en cada una que controlan el comportamiento del cliente en el sistema en ese estado.

Eliminar un cliente solo se puede hacer si el cliente no tiene servicios activos, facturas impagadas o transacciones no facturadas. Si el cliente tiene alguno de estos, deberá cerrar los servicios activos y asegurarse de que los pagos se realicen antes de poder eliminar al cliente, lo que a su vez archivará al cliente y todos los datos asociados, que pueden ser restaurados más tarde si es necesario.

Cliente - Detalle

## Sitio - Detalle

Los sitios son ubicaciones físicas donde se entregan servicios y pueden tener múltiples servicios asociados con ellos.

Se utilizan predominantemente para clientes comerciales, donde un solo cliente podría tener múltiples sitios, por ejemplo, un cliente con múltiples oficinas.

Tener múltiples sitios nos permite rastrear qué servicios están asociados con qué sitio, por ejemplo, si un cliente solicita un nuevo servicio para una nueva oficina, necesitamos asegurarnos de que entregamos el servicio correcto a la ubicación correcta. Esto nos permite rastrear servicios por sitio y facturarlos por separado si es necesario.

## Integración de Google Maps y Geocodificación

Cada sitio está integrado con **Google Maps** para asegurar datos de dirección precisos y geolocalización. La interfaz de usuario geocodifica automáticamente las direcciones y genera datos de ubicación.

### Cómo Funciona la Geocodificación de Direcciones:

Al agregar o editar un sitio, la interfaz proporciona dos métodos para establecer la ubicación:

#### 1. Búsqueda de Dirección (Recomendado)

- Utilice la barra de búsqueda en la parte superior del formulario
- Escriba una dirección y Google Maps Autocomplete sugiere coincidencias
- Seleccione la dirección correcta del menú desplegable
- El sistema completa automáticamente:
  - **Nombre del Sitio** - Nombre del lugar de Google Maps
  - **Línea de Dirección 1** - Número y nombre de la calle
  - **Línea de Dirección 2** - Subpremisa (número de unidad/suite)
  - **Ciudad** - Localidad
  - **Estado/Región** - Área administrativa
  - **Código Postal** - Código postal
  - **País** - Nombre del país

- **Latitud y Longitud** - Coordenadas precisas
- **Código Plus** - Código de Ubicación Abierto de 11 caracteres (por ejemplo, 8C3MFJV8+2F)
- **ID de Lugar de Google Maps** - Identificador único del lugar

## 2. Marcador de Mapa Arrastrable (Manual)

- Arrastre el pin en el mapa a la ubicación exacta
- El sistema realiza **geocodificación inversa** para obtener la dirección de las coordenadas
- Los mismos campos se completan automáticamente según la ubicación del pin
- Útil para áreas rurales o cuando la dirección es imprecisa

## Generación de Código Plus:

Los Códigos Plus se generan automáticamente a partir de la latitud/longitud utilizando la biblioteca de Código de Ubicación Abierto. Un Código Plus es un código corto (11 caracteres) que representa una ubicación precisa en cualquier parte del mundo.

### Ejemplo:

Campos autocompletados: ✓ Nombre del Sitio: "123 Main Street" ✓ Línea de Dirección 1: "123 Main Street" ✓ Ciudad: "Londres" ✓ Estado: "Gran Londres" ✓ País: "Reino Unido" ✓ Código Postal: "SW1A 1AA" ✓ Latitud: 51.5074 ✓ Longitud: -0.1278 ✓ Código Plus: "9C3XGPHC+3Q" ✓ ID de Lugar de Google: "ChIJdd4hrwug2EcRmSrV3Vo6lll"

## Requisitos de Validación:

Antes de guardar un sitio, el sistema valida:

- **Latitud y Longitud** deben estar configuradas (ya sea mediante búsqueda o arrastrando el marcador)
- **País** debe estar poblado (se establece por defecto en REACT\_APP\_DEFAULT\_COUNTRY si no está configurado)
- **Código Plus** debe tener 12 caracteres (11 + 1 para relleno)

Si la validación falla, verá un error:

## Retroalimentación Visual:

La interfaz muestra retroalimentación en tiempo real:

o

## Cómo se Utilizan los Datos de Ubicación

Los datos de ubicación geocodificados (latitud, longitud, Código Plus) se utilizan en todo OmniCRM para:

### 1. Entrega e Instalación de Servicios

- **Técnicos de Campo** - Acceso al Código Plus para navegar a la ubicación exacta del sitio
- **Programación de Instalaciones** - Asignar técnicos según la proximidad geográfica
- **Despliegue de Equipos** - Asegurar que el equipo correcto se entregue en el sitio correcto

### 2. Notificaciones de Interrupciones

- **Alertas Geocercadas** - Si hay una interrupción de red en un área específica, consultar sitios por radio de lat/long
- **Comunicaciones Dirigidas** - Enviar notificaciones de interrupción solo a los clientes

- afectados a través de Mailjet <integrations\_mailjet>
- **Páginas de Estado** - Mostrar un mapa de interrupciones con los sitios afectados

#### Ejemplo:

Consulta: SELECT \* FROM Customer\_Site  
WHERE distance(latitude, longitude, 51.5074, -0.1278) < 5

Resultado: 47 sitios afectados Acción: Enviar notificación de interrupción a 47 clientes

### 3. Informes y Análisis

- **Ingresos Geográficos** - Ingresos por ciudad, estado, región
- **Mapas de Densidad de Servicios** - Mapa de calor de ubicaciones de servicios
- **Planificación de Expansión** - Identificar áreas desatendidas

### 4. Clientes Comerciales Multisitio

- **Gestión de Sitios** - Rastrear qué servicios están en qué ubicaciones
- **Facturación Separada** - Facturar por sitio si es necesario
- **Asignación de Servicios** - Vincular servicios a sitios específicos durante el aprovisionamiento

### Sitios Rurales y Remotos

Para clientes en áreas rurales donde las direcciones pueden no existir o ser inexactas:

1. **Usar Arrastre de Mapa**
  - Acerca a la zona aproximada
  - Arrastrar el pin a la propiedad/edificio exacto
  - El sistema genera un Código Plus para esa ubicación precisa
2. **Compartir Código Plus**
  - Compartir el Código Plus con el cliente (por ejemplo, 8C3MFJV8+2F)
  - El cliente puede ingresar esto en Google Maps para ver la ubicación exacta
  - Los técnicos de campo utilizan el Código Plus para la navegación
3. **Notas de Dirección**
  - Utilizar el campo "Nota de Dirección" para direcciones adicionales
  - Ejemplo: "Gire a la izquierda en el granero rojo, 500 m después de la rejilla de ganado"
  - Notas visibles para los equipos de instalación

Sitio - Agregar con Google Maps

Consejo

Puede arrastrar el pin en el mapa a la ubicación correcta si la dirección no es precisa. El sistema geocodificará la ubicación y completará automáticamente todos los campos de dirección.

Sitio - Vista Detalle

### Contacto - Detalle

Los Contactos son personas asociadas con el cliente. Pueden ser contactos de facturación, contactos técnicos u otros tipos, y cada contacto tiene un tipo que influye en cómo manejamos el contacto.

Lista de Contactos del Cliente

Podemos tener múltiples contactos para un solo cliente, por ejemplo, un cliente con múltiples contactos de facturación, o un cliente con múltiples contactos técnicos.

Un buen ejemplo sería una empresa con un proveedor de servicios gestionados, que maneja el lado técnico de las cosas, y un contacto de facturación separado que maneja el lado financiero, o una familia donde cada miembro tiene su propio contacto pero no todos están autorizados para hacer cambios.

Del mismo modo, es posible que solo queramos enviar notificaciones de interrupción al contacto técnico, o solo enviar facturas al contacto de facturación, y el tipo de contacto nos permite controlar esto.

La lógica exacta de cómo se utilizan los tipos de contacto depende de su negocio, pero la idea básica es que cada contacto tiene un tipo que influye en cómo los manejamos, y que cada persona asociada con el cliente es un contacto.

Los contactos se sincronizan con la integración de Mailjet <integrations\_mailjet>, lo que nos permite enviar campañas de correo electrónico dirigidas según el tipo de contacto, ubicación, gasto mensual o servicios adquiridos, y gestionar todas las plantillas de correo electrónico utilizadas para comunicaciones transaccionales.

Contacto - Resumen Contacto - Formulario de Edición Contacto - Detalle



# Navegación

OmniCRM está diseñado desde cero para ser receptivo e intuitivo.

Esta guía te ayudará a navegar por el sistema y encontrar la información que necesitas.

Una vez que hayas iniciado sesión, la barra de navegación en el lado izquierdo de la pantalla te mostrará las secciones principales del sistema y las subsecciones dentro de cada una.

## Navegación - Barra de Navegación

En la parte superior derecha de la pantalla, verás el menú de usuario, que te permite cerrar sesión, cambiar tu contraseña o acceder a la configuración de tu usuario.

## Navegación - Menú de Usuario

Tienes enlaces a todas las aplicaciones web más utilizadas de tu organización en la barra de WebApps (esto puede adaptarse a las necesidades de tu organización).

## Navegación - WebApps

Cualquier alerta que tengas pendiente es visible en la parte superior derecha de la pantalla, y puedes hacer clic en la alerta para ver más información.

## Navegación - Alertas

Podemos cambiar el idioma del sistema haciendo clic en el menú desplegable de idioma en la parte superior derecha de la pantalla.

## Navegación - Idioma

Si eres un príncipe de las tinieblas, puedes cambiar a modo oscuro haciendo clic en el ícono de la luna en la parte superior derecha de la pantalla.

## Navegación - Modo Oscuro



# Resumen de la pestaña de facturación

La pestaña de facturación proporciona una vista unificada de toda la información financiera de un cliente, combinando métodos de pago, transacciones y facturas en una sola interfaz para una gestión de facturación eficiente.

Documentación relacionada: Payment Methods <payment\_methods>, Transactions <payments\_transaction>, Invoices <payments\_invoices>, Payment Processing <payments\_process>.

## Acceso a la pestaña de facturación

### Vista por cliente:

1. Navegar a **Customers** → **[Select Customer]**
2. Hacer clic en la pestaña **Billing**
3. Ver las tres secciones: Métodos de Pago, Transacciones y Facturas

### Vistas a nivel de sistema:

Los datos de facturación a nivel de sistema se pueden acceder por separado:

- **Billing** → **Transactions** - Todas las transacciones de todos los clientes
- **Billing** → **Invoices** - Todas las facturas de todos los clientes

### Portal de Auto-Cuidado:

Los clientes que acceden al Self-Care Portal <self\_care\_portal> ven la misma estructura de pestaña de facturación:

- Ver y gestionar sus métodos de pago
- Ver el historial de transacciones
- Ver y pagar facturas en línea

## Estructura de la pestaña de facturación

La pestaña de facturación está organizada en tres secciones principales, mostradas como tarjetas:

### Sección 1: Métodos de Pago

**Propósito:** Gestionar cómo el cliente paga por los servicios

#### Características clave:

- Ver todas las tarjetas de crédito guardadas
- Establecer método de pago predeterminado
- Agregar nuevos métodos de pago (a través de Stripe)
- Eliminar tarjetas expiradas o no utilizadas

Facturación del Cliente - Métodos de Pago  
Detalle de la Tarjeta de Métodos de Pago

**Documentación:** basics\_payment

## Sección 2: Transacciones

**Propósito:** Rastrear todos los cargos y créditos del cliente

**Características clave:**

- Ver estadísticas de transacciones (Total, No Facturadas, Este Mes, Último Mes)
- Listar todas las transacciones con filtrado por estado de anulación/factura
- Agregar transacciones manuales (cargos o créditos)
- Anular transacciones incorrectas
- Ver qué transacciones están facturadas vs no facturadas

**Documentación:** payments\_transaction

## Sección 3: Facturas

**Propósito:** Agrupar transacciones en facturas para que el cliente pague

**Características clave:**

- Ver estadísticas de facturas (Total, No Pagadas, Este Mes, Último Mes)
- Listar todas las facturas con filtrado por estado pagado/anulado
- Generar nuevas facturas a partir de transacciones no facturadas
- Descargar PDFs de facturas
- Enviar facturas por correo electrónico a los clientes
- Pagar facturas en línea (métodos de pago de Stripe o manuales)

Facturación del Cliente - Transacciones y Facturas

- Anular o reembolsar facturas

**Documentación:** payments\_invoices

## Flujo de datos entre secciones

Entender cómo fluye la información entre las tres secciones es crucial para una gestión de facturación efectiva.

### Diagrama de flujo

#### Relación Transacción → Factura

##### 1. Creación de Transacción:

Cuando se provisiona un servicio o se agrega un cargo manual:

- Transacción creada en la sección **Transactions**
- Estado de la transacción: **No Facturada**
- Campo `invoice_id` de la transacción es `null`

**Ejemplo:**

##### 2. Generación de Factura:

Cuando el personal genera una factura:

- Factura creada en la sección **Invoices**
- Todas las transacciones no facturadas dentro del rango de fechas agrupadas en la factura
- Campo **invoice\_id** de la transacción poblado
- Estado de la transacción cambia a: **Facturada**

**Ejemplo:**

### 3. Actualización de estadísticas de transacciones:

- Total de **Transacciones No Facturadas** disminuye
- Estadística de **Total de Facturas** aumenta
- Total de **Facturas No Pagadas** aumenta

## Relación Factura → Pago

### 1. Pago de Factura:

Cuando el cliente paga la factura:

- Pago procesado utilizando el **Método de Pago** guardado (tarjeta de Stripe)
- O se selecciona un método de pago manual (efectivo, POS, transferencia bancaria)
- Estado de la factura cambia a: **Pagada**

### 2. Transacción de Pago Creada:

Para pagos manuales (no Stripe):

- Transacción negativa creada automáticamente
- Título de la transacción: "Pago por Factura #1234"
- Monto de la transacción: -\$45.00 (negativo, acredita al cliente)
- Campo **invoice\_id** de la transacción: Enlaza a la factura pagada

**Ejemplo:**

### 3. Actualización de estadísticas:

- Total de **Facturas No Pagadas** disminuye
- **Total de Facturas Este Mes** sin cambios (la factura ya existía)

## Relación Método de Pago → Factura

### Flujo de Pago de Stripe:

1. El cliente agrega una tarjeta de crédito en **Payment Methods**
2. Tarjeta tokenizada a través de Stripe, almacenada de forma segura
3. Al pagar la factura, el cliente selecciona la tarjeta guardada
4. Stripe carga la tarjeta
5. Factura marcada como pagada
6. Campo **payment\_reference** poblado con el ID de intención de pago de Stripe

### Flujo de Pago Manual:

1. El cliente paga en efectivo/POS/transferencia bancaria (no se necesita método de pago)
2. El personal selecciona el método de pago en el modal de Pagar Factura
3. El personal ingresa el número de referencia (opcional)
4. Transacción negativa creada por el monto del pago
5. Factura marcada como pagada

# Flujos de trabajo completos de facturación

Estos flujos de trabajo demuestran cómo las tres secciones trabajan juntas para realizar tareas comunes.

## Flujo de trabajo 1: Configuración de nuevo cliente y primera factura

**Objetivo:** Configurar la facturación para un nuevo cliente y recoger el primer pago

### 1. Agregar Método de Pago:

- Navegar al cliente → pestaña **Billing**
- Sección **Payment Methods** → Hacer clic en "**Add Payment Method**"
- El cliente agrega una tarjeta de crédito a través de Stripe
- Tarjeta guardada como método de pago predeterminado

### 2. Verificar Transacciones:

- La sección **Transactions** muestra transacciones no facturadas:
  - Cargo de configuración del servicio: \$50.00
  - Servicio del primer mes: \$45.00
  - Total No Facturado: \$95.00

### 3. Generar Factura:

- Sección **Invoices** → Hacer clic en "**Generate Proforma Invoice**"
- Establecer rango de fechas para incluir la configuración y el primer mes
- Hacer clic en "**Generate Invoice**"
- Factura #INV-2025-001234 creada por \$95.00

### 4. Actualización de Transacciones:

- Ambas transacciones ahora muestran: Factura #INV-2025-001234
- Total de **Transacciones No Facturadas** ahora \$0.00

### 5. Enviar Factura por Correo:

- Hacer clic en el ícono de correo junto a la factura
- El cliente recibe un correo electrónico de la factura con PDF y enlace de pago

### 6. El Cliente Paga en Línea:

- El cliente hace clic en el enlace de pago en el correo
- Redirigido al portal de Auto-Cuidado
- Hacer clic en el botón "**Pay Invoice**"
- Seleccionar método de pago predeterminado
- Hacer clic en "**Pay Invoice**"
- Stripe carga la tarjeta

### 7. Actualización de la Factura:

- Estado de la factura cambia a "**Paid**"
- Total de **Facturas No Pagadas** disminuye en \$95.00

**Resultado:** Cliente completamente configurado con método de pago, primera factura pagada.

## Flujo de trabajo 2: Facturación mensual recurrente

**Objetivo:** Facturar a todos los clientes por el servicio mensual al final del mes

### 1. Servicios de Carga Automática:

- Llega el final del mes (31 de enero)
- El sistema de facturación crea automáticamente transacciones para todos los servicios recurrentes
- La sección **Transactions** muestra nuevas transacciones no facturadas

### 2. Revisar Transacciones No Facturadas:

- Navegar a la sección **Transactions**
- Filtrar: **Invoice Status: Not Invoiced**
- Revisar la lista de todas las transacciones listas para facturación
- Verificar que los montos y descripciones sean correctos

### 3. Generar Facturas:

- Navegar a **Billing → Invoices** (a nivel de sistema)
- Para cada cliente (o usar proceso por lotes):
  - Hacer clic en "**Generate Proforma Invoice**"
  - Seleccionar cliente
  - Fecha de Inicio: 2025-01-01
  - Fecha de Fin: 2025-01-31
  - Fecha de Vencimiento: 2025-02-15
  - Hacer clic en "**Generate Invoice**"

### 4. Actualización de Transacciones:

- Todas las transacciones ahora vinculadas a facturas
- Totales de **Transacciones No Facturadas** restablecidos a \$0.00

### 5. Enviar Todas las Facturas por Correo:

- Para cada factura, hacer clic en el ícono de correo
- Todos los clientes reciben facturas mensuales

### 6. Los Clientes Pagan:

- Los clientes con métodos de pago guardados pagan en línea a través de Auto-Cuidado
- El personal procesa pagos en efectivo/POS para los clientes que pagan en persona
- Total de **Facturas No Pagadas** disminuye a medida que se reciben los pagos

**Resultado:** Todos los clientes facturados por enero, facturas enviadas, pagos procesados.

## Flujo de trabajo 3: Manejo de crédito por problema de servicio

**Objetivo:** Acreditar al cliente por la interrupción del servicio, aplicar a factura no pagada

### 1. El Cliente Informa el Problema:

- El servicio estuvo inactivo durante 2 días
- El cliente merece un crédito de \$10

### 2. Agregar Transacción de Crédito:

- Navegar al cliente → pestaña **Billing** → sección **Transactions**
- Hacer clic en "**+ Add Transaction**"
- Tipo de Transacción: **Credit**
- Tipo de Crédito: **Cash Payment** (o tipo apropiado)
- Título: "Crédito por Interrupción de Servicio"
- Descripción: "Compensación por interrupción de 2 días 8-9 de enero"
- Costo Minorista: 10.00
- Hacer clic en "**Add Transaction**"

### 3. Transacción Creada:

- La transacción aparece en la lista con monto: **-\$10.00**
- Estado de la transacción: **No Facturada**
- Total de **Transacciones No Facturadas** ahora incluye -\$10.00

### 4. Aplicar a la Factura:

- Si el cliente ya tiene una factura no pagada:
  - La factura permanece no pagada con el monto original
  - El crédito se aplicará a la próxima generación de facturas
- Si se genera una nueva factura:
  - Sección **Invoices** → Hacer clic en "**Generate Proforma Invoice**"
  - Incluir rango de fechas con la transacción de crédito
  - Factura generada con crédito aplicado:

### 5. El Cliente Paga:

- El cliente paga el monto reducido: \$35.00
- Factura marcada como pagada

**Resultado:** Cliente acreditado por la interrupción, crédito aplicado a la próxima factura, se recolecta un pago menor.

## Flujo de trabajo 4: Método de Pago Expirado - Actualizar y Reintentar

**Objetivo:** La tarjeta del cliente expiró, causando un fallo en el pago - actualizar la tarjeta y reintentar el pago

### 1. Notificación de Fallo de Pago:

- El cliente intenta pagar la factura
- Stripe devuelve un error: "Tarjeta expirada"
- El pago falla, la factura permanece no pagada

### 2. Actualizar Método de Pago:

- El cliente navega a la pestaña **Billing**
- Sección **Payment Methods** → Hacer clic en "**Add Payment Method**"
- Ingresar detalles de la nueva tarjeta (fecha de expiración actualizada)
- Nueva tarjeta guardada

### 3. Establecer como Predeterminado:

- El cliente hace clic en "**Set as Default**" en la nueva tarjeta
- La tarjeta antigua se elimina automáticamente (si se desea)

### 4. Reintentar Pago:

- Navegar a la sección **Invoices**
- Localizar factura no pagada
- Hacer clic en el ícono "**Pay**"
- Se abre el modal de pago con la nueva tarjeta predeterminada preseleccionada
- Hacer clic en "**Pay Invoice**"
- Stripe carga la nueva tarjeta con éxito

### 5. Actualización de la Factura:

- Estado de la factura cambia a "**Paid**"
- Campo `payment_reference` poblado con el nuevo ID de intención de pago de Stripe

**Resultado:** Cliente actualizó el método de pago, factura pagada con éxito con la nueva tarjeta.

## Flujo de trabajo 5: Anular Factura Incorrecta y Re-Facturación

**Objetivo:** El personal generó una factura con transacciones incorrectas - anular y regenerar correctamente

### 1. Error Descubierto:

- Factura #INV-2025-001234 generada con rango de fechas incorrecto
- Incluyó transacciones del mes equivocado
- La factura está no pagada

### 2. Anular Factura:

- Navegar a la pestaña **Billing** → sección **Invoices**
- Localizar la factura incorrecta
- Hacer clic en el ícono de eliminar (⊖)
- Confirmar anulación
- Factura anulada

### 3. Transacciones Liberadas:

- Navegar a la sección **Transactions**
- Todas las transacciones de la factura anulada ahora muestran: **No Facturadas**
- Total de **Transacciones No Facturadas** aumenta
- Transacciones disponibles para nueva factura

### 4. Generar Factura Correcta:

- Sección **Invoices** → Hacer clic en "**Generate Proforma Invoice**"
- Establecer rango de fechas correcto
- Aplicar filtro si es necesario (por ejemplo, "Móvil" para factura solo móvil)
- Hacer clic en "**Generate Invoice**"
- Nueva factura creada con transacciones correctas

### 5. Verificar y Enviar por Correo:

- Revisar detalles de la nueva factura
- Verificar que se incluyan las transacciones correctas
- Hacer clic en el ícono de correo para enviar al cliente

**Resultado:** Factura incorrecta anulada, transacciones re-facturadas correctamente, el cliente recibe la factura corregida.

## Flujo de trabajo 6: Procesar Pago en Efectivo para Múltiples Facturas

**Objetivo:** El cliente paga múltiples facturas no pagadas con un solo pago en efectivo

### 1. El Cliente Llega con Efectivo:

- El cliente trae \$300 en efectivo para pagar facturas pendientes
- Navegar al cliente → pestaña **Billing**

### 2. Revisar Facturas No Pagadas:

- Sección **Invoices** → Filtrar: **Paid: Not yet Paid**
- Ver facturas no pagadas:

### 3. Pagar Primera Factura:

- Hacer clic en el ícono de pago en la factura #1234
- Se abre el modal de pago
- Seleccionar método de pago "**Cash**"
- Ingresar referencia: "Efectivo pagado 2025-02-10 - Recibo #001"
- Hacer clic en "**Pay Invoice**"
- Factura #1234 marcada como "**Paid**"

### 4. Pagar Facturas Restantes:

- Repetir el proceso para la factura #1235:
  - Referencia: "Efectivo pagado 2025-02-10 - Recibo #001"
- Repetir para la factura #1236:
  - Referencia: "Efectivo pagado 2025-02-10 - Recibo #001"

### 5. Verificar Transacciones:

- Navegar a la sección **Transactions**
- Se crean tres nuevas transacciones de pago:
- Todas vinculadas a las respectivas facturas

### 6. Actualizar Estadísticas:

- Sección **Invoices** → Total de **Facturas No Pagadas** disminuido en \$300.00
- Todas las facturas ahora muestran estado "**Paid**"

**Resultado:** El cliente pagó todas las facturas pendientes en efectivo, las transacciones de pago registradas con referencia de recibo.

## Mejores Prácticas

### Para Usuarios del Personal

#### Gestión de Transacciones:

- Agregar transacciones manuales de inmediato (no retrasar)
- Usar títulos y descripciones descriptivas para mayor claridad
- Vincular transacciones a servicios y sitios cuando sea aplicable
- Anular transacciones incorrectas antes de que sean facturadas

#### Generación de Facturas:

- Generar facturas a intervalos consistentes (por ejemplo, mensualmente el 1 del mes)
- Usar rangos de fechas cuidadosamente para evitar superposiciones o vacíos
- Usar filtros para crear facturas específicas de servicios cuando sea necesario
- Enviar facturas por correo electrónico inmediatamente después de la generación

- Revisar PDFs de facturas antes de enviarlas a los clientes

### **Procesamiento de Pagos:**

- Verificar que el método de pago sea válido antes de intentar el cargo
- Siempre ingresar números de referencia para pagos manuales (efectivo, POS, transferencia bancaria)
- Marcar facturas como pagadas inmediatamente después de recibir el pago
- Reembolsar solo a través de Stripe (crear crédito manual para otros métodos de pago)

### **Higiene de Datos:**

- Revisar regularmente las transacciones no facturadas
- Investigar transacciones anuladas para entender errores de facturación
- Monitorear facturas no pagadas y hacer seguimiento con los clientes
- Mantener métodos de pago actualizados (eliminar tarjetas expiradas)

## **Para Clientes (Portal de Auto-Cuidado)**

### **Métodos de Pago:**

- Mantener al menos un método de pago válido en archivo
- Actualizar métodos de pago antes de que las tarjetas expiren
- Establecer su tarjeta preferida como predeterminada

### **Pagos de Facturas:**

- Pagar facturas antes de la fecha de vencimiento para evitar cargos por demora
- Revisar detalles de la factura y transacciones antes de pagar
- Descargar PDFs de facturas para sus registros
- Contactar al soporte inmediatamente si la factura parece incorrecta

### **Revisión de Transacciones:**

- Revisar regularmente el historial de transacciones
- Informar cualquier cargo inesperado de inmediato
- Entender qué transacciones están facturadas vs no facturadas

## **Para Administradores**

### **Configuración del Sistema:**

- Configurar plantillas de correo electrónico de Mailjet para la entrega profesional de facturas
- Personalizar plantillas de PDF de facturas para que coincidan con la marca
- Configurar la integración de Stripe para un procesamiento de pagos seguro
- Configurar términos de pago y fechas de vencimiento

### **Monitoreo e Informes:**

- Usar widgets de estadísticas para monitorear la salud de la facturación
- Rastrear el total de **Transacciones No Facturadas** - debería disminuir después del ciclo de facturación
- Monitorear el total de **Facturas No Pagadas** - hacer seguimiento de pagos atrasados
- Revisar estadísticas de **Este Mes** vs **Último Mes** para tendencias

### **Automatización:**

- Automatizar cargos de servicios recurrentes a través de la configuración del producto
- Configurar generación automática de facturas para facturación recurrente (si está disponible)
- Configurar recordatorios por correo electrónico para facturas vencidas

## Problemas Comunes y Soluciones

### Problema: El Cliente No Puede Pagar la Factura

#### Síntomas:

- El cliente hace clic en el botón de pago pero no pasa nada
- Mensaje de error: "No se encontraron métodos de pago"

#### Diagnóstico:

1. Navegar al cliente → pestaña **Billing** → sección **Payment Methods**
2. Verificar si el cliente tiene métodos de pago guardados
3. Verificar si las tarjetas guardadas están expiradas

#### Solución:

- El cliente debe agregar un método de pago válido antes de pagar las facturas
- Guiar al cliente a la página de **Payment Methods** para agregar una tarjeta de crédito
- Verificar que la tarjeta sea aceptada (Visa, Mastercard, Amex, etc.)
- Reintentar el pago después de agregar la tarjeta

### Problema: Factura Generada con Transacciones Incorrectas

#### Síntomas:

- La factura incluye transacciones del período equivocado
- La factura falta transacciones esperadas
- Total de la factura incorrecto

#### Diagnóstico:

1. Abrir la factura en la sección **Invoices**
2. Revisar las transacciones incluidas en la factura
3. Verificar las fechas de las transacciones vs el rango de fechas de la factura
4. Verificar si se aplicó un filtro durante la generación

#### Solución:

- **Si la factura no está pagada:** Anular la factura, verificar que las transacciones estén no facturadas, regenerar con el rango de fechas correcto
- **Si la factura está pagada:** No se puede anular - crear una transacción de crédito por el monto incorrecto, generar factura corregida
- **Prevención:** Siempre revisar la sección **Transactions** antes de generar la factura para verificar que se incluirán las transacciones correctas

### Problema: Total de Transacciones No Facturadas No Disminuye

#### Síntomas:

- El widget de **Transacciones No Facturadas** muestra un monto alto
- La lista de transacciones muestra muchas transacciones no facturadas

- Las facturas mensuales ya se generaron

#### Diagnóstico:

1. Filtrar transacciones por **Invoice Status: Not Invoiced**
2. Revisar la lista de transacciones no facturadas
3. Verificar las fechas de las transacciones - pueden ser cargos recientes después de la última generación de facturas
4. Verificar si algunas transacciones están anuladas (no deberían contar para el total no facturado)

#### Solución:

- **Comportamiento esperado:** Las transacciones creadas después de la última generación de facturas permanecen no facturadas hasta el próximo ciclo de facturación
- **Si las transacciones antiguas están no facturadas:** Generar factura para esas transacciones con el rango de fechas apropiado
- **Si se contaron transacciones anuladas:** El sistema debería excluir automáticamente las transacciones anuladas - informar un error si no es así

### Problema: Método de Pago Sigue Siendo Rechazado

#### Síntomas:

- El pago de Stripe falla con "Tarjeta rechazada"
- El cliente informa que la tarjeta debería funcionar
- Múltiples intentos de reintento fallan

#### Diagnóstico:

1. Verificar el panel de Stripe para la razón del rechazo
2. Razones comunes de rechazo:
  - Fondos insuficientes
  - Tarjeta expirada
  - Tarjeta reportada como perdida/robada
  - Prevención de fraude bancario
  - Tarjeta internacional bloqueada

#### Solución:

- Pedir al cliente que contacte a su banco para autorizar el pago
- Probar un método de pago diferente (tarjeta diferente)
- Para el personal: Procesar como pago manual (efectivo, POS) y marcar la factura como pagada
- Verificar que la dirección de facturación coincide con la dirección de facturación de la tarjeta

## Atajos de Teclado y Consejos

#### Navegación:

- Acceder rápidamente a la pestaña de facturación: Página del cliente → tecla **B** (si los atajos de teclado están habilitados)

#### Filtrado:

- **Transacciones:** Hacer clic en el filtro "**Not Invoiced**" antes de generar facturas para ver

- exactamente qué se incluirá
- **Facturas:** Hacer clic en el filtro "**Not yet Paid**" para ver todas las facturas pendientes que requieren seguimiento

#### **Operaciones Masivas:**

- Seleccionar múltiples facturas (casilla de verificación) para eliminar/anular múltiples a la vez (solo personal)
- Usar la barra de búsqueda para encontrar rápidamente transacciones o facturas específicas por ID

#### **Acciones Rápidas:**

- Hacer clic en el título de la transacción/factura para ver todos los detalles en el modal
- Menú de acciones de clic derecho ( : ) para acceso rápido a opciones de descarga/enviar por correo/pagar

## **Documentación Relacionada**

- [basics\\_payment](#) - Métodos de pago e integración de Stripe
- [payments\\_transaction](#) - Transacciones en detalle
- [payments\\_invoices](#) - Facturas en detalle
- [integrations\\_mailjet](#) - Entrega de facturas por correo electrónico
- [csa\\_activity\\_log](#) - Visualización del historial de actividad de facturación



# Ansible Playbooks: Guía Detallada

Los productos de OmniCRM se provisionan utilizando **Ansible**, lo que permite la gestión automatizada de servicios basada en los requisitos específicos de cada producto y su inventario asociado.

## Cómo Funcionan Juntos los Playbooks y los Productos

**Concepto Crítico:** Los playbooks son los que realmente crean servicios en OmniCRM. Cuando asignas un playbook a un producto, estás definiendo **lo que sucede** cuando ese producto es provisionado, pero eso puede significar cosas diferentes para diferentes productos.

### Los Productos Activan los Playbooks

Cuando un producto es provisionado en OmniCRM:

1. La definición del producto especifica qué playbook ejecutar (a través del campo `provisioning_play`)
2. El producto pasa variables al playbook (a través de `provisioning_json_vars` y selecciones de inventario)
3. El playbook se ejecuta y hace lo que está programado para hacer
4. El playbook determina qué se crea (si es que se crea algo)

### Lo Que Pueden Hacer los Playbooks

Un solo playbook de aprovisionamiento puede:

#### Crear Múltiples Servicios

Un playbook de producto agrupado podría crear:

- Un registro de servicio de internet principal
- Un registro de servicio de complemento IPTV
- Un registro de servicio VoIP
- Todo con una acción de provisión de producto

#### Crear Ningún Servicio

Algunos playbooks no crean registros de servicio en absoluto:

- Un playbook que solo configura equipos CPE
- Un playbook que envía configuraciones a equipos de red
- Un playbook que actualiza sistemas externos

## Crear Un Servicio

El patrón más común:

- Crear un único registro de servicio para el cliente
- Vincular el inventario a ese servicio
- Configurar la facturación para ese servicio

## Modificar Servicios Existentes

Playbooks de recarga y complemento:

- No crean nuevos servicios
- Actualizan registros de servicio existentes (agregan datos, extienden la expiración, etc.)
- Agregan saldos a cuentas de facturación existentes

## Realizar Acciones Sin Registros de Servicio

Algunos playbooks son puramente operacionales:

- Restablecer saldos de cuentas
- Intercambiar elementos de inventario entre clientes
- Generar informes o configuraciones

## Ejemplo: Comportamientos Diferentes de Playbooks

```
# Producto 1: Servicio de SIM Móvil (crea 1 servicio)
# provisioning_play: play_simple_service
- Crea registro de servicio en CRM
- Crea cuenta de facturación en OCS
- Asigna inventario de tarjeta SIM y número de teléfono
- Envía correo electrónico de bienvenida

# Producto 2: Paquete de Internet (crea 3 servicios)
# provisioning_play: play_bundle_internet_tv_voice
- Crea registro de servicio de internet
- Crea registro de servicio IPTV
- Crea registro de servicio VoIP
- Vincula todos al mismo cliente
- Cuenta de facturación única para el paquete

# Producto 3: Recarga de Datos (crea 0 servicios)
# provisioning_play: play_topup_no_charge
- Encuentra servicio existente por service_id
- Agrega saldo de datos a cuenta OCS existente
- Actualiza fecha de expiración del servicio
- NO se crea nuevo servicio

# Producto 4: Configuración de CPE (crea 0 servicios)
# provisioning_play: play_prov_cpe_mikrotik
```

- Genera configuración de router
- Actualiza registro de inventario con configuración
- Envía configuración al equipo de soporte
- NO se crea servicio (solo configuración de equipo)

El punto clave: **El playbook define el comportamiento, el producto es solo un disparador.**

## Plays vs Tareas

Entender la distinción entre Plays y Tareas es fundamental para trabajar con los playbooks de OmniCRM.

### Play (Playbook)

Un flujo de trabajo completo de aprovisionamiento que orquesta múltiples tareas para lograr un objetivo comercial. Los plays son los playbooks de nivel superior almacenados en `OmniCRM-API/Provisioners/plays/` y se hacen referencia en las definiciones de productos.

Ejemplos:

- `play_simple_service.yaml` - Aprovisionar un servicio básico
- `play_topup_no_charge.yaml` - Aplicar una recarga gratuita a un servicio
- `play_prov_cpe_mikrotik.yaml` - Configurar equipos de cliente

### Tarea (Componente Reutilizable)

Un conjunto de operaciones autónomas y reutilizables que pueden ser incluidas por múltiples plays. Las tareas tienen el prefijo `task_` y viven en el mismo directorio.

Ejemplos:

- `task_welcome_email.yaml` - Enviar un correo electrónico de bienvenida a un cliente
- `task_activate_olt.yaml` - Activar equipos OLT
- `task_notify_ocs.yaml` - Enviar notificaciones al sistema de facturación

La relación entre ellos:

```
# play_simple_service.yaml (Un Play)
- name: Play de Aprovisionamiento Simple
  hosts: localhost
  tasks:
    - name: Bloque principal de aprovisionamiento
      block:
        - name: Crear servicio
          uri: ...
```

```

- name: Configurar facturación
  uri: ...

# Incluir tarea reutilizable
- include_tasks: task_welcome_email.yaml

# Incluir tareas post-aprovisionamiento
- include_tasks: post_provisioning_tasks.yaml

```

## Estructura y Anatomía del Playbook

Todos los playbooks de OmniCRM siguen una estructura consistente. Entender esta estructura es esencial para crear y mantener playbooks.

### Estructura Básica

Cada playbook comienza con estos encabezados estándar:

```

- name: Nombre Descriptivo del Playbook
  hosts: localhost          # Siempre localhost para OmniCRM
  gather_facts: no          # Deshabilitado por rendimiento
  become: False             # No escalar privilegios

tasks:
  - name: Bloque principal
    block:
      # Las tareas de aprovisionamiento van aquí

  rescue:
    # Las tareas de reversión/limpieza van aquí

```

### Explicación del Encabezado

#### **name**

Nombre descriptivo que se muestra en los registros de aprovisionamiento y en la interfaz de usuario. Esto aparece como `playbook_description` en el registro de provisión.

#### **hosts: localhost**

Todos los playbooks de OmniCRM se ejecutan en localhost ya que interactúan con sistemas remotos a través de APIs, no por SSH.

#### **gather\_facts: no**

La recopilación de hechos de Ansible está deshabilitada porque:

- No necesitamos información del sistema
- Agrega sobrecarga innecesaria
- Puede bloquear navegadores si se muestra en la salida de depuración

## **become: False**

No se necesita escalada de privilegios ya que estamos haciendo llamadas a la API, no modificando archivos del sistema.

## **Carga de Configuración**

Cada playbook debe cargar el archivo de configuración central:

```
tasks:
  - name: Incluir vars de crm_config
    ansible.builtin.include_vars:
      file: "../../crm_config.yaml"
      name: crm_config
```

Esto hace que la configuración esté disponible como `crm_config.ocs.cgrates`, `crm_config.crm.base_url`, etc.

El `crm_config.yaml` típicamente contiene:

```
ocs:
  cgrates: "10.0.1.100:2080"
  ocsTenant: "default_tenant"
crm:
  base_url: "https://crm.example.com"
```

## **Patrones de Acceso a Variables**

Las variables pueden venir de varias fuentes:

### **De la Definición del Producto:**

```
- name: Acceder a product_id pasado por OmniCRM
  debug:
    msg: "Aprovisionando producto {{ product_id }}"
```

### **De la Selección de Inventario:**

```
- name: Obtener ID de inventario para la Tarjeta SIM
  set_fact:
    sim_card_id: "{{ hostvars[inventory_hostname]['SIM Card'] | int }}"
  when: "'SIM Card' in hostvars[inventory_hostname]"
```

### **De las Respuestas de la API:**

```
- name: Obtener información del Producto desde la API de CRM
  uri:
    url: "http://localhost:5000/crm/product/product_id/{{ product_id }}
```

```

}"}
    method: GET
    headers:
        Authorization: "Bearer {{ access_token }}"
    return_content: yes
    register: api_response_product

- name: Usar el nombre del producto
  debug:
    msg: "El nombre del producto es {{ api_response_product.json.product_name }}"

```

## Patrones Comunes de Playbook

### Patrón de Aprovisionamiento de Servicios

Este es el patrón más común para crear nuevos servicios.

```

- name: Playbook de Aprovisionamiento de Servicios
  hosts: localhost
  gather_facts: no
  become: False

  tasks:
    - name: Bloque principal
      block:

        # 1. Cargar configuración
        - name: Incluir vars de crm_config
          ansible.builtin.include_vars:
            file: "../../crm_config.yaml"
            name: crm_config

        # 2. Obtener información del producto
        - name: Obtener información del Producto desde la API de CRM
          uri:
            url: "http://localhost:5000/crm/product/product_id/{{ product_id }}"
            method: GET
            headers:
                Authorization: "Bearer {{ access_token }}"
            return_content: yes
            validate_certs: no
            register: api_response_product

        # 3. Obtener información del cliente
        - name: Obtener información del Cliente desde la API de CRM
          uri:

```

```

        url: "http://localhost:5000/crm/customer/customer_id/{{customer_id }}"
        method: GET
        headers:
            Authorization: "Bearer {{ access_token }}"
        return_content: yes
        register: api_response_customer

# 4. Establecer hechos a partir de los datos recuperados
- name: Establecer hechos del paquete
  set_fact:
    package_name: "{{ api_response_product.json.product_name }}"
}
  package_comment: "{{ api_response_product.json.comment }}"
  setup_cost: "{{ api_response_product.json.retail_setup_cost }}"
}
  monthly_cost: "{{ api_response_product.json.retail_cost }}"

# 5. Generar identificadores únicos
- name: Generar UUID
  set_fact:
    uuid: "{{ 99999999 | random | to_uuid }}"

- name: Generar UUID de Servicio
  set_fact:
    service_uuid: "Service_{{ uuid[0:8] }}"

# 6. Crear cuenta en el sistema de facturación
- name: Crear cuenta en OCS/CGRateS
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    headers:
      Content-Type: "application/json"
    body:
      {
        "method": "ApierV2.SetAccount",
        "params": [
          {
            "Tenant": "{{ crm_config.ocs.ocsTenant }}",
            "Account": "{{ service_uuid }}",
            "ActionPlanIds": [],
            "ActionPlansOverwrite": true,
            "ExtraOptions": {
              "AllowNegative": false,
              "Disabled": false
            },
            "ReloadScheduler": true
          }
        ]
      }

```

```

        }]
    }
    status_code: 200
register: ocs_response

- name: Verificar creación de cuenta OCS
assert:
  that:
    - ocs_response.status == 200
    - ocs_response.json.result == "OK"

# 7. Agregar saldo inicial
- name: Agregar 0 Saldo Monetario
uri:
  url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
method: POST
body_format: json
body:
{
  "method": "ApierV1.AddBalance",
  "params": [
    {
      "Tenant": "{{ crm_config.ocs.ocsTenant }}",
      "Account": "{{ service_uuid }}",
      "BalanceType": "*monetary",
      "Categories": "*any",
      "Balance": {
        "ID": "Saldo Inicial",
        "Value": 0,
        "ExpiryTime": "+4320h",
        "Weight": 1,
        "Blocker": true
      }
    }
  ]
}
status_code: 200
register: balance_response

# 8. Crear registro de servicio en CRM
- name: Obtener fecha y hora actuales en formato ISO 8601
command: date --utc +%Y-%m-%dT%H:%M:%S%z
register: current_date_time

- name: Agregar Servicio a través de API
uri:
  url: "http://localhost:5000/crm/service/"
method: PUT
body_format: json
headers:

```

```

Content-Type: "application/json"
Authorization: "Bearer {{ access_token }}"
body:
{
  "customer_id": "{{ customer_id }}",
  "product_id": "{{ product_id }}",
  "service_name": "{{ package_name }} - {{ service_uuid }}",
  "service_type": "generic",
  "service_uuid": "{{ service_uuid }}",
  "service_billed": true,
  "service_taxable": true,
  "service_provisioned_date": "{{ current_date_time.stdout }}",
  "service_status": "Active",
  "wholesale_cost": "{{ api_response_product.json.wholesale_cost | float }}",
  "retail_cost": "{{ monthly_cost | float }}"
}
status_code: 200
register: service_creation_response

# 9. Agregar transacción de costo de configuración
- name: Agregar Transacción de Costo de Configuración a través
de API
uri:
  url: "http://localhost:5000/crm/transaction/"
  method: PUT
  headers:
    Content-Type: "application/json"
    Authorization: "Bearer {{ access_token }}"
  body_format: json
  body:
  {
    "customer_id": {{ customer_id | int }},
    "service_id": {{ service_creation_response.json.service_id | int }},
    "title": "{{ package_name }} - Costos de
Configuración",
    "description": "Costos de configuración para {{ package_comment }}",
    "invoice_id": null,
    "retail_cost": "{{ setup_cost | float }}"
  }
  return_content: yes
register: transaction_response

# 10. Incluir tareas post-aprovisionamiento

```

```

- include_tasks: post_provisioning_tasks.yaml

rescue:

# Sección de reversión/limpieza
- name: Imprimir todas las vars para depuración
  debug:
    var: hostvars[inventory_hostname]

- name: Eliminar cuenta en OCS
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
    {
      "method": "ApierV2.RemoveAccount",
      "params": [
        "Tenant": "{{ crm_config.ocs.ocsTenant }}",
        "Account": "{{ service_uuid }}",
        "ReloadScheduler": true
      ]
    }
    status_code: 200
  ignore_errors: True
  when: service_uuid is defined

- name: Eliminar Servicio de CRM si fue creado
  uri:
    url: "http://localhost:5000/crm/service/service_id/{{ service_creation_response.json.service_id }}"
    method: DELETE
    headers:
      Authorization: "Bearer {{ access_token }}"
    status_code: 200
  ignore_errors: True
  when: service_creation_response is defined

- name: Fallar si no es una desprovisión intencional
  assert:
    that:
      - action == "deprovision"

```

## Patrón de Recarga

Utilizado para agregar créditos, datos o tiempo a servicios existentes.

```
- name: Playbook de Recarga de Servicio
```

```

hosts: localhost
gather_facts: no
become: False

tasks:
  - name: Incluir vars de crm_config
    ansible.builtin.include_vars:
      file: "../../crm_config.yaml"
      name: crm_config

    # 1. Obtener información del servicio
    - name: Obtener información del Servicio desde la API de CRM
      uri:
        url: "http://localhost:5000/crm/service/service_id/{{ service_id }}"
      method: GET
      headers:
        Authorization: "Bearer {{ access_token }}"
      return_content: yes
      register: api_response_service

    # 2. Obtener información del producto (qué recargar)
    - name: Obtener información del Producto desde la API de CRM
      uri:
        url: "http://localhost:5000/crm/product/product_id/{{ product_id }}"
      method: GET
      headers:
        Authorization: "Bearer {{ access_token }}"
      return_content: yes
      register: api_response_product

    # 3. Extraer detalles del servicio
    - name: Establecer hechos del servicio
      set_fact:
        service_uuid: "{{ api_response_service.json.service_uuid }}"
        customer_id: "{{ api_response_service.json.customer_id }}"
        package_name: "{{ api_response_product.json.product_name }}"
        topup_value: "{{ api_response_product.json.retail_cost }}"

    # 4. Ejecutar acción en el sistema de facturación (recarga gratuita)
    - name: Ejecutar Acción para agregar créditos
      uri:
        url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
      method: POST
      body_format: json
      body:

```

```

{
  "method": "APIerSv1.ExecuteAction",
  "params": [
    "Tenant": "{{ crm_config.ocs.ocsTenant }}",
    "Account": "{{ service_uuid }}",
    "ActionsId": "Action_Topup_Standard"
  ]
}
status_code: 200
register: action_response

- name: Verificar que la acción se ejecutó correctamente
  assert:
    that:
      - action_response.status == 200
      - action_response.json.result == "OK"

# 5. Restablecer cualquier límite activado
- name: Restablecer ActionTriggers
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
      {
        "method": "APIerSv1.ResetAccountActionTriggers",
        "params": [
          "Tenant": "{{ crm_config.ocs.ocsTenant }}",
          "Account": "{{ service_uuid }}",
          "Executed": false
        ]
      }
  status_code: 200

# 6. Actualizar fechas del servicio
- name: Calcular nueva fecha de expiración
  command: "date --utc +%Y-%m-%dT%H:%M:%S%z -d '+30 days'"
  register: new_expiry_date

- name: Actualizar Servicio con nueva expiración
  uri:
    url: "http://localhost:5000/crm/service/{{ service_id }}"
    method: PATCH
    headers:
      Authorization: "Bearer {{ access_token }}"
      Content-Type: "application/json"
    body_format: json
    body:

```

```

        {
            "service_deactivate_date": "{{ new_expiry_date.stdout }}",
            "service_status": "Active"
        }

# 7. Opcional: Enviar notificación
- name: Enviar SMS de Notificación
  uri:
    url: "http://sms-gateway/api/send"
    method: POST
    body_format: json
    body:
      {
        "source": "CompanyName",
        "destination": "{{ customer_phone }}",
        "message": "Su servicio ha sido recargado. Nueva
expiración: {{ new_expiry_date.stdout }}"
      }
    status_code: 201
  ignore_errors: True

```

## Patrón de Aprovisionamiento de CPE

Utilizado para configurar equipos de cliente (routers, módems, ONTs).

```

- name: Playbook de Aprovisionamiento de CPE
  hosts: localhost
  gather_facts: no
  become: False

  tasks:
    - name: Incluir vars de crm_config
      ansible.builtin.include_vars:
        file: "../../crm_config.yaml"
        name: crm_config

# 1. Obtener elemento de inventario para CPE
- name: Establecer ID de inventario de CPE desde hostvars
  set_fact:
    cpe_inventory_id: "{{ hostvars[inventory_hostname]['WiFi
Router CPE'] | int }}"
  when: "'WiFi Router CPE' in hostvars[inventory_hostname]"

# 2. Obtener detalles de CPE del inventario
- name: Obtener datos de inventario para CPE
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/

```

```

inventory_id/{{ cpe_inventory_id }}"
    method: GET
    headers:
        Authorization: "Bearer {{ access_token }}"
    return_content: yes
    register: api_response_cpe

# 3. Obtener información del sitio del cliente
- name: Obtener información del Sitio desde la API
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/site/customer_id/{{ customer_id }}"
    method: GET
    headers:
        Authorization: "Bearer {{ access_token }}"
    return_content: yes
  register: api_response_site

# 4. Actualizar inventario de CPE con ubicación
- name: Parchear elemento de inventario de CPE con ubicación
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ cpe_inventory_id }}"
    method: PATCH
    body_format: json
    headers:
        Authorization: "Bearer {{ access_token }}"
    body:
    {
        "address_line_1": "{{ api_response_site.json.0.address_line_1 }}",
        "city": "{{ api_response_site.json.0.city }}",
        "state": "{{ api_response_site.json.0.state }}",
        "latitude": "{{ api_response_site.json.0.latitude }}",
        "longitude": "{{ api_response_site.json.0.longitude }}"
    }
  status_code: 200

# 5. Generar credenciales
- name: Establecer nombre de host de CPE
  set_fact:
    cpe_hostname: "CPE_{{ cpe_inventory_id }}"
    cpe_username: "admin_{{ cpe_inventory_id }}"

- name: Generar contraseña aleatoria
  set_fact:
    cpe_password: "{{ lookup('pipe', 'cat /dev/urandom | tr -dc a-zA-Z0-9 | head -c 16') }}"

```

```

# 6. Generar credenciales de WiFi
- name: Establecer SSID de WiFi
  set_fact:
    wifi_ssid: "Network_{{ cpe_inventory_id }}"

- name: Generar contraseña de WiFi
  set_fact:
    word_list:
      - apple
      - cloud
      - river
      - mountain
      - ocean

- name: Crear PSK de WiFi
  set_fact:
    random_word: "{{ word_list | random }}"
    random_number: "{{ 99999 | random(start=10000) }}"

- name: Combinar PSK de WiFi
  set_fact:
    wifi_psk: "{{ random_word }}{{ random_number }}"

# 7. Generar archivo de configuración
- name: Establecer nombre de archivo de configuración
  set_fact:
    config_name: "{{ cpe_hostname }}_{{ lookup('pipe', 'date
+%Y%m%d%H%M%S') }}.cfg"
    config_dest: "/tmp/{{ cpe_hostname }}_{{ lookup('pipe', 'date
+%Y%m%d%H%M%S') }}.cfg"

- name: Crear configuración desde plantilla
  template:
    src: "templates/cpe_router_config.j2"
    dest: "{{ config_dest }}"

# 8. Leer configuración generada
- name: Leer archivo de configuración
  ansible.builtin.slurp:
    src: "{{ config_dest }}"
    register: config_content

# 9. Actualizar inventario con información de aprovisionamiento
- name: Parchear inventario de CPE con configuración
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/
inventory_id/{{ cpe_inventory_id }}"

```

```

method: PATCH
body_format: json
headers:
  Authorization: "Bearer {{ access_token }}"
body:
{
  "itemtext3": "{{ wifi_ssid }}",
  "itemtext4": "{{ wifi_psk }}",
  "management_url": "{{ cpe_hostname }}",
  "management_username": "{{ cpe_username }}",
  "management_password": "{{ cpe_password }}",
  "config_content": "{{ config_content.content | b64decode }}",
  "inventory_notes": "Provisionado: {{ lookup('pipe', 'date +%Y-%m-%d') }}"
}
status_code: 200

# 10. Enviar configuración al equipo de soporte
- name: Enviar configuración por correo electrónico al soporte
  uri:
    url: "https://api.mailjet.com/v3.1/send"
    method: POST
    body_format: json
    headers:
      Content-Type: "application/json"
    body:
    {
      "Messages": [
        "From": {
          "Email": "provisioning@example.com",
          "Name": "Sistema de Aprovisionamiento"
        },
        "To": [
          {
            "Email": "support@example.com",
            "Name": "Equipo de Soporte"
          }
        ],
        "Subject": "Configuración de CPE - {{ cpe_hostname }}",
        "Attachments": [
          {
            "ContentType": "text/plain",
            "Filename": "{{ config_name }}",
            "Base64Content": "{{ config_content.content }}"
          }
        ]
      ]
    }
  user: "{{ mailjet_api_key }}"
  password: "{{ mailjet_api_secret }}"
  force_basic_auth: true

```

```
status_code: 200
```

## Patrón de Renovación Automática

Configurar cargos recurrentes automáticos o renovaciones utilizando ActionPlans de CGRateS.

```
# Parte de un playbook de recarga que configura la renovación
# automática

# 1. Normalizar parámetro auto_renew
- name: Normalizar auto_renew a booleano
  set_fact:
    auto_renew_bool: "{{ (auto_renew | string | lower) in ['true', '1', 'yes'] }}"

# 2. Crear acción para renovación automática
- name: Crear Acción para AutoRenovación
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
    {
      "method": "ApierV1.SetActions",
      "params": [
        {
          "ActionsId": "Action_AutoTopup_{{ service_uuid }}_{{ product_id }}",
          "Overwrite": true,
          "Actions": [
            {
              "Identifier": "*http_post",
              "ExtraParameters": "{{ crm_config.crm.base_url }}/crm/provision/simple_provision_addon/service_id/{{ service_id }}/{{ product_id }}/{{ product_id }}"
            },
            {
              "Identifier": "*cdrlog",
              "BalanceType": "*generic",
              "ExtraParameters": "{{ \"Category\": \"activation\", \"Destination\": \"Auto Renewal\" }}"
            }
          ]
        }
      ]
    }
  status_code: 200
  register: action_response
  when: auto_renew_bool
```

```

# 3. Crear ActionPlan mensual
- name: Crear ActionPlan para Renovación Mensual
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
      {
        "method": "ApierV1.SetActionPlan",
        "params": [
          {
            "Id": "ActionPlan_Monthly_{{ service_uuid }}_{{ product_id }}",
            "Tenant": "{{ crm_config.ocs.ocsTenant }}",
            "ActionPlan": [
              {
                "ActionsId": "Action_AutoTopup_{{ service_uuid }}_{{ product_id }}",
                "Years": "*any",
                "Months": "*any",
                "MonthDays": "*any",
                "WeekDays": "*any",
                "Time": "*monthly",
                "StartTime": "*now",
                "Weight": 10
              },
              {
                "Overwrite": true,
                "ReloadScheduler": true
              }
            ],
            "Overwrite": true,
            "ReloadScheduler": true
          }
        ],
        "status_code": 200
      when: auto_renew_bool

# 4. Asignar ActionPlan a la cuenta
- name: Asignar ActionPlan a la cuenta
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
      {
        "method": "ApierV2.SetAccount",
        "params": [
          {
            "Tenant": "{{ crm_config.ocs.ocsTenant }}",
            "Account": "{{ service_uuid }}",
            "ActionPlanIds": ["ActionPlan_Monthly_{{ service_uuid }}_{{ product_id }}"],
            "ActionPlansOverwrite": true,
            "ReloadScheduler": true
          }
        ]
      }

```

```

        }]
    }
  status_code: 200
when: auto_renew_bool

# 5. Eliminar ActionPlan si la renovación automática está
deshabilitada
- name: Eliminar ActionPlan de la cuenta
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
      {
        "method": "ApierV1.RemoveActionPlan",
        "params": [
          {
            "Tenant": "{{ crm_config.ocs.ocsTenant }}",
            "Id": "ActionPlan_Monthly_{{ service_uuid }}_{{ product_id
}}"
          }
        ]
      }
    status_code: 200
  ignore_errors: true
when: not auto_renew_bool

```

## Tareas Reutilizables

Las tareas reutilizables son pequeños playbooks autónomos que pueden ser incluidos por múltiples plays. Promueven la reutilización de código y la consistencia.

### Tarea de Correo Electrónico de Bienvenida

task\_welcome\_email.yaml - Envía un correo electrónico de bienvenida a nuevos clientes.

```

# Esta tarea espera que estas variables sean establecidas por el play
padre:
# - api_response_customer (detalles del cliente)
# - package_name (nombre del producto)
# - monthly_cost (costo recurrente)
# - setup_cost (costo único)

- name: Establecer configuración de correo electrónico
  set_fact:
    mailjet_api_key: "{{ lookup('env', 'MAILJET_API_KEY') }}"
    mailjet_api_secret: "{{ lookup('env', 'MAILJET_SECRET') }}"
    email_from: "noreply@example.com"

```

```
recipients: []

- name: Establecer asunto del correo electrónico y nombre del remitente
  set_fact:
    email_subject: ";Bienvenido a nuestro servicio!"
    email_from_name: "Equipo de Atención al Cliente"

- name: Preparar lista de destinatarios de los contactos del cliente
  loop: "{{ api_response_customer.json.contacts }}"
  set_fact:
    recipients: "{{ recipients + [ {'Email': item.contact_email, 'Name': item.contact_firstname ~ ' ' ~ item.contact_lastname} ] }}"
  loop_control:
    continue_if: item.contact_email == null

- name: Obtener nombre del primer contacto
  set_fact:
    first_contact: "{{ api_response_customer.json.contacts[0].contact_firstname }}"

- name: Enviar correo electrónico de bienvenida
  uri:
    url: "https://api.mailjet.com/v3.1/send"
    method: POST
    body_format: json
    headers:
      Content-Type: "application/json"
    body:
      {
        "Messages": [
          {
            "From": {
              "Email": "{{ email_from }}",
              "Name": "{{ email_from_name }}"
            },
            "To": "{{ recipients }}",
            "Subject": "{{ email_subject }}",
            "TextPart": "Estimado {{ first_contact }}, ¡bienvenido! Su servicio está listo.",
            "HTMLPart": "Estimado {{ first_contact }},  
<h3>¡Bienvenido!</h3><br/>Su servicio {{ package_name }} ahora está activo.<br/>Costo mensual: ${{ monthly_cost }}<br/>Costo de configuración: ${{ setup_cost }}<br/>Si tiene alguna pregunta, contacte a support@example.com"
          }
        ]
      }
    user: "{{ mailjet_api_key }}"
    password: "{{ mailjet_api_secret }}"
    force_basic_auth: true
    status_code: 200
```

```
register: email_response
```

## Tareas Post-Aprovisionamiento

`post_provisioning_tasks.yaml` - Limpieza estándar y notificaciones que se ejecutan después de cada provisión.

```
# Este archivo se incluye al final de la mayoría de los playbooks de aprovisionamiento
# Maneja operaciones comunes post-aprovisionamiento
- include_tasks: task_notify_ocs.yaml
```

El `task_notify_ocs.yaml` podría contener:

```
- name: Notificar a OCS de la finalización del aprovisionamiento
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
      {
        "method": "APIerSv1.ReloadCache",
        "params": [
          {
            "ArgsCache": "*all"
          }
        ]
      }
    status_code: 200
  ignore_errors: true
```

## Operaciones Comunes

### Trabajando con inventario

Recuperando detalles del inventario:

```
- name: Obtener ID de inventario de la Tarjeta SIM
  set_fact:
    sim_inventory_id: "{{ hostvars[inventory_hostname]['SIM Card'] | int }}"
  when: "'SIM Card' in hostvars[inventory_hostname]"

- name: Obtener detalles de la Tarjeta SIM
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ sim_inventory_id }}"
    method: GET
    headers:
```

```

    Authorization: "Bearer {{ access_token }}"
    return_content: yes
    register: sim_response

- name: Extraer detalles de la SIM
  set_fact:
    iccid: "{{ sim_response.json.iccid }}"
    imsi: "{{ sim_response.json.imsi }}"
    ki: "{{ sim_response.json.ki }}"

```

Asignando inventario al cliente:

```

- name: Asignar SIM al cliente
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ sim_inventory_id }}"
    method: PATCH
    headers:
      Authorization: "Bearer {{ access_token }}"
    body_format: json
    body:
      {
        "customer_id": {{ customer_id }},
        "service_id": {{ service_id }},
        "item_state": "Assigned"
      }
  status_code: 200

```

## Operaciones de Fecha y Hora

Obteniendo la fecha/hora actuales:

```

- name: Obtener fecha y hora actuales en formato ISO 8601
  command: date --utc +%Y-%m-%dT%H:%M:%S%z
  register: current_date_time

- name: Obtener solo la fecha de hoy
  set_fact:
    today: "{{ lookup('pipe', 'date +%Y-%m-%d') }}"

```

Calculando fechas futuras:

```

- name: Calcular fecha de expiración 30 días a partir de ahora
  command: "date --utc +%Y-%m-%dT%H:%M:%S%z -d '+30 days'"
  register: expiry_date

- name: Calcular fecha 90 días en el futuro
  command: "date --utc +%Y-%m-%d -d '+{{ days }} days'"

```

```
register: future_date
vars:
  days: 90
```

## Generando Valores Aleatorios

UUIDs e identificadores:

- `name`: Generar UUID  
`set_fact`:  
  `uuid: "{{ 99999999 | random | to_uuid }}"`
- `name`: Generar identificador de servicio  
`set_fact`:  
  `service_uuid: "SVC_{{ uuid[0:8] }}"`

Contraseñas aleatorias:

- `name`: Generar contraseña segura  
`set_fact`:  
  `password: "{{ lookup('pipe', 'cat /dev/urandom | tr -dc a-zA-Z0-9 | head -c 16') }}"`

Frases de contraseña memorables:

- `name`: Establecer lista de palabras  
`set_fact`:  
  `words:`
  - alpha
  - bravo
  - charlie
  - delta
  - echo
- `name`: Generar frase de contraseña  
`set_fact`:  
  `word: "{{ words | random }}"`  
  `number: "{{ 99999 | random(start=10000) }}"`
- `name`: Combinar en frase de contraseña  
`set_fact`:  
  `passphrase: "{{ word }}{{ number }}"`

## Trabajando con CGRateS/OCS

Creando cuentas:

- `name`: Crear cuenta de facturación

```

uri:
  url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body_format: json
  body:
  {
    "method": "ApierV2.SetAccount",
    "params": [
      "Tenant": "{{ crm_config.ocs.ocsTenant }}",
      "Account": "{{ service_uuid }}",
      "ActionPlanIds": [],
      "ActionPlansOverwrite": true,
      "ExtraOptions": {
        "AllowNegative": false,
        "Disabled": false
      },
      "ReloadScheduler": true
    ]
  }
  status_code: 200
register: account_response

```

Agregando saldos:

```

- name: Agregar saldo de datos
uri:
  url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body_format: json
  body:
  {
    "method": "ApierV1.AddBalance",
    "params": [
      "Tenant": "{{ crm_config.ocs.ocsTenant }}",
      "Account": "{{ service_uuid }}",
      "BalanceType": "*data",
      "Categories": "*any",
      "Balance": {
        "ID": "Paquete de Datos",
        "Value": 10737418240,
        "ExpiryTime": "+720h",
        "Weight": 10
      }
    ]
  }
  status_code: 200

```

Ejecutando acciones:

```

- name: Ejecutar acción de cobro
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body_format: json
  body:
  {
    "method": "APIerSv1.ExecuteAction",
    "params": [
      "Tenant": "{{ crm_config.ocs.ocsTenant }}",
      "Account": "{{ service_uuid }}",
      "ActionsId": "Action_Standard_Charge"
    ]
  }
  status_code: 200

```

Obteniendo información de la cuenta:

```

- name: Obtener detalles de la cuenta
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body_format: json
  body:
  {
    "method": "ApierV2.GetAccount",
    "params": [
      "Tenant": "{{ crm_config.ocs.ocsTenant }}",
      "Account": "{{ service_uuid }}"
    ]
  }
  status_code: 200
  register: account_info

```

Trabajando con Perfiles de Atributos:

```

- name: Obtener Perfil de Atributo
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body_format: json
  body:
  {
    "method": "APIerSv1.GetAttributeProfile",
    "params": [
      "Tenant": "{{ crm_config.ocs.ocsTenant }}",
      "ID": "ATTR_{{ service_uuid }}"
    ]
  }

```

```

  return_content: yes
  status_code: 200
  register: attr_response
  ignore_errors: true

- name: Extraer valor de atributo
  set_fact:
    phone_number: "{{ attr_response.json.result.Attributes |
  json_query(\"[?Path=='req.PhoneNumber'].Value[0].Rules\") | first
}}"
  when: attr_response is defined

```

## Lógica Condicional

Verificando si existen variables:

```

- name: Usar valor personalizado o predeterminado
  set_fact:
    monthly_cost: "{{ custom_cost | default(50.00) }}"

- name: Ejecutar solo si la variable está definida
  debug:
    msg: "El UUID del servicio es {{ service_uuid }}"
  when: service_uuid is defined

```

Condiciones booleanas:

```

- name: Aprovisionar equipo
  include_tasks: configure_cpe.yaml
  when: provision_cpe | default(false) | bool

- name: Saltar si es desprovisión
  assert:
    that:
      - action != "deprovision"
  when: action is defined

```

Múltiples condiciones:

```

- name: Tarea condicional compleja
  uri:
    url: "{{ endpoint }}"
    method: POST
  when:
    - service_uuid is defined
    - customer_id is defined
    - action != "deprovision"
    - enable_feature | default(true) | bool

```

## Bucles e Iteración

Bucles simples:

```
- name: Crear múltiples saldos
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
      {
        "method": "ApierV1.AddBalance",
        "params": [
          {
            "Account": "{{ service_uuid }}",
            "BalanceType": "{{ item.type }}",
            "Balance": {
              "Value": "{{ item.value }}"
            }
          }
        ]
      }
  loop:
    - { type: "*voice", value: 3600 }
    - { type: "*data", value: 10737418240 }
    - { type: "*sms", value: 100 }
```

Iterando sobre respuestas de API:

```
- name: Obtener todos los sitios del cliente
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/site/customer_id/{{ customer_id }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
  register: sites_response

- name: Configurar equipo en cada sitio
  debug:
    msg: "Configurando sitio en {{ item.address_line_1 }}"
  loop: "{{ sites_response.json }}"
```

## Manejo de Errores

Usando ignore\_errors:

```
- name: Notificación SMS opcional
  uri:
    url: "http://sms-gateway/send"
```

```
  method: POST
  body: {...}
  ignore_errors: true
```

Afirmaciones para validación:

```
- name: Verificar respuesta de API
  assert:
    that:
      - response.status == 200
      - response.json.result == "OK"
  fail_msg: "La llamada a la API falló: {{ response.json }}"
```

Manejo de errores condicional:

```
- name: Intentar obtener servicio existente
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/service/service_uuid/{{ service_uuid }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
  register: service_lookup
  failed_when: false

- name: Crear servicio si no existe
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/service/"
    method: PUT
    body: {...}
  when: service_lookup.status == 404
```

## Mejores Prácticas

### Nomenclatura de Variables

Usar nombres descriptivos y consistentes:

```
# Bueno
service_uuid: "SVC_12345"
customer_name: "John Smith"
monthly_cost: 49.99
```

```
# Malo
svc: "SVC_12345"
name: "John Smith"
cost: 49.99
```

Prefijar variables por fuente:

```
api_response_customer: {...}
api_response_product: {...}
cgr_account_info: {...}
```

## Depuración

Imprimir variables para solucionar problemas:

```
- name: Imprimir todas las variables
  debug:
    var: hostvars[inventory_hostname]

- name: Imprimir variable específica
  debug:
    msg: "UUID del servicio: {{ service_uuid }}"

- name: Imprimir respuesta de API
  debug:
    var: api_response_product.json
```

## Validación

Siempre validar respuestas críticas de API:

```
- name: Crear cuenta
  uri:
    url: "{{ billing_endpoint }}"
    method: POST
    body: {...}
  register: response

- name: Verificar creación de cuenta
  assert:
    that:
      - response.status == 200
      - response.json.result == "OK"
  fail_msg: "Fallo al crear cuenta: {{ response.json }}"
```

## Idempotencia

Diseñar tareas para ser ejecutables de forma segura:

```
# Verificar si el recurso existe primero
- name: Verificar si la cuenta existe
  uri:
    url: "{{ ocs_endpoint }}/get_account"
```

```

    method: POST
    body: {"Account": "{{ service_uuid }}"}
register: account_check
failed_when: false

# Solo crear si no existe
- name: Crear cuenta
  uri:
    url: "{{ ocs_endpoint }}/create_account"
    method: POST
    body: {...}
  when: account_check.status == 404

```

## Seguridad

Nunca codificar credenciales:

```

# Malo
mailjet_api_key: "abc123def456"

# Bueno - usar variables de entorno
mailjet_api_key: "{{ lookup('env', 'MAILJET_API_KEY') }}"

# Bueno - usar archivo de configuración
mailjet_api_key: "{{ crm_config.email.api_key }}"

```

Siempre usar HTTPS y autenticación:

```

- name: Llamar a API externa
  uri:
    url: "https://api.example.com/endpoint"
    method: POST
    headers:
      Authorization: "Bearer {{ access_token }}"
    validate_certs: yes

```

## Documentación

Documentar lógica compleja:

```

# Calcular cargo prorratizado por mes parcial
# Si el cliente se inscribe el día 15 y la facturación es el día 1,
# cobrar el 50% del costo mensual por los días restantes
- name: Calcular días hasta el final del mes
  command: "date -d 'last day of this month' +%d"
  register: days_in_month

- name: Obtener día actual

```

```

command: "date +%d"
register: current_day

- name: Calcular monto prorrteado
  set_fact:
    days_remaining: "{{ (days_in_month.stdout | int) -
(current_day.stdout | int) }}"
    pro_rata_cost: "{{ (monthly_cost | float) * (days_remaining | float) / (days_in_month.stdout | float) }}"

```

## Pruebas de Playbooks

### Enfoque de Pruebas

- Ejecutar en Seco Primero:** Probar con sistemas no productivos
- Verificar Variables:** Usar tareas de depuración para confirmar que todas las variables requeridas están presentes
- Verificar Respuestas:** Validar respuestas de API antes de continuar
- Pruebas de Reversión:** Fallar intencionalmente tareas para verificar que los bloques de rescate funcionen
- Pruebas de Desprovisión:** Probar con action: "deprovision" para verificar limpieza

Ejemplo de playbook de prueba:

```

- name: Probar Aprovisionamiento de Servicio
  hosts: localhost
  gather_facts: no

  tasks:
    - name: Verificar variables requeridas
      assert:
        that:
          - product_id is defined
          - customer_id is defined
          - access_token is defined
      fail_msg: "Faltan variables requeridas"

    - name: Probar conectividad de API
      uri:
        url: "http://localhost:5000/crm/health"
        method: GET
      register: health_check

    - name: Verificar chequeo de salud
      assert:
        that:
          - health_check.status == 200

```

## Errores Comunes

### Faltan conversiones de tipo:

```
# Incorrecto - puede ser cadena
customer_id: "{{ customer_id }}"

# Correcto - asegurar entero
customer_id: {{ customer_id | int }}
```

### No manejar variables indefinidas:

```
# Incorrecto - falla si no está definida
service_uuid: "{{ service_uuid }}"

# Correcto - proporcionar predeterminado
service_uuid: "{{ service_uuid | default('') }}"
```

### Olvidar validación:

```
# Incorrecto - no verifica respuesta
- name: Crear cuenta
  uri: ...
  register: response

# Correcto - validar respuesta
- name: Crear cuenta
  uri: ...
  register: response

- name: Verificar creación
  assert:
    that:
      - response.json.result == "OK"
```

## Flujo de Trabajo de Aprovisionamiento

Generalmente, el personal de Omnitouch trabajará con el cliente para:

1. Definir los requisitos del producto
2. Desarrollar los playbooks de Ansible necesarios para automatizar el proceso de aprovisionamiento
3. Probar los playbooks en un entorno de staging
4. Desplegar en producción

Esto asegura que cada servicio se despliegue de manera consistente y confiable, reduciendo el riesgo de errores y asegurando que todos los pasos necesarios se completen en el orden correcto.

# Variables de Ansible

Las variables pasadas a los playbooks de Ansible incluyen:

## Variables del Producto

Derivadas de las configuraciones de productos de OmniCRM y definen cómo se debe configurar el servicio.

## Variables de Inventory

Seleccionadas del inventario, estas incluyen elementos como módems, tarjetas SIM, bloques de direcciones IP o números de teléfono que son necesarios para el aprovisionamiento.

## Variables del Sistema

Agregadas automáticamente por OmniCRM:

- `product_id` - El producto que se está aprovisionando
- `customer_id` - El cliente que recibe el servicio
- `service_id` - El servicio que se está modificando (para recargas/cambios)
- `access_token` - JWT para autenticación de API

# Desprovisionamiento

Cuando un servicio ya no es necesario, los **Playbooks de Ansible** también se utilizan para desprovisionar el servicio utilizando el patrón de bloque `rescue`. Esto:

- Elimina cualquier configuración
- Libera inventario de nuevo al grupo
- Elimina cuentas de facturación
- Asegura que el sistema se mantenga limpio

# Reversión y Manejo de Errores

La función **block/rescue** de Ansible se emplea durante el aprovisionamiento y desprovisionamiento para manejar errores de manera elegante. Si una tarea falla en cualquier momento durante el aprovisionamiento, la sección de rescate revierte automáticamente los cambios para regresar a un estado consistente. Esto asegura confiabilidad y reduce el riesgo de despliegues parciales o fallidos.

Para obtener detalles completos sobre el sistema de aprovisionamiento, flujos de trabajo y autenticación, consulte [concepts\\_provisioning](#).



# OmniCRM API

Todas las funciones dentro de OmniCRM son accesibles a través de la API - No hay funcionalidad que esté disponible solo en la interfaz de usuario.

Esto te permite integrar OmniCRM con otros sistemas o automatizar tareas.

La API es una API RESTful y está asegurada utilizando múltiples métodos de autenticación, incluyendo tokens JWT, claves API y listas blancas de IP.

La documentación de la API está documentada utilizando Swagger, una herramienta que permite una fácil lectura, comprensión y prueba de la funcionalidad de la API.

La documentación de la API está disponible en la siguiente URL:

<<https://yourcrm/crm/docs/>>

Página Swagger de OmniCRM

## Métodos de Autenticación

OmniCRM soporta tres métodos de autenticación, cada uno diseñado para diferentes casos de uso:

1. **Tokens Bearer JWT** - Para sesiones de usuario interactivas (interfaz web, aplicaciones móviles)
2. **Claves API** - Para integraciones de servidor a servidor y scripts de automatización
3. **Lista Blanca de IP** - Para sistemas internos de confianza (servidores de aprovisionamiento, herramientas de monitoreo)

### Autenticación con Token Bearer JWT

Este es el método de autenticación principal para las sesiones de usuario. Los usuarios inician sesión con su correo electrónico y contraseña, reciben un token JWT y lo utilizan para solicitudes posteriores.

#### Casos de Uso:

- Autenticación de interfaz web
- Autenticación de aplicaciones móviles
- Acceso programático de corta duración

## Cómo Autenticarse:

Para iniciar sesión, envía un cuerpo JSON con la siguiente estructura a /crm/auth/login como una solicitud POST:

```
{  
  "email": "youruser@yourdomain.com",  
  "password": "yourpassword"  
}
```

La API devolverá un objeto JSON que contiene un campo `token`, que se utiliza para autenticar todas las solicitudes futuras. Además, la respuesta incluye un `refresh_token` que se puede utilizar para refrescar el token cuando expire, junto con los permisos y roles del usuario.

Puedes probar esto desde la página de Swagger seleccionando el endpoint /auth/login, completando tu Nombre de Usuario y Contraseña, y haciendo clic en el botón Try it out.

Página Swagger de OmniCRM

Para autorizar la sesión, copia el valor del token y haz clic en el botón "Authorize" en la parte superior derecha de la página de Swagger. Pega el token en el campo "Value", precedido por `Bearer` y haz clic en "Authorize".

Autenticación Bearer de OmniCRM

Ahora, todas las solicitudes posteriores estarán autenticadas con este token.

## Autenticación con Clave API

Las claves API proporcionan autenticación segura y de larga duración para integraciones de servidor a servidor y scripts de automatización sin requerir contraseñas de usuario.

### Casos de Uso:

- Sistemas de aprovisionamiento automatizados
- Herramientas de monitoreo y alerta
- Integración con sistemas externos
- Tareas programadas y trabajos cron

### Cómo Funcionan las Claves API:

Las claves API se configuran en el archivo `crm_config.yaml` y están asociadas con roles y permisos específicos. Cada clave API es una cadena aleatoria segura (mínimo 32 caracteres) que autentica las solicitudes cuando se pasa en el encabezado X-API-KEY.

## Configurando Claves API:

Las claves API deben ser añadidas a `crm_config.yaml` por un administrador con acceso al servidor:

```
api_keys:
  your-secure-api-key-here-minimum-32-chars:
    roles:
      - admin
    description: "Sistema de automatización de aprovisionamiento"
  another-api-key-for-monitoring-system:
    roles:
      - view_customer
      - view_service
    description: "Monitoreo y alerta"
```

## Usando Claves API:

Incluye la clave API en el encabezado `X-API-KEY` de tus solicitudes:

```
curl -X GET "https://yourcrm.com/crm/customers" \
  -H "X-API-KEY: your-secure-api-key-here-minimum-32-chars"
```

## Ejemplo en Python:

```
import requests

crm_url = 'https://yourcrm.com'
api_key = 'your-secure-api-key-here-minimum-32-chars'

headers = {
    "Content-Type": "application/json",
    "X-API-KEY": api_key
}

# Obtener Clientes
response = requests.get(crm_url + '/crm/customers', headers=headers)
for customer in response.json()['data']:
    print(customer)
```

## Mejores Prácticas:

- Genera claves API utilizando generadores aleatorios seguros criptográficamente (`openssl rand -base64 48`)
- Usa diferentes claves API para diferentes sistemas
- Documenta el propósito de cada clave API en el campo `description`
- Rota las claves API periódicamente
- Nunca comités claves API en el control de versiones
- Asigna los permisos mínimos necesarios a cada clave API

## Autenticación con Lista Blanca de IP

La lista blanca de IP permite que direcciones IP específicas accedan a la API sin autenticación. Esto es útil para sistemas internos de confianza en redes privadas.

### Casos de Uso:

- Servidores de aprovisionamiento internos
- Sistemas de monitoreo de red en VLANs de gestión
- Playbooks de Ansible ejecutándose en infraestructura controlada

### Configurando la Lista Blanca de IP:

Añade direcciones IP de confianza a `crm_config.yaml`:

```
ip_whitelist:  
  - 192.168.1.100  
  - 10.0.0.0/24  
  - 172.16.50.10
```

### Consideraciones de Seguridad:

- Usa la lista blanca de IP solo en redes privadas y seguras
- Nunca incluyas direcciones IP públicas en la lista blanca
- Usa los rangos de IP más específicos posibles
- Documenta por qué cada IP está en la lista blanca
- Audita regularmente las IPs en la lista blanca

## Ejemplo de Llamadas a la API con Python

Aquí hay un ejemplo de cómo iniciar sesión y recuperar una lista de clientes usando autenticación con token JWT:

```
import requests  
  
crm_url = 'https://yourcrm.com'  
session = requests.Session()  
  
print("Aprovisionando datos al servidor: " + str(crm_url))  
  
headers = {  
    "Content-Type": "application/json"  
}  
  
# Obtener Token de Autenticación  
response = session.post(crm_url + '/crm/auth/login', json={  
    "email": "youruser@yourdomain.com",  
    "password": "yourpassword"
```

```
, headers=headers)

print(response.status_code)
print(response.json())
assert response.status_code == 200

headers['Authorization'] = 'Bearer ' + response.json()['token']
print("Autenticado en CRM con éxito")

# Obtener Clientes
response = session.get(crm_url + '/crm/customers', headers=headers)
for customer in response.json()['data']:
    print(customer)
```



# Notas sobre la Facturación de Servicios / Productos CRM

## Nota

Para una guía completa de extremo a extremo que cubre la definición de productos, provisión, complementos y desprovisión con ejemplos detallados de Ansible y estrategia de precios, consulte [Complete Product Lifecycle Guide <guide\\_product\\_lifecycle>](#).

## Visión General de Productos y Servicios

### Producto (Elemento del Menú):

Un Producto es como un plato específico en el menú de un restaurante, como un "Spaghetti Carbonara."

Tiene una descripción clara, una lista de ingredientes (como pasta, crema, huevos, queso y tocino), y un precio.

En OmniCRM, un Producto contiene de manera similar los detalles de lo que se incluye — características, especificaciones y precios.

A menudo, los clientes pueden querer modificaciones, como "sin cebollas" o "agregar queso extra" a su comida. Dentro de OmniCRM, esto corresponde a personalizar un servicio antes de su creación. El nivel de personalizaciones o modificaciones a un servicio depende de usted (el operador) para definir.

En OmniCRM, los clientes o el personal pueden modificar un Producto para adaptarse mejor a las necesidades de un cliente específico, como mejorar su velocidad de Internet o agregar características específicas. Esta personalización se refleja en el Servicio específico proporcionado.

Un producto es esencialmente una oferta de la que los clientes pueden elegir ordenar, similar a leer y seleccionar un plato del menú.

Definiciones de Producto OmniCRM

### Catálogo de Productos (Menú del Restaurante):

El Catálogo de Productos es como todo el menú en un restaurante, que lista todos los platos disponibles — desde aperitivos hasta postres.

Es la colección completa de todo lo que el restaurante (o en su caso, el Proveedor de Servicios) tiene para ofrecer.

En el contexto empresarial, el Catálogo de Productos proporciona a los clientes todos los Productos disponibles, para que puedan elegir el que mejor se adapte a sus necesidades.

Interfaz de Gestión de Productos Vista de Edición del Catálogo de Productos

## **Servicio (Plato Preparado):**

Cuando un cliente ordena un artículo del menú, el plato se prepara en la cocina. Esto es similar a crear un Servicio a partir de un Producto.

En OmniCRM, cuando un cliente selecciona un Producto, se crea una instancia de ese Producto y se entrega como un Servicio.

Se personaliza y prepara específicamente para ese cliente, al igual que una comida preparada para un comensal.

Por ejemplo, cuando alguien selecciona el "Plan de Internet Bronce" del Catálogo de Productos, el sistema de provisión "cocina" una instancia de ese plan a partir de los ingredientes (direcciones IP, módems y puertos) — es decir, activa el plan y lo entrega al cliente específico.

## **Productos Agrupados (Comidas Combinadas):**

El Catálogo de Productos también puede ofrecer paquetes, como una comida combinada que incluye un aperitivo, plato principal y postre juntos a un precio especial.

En OmniCRM, los Productos agrupados combinan múltiples Productos individuales en un solo paquete conveniente — como un "Paquete de Esenciales para el Hogar" que incluye servicios de Internet, cable y teléfono a una tarifa con descuento.

Una vez seleccionado, este paquete se convierte en múltiples Servicios adaptados para el cliente.

## **Definiciones de Productos**

Un producto es una plantilla que se utiliza para crear un servicio / complemento / descuento / adicional, etc.

Dentro de la definición incluimos:

- Información sobre el producto (características, inclusiones, T&Cs, duración del contrato, ícono, etc.) que se muestra al usuario del CRM (Cliente o Personal).
- La lógica empresarial sobre quién puede comprar el producto (*Empresarial* o *Residencial*), si depende de tener un servicio principal provisionado (como complementos móviles solo disponibles para clientes con un servicio móvil), si puede ser ordenado directamente por un cliente a través del autoservicio o solo por un agente de servicio al cliente, y cuándo se puede comprar el producto (Permitiendo que un producto esté disponible solo por un tiempo determinado).
- Cuando se deben incluir elementos de Inventario (como Módems o Tarjetas SIM) estos se especifican como Lista de Elementos de Inventario, por ejemplo, el siguiente servicio requiere que se asigne una Tarjeta SIM y un Número de Teléfono:  

```
['SIM Card', 'Phone Number']
```

Estos se correlacionan con los *Inventory Items* `<administration_inventory>` definidos en el CRM.
- Referenciar un Playbook de Ansible para provisionar el servicio *Provisioning Play* `<concepts_ansible>` así como las variables que se pasan a Ansible. Estas variables a pasar son mágicas, en el sentido de que pueden ser variables como `service_id` que están definidas por el producto al que estamos agregando, o pueden ser como `ICCID` & `MSISDN` donde hemos seleccionado elementos de inventario que se pasan al asignar el inventario. El agrupamiento se maneja en el play de provisión para contener múltiples servicios, por

ejemplo, un producto de internet, TV y voz agrupados, puede provisionar un servicio para cada uno.

Definiciones de Producto OmniCRM

## Categorías de Productos y Tipos de Servicio

Los productos utilizan dos campos de clasificación para ayudar a organizar y filtrar ofertas:

### Categorías de Productos

El campo `category` controla dónde se muestran los productos en la interfaz de usuario. Los valores comunes incluyen:

- **standalone** - Se muestra como una opción de servicio base al crear un nuevo servicio
- **addon** - Se muestra al agregar a un servicio existente
- **bundle** - Se muestra como una opción de servicio agrupado (provisionado como un complemento a servicios existentes)
- **promo** - Ofertas promocionales especiales

Estas categorías son puramente organizativas y no dictan lo que se provisiona. El comportamiento real de provisión está determinado completamente por el playbook de Ansible referenciado en `provisioning_play`.

Por ejemplo: - Un producto `standalone` típicamente crea un nuevo objeto de servicio - Un producto `addon` o `bundle` se agrega típicamente a un servicio existente - Pero esto depende del implementador que escribe el playbook - podría crear múltiples objetos de servicio a partir de un complemento, o modificar servicios existentes a partir de un producto independiente si es necesario

La categoría simplemente controla el flujo de la interfaz de usuario y dónde los clientes/personal ven la opción del producto.

### Tipos de Servicio

El campo `service_type` categoriza qué tipo de servicio se está proporcionando.

Estos son completamente definidos por el usuario, pero los valores comunes incluyen:

- **mobile** - Servicios de teléfono móvil con voz, SMS y datos
- **fixed** - Servicios de internet fijos inalámbricos o por cable
- **fixed-voice** - Servicios de voz de línea fija (VoIP, línea terrestre)
- **hotspot** - Dispositivos móviles de hotspot o de alquiler
- **dongle** - Servicios de módem USB o dongle
- **voice** - Servicios solo de voz
- **data** - Servicios solo de datos

Al igual que las categorías, los tipos de servicio son personalizables según sus ofertas. Ayudan en:

- Filtrar qué complementos se aplican a qué servicios base
- Organizar productos en el portal del cliente
- Coincidir con los requisitos de inventario
- Determinar flujos de trabajo de provisión

Ejemplo: Un cliente con un servicio `mobile` puede ver complementos móviles, mientras que un cliente con un servicio `fixed` ve complementos de línea fija.

# Gestión de Productos

Los productos se gestionan a través de la página de Gestión de Productos, donde puede ver, buscar, filtrar y editar todos los productos disponibles.

Página de Lista de Productos

## Interfaz Modal de Producto

Hacer clic en cualquier producto abre una interfaz de pestañas mejorada que organiza todas las configuraciones del producto en grupos lógicos para una navegación y edición más fáciles.

Modal de Producto - Pestaña de Información Básica

El modal de gestión de productos presenta cinco pestañas organizadas:

1. **Información Básica** - Información central del producto (nombre, slug, categoría, ícono, características, términos)
2. **Precios** - Todos los campos relacionados con costos, incluidos costos recurrentes, costos de configuración y porcentaje de impuestos
3. **Configuración** - Configuraciones de renovación, tipos de clientes y dependencias
4. **Provisión** - Configuración del playbook de Ansible y requisitos de inventario
5. **Disponibilidad** - Rangos de fechas y marcas de tiempo del sistema

Modal de Producto - Pestaña de Precios

### Organización de la Pestaña de Precios:

La pestaña de precios agrupa los campos de costo en secciones lógicas:

- **Costos Recurrentes** - Costos minoristas y mayoristas mensuales uno al lado del otro
- **Costos de Configuración** - Tarifas de activación únicas para minoristas y mayoristas
- **Impuesto** - Configuración del porcentaje de impuestos con cálculo automático

### Características del Modo de Edición:

- **Selector de Íconos** - Buscar y seleccionar íconos de FontAwesome visualmente
- **Selector de Elementos de Inventario** - Seleccionar entre tipos de elementos de inventario disponibles
- **Selector de Fecha/Hora** - Selección fácil de ventanas de disponibilidad
- **Formato de Moneda** - Prefijo automático \$ para campos de costo
- **Selectores Desplegables** - Opciones predefinidas para categorías y campos booleanos

Modal de Producto - Modo de Edición de Información Básica

### Selector de Íconos:

Al editar el campo de ícono, aparece una interfaz de selección de íconos buscable que le permite navegar visualmente y seleccionar entre miles de íconos de FontAwesome.

Modal de Producto - Selector de Íconos

Características: \* Buscar íconos por palabra clave (por ejemplo, "llave", "móvil", "wifi") \* Previsualizar la apariencia del ícono en tiempo real \* Muestra el nombre de la clase del ícono como referencia \* Selección desplegable para acceso rápido

### Pestaña de Configuración:

La pestaña de configuración organiza las configuraciones del comportamiento del producto en grupos lógicos.

Modal de Producto - Pestaña de Configuración

### **Secciones de Configuración:**

- **Configuraciones de Renovación:**
  - Auto Renovar - Comportamiento de renovación predeterminado (Preguntar/Sí/No)
  - Permitir Auto Renovar - Si los clientes pueden habilitar la renovación automática
  - Días del Contrato - Duración mínima del contrato (por ejemplo, 30 para mensual, 365 para anual)
- **Tipos de Clientes:**
  - Residencial - Disponible para clientes consumidores
  - Empresarial - Disponible para clientes comerciales
- **Dependencias:**
  - Depende de la Lista - IDs de productos o tipos de servicio requeridos antes de que este producto pueda ser agregado
  - Usado para dependencias de complementos (por ejemplo, los complementos móviles requieren un servicio móvil activo)

### **Pestaña de Provisión:**

La pestaña de provisión maneja la automatización de Ansible y los requisitos de inventario.

Modal de Producto - Pestaña de Provisión

### **Campos de Provisión:**

- **Play de Provisión:**
  - Nombre del playbook de Ansible (sin extensión .yaml)
  - Debe existir en el directorio OmniCRM-API/Provisioners/plays/
  - Se llama cuando se crea, actualiza o desprovista el servicio
- **Variables JSON de Provisión:**
  - Variables predeterminadas pasadas al playbook de Ansible como JSON
  - Pueden ser sobreescritas durante la provisión
  - El playbook recibe estas más customer\_id, product\_id, service\_id, access\_token
- **Lista de Elementos de Inventario:**
  - Selector de múltiples opciones que muestra tipos de elementos de inventario disponibles
  - Ejemplos: Tarjeta SIM, Número de Teléfono, Módem Router, Dirección IPv4
  - El cliente/personal selecciona elementos específicos del inventario disponible durante el pedido
  - Los IDs de inventario seleccionados se pasan al playbook con el tipo de inventario como nombre de variable

### **Pestaña de Disponibilidad:**

La pestaña de disponibilidad controla cuándo se puede comprar el producto y muestra metadatos del sistema.

Modal de Producto - Pestaña de Disponibilidad

### **Configuraciones de Disponibilidad:**

- **Disponible Desde:**
  - Fecha/hora cuando el producto se vuelve disponible para la compra
  - Dejar vacío para disponibilidad inmediata

- Útil para preanunciar nuevos productos
- **Disponible Hasta:**
  - Fecha/hora cuando el producto ya no está disponible para la compra
  - Dejar vacío para disponibilidad indefinida
  - Perfecto para promociones por tiempo limitado o productos al final de su vida útil
- **Metadatos del Sistema (Solo Lectura):**
  - Creado - Marca de tiempo cuando se creó el producto por primera vez
  - Última Modificación - Marca de tiempo de la actualización más reciente
  - Mantenido automáticamente por el sistema

## Acciones del Modal:

- **Modo de Vista:**
  - Cerrar - Descartar modal
  - Clonar Producto - Crear una copia con el sufijo "\_clone"
  - Editar Producto - Cambiar a modo de edición
- **Modo de Edición/Creación:**
  - Cancelar - Descartar cambios y cerrar
  - Guardar Cambios - Crear o actualizar producto (botón grande para énfasis)

## Campos del Producto

El modelo de Producto contiene toda la información necesaria para definir una oferta y cómo debe ser provisionada. Estos campos se gestionan a través de la interfaz modal de gestión de productos descrita anteriormente.

### Información Básica

- **product\_id** - Identificador único asignado automáticamente por el sistema
- **product\_name** - Nombre que se muestra a los clientes y al personal en la interfaz de usuario
- **product\_slug** - Identificador único utilizado en URLs y llamadas API (minúsculas, sin espacios, usar guiones)
- **category** - Controla dónde aparece este producto en la interfaz de usuario:
  - **standalone** - Se muestra como una opción de servicio base al crear un nuevo servicio
  - **addon** - Se muestra al agregar a un servicio existente
  - **bundle** - Se muestra como una opción de servicio agrupado
  - **promo** - Ofertas promocionales especiales
- **service\_type** - Tipo de servicio que se proporciona (por ejemplo, móvil, fijo, voz fija, hotspot, dongle, voz, datos). Se utiliza para filtrar qué complementos se aplican a qué servicios.
- **comment** - Notas internas sobre el producto para referencia del personal solamente (no se muestra a los clientes)
- **icon** - Clase de ícono de FontAwesome mostrada en la interfaz de usuario (por ejemplo, `fa-solid fa-sim-card`)

### Campos de Precios

- **retail\_cost** - Cargo recurrente mensual facturado al cliente (establecer en 0 para compras únicas o productos prepagos)
- **wholesale\_cost** - Su costo mensual por proporcionar este servicio (utilizado para cálculos de margen)
- **retail\_setup\_cost** - Tarifa de activación o configuración única cobrada al cliente
- **wholesale\_setup\_cost** - Su costo único por configurar el servicio
- **tax\_percentage** - Porcentaje de impuestos aplicado a este producto (por ejemplo, 10 para 10%, 12.5 para 12.5%). Establecer en 0 para productos exentos de impuestos. Esta tasa de

impuestos se aplica automáticamente a las transacciones creadas a partir de este producto.

## Configuración de Impuestos del Producto

### Aplicación de Impuestos:

Cuando se crea una transacción a partir de este producto, el porcentaje de impuestos se copia automáticamente a la transacción y se calcula el monto del impuesto. Por ejemplo:

- Producto con 10% de impuestos, \$50.00 costo minorista → La transacción tiene \$5.00 de impuestos
- Producto con 0% de impuestos (exento de impuestos) → La transacción tiene \$0.00 de impuestos
- Sobrescritura manual de transacción → El personal puede cambiar el porcentaje de impuestos por transacción

## Visibilidad y Acceso del Cliente

- **enabled** - Si este producto está activo y disponible para compra (establecer en falso para ocultar sin eliminar)
- **residential** - Si los clientes residenciales (consumidores) pueden comprar este producto
- **business** - Si los clientes comerciales pueden comprar este producto
- **customer\_can\_purchase** - Si los clientes pueden auto-comprar a través del portal (verdadero) o si solo el personal puede agregarlo (falso)
- **available\_from** - Fecha/hora cuando este producto se vuelve disponible para compra (opcional)
- **available\_until** - Fecha/hora cuando este producto ya no está disponible para compra (opcional, útil para ofertas por tiempo limitado)

## Contrato y Renovación

- **contract\_days** - Duración mínima del contrato en días (por ejemplo, 30 para mensual, 365 para anual, 0 para sin contrato mínimo)
- **auto\_renew** - Comportamiento de renovación predeterminado:
  - **prompt** - Pregunta al cliente cada vez si desea renovar
  - **true** - Renueva automáticamente sin preguntar
  - **false** - Requiere renovación manual
- **allow\_auto\_renew** - Si los clientes pueden habilitar la renovación automática (establecer en falso para compras únicas)

## Contenido Visible para el Cliente

- **terms** - Términos y condiciones mostrados a los clientes antes de la compra (incluir limitaciones, reglas de caducidad, condiciones de uso)
- **features\_list** - Lista de características e inclusiones mostradas a los clientes (formato de lista de Python: ['Feature 1', 'Feature 2'])

## Configuración de Provisión

- **provisioning\_play** - Nombre del playbook de Ansible que provisona este servicio (sin extensión .yaml). Debe existir en `OmniCRM-API/Provisioners/plays/`.
- **provisioning\_json\_vars** - Variables predeterminadas pasadas al playbook de Ansible como JSON. Estas pueden ser sobrescritas al provisionar. El playbook recibe estas junto con `customer_id`, `product_id`, `service_id`, y `access_token`.
- **inventory\_items\_list** - Lista de elementos de inventario requeridos para este producto (por ejemplo, ['SIM Card', 'Mobile Number']). Cuando un cliente ordena, se le pedirá que seleccione elementos específicos del inventario disponible. Los IDs de inventario

- seleccionados se pasan al playbook con el tipo de inventario como el nombre de la variable.
- **relies\_on\_list** - Lista de IDs de productos o tipos de servicio que deben existir antes de que este producto pueda ser agregado. Se utiliza para dependencias de complementos (por ejemplo, los complementos móviles requieren un servicio móvil activo).

## Metadatos del Sistema

- **created** - Marca de tiempo cuando se creó el producto (establecido automáticamente)
- **last\_modified** - Marca de tiempo cuando se actualizó el producto por última vez (actualizado automáticamente)

# Ejemplos de Definiciones de Productos

## Producto Independiente (SIM Móvil)

Definiciones de Producto OmniCRM

```
{
  "product_id": 1,
  "product_slug": "Mobile-SIM",
  "product_name": "Mobile SIM Only",
  "category": "standalone",
  "service_type": "mobile",
  "provisioning_play": "play_psim_only",
  "provisioning_json_vars": "{\"iccid\": \"\", \"msisdn\": \"\"}",
  "inventory_items_list": "[\"SIM Card\", 'Mobile Number']",
  "retail_cost": 0,
  "retail_setup_cost": 0,
  "wholesale_cost": 3,
  "wholesale_setup_cost": 1,
  "contract_days": 0,
  "residential": true,
  "business": true,
  "enabled": true,
  "customer_can_purchase": true,
  "icon": "fa-solid fa-sim-card",
  "features_list": "[\"Australian Phone Number (04xxx)\", 'Fastest speeds', 'Best coverage', 'Roaming on the Mainland']",
  "terms": "Must be activated within 6 months. All credit lost if service is not used for 12 months.",
  "comment": "Physical SIM card for use with Mobile Phones"
}
```

Este producto independiente requiere dos elementos de inventario (Tarjeta SIM y Número Móvil) y crea un nuevo servicio cuando se provisiona.

## Producto Complementario (Plan de Datos Mensual)

```
{
  "product_slug": "norfone-mobile-prepaid-mini",
  "product_name": "Norfone Mini Plan",
  "category": "addon",
  "service_type": "mobile",
  "provisioning_play": "play_topup_charge_then_action",
  "provisioning_json_vars": "",
  "inventory_items_list": "[]",
  "retail_cost": 30,
```

```

"retail_setup_cost": 0,
"wholesale_cost": 5.84,
"contract_days": 30,
"residential": true,
"business": false,
"enabled": true,
"customer_can_purchase": true,
"auto_renew": "prompt",
"icon": "fa-solid fa-sim-card",
"features_list": ["'8GB of Ultra fast data', 'Unlimited Calls & Texts to Norfone users', '100 Minutes of Talk to Australia', '100 SMS to Australia', '30 Day Expiry']",
"terms": "Credit expires after 30 days. Once data, voice or sms is used up, you will need to top up to continue using the service.",
"comment": "Our smallest plan for light users"
}

```

Este producto complementario no requiere inventario y se aplica a un servicio existente. Cobra al cliente y agrega créditos/saldos a su servicio.

## Producto Agrupado (Paquete para Personas Mayores)

```

{
  "product_slug": "Bundle-Seniors",
  "product_name": "Seniors Bundle",
  "category": "bundle",
  "service_type": "fixed",
  "provisioning_play": "play_seniors_package",
  "provisioning_json_vars": "{\"IPTV_Service_ID\": \"SeniorBundle\"}",
  "inventory_items_list": ["'Modem Router']",
  "retail_cost": 30,
  "retail_setup_cost": 0,
  "wholesale_cost": 10,
  "wholesale_setup_cost": 11,
  "contract_days": 180,
  "residential": true,
  "business": false,
  "enabled": true,
  "icon": "fa-solid fa-person-walking-with-cane",
  "features_list": ["'20Mbps Download', '5Mbps Upload', 'Unlimited Data', 'Home Voice', 'TV: Extra +£5 per month', '£60 Installation Fee']",
  "terms": "6 Month Contract, must show senior citizen's card to qualify",
  "comment": "20Mbps/2Mbps GPON Service + IPTV + Phone"
}

```

Este producto agrupado provisóna múltiples servicios (Internet + IPTV + Teléfono) utilizando un solo playbook. Requiere un elemento de inventario (Módem Router).

## Producto Complementario (Recarga Simple)

```

{
  "product_slug": "Mobile-Topup-5",
  "product_name": "PAYG £5 Topup",
  "category": "addon",
  "service_type": "mobile",
  "provisioning_play": "play_topup_monetary",
  "provisioning_json_vars": "{\"service_id\": \"\"}",
  "inventory_items_list": []
}

```

```

"retail_cost": 5,
"retail_setup_cost": 0,
"wholesale_cost": 0,
"contract_days": 0,
"residential": true,
"business": false,
"enabled": true,
"customer_can_purchase": true,
"icon": "fa-solid fa-coins",
"features_list": "[‘£5 credit’, ‘Valid for 180 days’]",
"terms": "Valid for 180 days or until all credit is used. See our website for full rates",
"comment": "£5 to use for Calls, SMS & Data"
}

```

Este complemento simplemente agrega crédito monetario a un servicio existente. No se requiere inventario, y utiliza `service_id` para identificar qué servicio recargar.

## Cómo se Pasan las Variables a Ansible

Entender cómo fluyen las variables desde la definición del producto a través de la API hasta el playbook de Ansible es crítico para escribir playbooks de provisión efectivos.

### Fuentes de Variables y Fusión

Cuando se crea un trabajo de provisión, las variables provienen de múltiples fuentes y se fusionan en este orden (las fuentes posteriores sobrescriben las anteriores):

1. **Variables de provisioning\_json\_vars del Producto** - Variables predeterminadas de la definición del producto
2. **Cuerpo de la Solicitud** - Variables pasadas en la llamada API (pueden sobrescribir los valores predeterminados del producto)
3. **Variables Agregadas por el Sistema** - Agregadas automáticamente por el sistema de provisión
4. **Selecciones de Inventario** - IDs de elementos de inventario seleccionados (si `inventory_items_list` no está vacío)

### Proceso de Fusión de Variables

El sistema fusiona variables de todas las fuentes, con las fuentes posteriores sobrescribiendo las anteriores. Esto permite una personalización flexible en el momento de la provisión.

Por ejemplo, si su producto tiene:

```
"provisioning_json_vars": "{\"monthly_cost\": 50, \"data_gb\": 100}"
```

Y su solicitud API incluye:

```
{
  "product_id": 10,
  "customer_id": 456,
  "monthly_cost": 45,
  "custom_param": "value"
}
```

Las `extra_vars` finales pasadas a Ansible serán:

```
{
  "monthly_cost": 45,           # Sobrescrito de la solicitud
  "data_gb": 100,               # De provisioning_json_vars
  "product_id": 10,              # De la solicitud
  "customer_id": 456,             # De la solicitud
  "custom_param": "value",       # De la solicitud
  "access_token": "eyJ..."       # Agregado por el sistema
}
```

## Variables Agregadas por el Sistema

El sistema de provisión agrega automáticamente:

- `access_token` - Token JWT para autenticar llamadas API de regreso al CRM (de `g.access_token` para autenticación de IP/API key, o generado a partir de `refresh_token` para autenticación de usuario)
- `initiating_user` - El ID de usuario que activó la provisión (o el primer administrador para sistemas automatizados)
- Cualquier campo del cuerpo de la solicitud (`product_id`, `customer_id`, `service_id`, etc.)

## Variables de Inventario

Cuando un producto requiere elementos de inventario (por ejemplo, `inventory_items_list: "['SIM Card', 'Mobile Number']"`), el proceso funciona de la siguiente manera:

1. **Prompts de UI/API para selección** - El usuario selecciona elementos de inventario específicos del stock disponible
2. **IDs de Inventario se agregan a las variables** - Los IDs de los elementos de inventario seleccionados se agregan con el tipo de inventario como el nombre de la variable
3. **El playbook accede a los IDs de inventario** - El playbook de provisión puede recuperar los detalles completos del inventario desde la API del CRM

Por ejemplo, si un usuario selecciona: - Tarjeta SIM con `inventory_id: 789` - Número Móvil con `inventory_id: 101`

Las variables pasadas al playbook incluyen: - `SIM Card: 789` - `Mobile Number: 101`

El playbook puede usar estos IDs para obtener los registros completos de inventario (ICCID, IMSI, MSISDN, etc.) desde la API del CRM y usar esa información para provisionar el servicio en el equipo de red.

## Cómo Ansible Recibe Variables

El sistema de provisión pasa todas las variables fusionadas al playbook de Ansible como `extravars`. Dentro del playbook, estas variables están disponibles a través del sistema estándar de variables de Ansible y pueden ser utilizadas en tareas.

Las variables pueden ser referenciadas directamente en las tareas del playbook usando la sintaxis `{{ variable_name }}`. Por ejemplo, `{{ product_id }}`, `{{ customer_id }}`, `{{ monthly_cost }}`, etc.

## Variables Pasadas a Productos Complementarios

Cuando se provisiona un producto complementario, el sistema pasa automáticamente:

- `product_id` - El ID del producto complementario que se está provisionando
- `customer_id` - El cliente que posee el servicio

- `service_id` - El ID del servicio al que se está agregando este complemento (crítico para complementos)
- `access_token` - Token de autenticación para llamadas API
- Cualquier variable de `provisioning_json_vars`
- Cualquier variable adicional de la solicitud API

## Ejemplo de Flujo de Provisión de Complementos

Cuando un cliente agrega el complemento "Recarga de £5" a su servicio móvil (`service_id`: 123), el playbook recibe variables que incluyen:

- `product_id`: 45 (el producto de recarga)
- `customer_id`: 456 (el cliente)
- `service_id`: 123 (el servicio al que agregar crédito)
- `access_token`: Token de autenticación
- Más cualquier variable de `provisioning_json_vars` del producto

El playbook luego utiliza estas variables para:

1. **Obtener detalles del servicio** desde la API del CRM usando el `service_id`
2. **Extraer el UUID del servicio** y otra información del registro del servicio
3. **Agregar crédito al sistema de facturación** (OCS) usando el UUID del servicio
4. **Registrar la transacción** en el CRM para fines de facturación

Este flujo permite que el complemento identifique exactamente qué servicio modificar y aplicar los cambios de manera apropiada.

## Diferencia entre Variables de Productos Independientes y Complementarios

**Productos Independientes** reciben:

- `product_id` - El producto que se está provisionando
- `customer_id` - El cliente que ordena el servicio
- IDs de elementos de inventario (por ejemplo, SIM Card, Mobile Number) si el producto los requiere
- `access_token` - Para autenticación API

**Productos Complementarios** reciben:

- `product_id` - El producto complementario que se está provisionando
- `customer_id` - El cliente que posee el servicio
- `service_id` - **El ID del servicio existente a modificar** (esta es la clave de diferencia)
- `access_token` - Para autenticación API

La clave diferencia es `service_id` - esto le dice al playbook qué servicio existente modificar o agregar.

## Productos Agrupados

Los productos agrupados se provisionan como complementos, pero su playbook puede crear múltiples registros de servicio. Reciben las mismas variables que los complementos, incluyendo:

- `product_id` - El producto agrupado
- `customer_id` - El cliente
- `service_id` - Servicio principal (si corresponde)
- IDs de elementos de inventario (por ejemplo, Modem Router) si se requieren
- `access_token` - Para autenticación API

El playbook de agrupamiento (por ejemplo, `play_seniors_package`) luego crea múltiples servicios relacionados (Internet, IPTV, Teléfono) y los vincula entre sí.

## Servicios

Un servicio es una instancia de un producto que pertenece a un cliente, por la que se le factura.

Es esencialmente un enlace a una cuenta de OCS </glossary> (Sistema de Facturación En Línea) que maneja la generación de cargos y los saldos y usos reales de la cuenta. El OCS es impulsado por CGRateS y gestiona saldos monetarios, saldos unitarios (datos, voz, SMS), Planes de Acción para renovación automática y ThresholdS para límites de gasto.

## Agregar un Servicio: Selección y Filtrado de Productos

Al agregar un servicio a un cliente (ya sea un nuevo servicio independiente o un complemento a un servicio existente), el sistema muestra los productos disponibles en una interfaz de carrusel. Los productos mostrados se filtran en función de varios criterios:

### Filtrado de Productos para Servicios Independientes

Al crear un nuevo servicio para un cliente, la interfaz de usuario muestra productos filtrados por:

1. **Tipo de Cliente** - Los productos se categorizan como:
  - **Individual (Residencial)**: Productos donde `residential = true` o `business = false`
  - **Empresarial**: Productos donde `business = true`
2. **Categoría** - Los productos se separan en:
  - **Planes de Servicio**: Productos con `category = standalone` o `bundle`
  - **Complementos**: Productos con `category = addon` (mostrados en un carrusel separado)
3. **Disponibilidad** - Los productos solo se muestran si:
  - `enabled = true` - El producto está activo y no deshabilitado
  - La fecha actual está entre `available_from` y `available_until` - El producto está dentro de su ventana de disponibilidad
  - `customer_can_purchase = true` (si el cliente se está auto-comprando) - El producto permite la compra directa por parte del cliente

Nota

**Filtrado a Nivel de API:** La API filtra automáticamente los productos por estado habilitado y fechas de disponibilidad en dos niveles:

- **Puntos de Finalización de Compra/Selección** (`/crm/product/`) - Usado por el modal de Complementos y PlanList para la selección de productos. Filtra automáticamente para mostrar SOLO productos habilitados dentro de su rango de fechas de disponibilidad. Esto asegura que los clientes y el personal solo puedan seleccionar productos que están actualmente disponibles para la compra.
- **Puntos de Finalización de Gestión** (`/crm/product/paginated`) - Usado por la página de Gestión de Productos. Muestra TODOS los productos, incluidos los deshabilitados y fuera de las fechas de disponibilidad, permitiendo a los administradores gestionar el catálogo completo de productos, incluidos los productos inactivos.

Pase `include_disabled=true` al punto final base del producto para omitir el filtrado (solo para uso administrativo).

La interfaz de usuario muestra carruseles separados para:

- **Planes de Servicio Individuales** - Productos residenciales para clientes consumidores
- **Planes de Servicio Empresariales** - Productos comerciales para clientes empresariales
- **Complementos Individuales** - Paquetes de complementos residenciales
- **Complementos Empresariales** - Paquetes de complementos comerciales

## Filtrado de Productos para Servicios Complementarios

Al agregar un complemento a un **servicio existente**, se aplica un filtrado adicional:

1. **Coincidencia de Tipo de Servicio** - Solo se muestran complementos con `service_type` coincidente:
  - Si el servicio existente tiene `service_type = "mobile"`, solo se muestran complementos con `service_type = "mobile"`
  - Esto asegura que los clientes móviles solo vean complementos móviles, los clientes de internet solo vean complementos de internet, etc.
2. **Verificación de Dependencias** - Si un complemento tiene una `relies_on_list`:
  - El sistema verifica si el cliente tiene los productos/servicios requeridos
  - Solo se muestran los complementos cuyas dependencias están satisfechas
3. **Filtro de Mismo Tipo de Cliente** - Los complementos aún se filtran por `residential` frente a `business` para coincidir con el tipo de cliente

## Ejemplo de Escenario de Filtrado

Para un cliente empresarial con un servicio móvil existente (`service_type = "mobile"`):

- **Productos Independientes Mostrados:** Todos los productos independientes/agrupados empresariales (`business = true, category != "addon"`)
- **Productos Complementarios Mostrados:** Solo complementos móviles empresariales (`business = true, category = "addon", service_type = "mobile"`)
- **Productos Ocultos:** Productos residenciales, complementos para otros tipos de servicio (internet, voz, etc.), productos deshabilitados

## Campos del Servicio

El modelo de Servicio contiene campos que rastrean la instancia del servicio provisionado y su relación con el cliente, producto y sistema de facturación.

### Información Básica del Servicio

- **service\_id** - Identificador único asignado automáticamente por el sistema (solo lectura)
- **customer\_id** - Enlace al cliente que posee este servicio (solo lectura después de la creación)
- **product\_id** - Enlace al producto del que se creó este servicio (solo lectura después de la creación)
- **service\_name** - Nombre que se muestra a los clientes (editable)
- **service\_type** - Tipo de servicio: móvil, internet, voip, iptv, agrupado, etc. (editable)
- **service\_uuid** - Identificador único utilizado en OCS/CGRateS para facturación (solo lectura, auto-generado)
- **icon** - Clase de ícono de FontAwesome para mostrar en el portal de autoservicio (editable)

### Estado y Fechas del Servicio

- **service\_status** - Estado actual: Activo, Inactivo, Suspendido, etc. (editable)
- **service\_provisioned\_date** - Cuando se provisionó el servicio por primera vez (establecido automáticamente, solo lectura)
- **service\_active\_date** - Cuando el servicio se volvió activo (editable)
- **service\_deactivate\_date** - Cuando el servicio expira o será desactivado (editable)

- **contract\_end\_date** - Fecha de finalización del compromiso del contrato (editable)

## Facturación y Precios

- **retail\_cost** - Cargo recurrente mensual al cliente (editable)
- **wholesale\_cost** - Su costo por proporcionar el servicio (editable)
- **service\_billed** - Si este servicio aparece en las facturas (editable, predeterminado: verdadero)
- **service\_taxable** - Si se aplican impuestos a este servicio (editable, predeterminado: verdadero)
- **invoiced** - Si el servicio ha sido facturado (establecido automáticamente por el sistema de facturación)
- **promo\_code** - Código promocional utilizado cuando se creó el servicio (editable)

## Visibilidad del Cliente

- **service\_visible\_to\_customer** - Si el cliente puede ver este servicio en el portal de autoservicio (editable, predeterminado: verdadero)
- **service\_usage\_visible\_to\_customer** - Si el cliente puede ver detalles de uso/saldo (editable, predeterminado: verdadero)

## Configuración de Provisión

- **provisioning\_play** - Playbook de Ansible utilizado para provisionar este servicio (heredado del producto, solo lectura)
- **provisioning\_json\_vars** - Variables pasadas al playbook de provisión (heredadas del producto, solo lectura)
- **deprovisioning\_play** - Playbook de Ansible que se ejecuta cuando se desprovisiona el servicio (solo lectura)
- **deprovisioning\_json\_vars** - Variables para el playbook de desprovisión (solo lectura)

## Relaciones del Servicio

- **bundled\_parent** - Si este servicio es parte de un paquete, el service\_id del servicio principal (solo lectura)
- **site\_id** - Enlace al sitio/ubicación física donde se entrega el servicio (editable)

## Notas y Metadatos

- **service\_notes** - Notas internas sobre el servicio para referencia del personal (editable)
- **created** - Marca de tiempo cuando se creó el servicio (establecido automáticamente, solo lectura)
- **last\_modified** - Marca de tiempo de la última actualización (actualizado automáticamente, solo lectura)

## Campos Editables vs Solo Lectura

### Editables a través de API/UI:

Los servicios pueden ser actualizados a través de PATCH /crm/service/{service\_id} con estos campos:

- service\_name, service\_type, service\_status
- service\_notes
- retail\_cost, wholesale\_cost
- service\_billed, service\_taxable
- service\_visible\_to\_customer, service\_usage\_visible\_to\_customer
- service\_active\_date, service\_deactivate\_date, contract\_end\_date

- icon, promo\_code, site\_id

### **Solo Lectura (Auto-Establecidos):**

Estos campos no pueden ser modificados directamente después de la creación:

- service\_id, customer\_id, product\_id
- service\_uuid (generado durante la provisión)
- service\_provisioned\_date
- provisioning\_play, provisioning\_json\_vars
- deprovisioning\_play, deprovisioning\_json\_vars
- bundled\_parent
- invoiced (gestionado por el sistema de facturación)
- created, last\_modified (gestionados automáticamente)

## **Provisión de Productos en Servicios**

El proceso de provisión convierte un Producto (plantilla) en un Servicio (instancia específica del cliente) a través de una serie de pasos coordinados que involucran la interfaz web, API y playbooks de Ansible.

### **Flujo de Provisión de Alto Nivel**

1. **Configuración de Pre-Provisión** - Producto creado en la API con configuración de provisión, y los correspondientes playbooks de Ansible escritos y probados
2. **Selección de Servicio** - Desde la Página del Cliente, el personal o el cliente selecciona "Agregar Servicio"
3. **Filtrado de Productos** - Los productos mostrados se filtran en función de:
  - Tipo de cliente (residencial/empresarial)
  - Servicios existentes (para dependencias de complementos en `relies_on_list`)
  - Fechas de disponibilidad (`available_from/available_until`)
  - Flags `enabled` y `customer_can_purchase`
4. **Personalización** - Opción para sobrescribir variables de provisión (para ajustes de precios, configuraciones personalizadas, etc.)
5. **Selección de Inventario** - Si el producto requiere inventario (`inventory_items_list` no está vacío), el usuario selecciona elementos específicos (por ejemplo, qué tarjeta SIM, qué número de teléfono)
6. **Inicio de Provisión** - Cuando se hace clic en el botón "Provisionar", la API crea un trabajo de provisión

### **Flujo Detallado de Integración de API y Ansible**

Cuando se provisiona un servicio, ocurre la siguiente secuencia:

#### **Paso 1: Creación del Trabajo de Provisión (/routes/service.py)**

La API recibe la solicitud de provisión y llama a `create_provisioning_job()` desde `services/provisioning_service.py` con:

- `provisioning_play` - Nombre del playbook de Ansible (por ejemplo, `play_psim_only`)
- `provisioning_json_vars` - Cadena JSON de variables del producto o sobrescritas por la solicitud
- `customer_id` - ID del cliente que ordena el servicio
- `product_id` - ID del producto que se está provisionando
- `service_id` - (Opcional) ID del servicio existente para complementos
- Selecciones de Inventario - IDs de elementos de inventario seleccionados

## Paso 2: Ensamblaje de Variables (services/provisioning\_service.py)

El servicio de provisión fusiona variables de múltiples fuentes en este orden:

1. provisioning\_json\_vars del Producto (valores predeterminados de la definición del producto)
2. Parámetros del cuerpo de la solicitud (pueden sobrescribir los valores predeterminados del producto)
3. Variables agregadas por el sistema:
  - access\_token - Token JWT para autenticación API de regreso al CRM
  - initiating\_user - ID de usuario que activó la provisión
  - customer\_id, product\_id, service\_id
4. Selecciones de Inventario - Agregadas como pares {inventory\_type: inventory\_id}

Ejemplo de variables fusionadas:

```
{  
    "customer_id": 123,  
    "product_id": 456,  
    "service_id": 789,          # Solo para complementos  
    "SIM Card": 1001,          # De la selección de inventario  
    "Mobile Number": 1002,     # De la selección de inventario  
    "monthly_cost": 30,        # De provisioning_json_vars  
    "data_gb": 50,             # De provisioning_json_vars  
    "access_token": "eyJ...:",  # Agregado por el sistema para callbacks API  
    "initiating_user": 5       # Agregado por el sistema  
}
```

## Paso 3: Creación del Registro de Provisión (models.py - Modelo de Provisión)

Se crea un registro de Provision en la base de datos con:

- provision\_id - Identificador único para seguimiento
- provisioning\_play - Nombre del archivo del playbook
- provisioning\_json\_vars - Variables fusionadas como cadena JSON
- task\_count - Número de tareas en el playbook (extraído de YAML)
- provisioning\_status - Código de estado (initialmente establecido en 1 = en ejecución, luego actualizado a 0 = éxito, 2 = fallido, o puede permanecer en 1 si aún está en progreso)
- product\_id, customer\_id, service\_id - Referencias de contexto

## Paso 4: Ejecución en Segundo Plano del Playbook (Provisioners/playbook\_runner\_v2.py)

La API genera un hilo en segundo plano que:

1. Carga el YAML del playbook desde OmniCRM-API/Provisioners/plays/{playbook\_name}.yaml
2. Llama a ansible\_runner.run() con:
  - playbook - Ruta al archivo YAML cargado
  - extravars - Todas las variables fusionadas (pasadas a Ansible)
  - inventory - Establecido en 'localhost,' (ejecución local)
  - event\_handler - Manejador personalizado para capturar eventos de ejecución de tareas
3. Monitorea la ejecución del playbook en tiempo real

## Paso 5: Captura de Eventos y Registro (ProvisioningEventHandler)

A medida que se ejecuta cada tarea de Ansible, se capturan eventos y se almacenan como

registros de Provision\_Event:

- `event_name` - Nombre de la tarea del playbook
  - `event_number` - Número de secuencia
  - `provisioning_status` - Código de estado que indica el resultado de la tarea:
    - **0** = Éxito - La tarea se completó con éxito
    - **1** = En Ejecución - La tarea se está ejecutando actualmente
    - **2** = Fallido - Fallo crítico que detiene la provisión
    - **3** = Fallido (ignorando) - La tarea falló pero se ignoraron los errores  
(`ignore_errors: true` en el playbook)
  - `provisioning_result_json` - Resultados de la tarea con datos sensibles redactados

El manejador de eventos elimina automáticamente contraseñas, claves, secretos y otros datos sensibles de los registros.

## Paso 6: Ejecución del Playbook de Ansible (Provisioners/plays/\*.yaml)

El playbook de Ansible se ejecuta localmente y generalmente realiza estas acciones:

1. **Obtener Definición del Producto** - Solicitud GET a /crm/product/product\_id/{{product\_id }} usando {{ access\_token }}
  2. **Obtener Información del Cliente** - Solicitud GET a /crm/customer/customer\_id/{{customer\_id }}
  3. **Procesar Elementos de Inventario** (si es necesario) - Solicitud GET a /crm/inventory/inventory\_id/{{ inventory\_id }} para cada elemento seleccionado para recuperar detalles completos (ICCID, MSISDN, números de serie, etc.)
  4. **Configurar Sistemas Externos** - Hacer llamadas API a:
    - HSS (Servidor de Suscriptores en el Hogar) para provisión de suscriptores
    - IMS (Subsistema Multimedia IP) para registro de voz
    - CGRateS/OCS para creación de cuentas, configuración de cargos, planes de tarifas
    - Servidores ENUM para mapeo de números de teléfono
    - Equipos de red (enrutadores, conmutadores, etc.)
  5. **Agregar Costos de Configuración** (si corresponde) - POST a /crm/transaction/ para registrar cargos únicos
  6. **Cargar al Cliente** - POST a OCS/CGRateS para cargar retail\_setup\_cost si está configurado
  7. **Crear Cuenta OCS** - POST a OCS/CGRateS para crear una cuenta de facturación con UUID
  8. **Configurar Cargos Recurrentes** - Crear Acciones y Planes de Acción en OCS/CGRateS para cargos recurrentes mensuales
  9. **Crear Registro de Servicio** - PUT/POST a /crm/service/ para crear el registro de servicio en CRM:

```
{  
  "customer_id": 123,  
  "product_id": 456,  
  "service_name": "Mobile SIM - 0412345678",  
  "service_uuid": "generated-uuid-for-ocs",  
  "service_status": "Active",  
}
```

```

    "service_type": "mobile",
    "retail_cost": 30,
    "wholesale_cost": 5,
    "provisioning_play": "play_psim_only",
    "provisioning_json_vars": "..."
}

```

10. **Asignar inventario** - PATCH a /crm/inventory/inventory\_id/{{ inventory\_id }} para marcar el inventario como "Asignado" al servicio
11. **Enviar Notificaciones** (opcional) - Correo electrónico o SMS al cliente con detalles del servicio

#### Paso 7: Finalización y Actualización de Estado

Cuando el playbook se completa:

- **Éxito:** Provision.provisioning\_status actualizado a **0** (Éxito)
- **Fallo Crítico:** Provision.provisioning\_status actualizado a **2** (Fallido), y correo electrónico de fallo enviado a `crm_config.provisioning.failure_list`
- **Fallos No Críticos:** Las tareas que fallan con `ignore_errors: true` se marcan con estado **3** (Fallido pero ignorado) y no detienen la provisión

El servicio provisionado ahora es visible en el CRM y está activo para el cliente (si la provisión tuvo éxito).

### Diferencias Clave: Provisión Independiente vs Complementaria vs Agrupada

#### Productos Independientes (category: standalone):

- Reciben `customer_id` y `product_id`
- Típicamente requieren elementos de inventario (tarjetas SIM, números de teléfono, módems)
- Crean un registro de servicio **nuevo** a través de la API PUT /crm/service/
- Provisionan nuevos recursos en sistemas externos (HSS, OCS, equipos de red)
- Ejemplo: Nueva activación de SIM móvil, nueva conexión a internet

#### Productos Complementarios (category: addon):

- Reciben `customer_id`, `product_id` y **service\_id** (servicio existente a modificar)
- Típicamente NO requieren inventario (o inventario mínimo)
- **Modifican un servicio existente** o agregan cargos a una cuenta OCS existente
- Pueden ejecutar acciones en OCS (agregar paquete de datos, agregar crédito, habilitar función)
- No crean nuevos registros de servicio (o crean registros de servicio secundarios vinculados al principal)
- Ejemplo: Recarga de plan de datos mensual, paquete de roaming internacional, crédito extra

#### Productos Agrupados (category: bundle):

- Similar a los complementos en términos de variables recibidas
- Pueden requerir algunos elementos de inventario (por ejemplo, módem para paquete de hogar)
- Crean **múltiples** registros de servicio relacionados (Internet + TV + Teléfono)
- Provisionan múltiples recursos a través de diferentes sistemas
- Vinculan servicios en CRM para facturación/gestión unificada
- Ejemplo: Paquete de hogar (Internet + IPTV + teléfono VoIP)

## Requisitos del Playbook de Provisión

Para que un playbook funcione correctamente, debe:

1. **Estar ubicado en** OmniCRM-API/Provisioners/plays/{playbook\_name}.yaml
2. **Aceptar variables** a través de extravars de Ansible (accedidas como {{ variable\_name }})
3. **Autenticar llamadas API** usando Authorization: Bearer {{ access\_token }} en el encabezado
4. **Manejar fallos de manera adecuada** usando bloques rescue y ignore\_errors donde sea apropiado
5. **Crear registro de servicio** para productos independientes, o modificar el servicio existente para complementos
6. **Asignar inventario** si se seleccionaron elementos de inventario
7. **Devolver mensajes de error significativos** a través del módulo fail cuando ocurran errores críticos

## Variables Comunes Disponibles en Playbooks

Cada playbook recibe estas variables:

- customer\_id - Entero, cliente que ordena el servicio
- product\_id - Entero, producto que se está provisionando
- service\_id - Entero (solo complementos/agrupados), servicio existente a modificar
- access\_token - Cadena, token JWT para autenticación API de CRM
- initiating\_user - Entero, usuario que activó la provisión
- Más cualquier ID de elemento de inventario: {{ inventory\_type }}: inventory\_id
- Más cualquier variable de provisioning\_json\_vars
- Más cualquier variable pasada en la solicitud de provisión

Los playbooks pueden usar estos para:

- Obtener detalles completos del producto: GET /crm/product/product\_id/{{ product\_id }}
- Obtener detalles del cliente: GET /crm/customer/customer\_id/{{ customer\_id }}
- Obtener detalles del inventario: GET /crm/inventory/inventory\_id/{{ SIM\_Card }}
- Crear transacciones: POST /crm/transaction/
- Crear servicios: PUT /crm/service/
- Actualizar servicios: PATCH /crm/service/{{ service\_id }}
- Asignar inventario: PATCH /crm/inventory/inventory\_id/{{ inventory\_id }}

## Ejemplo: Flujo de Playbook de Complemento Simple

Para un complemento de recarga de datos móviles:

1. El playbook recibe: customer\_id, product\_id, service\_id, access\_token
2. Obtener detalles del servicio: GET /crm/service/{{ service\_id }} para obtener service\_uuid
3. Obtener detalles del producto: GET /crm/product/product\_id/{{ product\_id }} para obtener precios y cantidad de datos
4. Cargar al cliente en OCS: POST a CGRateS para deducir retail\_cost del saldo
5. Agregar crédito de datos en OCS: POST a CGRateS para agregar saldo de datos con fecha de caducidad
6. Registrar la transacción en CRM: POST /crm/transaction/ con detalles de carga
7. Completar con éxito

Todo el proceso se rastrea en las tablas Provision y Provision\_Event para fines de depuración y

auditoría.

## Participación de OCS

OCS (Sistema de Facturación En Línea), implementado a través de CGRateS, maneja toda la facturación en tiempo real y el seguimiento de uso para servicios. El registro de servicio del CRM actúa como un puntero a la cuenta OCS, que gestiona:

- **Cargos recurrentes** - Tarifas mensuales, alquiler de DID, cargos de suscripción
  - **Facturación basada en uso** - Llamadas de voz por minuto, datos por MB, cargos por SMS
  - **Gestión de saldos** - Saldos monetarios (crédito prepago) y saldos unitarios (GB de datos, minutos de voz, conteo de SMS)
  - **Conversión de saldos** - Convertir saldos monetarios en saldos unitarios (por ejemplo, gastar \$30 para obtener un paquete de datos de 10GB)
  - **Estado de la cuenta** - Activo, suspendido, deshabilitado según límites de crédito y umbrales

El registro de servicio del CRM contiene metadatos y configuración (cliente, producto, precios, visibilidad), mientras que OCS contiene el estado de facturación en vivo (saldos, uso, cargos).

## Recuperación de Uso y Saldos del Servicio

La información de uso del servicio se recupera de OCS/CGRateS y se muestra a clientes y personal en tiempo real.

## **Cómo se Recupera el Uso**

Cuando se solicita el uso de un servicio (a través de la interfaz o API), ocurre el siguiente flujo:

1. **Solicitud API** - El frontend llama a GET /crm/service/{service\_id} o ve detalles del servicio en la interfaz
  2. **Búsqueda del Servicio** - La API recupera el registro del servicio de la base de datos, extrae service\_uuid
  3. **Llamadas API de CGRateS** - El módulo cgrates\_service.py realiza dos llamadas a CGRateS:
    1. **Get\_Balance(service\_uuid)** - Recupera el saldo de la cuenta con BalanceMap
      - Devuelve saldos organizados por tipo: DATOS, VOZ, SMS, MONETARIO, DATA\_DONGLE
      - Cada saldo incluye: ID, Valor, Fecha de Expiración, Peso, DestinationIDs
      - El sistema agrega campos legibles por humanos: custom\_Name\_hr, custom\_Expiration, custom\_Description\_String
    2. **Get\_ActionPlans(service\_uuid)** - Recupera planes de acción de renovación automática activos (cubiertos en la siguiente sección)
  4. **Edición de Recursos** - Los datos de CGRateS se sincronizan con este del servicio

```
"service_id": 123,  
"service_name": "Mobile Service",  
"service_uuid": "abc-123-def",  
"cgbrates": {  
    "BalanceMap": {  
        "DATA": [{
```

```

        "ID": "DATA_10GB",
        "Value": 5368709120,
        "ExpirationDate": "2025-02-01T00:00:00Z",
        "custom_Name_hr": "10GB Data Pack",
        "custom_Expiration": "Feb 1, 2025",
        "custom_Description_String": "5 GB remaining"
    ],
    "VOICE": [
        {
            "ID": "VOICE_UNLIMITED",
            "Value": 99999999,
            "custom_Name_hr": "Unlimited Calls",
            "custom_Description_String": "Unlimited minutes"
        }
    ],
    "MONETARY": [
        {
            "ID": "PREPAID_CREDIT",
            "Value": 25.50,
            "custom_Description_String": "$25.50 credit"
        }
    ],
    "ActionPlans": [...]
}
}

```

5. **Visualización en la Interfaz** - Los componentes del frontend muestran los datos de uso:

- **ServiceUsage.js** - Componente principal de visualización de uso con actualización automática cada 3 segundos
- **UsageCard.js** - Tarjetas de resumen para cada tipo de saldo
- **UsageProgress.js** - Barras de progreso que muestran el porcentaje usado/restante
- Los saldos están codificados por colores y formateados para facilitar la lectura

## Estructura de Datos de Uso

Cada saldo en el BalanceMap contiene:

### Campos Nativos de CGRateS:

- **ID** - Identificador único para el saldo (por ejemplo, "DATA\_10GB\_2025\_01")
- **Value** - Monto del saldo:
  - Para DATOS: bytes (5368709120 = 5 GB)
  - Para VOZ: segundos (3600 = 1 hora)
  - Para SMS: conteo (100 = 100 mensajes)
  - Para MONETARIO: unidades de moneda (25.50 = \$25.50)
- **ExpirationDate** - Marca de tiempo ISO 8601 cuando expira el saldo
- **Weight** - Prioridad para el consumo del saldo (peso más alto consumido primero)
- **DestinationIDs** - Destinos a los que se aplica este saldo (por ejemplo, ["AU", "INTERNATIONAL"])

### Campos Legibles por Humanos (agregados por CRM):

- **custom\_Name\_hr** - Nombre legible por humanos extraído del ID
- **custom\_Expiration** - Fecha de expiración formateada (por ejemplo, "15 de enero de 2025" o "en 11 días")
- **custom\_Description\_String** - Descripción del saldo legible por humanos:
  - DATOS: "5 GB restantes" o "10 GB en total"
  - VOZ: "60 minutos restantes" o "Ilimitado"
  - SMS: "50 SMS restantes"
  - MONETARIO: "\$25.50 de crédito"

## Control de Visibilidad del Uso

La visibilidad del uso del servicio se controla mediante dos campos:

- **service\_visible\_to\_customer** - Si es falso, el servicio se oculta por completo del portal de autoservicio del cliente
- **service\_usage\_visible\_to\_customer** - Si es falso, el servicio es visible pero los detalles de uso/saldo están ocultos (el cliente puede ver que tiene el servicio, pero no cuánto ha usado)

Esto permite a los operadores:

- Ocultar servicios internos/de prueba de los clientes
- Mostrar que el servicio existe sin revelar el uso (útil para planes ilimitados o servicios sensibles)
- Visualización completamente transparente del uso (predeterminado)

## Actualizaciones de Uso en Tiempo Real

La interfaz web actualiza automáticamente los datos de uso:

- **Intervalo:** Cada 3 segundos (configurable en el componente ServiceUsage)
- **Método:** Realiza una consulta GET /crm/service/{service\_id} que obtiene datos en vivo de CGRateS
- **Eficiencia:** Solo las vistas de servicios activos se actualizan; las vistas de lista utilizan datos en caché

Esto asegura que los clientes y el personal vean actualizaciones de saldo casi en tiempo real a medida que ocurre el uso.

## Cargos Recurrentes / AutoRenovación

Los cargos recurrentes, como un Cargo de Servicio Mensual, o un Cargo por DID, se crean primero como Acciones dentro de OCS y toman el formato Action\_ServiceUUID\_ServiceName\_WhatitDoes.

Por ejemplo, para un servicio GPON de \$60 por mes que incluye 1TB de uso, la Acción se vería algo así:

Action\_kj49-adsf-1234-9742\_60\_GPON\_1TB\_MonthlyExpiry

1. Restablecer Saldo Monetario a \$0
2. Enviar un POST HTTP a /simple\_provision en el CRM para provisionar algo
3. Agregar un Crédito por 1TB de Uso que expira en 1 mes

Si queremos que el MRC sea recurrente (lo hacemos), entonces creariamos un ActionPlan llamado ActionPlan\_{{ service\_uuid }}\_Monthly\_Charge que tendría el tiempo establecido en *mensual* para activarse cada mes, y asignariamos el ActionPlan a la cuenta.

Podemos establecer, según el parámetro *Año / Meses / Días*, una fecha de caducidad para cuándo se detendrá el MRC también, por ejemplo, para un servicio fijo de 12 meses que se detuvo después de este punto.

Dado que las Acciones y ActionPlans son únicas para el servicio, no comparten nada con otros servicios.

Esto significa que podemos provisionarlos con valores ajustados, y no impactará a ningún otro servicio.

## Complementos y Adicionales

Los complementos / Adicionales, como comprar datos extra, paquetes de roaming, minutos internacionales, etc., se manejan de manera muy similar. Se crea una Acción para hacer lo que se necesita, como cobrar un valor monetario y luego otorgar un saldo unitario con una fecha de caducidad establecida.

En lugar de usar ActionPlans para agregar esto automáticamente a la cuenta, simplemente activamos ExecuteAction para la Acción que acabamos de crear una vez desde dentro de Ansible.

## Agregar Saldos Monetarios Prepagados

Para saldos monetarios prepagados, como un plan PAYG de \$10, esto se agrega como un saldo monetario, pero con una prioridad más alta.

El límite de crédito en estos servicios para el saldo predeterminado sería \$0.

## Límites de Crédito / Prevención de Gastos Excesivos

ThresholdS se utilizan en cada cuenta para establecer el gasto máximo para un período de tiempo dado.

Para clientes PAYG / Prepagados, esto es \$0.

## Interactuando con OCS a través del CRM

Para cada Servicio, puede ver los Balances y ActionPlans, Actions y ThresholdS de OCS desde dentro de la API del CRM.

Los ActionPlans pueden ser eliminados según sea necesario desde la API del CRM, ejecutados a través de Playbooks de Ansible. Los ActionPlans pueden ser agregados según sea necesario, desde el CRM, agregando un Complemento/Servicio y ejecutados a través de Playbooks de Ansible.

Las cuentas de OCS pueden ser deshabilitadas, lo que detendrá la ejecución de ActionPlans y de los servicios que puedan ser consumidos.

Para Límites de Crédito, se establece un valor de ThresholdS según la política del producto.

## Visualización y Gestión de ActionPlans en el CRM

Los ActionPlans (configuraciones de renovación automática) se muestran y gestionan a través de la interfaz del CRM, permitiendo al personal y a los clientes ver las próximas renovaciones automáticas y gestionarlas.

## Cómo se Recuperan y Muestran los ActionPlans

Al ver un servicio en el CRM, los ActionPlans se recuperan y muestran automáticamente:

1. **Llamada API** - Cuando se llama a GET /crm/service/{service\_id}, la API:

- Recupera el registro del servicio de la base de datos
- Extrae el service\_uuid (identificador de cuenta OCS)
- Llama a get\_cgrates\_action\_plans\_by\_service\_uuid(service\_uuid) desde

- cgrates\_service.py
- Esto llama internamente a `ocs.Get_ActionPlans(service_uuid)` para recuperar ActionPlans de CGRateS

## 2. Estructura de Datos de ActionPlan

- Cada ActionPlan devuelto contiene:

```
{
  "ActionPlanId": "ServiceID_abc-123-def__ProductID_456_...",
  "PlanName": "Monthly_Renewal_Plan",
  "NextExecTime": "2025-02-01T00:00:00Z",
  "custom_NextExecTime_hr": "in 11 days",
  "ActionPlanId_split_dict": {
    "ServiceID": "abc-123-def",
    "ProductID": 456,
    "CustomerID": 789,
    ...
  }
}
```

- ActionPlanId - Identificador único que contiene información codificada sobre el servicio/producto/cliente
- PlanName - Nombre del plan de acción (típicamente el nombre del playbook de renovación)
- NextExecTime - Marca de tiempo ISO cuando se ejecutará el ActionPlan a continuación
- custom\_NextExecTime\_hr - Tiempo legible por humanos hasta la ejecución (por ejemplo, "en 11 días", "mañana", "1 de febrero de 2025")
- ActionPlanId\_split\_dict - Diccionario con componentes analizados del ActionPlanId

## 3. Visualización en la Interfaz

- Los ActionPlans se muestran en el componente **ActionPlansTable**:

### Columnas de la Tabla:

- Nombre del Producto** - Recuperado buscando ProductID del ActionPlanId
- Costo** - Muestra `retail_cost` de la definición del producto
- Fecha de Renovación** - Muestra `custom_NextExecTime_hr` (legible por humanos)
- Acciones** - Dos botones:
  - Renovar Ahora** - Provisionar inmediatamente el complemento/renovación (salta la espera de la ejecución programada)
  - Eliminar Renovación Automática** - Cancela la renovación automática

### Cuando No Existen ActionPlans:

- La tabla muestra el mensaje: "No hay renovación automática habilitada para este servicio"
- El cliente puede agregar complementos de renovación automática para habilitar la renovación automática

## Gestión de ActionPlans

El personal y los clientes pueden gestionar ActionPlans a través de la interfaz:

### Eliminando un ActionPlan (Cancelando la Renovación Automática):

- Haga clic en el botón "Eliminar Renovación Automática" en ActionPlansTable
- Aparece un modal de confirmación: "¿Está seguro de que desea eliminar esta renovación

- automática?"
3. Al confirmar, el frontend llama a: `DELETE /crm/oam/remove_action_plan/{action_plan_id}`
  4. La API elimina el ActionPlan de CGRateS a través de `ocs.Remove_ActionPlan()`
  5. La actividad se registra: "Eliminado ActionPlan {ActionPlanId} del servicio {service\_id}"
  6. El ActionPlan desaparece de la tabla

### **Renovando Inmediatamente (Renovación Manual):**

1. Haga clic en el botón "Renovar Ahora" en ActionPlansTable
2. Aparece un modal de confirmación: "¿Está seguro de que desea renovar esto ahora?"
3. Al confirmar, el sistema:
  - Extrae `product_id` del `ActionPlanId`
  - Crea un nuevo trabajo de provisión para ese producto
  - Provisiona el complemento inmediatamente (ejecuta el playbook de provisión)
  - El servicio recibe los beneficios del complemento sin esperar a la renovación programada
4. El modal de estado de provisión muestra el progreso
5. Al éxito, los saldos se actualizan inmediatamente

### **Agregando Renovación Automática:**

La renovación automática se habilita al provisionar un producto complementario que tiene `auto_renew` establecido:

- Productos con `auto_renew = "true"` - Crean automáticamente ActionPlans durante la provisión
- Productos con `auto_renew = "prompt"` - Preguntan al cliente si desea renovación automática (diálogo modal)
- Productos con `auto_renew = "false"` - Nunca crean ActionPlans (compra única)

El playbook de provisión crea el ActionPlan en CGRateS con:

- `ActionPlanId` único que codifica los IDs de servicio, producto y cliente
- Programa de renovación (mensual, anual, intervalo personalizado)
- Acción a ejecutar (típicamente reprovisionar el mismo complemento)
- Fecha de caducidad (si el contrato tiene un término fijo)

### **Convención de Nombres de ActionPlan**

Los ActionPlans siguen una convención de nombres estandarizada para codificar metadatos:

#### **Formato:**

`ServiceID_{service_uuid}__ProductID_{product_id}__CustomerID_{customer_id}__...`

#### **Ejemplo:**

`ServiceID_abc-123-def__ProductID_456__CustomerID_789__MonthlyRenewal`

Esta codificación permite al CRM:

- Identificar a qué servicio pertenece el ActionPlan
- Buscar detalles del producto (nombre, precios) para mostrar
- Rastre



# Sistema de Aprovisionamiento

OmniCRM utiliza **Ansible** para automatizar el aprovisionamiento, la configuración y la desprovisión de los servicios al cliente. El sistema de aprovisionamiento está diseñado para ser flexible, permitiendo flujos de trabajo complejos mientras mantiene la consistencia y la fiabilidad.

Eventos de Aprovisionamiento Recientes

Nota

Para una guía completa del viaje del producto al servicio con ejemplos detallados de playbooks de Ansible, estrategias de precios y escenarios del mundo real, consulte [Complete Product Lifecycle Guide <guide\\_product\\_lifecycle>](#).

## Descripción General

Cuando se pide un producto o se necesita configurar un servicio, OmniCRM crea un **Trabajo de Aprovisionamiento** que ejecuta uno o más playbooks de Ansible. Estos playbooks interactúan con varios sistemas de backend (OCS/CGRateS, equipos de red, APIs, etc.) para aprovisionar completamente el servicio.

El sistema de aprovisionamiento admite dos flujos de trabajo principales:

1. **Aprovisionamiento Estándar** - Activado por el personal o los clientes a través de la UI/API
2. **Aprovisionamiento Simple** - Activado por sistemas externos como OCS para operaciones automatizadas

## Valores de Estado de Aprovisionamiento

Los trabajos de aprovisionamiento y las tareas individuales pueden tener los siguientes estados:

- **Estado 0 (Éxito)** - El trabajo de aprovisionamiento se completó con éxito
- **Estado 1 (En Ejecución)** - El trabajo de aprovisionamiento o la tarea se están ejecutando actualmente
- **Estado 2 (Fallido - Crítico)** - Ocurrió un fallo crítico que causó que el aprovisionamiento fallara
- **Estado 3 (Fallido - Ignorado)** - Una tarea falló pero tenía `ignore_errors: true`, por lo que el aprovisionamiento continuó

Cuando un trabajo de aprovisionamiento falla, OmniCRM envía notificaciones por

correo electrónico a la lista de notificación de fallos configurada con información detallada sobre el error.

## Cómo los Productos Impulsan el Aprovisionamiento

La definición de **Producto** es el plano de lo que se aprovisiona y cómo. Cuando un usuario selecciona un producto para aprovisionar, el sistema lee varios campos clave de la definición del producto para determinar qué hacer.

### Campos del Producto Utilizados en el Aprovisionamiento

Una definición de producto contiene:

- `provisioning_play` - El nombre del playbook de Ansible a ejecutar (sin la extensión `.yaml`)
- `provisioning_json_vars` - Cadena JSON que contiene variables predeterminadas para pasar a Ansible
- `inventory_items_list` - Lista de tipos de inventario que deben ser asignados (por ejemplo, `['SIM Card', 'Mobile Number']`)
- `product_id`, `product_name`, campos de precios - Pasados automáticamente al playbook

### Ejemplo de Definición de Producto

```
{  
  "product_id": 1,  
  "product_slug": "Mobile-SIM",  
  "product_name": "Mobile SIM Only",  
  "provisioning_play": "play_psim_only",  
  "provisioning_json_vars": "{\"iccid\": \"\", \"msisdn\": \"\"}",  
  "inventory_items_list": "[\"SIM Card\", \"Mobile Number\"]",  
  "retail_cost": 0,  
  "retail_setup_cost": 0,  
  "wholesale_cost": 3,  
  "wholesale_setup_cost": 1  
}
```

### Del Producto al Trabajo de Aprovisionamiento

Cuando se inicia el aprovisionamiento, el sistema:

1. **Carga el playbook** especificado en `provisioning_play`

El sistema busca `OmniCRM-API/Provisioners/plays/play_psim_only.yaml`

## 2. **Fusiona variables** de múltiples fuentes en `extra_vars`:

1. **Desde provisioning\_json\_vars**: `{"iccid": "", "msisdn": ""}`
2. **Desde el cuerpo de la solicitud**: Cualquier variable adicional que el usuario/API proporcione
3. **Desde campos del producto**: `product_id`, `customer_id`, etc.
4. **Desde autenticación**: `access_token` o configuración para `refresh_token`

## 3. **Asigna inventario** basado en `inventory_items_list`

Antes de ejecutar el playbook, la UI/API solicita la selección de inventario:

- **SIM Card** - El usuario selecciona una SIM disponible del inventario
- **Mobile Number** - El usuario selecciona un número de teléfono disponible

Los IDs de inventario seleccionados se añaden a `extra_vars` con el tipo de inventario como clave:

```
extra_vars = {
    "product_id": 1,
    "customer_id": 456,
    "SIM Card": 789,           # inventory_id de la SIM
seleccionada
    "Mobile Number": 101,     # inventory_id del número de
teléfono seleccionado
    "iccid": "",             # Desde provisioning_json_vars
    "msisdn": "",             # Desde provisioning_json_vars
    "access_token": "eyJ..."
```

## 4. **Pasa todo a Ansible** a través de `hostvars[inventory_hostname]`

Dentro del playbook, las variables son accesibles como:

```
- name: Get inventory_id for SIM Card
  set_fact:
    inventory_id_sim_card: "{{ hostvars[inventory_hostname]['SIM
Card'] | int }}"
  when: "'SIM Card' in hostvars[inventory_hostname]"
```

# Cómo los Playbooks Usan Variables de Inventario

Una vez que el playbook tiene los IDs de inventario, recupera los detalles completos del inventario desde la API:

```
- name: Get SIM Card Details from Inventory
  uri:
```

```

  url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ inventory_id_sim_card }}"
  method: GET
  headers:
    Authorization: "Bearer {{ access_token }}"
  return_content: yes
  register: sim_card_response

- name: Extract ICCID and IMSI from inventory
  set_fact:
    iccid: "{{ sim_card_response.json.iccid }}"
    imsi: "{{ sim_card_response.json.imsi }}"

- name: Get Phone Number Details from Inventory
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ inventory_id_phone_number }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
    return_content: yes
  register: phone_number_response

- name: Extract MSISDN
  set_fact:
    msisdn: "{{ phone_number_response.json.msisdn }}"

```

El playbook puede luego usar estos valores para:

- Aprovisionar la tarjeta SIM en el HSS con el IMSI
- Configurar el número de teléfono en el sistema de facturación
- Asignar los elementos de inventario al cliente
- Crear el registro de servicio con estos detalles

## Ejemplo del Mundo Real: Aprovisionamiento de SIM Móvil

Desde `play_psim_only.yaml`, así es como utiliza los datos del producto y del inventario:

```

- name: Get Product information from CRM API
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/product/product_id/{{ product_id }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
    return_content: yes
  register: api_response_product

```

```
- name: Set package facts from product
  set_fact:
    package_name: "{{ api_response_product.json.product_name }}"
    package_comment: "{{ api_response_product.json.comment }}"
    setup_cost: "{{ api_response_product.json.retail_setup_cost }}"
    monthly_cost: "{{ api_response_product.json.retail_cost }}"

- name: Set inventory_id_sim_card if SIM Card was selected
  set_fact:
    inventory_id_sim_card: "{{ hostvars[inventory_hostname]['SIM
Card'] | int }}"
  when: "'SIM Card' in hostvars[inventory_hostname]"

- name: Set inventory_id_phone_number if Mobile Number was selected
  set_fact:
    inventory_id_phone_number: "{{ hostvars[inventory_hostname]['Mobile Number'] | int }}"
  when: "'Mobile Number' in hostvars[inventory_hostname]"

- name: Get SIM Card details from inventory
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ inventory_id_sim_card }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
    return_content: yes
  register: sim_inventory_response

- name: Get Phone Number details from inventory
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ inventory_id_phone_number }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
    return_content: yes
  register: phone_inventory_response

- name: Extract values from inventory
  set_fact:
    iccid: "{{ sim_inventory_response.json.iccid }}"
    imsi: "{{ sim_inventory_response.json.imsi }}"
    msisdn: "{{ phone_inventory_response.json.msisdn }}"
    ki: "{{ sim_inventory_response.json.ki }}"
    opc: "{{ sim_inventory_response.json.opc }}"
```

```

- name: Provision subscriber on HSS
  uri:
    url: "http://{{ hss_server }}/subscriber/{{ imsi }}"
  method: PUT
  body_format: json
  body:
    {
      "imsi": "{{ imsi }}",
      "msisdn": "{{ msisdn }}",
      "ki": "{{ ki }}",
      "opc": "{{ opc }}",
      "enabled": true
    }
  status_code: 200

- name: Assign inventory to customer
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ inventory_id_sim_card }}"
  method: PATCH
  headers:
    Authorization: "Bearer {{ access_token }}"
  body_format: json
  body:
    {
      "customer_id": {{ customer_id }},
      "item_state": "Assigned"
    }
  status_code: 200

```

Esto demuestra el flujo completo:

1. La definición del producto especifica `provisioning_play`:  
"play\_psim\_only"
2. El producto requiere `inventory_items_list`: ['SIM Card', 'Mobile Number']
3. El usuario selecciona elementos de inventario durante el aprovisionamiento
4. Los IDs de inventario se pasan al playbook como `extra_vars`
5. El playbook recupera los detalles completos del inventario desde la API
6. El playbook utiliza los datos del inventario para configurar el equipo de red
7. El playbook marca el inventario como asignado al cliente

## Rollback y Limpieza: Patrón de Mejores Prácticas

**Mejor Práctica Crítica:** El mismo playbook debe manejar tanto el rollback de aprovisionamiento fallido como la desprovisión intencionada utilizando la estructura `block` y `rescue` de Ansible.

## Estructura del Playbook

Desde play\_psim\_only.yaml:

```
- name: OmniCore Service Provisioning 2024
  hosts: localhost
  gather_facts: no
  become: False

  tasks:
    - name: Main block
      block:
        # --- TAREAS DE APROVISIONAMIENTO ---
        - name: Get Product information
          uri: ...

        - name: Create account in OCS
          uri: ...

        - name: Provision subscriber on HSS
          uri: ...

        - name: Create service record
          uri: ...

        # ... muchas más tareas de aprovisionamiento ...

  rescue:
    # --- TAREAS DE LIMPIEZA ---
    # Esta sección se ejecuta cuando:
    # 1. Cualquier tarea en el bloque falla (rollback)
    # 2. action == "deprovision" (limpieza intencionada)

    - name: Get Inventory items linked to this service
      uri:
        url: "{{ crm_config.crm.base_url }}/crm/inventory/
customer_id/{{ customer_id }}"
        method: GET
      register: inventory_api_response
      ignore_errors: True

    - name: Return inventory to pool
      uri:
        url: "{{ crm_config.crm.base_url }}/crm/inventory/
inventory_id/{{ item.inventory_id }}"
        method: PATCH
      body_format: json
      body:
```

```

        service_id: null
        customer_id: null
        item_state: "Used"
with_items: "{{ inventory_api_response.json.data }}"
ignore_errors: True

- name: Delete Account from Charging
  uri:
    url: "http://{{ crm_config.ocs.0CS }}/jsonrpc"
  method: POST
  body:
    {
      "method": "ApierV1.RemoveAccount",
      "params": [
        {
          "Tenant": "{{ crm_config.ocs.ocsTenant }}",
          "Account": "{{ service_uuid }}"
        }
      ]
    }
  ignore_errors: True

- name: Delete Attribute Profile
  uri:
    url: "http://{{ crm_config.ocs.0CS }}/jsonrpc"
  method: POST
  body:
    {
      "method": "APIerSv1.RemoveAttributeProfile",
      "params": [
        {
          "ID": "ATTR_ACCOUNT_{{ service_uuid }}"
        }
      ]
    }
  ignore_errors: True

- name: Remove Resource Profile
  uri: ...
  ignore_errors: True

- name: Remove Filters
  uri: ...
  ignore_errors: True

- name: Deprovision Subscriber from HSS
  uri:
    url: "{{ item.key }}/subscriber/{{ item.value.subscriber_id }}"
    method: DELETE
  loop: "{{ hss_subscriber_data | dict2items }}"
  ignore_errors: True

```

```

when:
  - deprovision_subscriber | bool == true

  - name: Patch Subscriber to Dormant State
    uri:
      url: "{{ item.key }}/subscriber/{{ item.value.subscriber_id }}"
    method: PATCH
    body:
      {
        "enabled": true,
        "msisdn": "9999{{ imsi[-10:] }}", # Número ficticio
        "ue_ambr_dl": 999999, # Inusualmente
alto
        "ue_ambr_ul": 999999
      }
    loop: "{{ hss_subscriber_data | dict2items }}"
    when:
      - deprovision_subscriber | default(false) | bool == false

    # La afirmación final determina el éxito o el fracaso
    - name: Set status to "Success" if Manual deprovision / Fail
if failed provision
  assert:
    that:
      - action == "deprovision"

```

## Por qué Este Patrón es una Mejores Prácticas

### 1. Sin Duplicación de Código

Las mismas tareas de limpieza manejan ambos escenarios:

- **Fallo de Aprovisionamiento (Rollback):** Si alguna tarea en el block falla, la sección rescue se ejecuta automáticamente
- **Desprovisión Intencionada:** Cuando se llama con `action: "deprovision"`, el playbook salta inmediatamente a rescue

### 2. Limpieza Completa Garantizada

Cuando un aprovisionamiento falla a mitad de camino, la sección de rescate asegura:

- Se eliminan todas las cuentas de OCS creadas
- Se eliminan todas las entradas de equipos de red configurados
- Se devuelve el inventario asignado al pool
- Se eliminan o se establecen en estado inactivo los suscriptores de HSS
- No queda aprovisionamiento parcial en ningún sistema

Esto previene recursos "huérfanos" que:

- Consumir inventario sin ser rastreados
- Crear cuentas de facturación que no están vinculadas a servicios
- Causar confusión durante la resolución de problemas
- Desperdiciar recursos de red

### 3. Manejo de Fallos Elegante con ignore\_errors

Observe que cada tarea de limpieza utiliza `ignore_errors: True`. Esto es intencional porque:

- Durante el rollback, algunos recursos pueden no haberse creado aún
- Queremos intentar todas las tareas de limpieza incluso si algunas fallan
- La afirmación final determina el éxito/fallo general

Por ejemplo, si el aprovisionamiento falla en "Crear cuenta en OCS", la limpieza intentará:

- Eliminar la cuenta de OCS (fallará, pero se ignorará)
- Eliminar perfiles de atributos (fallará, pero se ignorará)
- Devolver el inventario (tendrá éxito)
- Eliminar el suscriptor de HSS (puede que no exista, se ignorará)

### 4. Distinguir Desprovisión de Rollback

La afirmación final al final de rescue es ingeniosa:

```
- name: Set status to "Success" if Manual deprovision / Fail if failed provision
  assert:
    that:
      - action == "deprovision"
```

Esto significa:

- **Si ``action == "deprovision"``:** La afirmación pasa, el playbook tiene éxito (estado 0)
- **Si ``action`` no está establecido o != "deprovision":** La afirmación falla, el playbook falla (estado 2)

Así que el mismo código de limpieza resulta en diferentes estados de trabajo de aprovisionamiento dependiendo de la intención.

### 5. Limpieza Condicional Basada en el Tipo de Servicio

Algunas tareas de limpieza utilizan condicionales para manejar diferentes escenarios:

```

- name: Deprovision Subscriber from HSS
  uri: ...
  when:
    - deprovision_subscriber | bool == true

- name: Patch Subscriber to Dormant State
  uri: ...
  when:
    - deprovision_subscriber | default(false) | bool == false

```

Esto permite una limpieza flexible:

- **Eliminación completa:** Cuando las SIM están dedicadas a clientes (deprovision\_subscriber: true)
- **Estado inactivo:** Cuando las SIM son reutilizables y deben permanecer en HSS (deprovision\_subscriber: false)

## Cómo Usar Este Patrón

### Para Aprovisionamiento:

```
{
  "product_id": 1,
  "customer_id": 456,
  "provisioning_play": "play_psim_only"
}
```

Si el aprovisionamiento falla, se produce un rollback automático a través de rescue.

### Para Desaprovisionamiento:

```
{
  "service_id": 123,
  "service_uuid": "Service_abc123",
  "action": "deprovision",
  "provisioning_play": "play_psim_only"
}
```

El playbook salta directamente a la sección rescue, ejecuta toda la limpieza y tiene éxito.

## Resumen de Beneficios

- ◊ **Fuente única de verdad:** Un playbook maneja aprovisionamiento y desaprovisionamiento
- ◊ **Operaciones atómicas:** O completamente aprovisionado o completamente limpiado
- ◊ **Sin recursos huérfanos:** Aprovisionamientos fallidos no dejan rastro
- ◊ **Mantenimiento más fácil:** Cambios en la lógica de

aprovisionamiento se aplican automáticamente a la limpieza ◊ **Errores reducidos**: Sin posibilidad de que el código de aprovisionamiento y desprovisión se desincronice ◊ **Probable**: Se puede probar la lógica de desprovisión ejecutando con `action: "deprovision"`

Este patrón debe seguirse en todos los playbooks de aprovisionamiento para garantizar fiabilidad y consistencia.

## Sobrescribir Variables del Producto

Las `provisioning_json_vars` pueden ser sobrescritas en el momento del aprovisionamiento. Por ejemplo, un producto podría definir:

```
{  
  "provisioning_json_vars": "{\"monthly_cost\": 50,  
  \"data_limit_gb\": 100}"  
}
```

Pero al aprovisionar, puede sobrescribir estas:

```
{  
  "product_id": 1,  
  "customer_id": 456,  
  "monthly_cost": 45,  
  "data_limit_gb": 150  
}
```

Las `extra_vars` fusionadas utilizarán los valores sobrescritos. Esto permite:

- Precios personalizados para clientes específicos
- Diferentes límites de datos según promociones
- Pruebas con diferentes parámetros sin modificar el producto

## Productos Sin Inventario

No todos los productos requieren inventario. Por ejemplo, un complemento de datos o un interruptor de función podrían tener:

```
{  
  "product_id": 10,  
  "product_name": "Extra 10GB Data",  
  "provisioning_play": "play_local_data_addon",  
  "provisioning_json_vars": "{\"data_gb\": 10}",  
  "inventory_items_list": []  
}
```

En este caso, el playbook recibe:

```

extra_vars = {
    "product_id": 10,
    "customer_id": 456,
    "service_id": 123, # Servicio al que se agrega datos
    "data_gb": 10,
    "access_token": "eyJ... "
}

```

El playbook simplemente agrega los datos al servicio existente sin necesidad de elementos de inventario.

## Flujo de Trabajo de Aprovisionamiento Estándar

El aprovisionamiento estándar se inicia cuando:

- Un miembro del personal agrega un servicio a un cliente desde la UI
- Un cliente pide un servicio a través del portal de autoservicio
- La API se llama directamente con PUT /crm/provision/

### Cuando Haces Clic en "Aprovisionar"

Aquí está el flujo completo que ocurre cuando un usuario hace clic en el botón "Aprovisionar":

#### 1. La UI Muestra la Selección de Producto

El usuario selecciona un producto del catálogo de productos. El producto contiene:

- provisioning\_play - Qué playbook de Ansible ejecutar
- inventory\_items\_list - Inventario requerido (por ejemplo, ['SIM Card', 'Mobile Number'])
- provisioning\_json\_vars - Variables predeterminadas

#### 2. Selector de Inventario (Si es Necesario)

Si inventory\_items\_list no está vacío, aparece un modal que muestra menús desplegables para cada tipo de inventario. El usuario debe seleccionar elementos de inventario disponibles antes de continuar.

#### 3. Botón de Aprovisionamiento Clicado

JavaScript envía una solicitud PUT /crm/provision/:

```

PUT /crm/provision/
Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...
Content-Type: application/json

```

```
{
  "product_id": 42,
  "customer_id": 123,
  "SIM Card": 5001,
  "Mobile Number": 5002
}
```

## 4. La API Recibe la Solicitud

El endpoint de aprovisionamiento (routes/provisioning.py):

- Valida la autenticación (token Bearer, clave API o lista blanca de IP)
- Verifica que el usuario tenga permiso CREATE\_PROVISION
- Extrae initiating\_user del token
- Carga la definición del producto desde la base de datos
- Recupera la ruta del playbook: OmniCRM-API/Provisioners/plays/play\_psim\_only.yaml

## 5. Variables Fusionadas

El sistema combina variables de múltiples fuentes:

```
# Desde el producto
product_vars = json.loads(product['provisioning_json_vars'])
# Desde el cuerpo de la solicitud
request_vars = request.json
# Agregado por el sistema
system_vars = {
    'product_id': 42,
    'customer_id': 123,
    'access_token': g.access_token, # Ver sección de autenticación a
continuación
    'initiating_user': 7
}
# Final fusionado
extra_vars = {**product_vars, **request_vars, **system_vars}
```

## 6. Registro de Aprovisionamiento Creado

Registro en la base de datos creado con estado 1 (En Ejecución):

```
provision = {
    'provision_id': 456,
    'customer_id': 123,
    'product_id': 42,
    'provisioning_play': 'play_psim_only',
    'provisioning_json_vars': json.dumps(extra_vars),
    'provisioning_status': 1, # En Ejecución
    'task_count': 85,
```

```
        "initiating_user": 7,
        "created": "2025-01-10T14:30:00Z"
    }
```

## 7. Hilo en Segundo Plano Generado

```
run_playbook_in_background(
    playbook='plays/play_psim_only.yaml',
    extra_vars=extra_vars,
    provision_id=456,
    refresh_token=refresh_token # Para la renovación de token
durante la ejecución
)
```

## 8. La API Devuelve Inmediatamente

Respuesta devuelta a la UI con provision\_id:

```
{
    "provision_id": 456,
    "provisioning_status": 1,
    "message": "Provisioning job created"
}
```

## 9. La UI Polls para Actualizaciones

La UI comienza a consultar GET /crm/provision/provision\_id/456 cada 3 segundos para verificar el estado. La respuesta incluye:

```
{
    "provision_id": 456,
    "provisioning_status": 1,
    "task_count": 12,
    "provisioning_result_json": [
        {
            "event_number": 1,
            "event_name": "Get Product information from CRM API",
            "provisioning_status": 0,
            "timestamp": "2024-01-15T10:30:05"
        },
        {
            "event_number": 2,
            "event_name": "Assign SIM Card from inventory",
            "provisioning_status": 1,
            "timestamp": "2024-01-15T10:30:07"
        }
    ]
}
```

## 10. Ansible Ejecuta en Segundo Plano

El playbook ejecuta tareas secuencialmente:

- Cada finalización de tarea crea un registro `Provision_Event` en la base de datos
- El evento incluye: nombre de la tarea, estado (0=éxito, 2=fallido, 3=fallido pero ignorado), resultado JSON
- La UI muestra el progreso en tiempo real mostrando tareas completadas y la tarea actualmente en ejecución
- Las tareas fallidas muestran mensajes de error en los detalles del evento

### Seguimiento en la UI:

Mientras el aprovisionamiento está en ejecución (estado 1), los usuarios pueden ver:

- **Página de Detalles del Servicio** - Muestra la insignia de estado de aprovisionamiento (En Ejecución/Éxito/Fallo)
- **Registro de Actividad** - Enumera todos los eventos de aprovisionamiento con marcas de tiempo
- **Vista de Detalles de Aprovisionamiento** - Muestra el progreso tarea por tarea con expandir/colapsar para detalles

Ejemplo de visualización:

Estado de Aprovisionamiento: En Ejecución (8 de 12 tareas completadas)

✓ Obtener información del producto de la API de CRM ✓ Obtener detalles del cliente ✓ Asignar tarjeta SIM del inventario (ICCID: 8991101200003204510) ✓ Asignar número de teléfono (555-0123) ✕ Crear cuenta en OCS/CGRateS (en progreso...) ✦ Configurar políticas de red ✦ Crear registro de servicio ...

## 11. El Aprovisionamiento Se Completa

Estado final establecido:

- `provisioning_status: 0` - Éxito
- `provisioning_status: 2` - Fallo (error crítico)

La UI deja de consultar y muestra el resultado:

- **Éxito:** Marca de verificación verde, servicio marcado como Activo, el usuario puede ver los detalles del servicio
- **Fallo:** X roja, mensaje de error mostrado, opción para reintentar o contactar soporte
- **Notificación por correo electrónico:** Si hay fallo, se envía un correo electrónico a `provisioning.failure_list` en la configuración

# Autenticación y Autorización

## Seguimiento de Usuarios

Cada trabajo de aprovisionamiento rastrea qué usuario lo inició:

- **Iniciado por el Usuario:** El campo `initiating_user` se establece en la ID del usuario de su token JWT
- **Autenticación por Clave API:** Utiliza la ID del primer usuario administrador
- **Autenticación por Lista Blanca de IP:** Utiliza la ID del primer usuario administrador

## Verificaciones de Permisos

El sistema verifica permisos antes de permitir el aprovisionamiento:

- El personal necesita el permiso `CREATE_PROVISION`
- Los clientes solo pueden aprovisionar servicios para su propia cuenta (`VIEW_OWN_PROVISION` permiso)

## Cómo Ansible se Autentica con la API de CRM

Los playbooks de Ansible necesitan hacer llamadas API autenticadas de vuelta a la CRM (para obtener detalles del producto, crear servicios, actualizar inventario, etc.). La autenticación se maneja a través de **tokens Bearer** pasados al playbook.

La fuente del `access_token` depende del método de autenticación utilizado para llamar a la API de aprovisionamiento:

### Método 1: Inicio de Sesión del Usuario (Token Bearer)

Cuando un usuario inicia sesión a través de la UI web:

1. El usuario se autentica: `POST /crm/auth/login`
2. Recibe un `access_token` JWT (de corta duración, 15-30 min) y un `refresh_token` (de larga duración)
3. Realiza la solicitud de aprovisionamiento con el token Bearer en el encabezado
4. La API de aprovisionamiento extrae el token del encabezado `Authorization: Bearer ...`
5. Se almacena en `g.access_token` (contexto de solicitud de Flask)
6. Se pasa a Ansible como variable `access_token`

### Código (permissions.py):

```
# Extraer token Bearer del encabezado
auth_header = request.headers.get('Authorization', '')
if auth_header.startswith('Bearer '):
```

```

bearer_token = auth_header[7:]
# Validar y decodificar
decoded_token = jwt.decode(bearer_token, secret_key,
algorithms=['HS256'])
# Almacenar para aprovisionamiento
g.access_token = bearer_token

```

### Código (provisioning.py):

```

if "access_token" in g:
    json_data['access_token'] = g.access_token
run_playbook(playbook_path, extra_vars=json_data,
provision_id=provision_id)

```

### Método 2: Clave API (Encabezado X-API-KEY)

Para sistemas automatizados que utilizan claves API:

1. El sistema realiza la solicitud: PUT /crm/provision/ con el encabezado X-API-KEY: your-api-key...
2. La API de aprovisionamiento valida la clave API contra `crm_config.yaml`
3. **Genera un nuevo token JWT sobre la marcha** para el primer usuario administrador
4. Se almacena en `g.access_token`
5. Se pasa a Ansible

### ¿Por qué Generar un Token?

Las claves API son cadenas, no JWTs. Los playbooks llaman a los endpoints de API esperando autenticación JWT. Así que:

- Valida la clave API
- Si es válida y tiene rol de `admin`, genera un JWT temporal
- Utiliza la ID del primer usuario administrador como sujeto del JWT
- El token permite que el playbook realice llamadas API autenticadas

### Código (permissions.py):

```

def handle_api_key_auth(f, api_key, *args, **kwargs):
    if not secure_compare_api_key(api_key):
        return {'message': 'Invalid API key'}, 401

    API_KEYS = yaml_config.get('api_keys', {})
    if api_key in API_KEYS:
        if 'admin' in API_KEYS[api_key].get('roles', []):
            admin_user_id = retrieve_first_admin_user_id()
            access_token =
create_access_token(identity=str(admin_user_id))
            g.access_token = access_token

```

### Método 3: Lista Blanca de IP

Para sistemas internos de confianza en redes privadas:

1. El sistema realiza la solicitud desde una IP en la lista blanca (por ejemplo, 192.168.1.100)
2. La API de aprovisionamiento verifica la IP del cliente contra `ip_whitelist` en `crm_config.yaml`
3. Si está en la lista blanca, **genera un nuevo token JWT** para el primer usuario administrador
4. Se almacena en `g.access_token`
5. Se pasa a Ansible

#### Código (permissions.py):

```
def handle_ip_auth(f, *args, **kwargs):  
    client_ip = get_real_client_ip()  
    if not is_ip_whitelisted(client_ip):  
        return {'message': 'Access denied'}, 403  
  
    admin_user_id = retrieve_first_admin_user_id()  
    access_token = create_access_token(identity=str(admin_user_id))  
    g.access_token = access_token
```

### Usando el Token en los Playbooks

Cada llamada API en el playbook incluye el token:

```
- name: Get Product Details  
  uri:  
    url: "http://localhost:5000/crm/product/product_id/{{ product_id }}"  
  headers:  
    Authorization: "Bearer {{ access_token }}"  
  
- name: Create Service Record  
  uri:  
    url: "http://localhost:5000/crm/service/"  
    method: PUT  
  headers:  
    Authorization: "Bearer {{ access_token }}"  
  body:  
    customer_id: "{{ customer_id }}"  
    service_name: "Mobile Service"
```

### Expiración y Renovación del Token

Los playbooks de larga duración (5-10 minutos) pueden superar el `access_token` (expiración de 15-30 min). Para aprovisionamientos iniciados por el usuario, el

sistema pasa tanto `access_token` como `refresh_token`:

```
refresh_token = request.cookies.get('refresh_token')
run_playbook(playbook_path, extra_vars, provision_id,
refresh_token=refresh_token)
```

Si `access_token` expira, el ejecutor del playbook puede:

1. Detectar respuesta 401 No Autorizado
2. Llamar a POST /crm/auth/refresh con `refresh_token`
3. Recibir un nuevo `access_token`
4. Reintentar la solicitud fallida

Para la autenticación por clave API/lista blanca de IP, los tokens generados pueden tener una expiración más larga (1-2 horas) ya que estos son sistemas automatizados de confianza.

## El Proceso de Aprovisionamiento

### 1. Creación del Trabajo

Cuando se recibe una solicitud de aprovisionamiento, el sistema:

- Valida la solicitud y verifica permisos
- Carga el playbook de Ansible especificado en la definición del producto
- Crea un registro `Provision` en la base de datos con estado 1 (En Ejecución)
- Extrae variables de la definición del producto y del cuerpo de la solicitud
- Captura tokens de autenticación para el acceso a la API

### 2. Manejo de Tokens

Los playbooks de Ansible necesitan autenticarse con la API de CRM para recuperar datos y realizar cambios. El sistema de aprovisionamiento maneja esto de dos maneras:

- **Token Bearer (JWT):** Para aprovisionamientos iniciados por el usuario, el `refresh_token` de la solicitud se utiliza para generar tokens de acceso frescos durante la ejecución del playbook
- **Autenticación por Clave API/IP:** Para sistemas automatizados, se pasa un `access_token` directamente al playbook a través de `g.access_token`

### 3. Ejecución en Segundo Plano

El playbook se ejecuta en un hilo en segundo plano utilizando `playbook_runner_v2`. Esto permite que la API devuelva inmediatamente

mientras el aprovisionamiento continúa de manera asíncrona.

Durante la ejecución:

- Cada finalización/fallo de tarea crea un registro `Provision_Event`
- El controlador de eventos monitorea fallos críticos vs. ignorados
- Se escriben actualizaciones de estado en tiempo real en la base de datos
- La UI puede consultar actualizaciones a través de `GET /crm/provision/provision_id/<id>`

## 4. Ejecución del Playbook

El playbook de Ansible típicamente realiza estas operaciones:

- Recupera información del producto desde la API
- Recupera información del cliente desde la API
- Asigna elementos de inventario (tarjetas SIM, direcciones IP, números de teléfono, etc.)
- Crea cuentas en OCS/OCS
- Configura equipos de red
- Crea el registro de servicio en la API de CRM
- Agrega transacciones de costo de configuración
- Envía correos electrónicos/SMS de bienvenida a los clientes

## 5. Manejo de Errores

Los playbooks de Ansible utilizan secciones `block` y `rescue` para el rollback:

- Si una tarea crítica falla, la sección de rescate elimina el aprovisionamiento parcial
- Las tareas con `ignore_errors: true` se marcan como estado 3 y no fallan el trabajo
- Errores fatales (errores de sintaxis YAML, fallos de conexión) crean un evento de error especial con información de depuración

## Ejemplo: Playbook de Aprovisionamiento Estándar

Aquí hay un ejemplo de `play_simple_service.yaml`:

```
- name: Simple Provisioning Play
  hosts: localhost
  gather_facts: no
  become: False

  tasks:
    - name: Main block
      block:
```

```

- name: Get Product information from CRM API
  uri:
    url: "http://localhost:5000/crm/product/product_id/{{ product_id }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
    return_content: yes
    validate_certs: no
  register: api_response_product

- name: Set package facts
  set_fact:
    package_name: "{{ api_response_product.json.product_name }}"
  }
  setup_cost: "{{ api_response_product.json.retail_setup_cost }}"
  }
  monthly_cost: "{{ api_response_product.json.retail_cost }}"
}

- name: Generate Service UUID
  set_fact:
    service_uuid: "Service_{{ 99999999 | random | to_uuid }}"

- name: Create account in OCS
  uri:
    url: "http://{{ crm_config.ocs.0CS }}/jsonrpc"
    method: POST
    body_format: json
    body:
      {
        "method": "ApierV2.SetAccount",
        "params": [
          {
            "Tenant": "{{ crm_config.ocs.ocsTenant }}",
            "Account": "{{ service_uuid }}",
            "ActionPlanIds": [],
            "ExtraOptions": { "AllowNegative": false,
"Disabled": false }
          }
        ]
      }
    status_code: 200
  register: response

- name: Add Service via API
  uri:
    url: "http://localhost:5000/crm/service/"
    method: PUT
    body_format: json
    headers:

```

```

        Authorization: "Bearer {{ access_token }}"
body:
{
  "customer_id": "{{ customer_id }}",
  "product_id": "{{ product_id }}",
  "service_name": "Service: {{ service_uuid }}",
  "service_uuid": "{{ service_uuid }}",
  "service_status": "Active",
  "retail_cost": "{{ monthly_cost | float }}"
}
status_code: 200
register: service_creation_response

- name: Add Setup Cost Transaction
uri:
  url: "http://localhost:5000/crm/transaction/"
  method: PUT
  headers:
    Authorization: "Bearer {{ access_token }}"
  body_format: json
  body:
{
  "customer_id": {{ customer_id | int }},
  "service_id": {{ service_creation_response.json.service_id | int }},
  "title": "{{ package_name }} - Setup Costs",
  "retail_cost": "{{ setup_cost | float }}"
}
register: api_response_transaction

rescue:
- name: Remove account in OCS
uri:
  url: "http://{{ crm_config.ocs.OCS }}/jsonrpc"
  method: POST
  body_format: json
  body:
{
  "method": "ApierV2.RemoveAccount",
  "params": [
    {
      "Tenant": "{{ crm_config.ocs.ocsTenant }}",
      "Account": "{{ service_uuid }}"
    }
  ]
}
status_code: 200

- name: Fail the provision
assert:

```

```
that:  
- false
```

Este playbook demuestra el flujo típico:

1. Recuperar detalles del producto desde la API de CRM
2. Generar un UUID de servicio único
3. Crear la cuenta de facturación en OCS
4. Crear el registro del servicio a través de la API de CRM
5. Agregar transacciones de costo de configuración
6. Si algo falla, la sección `rescue` elimina la cuenta de OCS

## Flujo de Trabajo de Aprovisionamiento Simple

El aprovisionamiento simple está diseñado para sistemas automatizados que necesitan activar el aprovisionamiento sin interacción del usuario. El caso de uso más común es el aprovisionamiento de complementos activado por OCS a través de ActionPlans.

### Endpoints de Aprovisionamiento Simple

OmniCRM proporciona dos endpoints de aprovisionamiento simple:

- `POST /crm/provision/simple_provision_addon/service_id/<id>/product_id/<id>`

Para aprovisionamiento automatizado de complementos (por ejemplo, cargos recurrentes, recargas automáticas)

- `POST /crm/provision/simple_provision_addon_recharge/service_id/<id>/product_id/<id>`

Para operaciones de recarga rápida que necesitan retroalimentación inmediata

### Autenticación para Aprovisionamiento Simple

Los endpoints de aprovisionamiento simple utilizan **listas blancas de IP o claves API** para la autenticación:

- La IP de origen de la solicitud se verifica contra `ip_whitelist` en `crm_config.yaml`
- O se puede proporcionar una clave API de `api_keys` en `crm_config.yaml`
- Se genera un token de acceso y se pasa al playbook a través de `g.access_token`

## Ejemplo: Callback de ActionPlan de OCS

OCS puede configurarse para llamar al endpoint de aprovisionamiento simple al ejecutar acciones recurrentes:

```
{  
  "method": "ApierV1.SetActionPlan",  
  "params": [  
    {"  
      "Id": "ActionPlan_Service123_Monthly_Charge",  
      "ActionsId": "Action_Service123_Add_Monthly_Data",  
      "Timing": {  
        "Years": [],  
        "Months": [],  
        "MonthDays": [1],  
        "Time": "00:00:00Z"  
      },  
      "Weight": 10,  
      "ActionTriggers": [  
        {  
          "ThresholdType": "*min_event_counter",  
          "ThresholdValue": 1,  
          "ActionsID": "Action_Service123_HTTP_Callback"  
        }  
      ]  
    }]  
}
```

La acción realiza un POST HTTP a:

Esto activa el playbook asociado (por ejemplo, `play_topup_no_charge.yaml`) que agrega datos/créditos al servicio.

## Ejemplo: Playbook de Recarga Simple

Desde `play_topup_monetary.yaml`:

```
- name: Mobile Topup Monetary - 2024  
  hosts: localhost  
  gather_facts: no  
  become: False  
  
  tasks:  
    - name: Get Product information from CRM API  
      uri:  
        url: "http://localhost:5000/crm/product/product_id/{{  
product_id }}"  
        method: GET  
        headers:
```

```

        Authorization: "Bearer {{ access_token }}"
        return_content: yes
        register: api_response_product

- name: Get Service information from CRM API
  uri:
    url: "http://localhost:5000/crm/service/service_id/{{ service_id }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
      return_content: yes
  register: api_response_service

- name: Set service facts
  set_fact:
    service_uuid: "{{ api_response_service.json.service_uuid }}"
    customer_id: "{{ api_response_service.json.customer_id }}"
    package_name: "{{ api_response_product.json.product_name }}"
    monthly_cost: "{{ api_response_product.json.retail_cost }}"

- name: Get Customer Payment Method
  uri:
    url: "http://localhost:5000/crm/stripe/customer_id/{{ customer_id }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
      return_content: yes
  register: api_response_stripe

- name: Charge customer
  uri:
    url: "http://localhost:5000/crm/stripe/charge_card/{{ customer_stripe_id }}"
    method: POST
    headers:
      Authorization: "Bearer {{ access_token }}"
    body_format: json
    body:
      {
        "retail_cost": "{{ monthly_cost }}",
        "description": "{{ package_name }} topup",
        "customer_id": "{{ customer_id | int }}",
        "service_id": "{{ service_id | int }}"
      }
  register: api_response_stripe

```

```

- name: Add monetary balance to OCS
  uri:
    url: "http://{{ crm_config.ocs.0CS }}/jsonrpc"
    method: POST
    body_format: json
    body:
    {
      "method": "ApierV1.AddBalance",
      "params": [
        {
          "Tenant": "{{ crm_config.ocs.ocsTenant }}",
          "Account": "{{ service_uuid }}",
          "BalanceType": "*monetary",
          "Balance": {
            "Value": "{{ monthly_cost | float * 100 }}",
            "ExpiryTime": "+4320h"
          }
        }
      ]
    }
  status_code: 200

- name: Add Transaction to CRM
  uri:
    url: "http://localhost:5000/crm/transaction/"
    method: PUT
    headers:
      Authorization: "Bearer {{ access_token }}"
    body_format: json
    body:
    {
      "customer_id": {{ customer_id | int }},
      "service_id": {{ service_id | int }},
      "title": "{{ package_name }}",
      "retail_cost": "{{ monthly_cost | float }}"
    }
  status_code: 201

- name: Send Notification SMS
  uri:
    url: "http://sms-gateway/SMS/plaintext/{{ api_key }}"
    method: POST
    body_format: json
    body:
    {
      "source_msisdn": "YourCompany",
      "destination_msisdn": "{{ customer_phone }}",
      "message_body": "Thanks for topping up {{ monthly_cost }}!"
    }
  status_code: 201

```

```
ignore_errors: True
```

Este playbook:

1. Obtiene detalles del servicio y del producto desde la API
2. Recupera el método de pago del cliente
3. Cobra al cliente a través de la API de Stripe
4. Agrega saldo monetario a OCS
5. Registra la transacción en la CRM
6. Envía un SMS de confirmación (con `ignore_errors: True` para que los fallos no fallen el trabajo)

## Cadenas de Aprovisionamiento

Para productos complejos que requieren múltiples pasos de aprovisionamiento, OmniCRM admite **cadenas de aprovisionamiento**. Una cadena ejecuta múltiples playbooks secuencialmente, pasando contexto entre ellos.

Ejemplo de caso de uso: Un servicio agrupado que aprovisiona:

1. Servicio de internet base (crea el registro de servicio principal)
2. Complemento IPTV (utiliza el `service_id` del paso 1)
3. Complemento de IP estática (utiliza el `service_id` del paso 1)

El servicio de aprovisionamiento automáticamente:

- Consulta la base de datos para el `service_id` creado por el primer playbook
- Lo inyecta en las `extra_vars` para los playbooks subsiguientes
- Rastreará cada playbook como un registro Provision separado

## Razones de Fallo y Depuración

Cuando el aprovisionamiento falla, el sistema captura información detallada para ayudar a diagnosticar el problema.

### Escenarios Comunes de Fallo

#### Fallos de Tareas Críticas (Estado 2)

Estos causan que todo el trabajo de aprovisionamiento falle:

- Llamadas API que devuelven códigos de estado inesperados
- Fallos de afirmaciones (por ejemplo, `assert: that: response.status == 200`)
- Elementos de inventario requeridos que faltan
- Equipos de red inalcanzables

- Credenciales inválidas o tokens expirados
- Errores de OCS/OCS

## Fallos Ignorados (Estado 3)

Estos se registran pero no fallan el trabajo:

- Notificaciones SMS/correo electrónico opcionales que fallan
- Búsquedas de datos no críticas (marcadas con `ignore_errors: True`)
- Operaciones de limpieza durante la desprovisión

## Errores Fatales

Estos impiden que el playbook se ejecute en absoluto:

- Errores de sintaxis YAML en el playbook
- Variables indefinidas en el playbook
- Archivos de playbook faltantes
- Fallos de conexión al controlador de Ansible

Cuando ocurre un error fatal, el sistema crea un evento de error especial que contiene:

- El código de salida de Ansible
- Salida completa (contiene detalles de errores de sintaxis)
- Salida de error completa (contiene errores de tiempo de ejecución)
- Una lista de causas comunes para ese tipo de fallo
- Todas las variables pasadas al playbook

## Correos Electrónicos de Notificación de Errores

Cuando el aprovisionamiento falla (estado 2), se envía automáticamente un correo electrónico a la lista de notificación de fallos configurada (`provisioning.failure_list` en `crm_config.yaml`).

El correo electrónico incluye:

- Información del cliente
- Detalles del producto/servicio
- Resultados de tareas codificados por colores:
  - **Verde**: Tareas exitosas
  - **Naranja**: Tareas fallidas pero ignoradas
  - **Rojo**: Fallos críticos
- Para fallos críticos: Salida de depuración completa que incluye cuerpos de solicitud/respuesta
- Para errores fatales: Salida de Ansible, mensajes de error y causas comunes

# Monitoreo de Trabajos de Aprovisionamiento

## API de Estado de Aprovisionamiento

Para verificar el estado de un trabajo de aprovisionamiento:

```
GET /crm/provision/provision_id/<id>
Authorization: Bearer <token>
```

La respuesta incluye:

```
{
  "provision_id": 123,
  "customer_id": 456,
  "customer_name": "John Smith",
  "product_id": 10,
  "provisioning_status": 0,
  "provisioning_play": "play_psim_only",
  "playbook_description": "OmniCore Service Provisioning 2024",
  "task_count": 85,
  "provisioning_result_json": [
    {
      "event_number": 1,
      "event_name": "Get Product information from CRM API",
      "provisioning_status": 1,
      "provisioning_result_json": "..."
    },
    {
      "event_number": 2,
      "event_name": "Create account in OCS",
      "provisioning_status": 1,
      "provisioning_result_json": "..."
    }
  ]
}
```

## Listando Trabajos de Aprovisionamiento

Para obtener una lista paginada de todos los trabajos de aprovisionamiento:

```
GET /crm/provision/?page=1&per_page=20&sort=provision_id&order=desc
Authorization: Bearer <token>
```

Admite filtrado:

```
GET /crm/provision/?filters={"provisioning_status": [2]}&search=Mobile
Authorization: Bearer <token>
```

Esto devuelve solo trabajos fallidos (estado 2) donde la descripción contiene "Mobile".

## Mejores Prácticas

### Diseño del Playbook

- **Siempre use block/rescue:** Asegúrese de que el aprovisionamiento parcial pueda ser revertido
- **Use ignore\_errors con juicio:** Solo para operaciones verdaderamente opcionales
- **Registre variables importantes:** Utilice tareas debug para registrar valores clave para la resolución de problemas
- **Valide respuestas:** Utilice assert para verificar que las respuestas de la API sean las esperadas
- **Idempotencia:** Diseñe playbooks para que sean seguros de volver a ejecutar

### Autenticación

- **Aprovisionamiento iniciado por el usuario:** Siempre use refresh\_token para playbooks de larga duración
- **Aprovisionamiento automatizado:** Use lista blanca de IP o claves API con tokens de acceso generados
- **Expiración del token:** El refresh\_token asegura que los tokens de acceso se regeneren según sea necesario

### Manejo de Errores

- **Proporcione contexto:** Incluya customer\_id, service\_id y detalles de operación en los mensajes de error
- **Notificar adecuadamente:** Fallos críticos desencadenan correos electrónicos, pero no spam para fallos esperados
- **Información de depuración:** Capture cuerpos completos de solicitud/respuesta en los registros de Provision\_Event

### Seguridad

- **Valide entradas:** Verifique customer\_id, product\_id, service\_id antes de aprovisionar
- **Verificaciones de permisos:** Verifique que los usuarios solo puedan aprovisionar para clientes autorizados
- **Datos sensibles:** Utilice el sistema de redacción para eliminar contraseñas/claves de los registros
- **Lista blanca de IP:** Restringa los endpoints de simple\_provision solo a sistemas de confianza

## Rendimiento

- **Ejecución en segundo plano:** Nunca bloquee las respuestas de la API esperando el aprovisionamiento
- **Intervalos de polling:** La UI debe consultar actualizaciones de estado cada 2-5 segundos
- **Tareas paralelas:** Utilice la paralelización nativa de Ansible para operaciones independientes
- **Actualizaciones de base de datos:** El controlador de eventos actualiza la base de datos en tiempo real, no es necesario consultar durante la ejecución

## Documentación Relacionada

- [concepts\\_ansible](#) - Conceptos generales de aprovisionamiento de Ansible
- [concepts\\_api](#) - Autenticación y uso de la API de CRM
- [concepts\\_products\\_and\\_services](#) - Definiciones de productos y servicios
- [administration\\_inventory](#) - Gestión de inventario para aprovisionamiento



# Registro de Actividad del Cliente

Cada cambio realizado en un cliente, contacto, sitio, servicio y parámetros financieros, como Transacciones, Facturas y Métodos de Pago, se registra en el Registro de Actividad.

Esto nos permite rastrear los cambios realizados en el sistema, quién los hizo y cuándo se realizaron, y es útil para auditar cambios y rastrear problemas; por ejemplo, si un cliente dice que nunca recibió una factura, podemos consultar el Registro de Actividad para ver si fue enviada, o si un contacto fue eliminado, podemos ver quién lo eliminó y cuándo.

El Registro de Actividad es una lista cronológica de cambios, con los cambios más recientes en la parte superior y los cambios más antiguos más abajo en la lista, que se puede filtrar desde las pestañas.

Los registros del registro de actividad no se pueden eliminar, pero se pueden filtrar, y los detalles se pueden ver para ver qué se cambió y por quién.

Registro de Actividad del Cliente



# Agregar un Servicio

Los servicios son las cosas por las que facturamos al cliente, pueden ser servicios de internet en casa, servicios móviles, o incluso servicios abstractos como arrendar un subred o proporcionar electricidad medida a un rack.

Un Servicio es solo una instancia de un Product <concepts\_products\_and\_services> para un cliente dado, que se selecciona del catálogo de productos y se provisiona para el cliente.

Si aún no has created a customer <basics\_create\_customer>, necesitarás hacer eso primero, ya que los servicios están vinculados a los clientes, también necesitarás definir un método de pago para el cliente, ya que los servicios generan cargos que deben ser pagados.

Los clientes pueden provisionar sus propios servicios (si les permitimos hacerlo), o el personal de Servicio al Cliente puede provisionar servicios para el cliente.

Basado en la definición del producto, hay reglas sobre quién puede comprar un producto, como permitir solo a clientes comerciales comprar un producto comercial, o permitir solo a clientes con un servicio móvil comprar un complemento móvil.

Los servicios también pueden tener componentes de uso, como uso de datos, minutos de llamada, u otros cargos basados en el uso, y pueden tener múltiples cargos asociados con ellos, como cargos mensuales, cargos únicos, o cargos por uso, podemos ver esto desde el botón "Uso".

Muchos servicios soportan Complementos, por ejemplo, un servicio móvil tiene todas las recargas disponibles en el menú de Complementos, y un servicio de internet en casa podría tener una dirección IP estática o datos adicionales disponibles como un complemento, nuevamente todo esto está definido en el product catalog<concepts\_products\_and\_services>.

Servicio - Provisionamiento y Cargos



# Flujo de Trabajo para Asignar Planes

La función Asignar Plan permite al personal **proveer servicios para clientes** seleccionando productos del catálogo e iniciando el proceso de aprovisionamiento. Este es el método principal para crear nuevos servicios cuando los clientes no se autoaprovisionan.

## Descripción General

Asignar un plan implica:

1. Seleccionar un cliente
2. Elegir un producto del catálogo
3. Configurar los requisitos de inventario (tarjetas SIM, equipos, etc.)
4. Establecer parámetros del servicio (renovación automática, campos personalizados)
5. Iniciar el aprovisionamiento
6. Monitorear el progreso del aprovisionamiento

Este flujo de trabajo se utiliza para todos los tipos de servicio: móvil, internet, IPTV y VoIP.

## Accediendo a Asignar Plan

**Desde el Catálogo de Productos:**

**Desde la Página del Cliente:**

**Desde la Página de Complementos:**

La interfaz de asignar plan se abre en un modal o página dedicada.

## Flujo de Trabajo Paso a Paso

### Paso 1: Navegar por el Catálogo de Productos

El catálogo de productos muestra los productos disponibles agrupados por categoría y tipo de cliente.

Catálogo de Productos - Vista del Cliente  
Modal de Carrusel de Complementos

**Alternar Vistas:**

Categorías: • Planes de Servicio • Complementos

Alternar entre filtros Individual y Empresarial filtra los productos para mostrar solo aquellos disponibles para el tipo de cliente seleccionado.

**Tarjetas de Producto:**

Los productos se muestran en un carrusel o cuadrícula:

Haz clic en "**Asignar a Cliente**" para continuar.

## Paso 2: Seleccionar Cliente

Si no estás ya en una página de cliente, se te pedirá que selecciones un cliente.

### Búsqueda de Clientes:

Buscar Clientes: [John ▼]

Clientes coincidentes: • John Smith (ID: 123) • John Doe (ID: 456) • Johnson Enterprises (ID: 789)

Escribe para buscar por:

- Nombre del cliente
- ID del cliente
- Dirección de correo electrónico
- Número de teléfono

Selecciona el cliente del menú desplegable.

### Advertencia

Asegúrate de haber seleccionado el cliente correcto antes de continuar. Asignar un plan al cliente incorrecto requiere intervención manual para corregirlo.

## Paso 3: Configurar Inventario (si es necesario)

Si el producto requiere elementos de inventario (definidos en `inventory_items_list`), aparecen seleccionadores de inventario.

Selector de Inventario de Servicio

### Ejemplo de Selector de Inventario:

Elementos de Inventario Requeridos:

Tarjeta SIM \*

Tarjetas SIM Disponibles: • SIM-00123 - ICCID: 8944...0001 (Nueva) • SIM-00124 - ICCID: 8944...0002 (Nueva) • SIM-00125 - ICCID: 8944...0003 (Nueva)

Número Móvil \*

Números Disponibles: • +44 7700 900123 (Reservado) • +44 7700 900124 (Disponible) • +44 7700 900125 (Disponible)

### Reglas de Selección de Inventario:

- El asterisco (\*) indica un campo requerido
- Solo se muestran elementos de inventario disponibles (estado: "En Stock" o "Nuevo")
- Los menús desplegables se cargan dinámicamente según los nombres de las plantillas de inventario
- Una vez seleccionados, los elementos se reservan temporalmente

### Qué Sucede:

- Los elementos de inventario seleccionados se pasan como variables al libro de jugadas de aprovisionamiento

- Durante el aprovisionamiento, los elementos se asignan al servicio y al cliente
- El estado del elemento cambia de "En Stock" a "Asignado"

## Paso 4: Configurar Renovación Automática (opcional)

Para servicios recurrentes, se te puede pedir que establezcas la renovación automática:

¿Te gustaría habilitar la renovación automática para este servicio?

Cuando está habilitado, este servicio se renovará automáticamente al final de cada período de facturación y se cargará al método de pago predeterminado del cliente.

[ No ] [ Sí ]

### Comportamiento de Renovación Automática:

- **Sí:** El servicio se renueva automáticamente, se carga al cliente mensualmente
- **No:** El servicio expira al final del período del contrato, se requiere renovación manual

**Mejor Práctica:** Predeterminar "Sí" para servicios de consumo, "No" para servicios únicos o cuando el cliente solicita control manual.

## Paso 5: Revisar y Confirmar

La pantalla de revisión muestra todas las selecciones antes del aprovisionamiento:

Cliente: John Smith (ID: 123) Producto: Móvil - Plan de 20GB

Inventario: • Tarjeta SIM: SIM-00123 (ICCID: 8944...0001) • Número Móvil: +44 7700 900123

Precios: • Cuota de Configuración: £0.00 • Costo Mensual: £15.00

Renovación Automática: Sí

[Cancelar] [Confirmar y Aprovisionar]

Haz clic en "**Confirmar y Aprovisionar**" para iniciar el proceso de aprovisionamiento.

## Paso 6: Progreso del Aprovisionamiento

El modal de aprovisionamiento muestra el progreso en tiempo real:

✓ Validando cuenta del cliente ✓ Asignando Tarjeta SIM (ICCID: 8944...0001) ✓  
Asignando Número Móvil (+44 7700 900123) ⚡ Configurando cuenta OCS (en progreso...)  
◇ Creando registro de servicio ◇ Enviando correo electrónico de bienvenida

Progreso: 3 de 6 tareas completadas

Progreso del Aprovisionamiento - Vista Detallada

El modal consulta la API de aprovisionamiento cada 0.2 segundos para actualizaciones de estado.

### Indicadores de Progreso:

- Completado con éxito
- ⚡ Actualmente en ejecución
- ◇ Pendiente (no iniciado)

- ✗ Fallido (si ocurren errores)

Ejemplo de Aprovisionamiento Fallido

## Paso 7: Finalización

### Éxito:

Servicio aprovisionado con éxito para John Smith

ID del Servicio: 789 Nombre del Servicio: Móvil - +44 7700 900123 Estado: Activo

[Ver Servicio] [Cerrar]

Haz clic en "**Ver Servicio**" para abrir la página de detalles del servicio.

### Fallo:

Si el aprovisionamiento falla:

Error: No se pudo conectar a OCS

El registro de servicio ha sido creado pero el aprovisionamiento no se completó. Por favor, revisa el error y vuelve a intentarlo.

ID de Aprovisionamiento: 456

[Ver Registros] [Reintentar] [Cerrar]

- **Ver Registros:** Abre los detalles del aprovisionamiento con mensajes de error
- **Reintentar:** Intenta el aprovisionamiento nuevamente
- **Cerrar:** Sale del modal (el registro de servicio permanece pero no está activado)

## Casos Especiales

### Agregar Complementos a un Servicio Existente

Al asignar un complemento (categoría: "addon") a un cliente que ya tiene un servicio:

1. **Detección Automática del Servicio:**
  - El sistema encuentra los servicios existentes del cliente
  - Filtra por service\_type (el complemento móvil solo se muestra para servicios móviles)
  - Si el cliente tiene múltiples servicios coincidentes, se le pide seleccionar cuál
2. **No se Crea un Nuevo Servicio:**
  - El complemento se aprovisiona contra el service\_id existente
  - Utiliza la cuenta OCS del servicio existente
  - El inventario (si lo hay) se asigna al servicio existente
3. **Libro de Jugadas de Aprovisionamiento:**
  - Libro de jugadas diferente al de servicios independientes
  - Típicamente agrega saldo, características o equipos a la cuenta existente

### Ejemplo:

Servicios Móviles Existentes: • Móvil - +44 7700 900123 (ID: 789) • Móvil - +44 7700 900456 (ID: 790)

¿Qué servicio debería recibir este complemento? [Móvil - +44 7700 900123 ▼]

[Cancelar] [Continuar]

## Aprovisionamiento para Clientes Empresariales

Los clientes empresariales pueden tener requisitos adicionales:

- **Selección de Sitio** - Elegir qué ubicación empresarial recibe el servicio
- **Asignación de Contacto** - Designar contactos de facturación/técnicos
- **Campos Personalizados** - Números de cuenta, centros de costo, números de orden de compra

### Ejemplo de Flujo Empresarial:

Seleccionar Sitio de Instalación: [Oficina de Londres - 123 Main St ▼]

Contacto de Facturación: [Jane Doe - <[jane@acme.com](mailto:jane@acme.com)> ▼]

Contacto Técnico: [Bob Smith - <[bob@acme.com](mailto:bob@acme.com)> ▼]

Número de Orden de Compra: [PO-2025-001234\_\_\_\_\_]

[Cancelar] [Continuar]

## Asignación de Servicio en Masa

Para asignar el mismo plan a múltiples clientes (por ejemplo, migraciones masivas):

1. Usar importación CSV (si está disponible)
2. O asignar individualmente con configuraciones de plantilla
3. El inventario debe estar disponible en masa
4. Monitorear la cola de aprovisionamiento para evitar sobrecarga

## Flujos de Trabajo Comunes

### Flujo de Trabajo 1: Nuevo Servicio Móvil

1. El cliente entra en la tienda queriendo un servicio móvil
2. El personal abre **Productos → Planes**
3. Alterna a tipo de cliente **Individual**
4. Selecciona "**Móvil - Plan de 20GB**"
5. Haz clic en "**Asignar a Cliente**"
6. Busca al cliente por teléfono: "+1234567890"
7. Selecciona a **John Smith** de los resultados
8. Elige la Tarjeta SIM del selector de inventario
9. Elige un número móvil disponible
10. Habilita **renovación automática**
11. Confirma y aprovisiona
12. Observa el progreso hasta completar
13. Entrega la SIM al cliente con el paquete de bienvenida

### Flujo de Trabajo 2: Agregar Internet a un Cliente Existente

1. Navega a la página del cliente: **Clientes → John Smith**
2. Haz clic en la pestaña **Servicios**
3. Haz clic en el botón "**Agregar Servicio**"

4. Navega por los planes de internet
5. Selecciona "**Fibra - 100Mbps**"
6. Selecciona el sitio de instalación (si es empresarial/múltiples sitios)
7. Elige el módem CPE del inventario
8. Establece la fecha de instalación (si es necesario)
9. Aprovisiona el servicio
10. Crea un ticket de instalación

### **Flujo de Trabajo 3: Asignar Complemento de Recarga de Datos**

1. El cliente llama: "Necesito más datos"
2. El personal busca al cliente en la búsqueda global
3. Abre la pestaña de servicios del cliente
4. Haz clic en "Complementos" junto al servicio móvil
5. Selecciona "Aumento de Datos de 5GB"
6. Pago autorizado
7. El complemento se aprovisiona inmediatamente
8. El cliente recibe un aumento de datos instantáneo

## **Solución de Problemas**

### **"No hay productos disponibles"**

- **Causa:** Los filtros excluyen todos los productos
- **Solución:**
  - Alternar tipo de cliente (Individual vs Empresarial)
  - Verificar que el catálogo de productos tenga productos habilitados
  - Verificar que los productos coincidan con la elegibilidad del cliente

### **"No hay inventario disponible"**

- **Causa:** Elementos de inventario agotados o todos asignados
- **Solución:**
  - Agregar más elementos de inventario al sistema
  - Verificar el estado del elemento (debería estar "Nuevo" o "En Stock")
  - Verificar que los nombres de las plantillas de inventario coincidan con los requisitos del producto

### **"Cliente no encontrado"**

- **Causa:** El cliente no existe o el término de búsqueda es incorrecto
- **Solución:**
  - Crear el cliente primero
  - Probar diferentes términos de búsqueda (ID, correo electrónico, teléfono)
  - Verificar errores tipográficos

### **El aprovisionamiento se queda atascado**

- **Causa:** Error en el libro de jugadas o sistema externo inalcanzable
- **Solución:**
  - Esperar el tiempo de espera (típicamente 5 minutos)
  - Verificar los registros de aprovisionamiento para errores específicos
  - Verificar que los sistemas OCS y de red estén en línea
  - Reintentar el aprovisionamiento después de solucionar el problema

### **El aprovisionamiento tiene éxito pero el servicio no funciona**

- **Causa:** Cuenta OCS creada pero la red no se actualizó
- **Solución:**
  - Verificar que OCS tenga la cuenta
  - Verificar que la SIM esté activada en HLR/HSS
  - Verificar el aprovisionamiento de la red (RADIUS, DPI, etc.)
  - Revisar las tareas del libro de jugadas para pasos omitidos

## Mejores Prácticas

### Antes de Asignar:

- Verificar que el cliente tenga un método de pago válido en el archivo
- Confirmar la elegibilidad del cliente para el producto (residencial vs empresarial)
- Asegurarse de que el inventario requerido esté disponible
- Revisar los términos y precios del producto con el cliente

### Durante la Asignación:

- Verificar la selección del cliente antes de confirmar
- Seleccionar los elementos de inventario correctos (verificar números de serie)
- Habilitar la renovación automática por conveniencia (a menos que el cliente objete)
- Monitorear el progreso del aprovisionamiento hasta la finalización

### Después de la Asignación:

- Verificar que el servicio aparezca en la lista de servicios del cliente
- Comprobar que el estado del servicio sea "Activo"
- Confirmar que el inventario esté asignado correctamente
- Enviar un correo electrónico de bienvenida o instrucciones al cliente
- Probar el servicio si es posible (hacer una llamada de prueba, verificar datos)

### Para Complementos:

- Confirmar que el complemento sea compatible con el servicio existente
- Explicar la facturación (única vs recurrente)
- Verificar el pago antes de aprovisionar
- Comprobar que el saldo se actualice inmediatamente después de aprovisionar el complemento

## Documentación Relacionada

- [csa\\_add\\_service](#) - Descripción general de los servicios
- [concepts\\_products\\_and\\_services](#) - Conceptos del catálogo de productos
- [administration\\_inventory](#) - Gestión de elementos de inventario
- [concepts\\_provisioning](#) - Detalles del sistema de aprovisionamiento
- [guide\\_product\\_lifecycle](#) - Ciclo de vida completo del producto, incluyendo aprovisionamiento



# Modificación de un Servicio

Los servicios pueden ser modificados por el cliente final a través del Self-Care Portal <self\_care\_portal>, o por un administrador a través del portal de administración.

Una vez que un servicio está provisionado, puedes modificar sus parámetros, agregar mejoras o cambiar configuraciones.

## Edición de Parámetros del Servicio

Los parámetros básicos del servicio pueden ser modificados haciendo clic en el botón **Editar** en la página de detalles del servicio.

Modal de Edición del Servicio

### Campos Editables:

- Nombre del Servicio
- Estado del Servicio (Activo, Inactivo, Suspendido)
- Notas del Servicio
- Configuraciones de Auto-Renovación
- Artículos de inventario vinculados
- Campos personalizados

Los cambios se guardan inmediatamente y se reflejan en el portal de auto-cuidado del cliente.

## Agregar Complementos al Servicio

Los complementos mejoran los servicios existentes con características adicionales, asignaciones de datos o hardware. Los tipos de complementos comunes incluyen:

- **Recargas de datos** - Asignación de datos adicional (por ejemplo, "5GB Data Boost")
- **Actualizaciones de características** - Capacidades adicionales (por ejemplo, "Llamadas Internacionales")
- **Alquiler de hardware** - Adiciones de equipo (por ejemplo, "Módem WiFi 6")
- **Servicios premium** - Características mejoradas (por ejemplo, "Dirección IP Estática")

## Accediendo al Catálogo de Complementos

Desde una página de detalles del servicio:

1. Navega al servicio que deseas mejorar
2. Haz clic en el botón "**Agregar Complemento**" o "**Explorar Complementos**"
3. Se abre el catálogo de complementos, filtrado para mostrar solo complementos compatibles

### Filtrado Automático:

El sistema filtra automáticamente los complementos según:

- **Tipo de Servicio** - Solo muestra complementos que coinciden con el tipo de servicio (móvil, internet, voip, etc.)
- **Tipo de Cliente** - Filtra por cliente residencial vs. cliente empresarial
- **Compatibilidad del Servicio** - Verifica si el servicio cumple con los requisitos del complemento
- **Disponibilidad** - Solo muestra productos habilitados

Por ejemplo, si estás viendo un servicio móvil residencial, solo verás complementos móviles marcados para clientes residenciales.

## Visualización de Complementos

Los complementos se muestran en un carrusel interactivo que muestra:

Catálogo de Complementos del Servicio

### Visualización de Tarjeta de Producto:

#### Información Mostrada:

- Icono del producto
- Nombre del producto
- Lista de características (puntos de viñeta)
- Costo de configuración
- Costo mensual/recurrido
- Enlace a términos y condiciones
- Botón "Agregar al Servicio"

## Provisionamiento de un Complemento

### Paso 1: Seleccionar Complemento

Haz clic en la tarjeta del complemento deseado, luego haz clic en "**Agregar al Servicio**"

## **Paso 2: Seleccionar Inventario (si es necesario)**

Si el complemento requiere inventario físico (por ejemplo, alquiler de hardware), aparece un selector de inventario:

Módems Disponibles: • Modem-12345 - TP-Link AX1800 (Nuevo) • Modem-12346 - TP-Link AX1800 (Nuevo) • Modem-12347 - Netgear RAX40 (Nuevo)

Selecciona el artículo de inventario específico para asignar a este servicio.

## **Paso 3: Configurar Auto-Renovación (opcional)**

Para complementos recurrentes, puede que se te pregunte:

¿Te gustaría habilitar la auto-renovación para este complemento?

[ No ] [ Sí ]

- **Sí** - El complemento se renueva automáticamente cada período de facturación
- **No** - Compra única, el cliente debe renovar manualmente

## **Paso 4: Confirmar y Provisionar**

Revisa los detalles del complemento y haz clic en "**Confirmar**"

El modal de provisionamiento aparece mostrando el progreso en tiempo real:

✓ Validando pago ✓ Asignando inventario ◇ Configurando servicio (en progreso...) ◇ Creando transacción ◇ Enviando correo electrónico de confirmación

Una vez completado, el complemento aparece en la lista de complementos del servicio.

### **Provisionamiento Detrás de Escenas:**

Cuando agregas un complemento, el sistema:

1. Valida que el cliente pueda comprar el complemento
2. Ejecuta el playbook de Ansible del complemento (`provisioning_play`)
3. Asigna cualquier artículo de inventario requerido al servicio
4. Crea un registro de transacción para la facturación
5. Actualiza la configuración del servicio (OCS, sistemas de red, etc.)
6. Envía una notificación de confirmación al cliente

Para detalles técnicos sobre el provisionamiento de complementos, consulta [Complete Product Lifecycle Guide - Adding Addons](#)

<guide\_product\_lifecycle>.

## Visualización de Complementos Activos

Los complementos activos aparecen en la página de detalles del servicio en la sección "**Complementos**":

### Ejemplo de Visualización:

◊ 5GB Data Boost

Agregado: 10 Ene 2025 Expira: 17 Ene 2025 Estado: Activo Costo: £5.00

◊ Alquiler de Módem WiFi 6

Agregado: 01 Dic 2024 Inventario: Modem-12345 Auto-Renovación: Sí

Mensual: £10.00

## Eliminación de Complementos

Para eliminar un complemento de un servicio:

1. Navega a la página de detalles del servicio
2. Localiza el complemento en la sección "**Complementos Activos**"
3. Haz clic en el "**Eliminar**" o en el **ícono de papelera** junto al complemento
4. Confirma la eliminación

### Qué Sucedé:

- **Complementos de hardware** - Se ejecuta el playbook de desprovisionamiento, el inventario se marca para devolución
- **Complementos virtuales** - Los beneficios se eliminan inmediatamente
- **Complementos de auto-renovación** - Cancela renovaciones futuras
- **Transacción creada** - Acredita cualquier monto prorrteado si es aplicable

### Advertencia

Eliminar complementos de hardware (módems, CPE, etc.) generalmente requiere que el equipo sea devuelto. El sistema marcará el inventario para devolución y puede enviar instrucciones de devolución al cliente.

## Escenarios Comunes de Complementos

### Escenario 1: Cliente Quedándose Sin Datos

1. El cliente contacta soporte: "He utilizado mi asignación mensual"
2. El personal navega al servicio móvil del cliente
3. Hace clic en "Agregar Complemento"
4. Selecciona "5GB Data Boost"

5. Provisiona inmediatamente
6. El cliente recibe una recarga de datos instantánea

### **Escenario 2: Empresa Necesita IP Estática**

1. El cliente empresarial solicita una IP estática para VPN
2. El personal abre el servicio de internet del cliente
3. Navega por los complementos, selecciona "Dirección IP Estática"
4. El sistema provisiona la IP del grupo disponible (inventario)
5. Configura el enrutamiento en el equipo de red
6. La empresa recibe los detalles de configuración de la IP

### **Escenario 3: Alquiler de Equipo para Servicio de Fibra**

1. El cliente se inscribe para internet de fibra
2. Durante el provisionamiento, el cliente opta por el alquiler de módem
3. El personal agrega el complemento "Módem WiFi 6"
4. Selecciona el módem disponible del inventario
5. El módem se envía al cliente
6. Se agrega un cargo recurrente de £10/mes a la cuenta

## **Solución de Problemas**

### **"No hay complementos compatibles disponibles"**

- **Causa:** No hay complementos que coincidan con el tipo de servicio o tipo de cliente
- **Solución:** Verifica que el catálogo de productos tenga complementos con configuraciones de `service_type` y `residential/business` coincidentes

### **Fallo en el provisionamiento de complementos**

- **Causa:** Error en el playbook de provisionamiento o inventario no disponible
- **Solución:**
  - Verifica los registros de provisionamiento para un error específico
  - Verifica que los artículos de inventario estén en stock (si es necesario)
  - Revisa los registros del playbook en el sistema de provisionamiento

### **El selector de inventario no muestra artículos**

- **Causa:** No hay artículos de inventario disponibles del tipo requerido
- **Solución:**
  - Agrega artículos de inventario al sistema
  - Verifica que los artículos existentes no estén todos asignados o dañados
  - Verifica que el nombre de la plantilla de inventario coincida con

`inventory_items_list` exactamente

## La auto-renovación no funciona

- **Causa:** La bandera de auto-renovación no está configurada o el método de pago ha expirado
- **Solución:**
  - Verifica que el servicio tenga `auto_renew: true`
  - Verifica que el cliente tenga un método de pago válido en el archivo
  - Revisa los trabajos programados en el sistema de facturación

## Documentación Relacionada

- `guide_product_lifecycle` - Flujo completo de provisionamiento de complementos
- `csa_add_service` - Creación de nuevos servicios
- `concepts_products_and_services` - Conceptos de productos y servicios
- `administration_inventory` - Gestión de inventario para complementos de hardware



# Gestión de Servicios

La interfaz de Gestión de Servicios proporciona herramientas completas para ver, filtrar y gestionar los servicios al cliente en toda su organización.

Documentación relacionada: Adding Services <csa\_add\_service>, Service Usage <csa\_service\_usage>, Modifying Services <csa\_modify>, Products & Services <concepts\_products\_and\_services>.

## Accediendo a la Lista de Servicios

Navegue a:

O directamente:

Esto muestra todos los servicios con capacidades de filtrado, ordenación y gestión masiva.

## Visión General de la Lista de Servicios

La lista de servicios muestra los servicios en una tabla con:

Vista de la Lista de Servicios

### Columnas:

- **ID del Servicio** - Identificador único
- **Nombre del Servicio** - Nombre descriptivo (por ejemplo, "Móvil - +44 7700 900123")
- **Nombre del Cliente** - Cliente asociado (enlace clicable)
- **Tipo de Servicio** - móvil, iptv, internet, voip
- **Estado** - Activo, Inactivo, Suspendido
- **Fecha de Inicio** - Cuando comenzó el servicio
- **Fecha de Finalización** - Cuando expira/termina el servicio
- **Costo Mensual** - Precio de venta
- **Acciones** - Ver, Editar, Complementos, Uso, Eliminar

## Filtrando Servicios

### Pestañas de Estado

Filtre rápidamente los servicios por estado utilizando la barra de pestañas:

- **Todos los Resultados** - Muestra todos los servicios independientemente del estado
- **Activos** - Servicios actualmente activos (vista predeterminada)
- **Inactivos** - Servicios cancelados o expirados
- **Suspendidos** - Servicios deshabilitados temporalmente (no pago, fraude, etc.)

Hacer clic en una pestaña actualiza la lista al instante. La pestaña activa está resaltada.

### Filtro por Tipo de Servicio

Filtrar por tipo de servicio:

Seleccione uno o varios tipos para mostrar solo los servicios que coincidan.

### **Casos de Uso:**

- **Móvil** - Ver todos los servicios de SIM móvil
- **Internet** - Mostrar fibra, DSL y inalámbrico fijo
- **IPTV** - Mostrar servicios de suscripción de TV
- **VoIP** - Listar servicios de voz sobre IP

### **Filtro por Cliente**

Buscar y filtrar por cliente:

Comience a escribir el nombre de un cliente para ver resultados coincidentes:

Seleccione un cliente para mostrar solo sus servicios.

**Caso de Uso:** Ver rápidamente todos los servicios de un cliente específico.

### **Búsqueda**

Búsqueda global en todos los campos de servicio:

Busca:

- Nombre del servicio
- ID del servicio
- Nombre del cliente
- Números de teléfono (para servicios móviles)
- UUID del servicio

**Ejemplo:** Busque "0770" para encontrar todos los servicios con números de teléfono que contengan "0770".

### **Ordenando Servicios**

Haga clic en cualquier encabezado de columna para ordenar por ese campo:

#### **Columnas Ordenables:**

- ID del Servicio (predeterminado: más nuevo primero)
- Nombre del Servicio (alfabético)
- Nombre del Cliente (alfabético)
- Tipo de Servicio (alfabético)
- Fecha de Inicio (cronológico)
- Fecha de Finalización (cronológico)
- Costo Mensual (numérico)

#### **Dirección de Ordenación:**

- Haga clic una vez: Ascendente (A-Z, más antiguo-más nuevo, más bajo-más alto)
- Haga clic dos veces: Descendente (Z-A, más nuevo-más antiguo, más alto-más bajo)
- La columna de ordenación activa muestra el indicador ▲ o ▼

#### **Desplegable de Ordenación:**

El desplegable en la parte superior derecha proporciona preajustes de ordenación rápida:

# Visualizando Detalles del Servicio

Haga clic en cualquier nombre de servicio para abrir la vista detallada del servicio.

## Pestañas de Detalle del Servicio:

1. **Resumen** - Resumen del servicio, estado, fechas, precios
2. **Inventario** - Equipos asignados (tarjetas SIM, módems, etc.)
3. **Transacciones** - Cargos, créditos, pagos
4. **Uso** - Estadísticas de uso de datos/voz/SMS
5. **Complementos** - Complementos activos y disponibles
6. **Registro de Actividad** - Historial de cambios

## Acciones Rápidas

Desde la lista de servicios, haga clic en el menú de acciones ( : ) para operaciones rápidas:

- **Ver** - Abre los detalles del servicio
- **Editar** - Modificar parámetros del servicio
- **Complementos** - Navegar y agregar mejoras al servicio
- **Uso** - Ver uso y saldo actuales
- **Eliminar** - Cancelar/eliminar servicio

# Operaciones Masivas

Seleccione múltiples servicios utilizando casillas de verificación para realizar acciones masivas.

## Seleccionando Servicios

### Selección Individual:

Haga clic en la casilla de verificación junto a cada servicio que desee seleccionar.

### Seleccionar Todo:

Haga clic en la casilla de verificación en el encabezado de la tabla para seleccionar todos los servicios visibles.

Servicio 1  Servicio 2  Servicio 3

### Contador de Selección:

La interfaz muestra cuántos servicios están seleccionados:

## Acciones Masivas

Una vez que se seleccionan los servicios, aparecen botones de acción masiva:

### Eliminar Múltiples Servicios:

1. Seleccione los servicios a eliminar
2. Haga clic en el botón "**Eliminar Seleccionados**"
3. Confirme la eliminación en el modal

Advertencia

La eliminación masiva es permanente y:

- Cancelará todos los servicios seleccionados
- Marcará el inventario como no asignado
- Detendrá todos los cargos recurrentes
- Creará entradas en el registro de actividad

#### **Casos de Uso:**

- Limpiar servicios de prueba
- Cancelar servicios para cuentas de clientes cerradas
- Eliminar entradas duplicadas o erróneas

**Mejor Práctica:** Utilice filtros para reducir la lista antes de realizar operaciones masivas para evitar eliminaciones accidentales.

## **Gestión del Estado del Servicio**

### **Tipos de Estado**

Los servicios pueden tener tres estados:

#### **Activo**

- El servicio está operativo
- Se aplican cargos
- El cliente puede usar el servicio
- Se muestra con una insignia verde

#### **Inactivo**

- El servicio ha sido cancelado o ha expirado
- No se aplican cargos
- El cliente no puede usar el servicio
- Se muestra con una insignia gris
- Inventario marcado para devolución/refurbishment

#### **Suspendido**

- Servicio deshabilitado temporalmente
- Puede o no cobrar (configurable)
- El cliente no puede usar el servicio
- Se muestra con una insignia naranja/amarilla
- Razones comunes: No pago, investigación de fraude, solicitud del cliente

## **Cambiando el Estado del Servicio**

Para cambiar el estado de un servicio:

1. Abra los detalles del servicio
2. Haga clic en el botón "**Editar**"
3. Cambie el desplegable "**Estado del Servicio**"
4. Haga clic en "**Guardar**"

#### **Qué Sucede:**

- **Activo → Inactivo:** Activa el libro de jugadas de desprovisionamiento (si está configurado)

- **Activo → Suspendido:** Desactiva el servicio en OCS/red pero mantiene el registro
- **Suspendido → Activo:** Reactiva el servicio, reanuda la facturación
- **Inactivo → Activo:** Puede activar el reprovisionamiento (usar con precaución)

## Vista de Uso del Servicio

Haga clic en "**Uso**" en el menú de acciones para abrir el modal de uso.

Modal de Uso del Servicio

### Información Mostrada:

- **Saldo** - Crédito restante o valor prepagado
- **Uso de Datos** - Usado vs. asignado (barra de progreso)
- **Uso de Voz** - Minutos usados vs. asignación del plan
- **Uso de SMS** - Mensajes enviados vs. asignación
- **Fecha de Expiración** - Cuando expira el saldo/plan actual

### Acciones:

- **Recargar** - Agregar crédito o datos (para servicios prepagados)
- **Ver Detalles** - Ver desglose detallado del uso
- **Exportar** - Descargar informe de uso (si está disponible)

## Próximas Renovaciones Automáticas

La vista de Próximas Renovaciones Automáticas proporciona una interfaz centralizada para monitorear y gestionar todas las renovaciones de servicios programadas en su organización.

### Accediendo a Renovaciones Automáticas

Navegue a:

O directamente:

Esto muestra todos los servicios con renovación automática programada, ordenados por la próxima fecha de renovación.

## Visión General de Renovaciones Automáticas

La lista de renovaciones automáticas muestra las renovaciones programadas en una tabla con:

### Columnas:

- **Cliente** - Nombre del cliente (enlace clicable a la vista general del cliente)
- **Servicio** - Nombre del servicio (enlace clicable a la vista general del cliente)
- **Producto** - Nombre del producto/plan que se renueva
- **Costo** - Costo de renovación (del precio de venta del producto)
- **Renueva En** - Fecha y hora de la próxima renovación con formato legible para humanos
- **Estado** - Estado del servicio (Activo, Suspendido, etc.)
- **Acciones** - Renovar ahora o eliminar la renovación automática

### Ejemplo de Visualización:

Tablero de Próximas Renovaciones Automáticas

## Cómo Funciona la Renovación Automática

La renovación automática se programa en CGRateS (el sistema de facturación) cuando se crea o modifica un servicio. El sistema:

1. **Programa la Acción** - Crea un ActionPlan en CGRateS con la fecha de renovación
2. **Monitorea el Saldo** - Verifica si el cliente tiene saldo suficiente antes de la renovación
3. **Ejecuta la Renovación** - En la fecha programada, renueva automáticamente el servicio
4. **Actualiza los Registros** - Crea transacciones, actualiza las fechas de servicio, y registra la actividad

### Fuente de Datos:

La vista de Próximas Renovaciones Automáticas consulta CGRateS directamente utilizando el ApierV1.GetScheduledActions API y enriquece los datos con información de cliente y servicio de la base de datos CRM.

### Renovar Ahora

Para activar manualmente una renovación antes de la fecha programada:

1. Haga clic en el botón **Renovar Ahora** para el servicio
2. Revise los detalles de la renovación en el modal de confirmación:

Cliente: Acme Corp Servicio: FixedWireless\_75628fa5 Producto: Internet Hogar  
Mega Costo: \$89.99

Próxima renovación programada: 11 de noviembre de 2025 10:45 AM

[Cancelar] [Confirmar Renovación]

3. Haga clic en **Confirmar Renovación** para procesar de inmediato
4. Se crea y ejecuta un trabajo de aprovisionamiento
5. Monitoree el progreso en el modal de estado de aprovisionamiento

### Qué Sucedé:

- El servicio se recarga con las asignaciones del producto
- El saldo se actualiza en CGRateS
- Se crea un registro de transacción
- La fecha de finalización del servicio se extiende
- Se crea una entrada en el registro de actividad

### Casos de Uso:

- El cliente solicita una renovación anticipada
- Resolver problemas de servicio restableciendo asignaciones
- Probar procesos de renovación
- El cliente quiere usar el servicio antes de la renovación programada

### Eliminar Renovación Automática

Para cancelar la renovación automática de un servicio:

1. Haga clic en el botón **Eliminar Renovación Automática** para el servicio

2. Confirme la eliminación en el modal:

Esto eliminará la renovación automática programada para:

Cliente: Acme Corp Servicio: FixedWireless\_75628fa5 Próxima Renovación: 11 de noviembre de 2025 10:45 AM

El servicio no se renovará automáticamente. Deberá renovar manualmente o el servicio expirará en la fecha de finalización.

[Cancelar] [Eliminar Renovación Automática]

3. Haga clic en "**Eliminar Renovación Automática**" para confirmar

4. El ActionPlan se elimina de CGRateS

5. El servicio expirará naturalmente a menos que se renueve manualmente

**Qué Sucede:**

- El ActionPlan se elimina de CGRateS
- El estado del servicio permanece sin cambios
- El servicio expirará en su fecha de finalización actual
- El cliente debe renovar manualmente o el servicio se detendrá

**Casos de Uso:**

- El cliente está cancelando el servicio
- Cambiando a un proceso de renovación manual
- El plan de servicio está cambiando
- El cliente solicitó detener los cargos automáticos

Advertencia

Eliminar la renovación automática significa que el servicio expirará a menos que se renueve manualmente. Los clientes no recibirán continuación automática del servicio.

## **Entendiendo la Visualización**

**Columna de Cliente:**

- Muestra el nombre del cliente como un enlace clicable
- Enlaces a la página de vista general del cliente
- Muestra "N/A" si el servicio no está vinculado a un cliente (servicio huérfano)

**Columna de Servicio:**

- Muestra el nombre/UUID del servicio
- Enlaces a la página de vista general del cliente
- Muestra el UUID del servicio si el nombre no está establecido
- Muestra "N/A" si el servicio no se puede encontrar en la base de datos

**Columna de Producto:**

- Nombre del producto del ActionPlan de CGRateS
- Extraído de los metadatos de la acción programada
- Siempre muestra el nombre del producto incluso si el servicio no se encuentra

### **Columna de Costo:**

- Costo de venta de la tabla de Productos en CRM
- Monto que se cobrará en la renovación
- Muestra "N/A" si el producto no se encuentra en la base de datos

### **Columna de Renueva En:**

- Fecha y hora exactas de la próxima renovación
- Tiempo relativo legible para humanos (por ejemplo, "3 horas a partir de ahora", "2 días a partir de ahora")
- Consciencia de zona horaria (utiliza la zona horaria del servidor CGRateS)

### **Columna de Estado:**

- Estado actual del servicio de la base de datos CRM
- **Activo** (Verde) - El servicio está operativo
- **Suspendido** (Naranja) - El servicio está deshabilitado temporalmente
- **Desconocido** (Gris) - Servicio no encontrado en la base de datos o estado no establecido

## **Resolución de Problemas**

### **Servicios que muestran "N/A" para Cliente/Servicio**

- **Causa:** El UUID del servicio en CGRateS no coincide con service\_uuid en la base de datos CRM
- **Solución:**
  - Verifique que el servicio exista en la base de datos
  - Verifique que el formato de service\_uuid coincida: ServiceType\_UUID (por ejemplo, FixedWireless\_75628fa5)
  - El servicio puede haber sido eliminado de CRM pero aún está programado en CGRateS

### **Costo mostrando "N/A"**

- **Causa:** El ID del producto en CGRateS no existe en la tabla de Productos de CRM
- **Solución:**
  - Verifique que el producto exista en la base de datos
  - Verifique que product\_id en el ActionPlan coincida con un producto en CRM
  - El producto puede haber sido eliminado

### **"Renueva En" no muestra fecha**

- **Causa:** NextRunTime no presente en la respuesta de CGRateS
- **Solución:**
  - Verifique la configuración del ActionPlan de CGRateS
  - Verifique que ActionTiming esté configurado correctamente
  - Revise los registros de CGRateS en busca de errores

### **"Renovar Ahora" falla**

- **Causa:** Varios errores de aprovisionamiento
- **Resolución de Problemas:**
  - Revise el modal de estado de aprovisionamiento para obtener detalles de errores
  - Verifique que los libros de jugadas de Ansible estén configurados correctamente
  - Verifique la conectividad de OCS
  - Revise los registros de aprovisionamiento

## Renovación automática eliminada pero aún mostrando

- **Causa:** Retraso en caché o problema de sincronización de CGRateS
- **Solución:**
  - Haga clic en el botón **"Actualizar"** para recargar los datos
  - Verifique que el ActionPlan se haya eliminado realmente en CGRateS
  - Verifique la conectividad de la API de CGRateS

## Paginación

Los servicios se muestran en páginas para mejorar el rendimiento:

◀ Anterior 1 [2] 3 4 5 Siguiente ►

Elementos por página: [10 ▼]

- 10
- 25
- 50
- 100

### Controles:

- **Anterior/Siguiente** - Navegar entre páginas
- **Números de Página** - Saltar a una página específica
- **Elementos por Página** - Ajustar cuántos servicios se muestran por página

**Consejo de Rendimiento:** Utilice filtros para reducir el total de resultados en lugar de aumentar los elementos por página.

## Insignias e Indicadores de Servicio

Los indicadores visuales ayudan a identificar rápidamente los estados del servicio:

### Insignias de Estado:

#### Indicador de Renovación Automática:

Los servicios con renovación automática habilitada muestran:

#### Expirando Pronto:

Los servicios que expiran dentro de 7 días muestran:

#### Atrasado:

Los servicios con saldo pendiente muestran:

## Integración con CGRateS (Avanzado)

Para servicios integrados con CGRateS (el motor de facturación y calificación), los administradores pueden gestionar configuraciones avanzadas directamente desde la vista del servicio. Esto incluye atributos, filtros y ver sesiones activas.

Nota

Las funciones de integración de CGRateS requieren el permiso **cgrates\_api\_access**. Solo los administradores tienen acceso a estas funciones por defecto. Consulte rbac para la configuración de permisos.

## Aprovisionamiento Automático vs. Gestión Manual

En operación normal, los atributos y filtros de CGRateS se aprovisionan automáticamente por Ansible durante el flujo de trabajo de aprovisionamiento inicial del servicio. Cuando se crea un nuevo servicio, los libros de jugadas de aprovisionamiento:

- Crean la cuenta de servicio en CGRateS
- Configuran atributos (IMSI, MSISDN, identificadores de cuenta, perfiles de velocidad, etc.)
- Configuran filtros para asegurar la calificación correcta
- Aplican los planes de calificación apropiados

Sin embargo, hay casos en los que puede necesitar modificar estas configuraciones después de que se haya aprovisionado el servicio:

- **Cambios en el Perfil de Velocidad** - El cliente actualiza/disminuye el ancho de banda (MaxBitrateDL/UL)
- **Ajustes de Políticas** - Cambiar políticas de QoS o reglas de modelado de tráfico (PcefPolicyName)
- **Cambios en el Número de Teléfono** - Actualizar MSISDN u otros identificadores
- **Resolución de Problemas** - Corregir configuraciones incorrectas o probar diferentes ajustes
- **Configuraciones Especiales** - Aplicar atributos personalizados que no forman parte de el aprovisionamiento estándar

La interfaz de gestión manual permite a los administradores realizar estos cambios directamente sin volver a ejecutar todo el flujo de trabajo de aprovisionamiento. Esto es particularmente útil para:

- **Cambios Rápidos** - Modificar un solo atributo sin esperar por el aprovisionamiento
- **Pruebas** - Experimentar con diferentes configuraciones
- **Soporte al Cliente** - Resolver problemas sobre la marcha durante llamadas de soporte
- **Configuraciones Personalizadas** - Aplicar ajustes específicos del servicio no cubiertos por plantillas

### Advertencia

Los cambios manuales en la configuración de CGRateS eluden el flujo de trabajo estándar de aprovisionamiento. Asegúrese de comprender el impacto de sus cambios, ya que configuraciones incorrectas pueden afectar la facturación y la funcionalidad del servicio. Todos los cambios se registran en el feed de actividad del cliente para fines de auditoría.

## Accediendo a las Funciones de CGRateS

Al ver o editar un servicio que está aprovisionado en CGRateS, aparecen tres secciones colapsables en la parte inferior del formulario del servicio:

- **Atributos de CGRateS** - Configurar atributos específicos del servicio
- **Filtros de CGRateS** - Definir reglas de filtrado para el servicio
- **Sesiones Activas** - Ver sesiones activas en tiempo real

Cada sección está colapsada por defecto para mantener la interfaz limpia. Haga clic en el encabezado de la sección para expandir y ver/editar la configuración.

### Vista Colapsada de Secciones de CGRateS

Las secciones colapsadas muestran insignias de conteo que indican cuántos atributos, filtros o sesiones activas existen para el servicio.

## Atributos de CGRateS

Los atributos le permiten definir campos personalizados y transformaciones que se aplican a los eventos de calificación para este servicio específico.

**Formato del ID de Atributo:** ATTR\_ACCOUNT\_{service\_uuid}

**Ejemplo:** Para el servicio con UUID Mobile\_SIM\_c2880638, el ID del perfil de atributo es ATTR\_ACCOUNT\_Mobile\_SIM\_c2880638

### Gestión de Atributos:

1. Abra la vista de edición del servicio
2. Expanda la sección "**Atributos de CGRateS**"
3. Haga clic en el botón "**Editar Atributos**"
4. Agregue/modifique/elimine atributos según sea necesario
5. Haga clic en "**Guardar Atributos**"

Vista de Edición de Atributos de CGRateS

### Campos de Atributo:

- **Ruta** - El campo a modificar (por ejemplo, \*req.Account, \*req.IMSI)
- **Tipo** - Cómo se establece el valor:
  - \*constant - Establece un valor fijo
  - \*variable - Captura el valor de los campos del evento utilizando RSRParser
  - \*composed - Anexa el valor en lugar de sobrescribir
  - \*usage\_difference - Calcula la duración entre dos campos
  - \*sum - Suma múltiples valores
  - \*value\_exponent - Calcula el exponente de un campo
- **Reglas** - El/los valor(es) a aplicar (puede tener múltiples reglas por atributo)

### Ejemplo de Configuración de Atributo:

```
{  
  "Path": "*req.Account",  
  "Type": "*constant",  
  "Value": [{"Rules": "Mobile_SIM_474a380a"}]  
}
```

### Casos de Uso Comunes:

- Establecer identificador de cuenta para calificación
- Mapear IMSI/MSISDN al servicio
- Configurar límites de ancho de banda (MaxBitrateDL/UL)
- Establecer nombres de políticas (PcefPolicyName)
- Transformar o enriquecer eventos de calificación

### Registro de Actividad:

Todas las modificaciones de atributos se registran en el feed de actividad del cliente con todos los detalles de los cambios realizados.

## Filtros de CGRateS

Los filtros definen reglas de coincidencia que determinan cuándo se debe aplicar la configuración de este servicio durante la calificación.

**Formato del ID de Filtro:** FLTR\_ACCOUNT\_{service\_uuid}

**Ejemplo:** Para el servicio con UUID Mobile\_SIM\_c2880638, el ID del filtro es FLTR\_ACCOUNT\_Mobile\_SIM\_c2880638

### Gestión de Filtros:

1. Abra la vista de edición del servicio
2. Expanda la sección "**Filtros de CGRateS**"
3. Haga clic en el botón "**Editar Filtros**"
4. Agregue/modifique/elimine reglas de filtro
5. Haga clic en "**Guardar Filtros**"

### Campos de Regla de Filtro:

- **Elemento** - El campo contra el que coincidir (por ejemplo, `~*req.Account`, `~*req.Destination`)
- **Tipo** - Tipo de coincidencia:
  - `*string` - Coincidencia exacta de cadena
  - `*prefix` - Comienza con el valor especificado
  - `*suffix` - Termina con el valor especificado
  - `*empty` - El campo está vacío
  - `*exists` - El campo existe
  - `*notexists` - El campo no existe
  - `*timings` - Coincidir patrones de tiempo/fecha
  - `*destinations` - Coincidir patrones de destino
  - `*rsr` - Coincidencia de campo RSR
  - `*gt / *gte / *lt / *lte` - Comparaciones numéricas
- **Valores** - El/los valor(es) a coincidir (puede tener múltiples valores por regla)

### Ejemplo de Configuración de Filtro:

```
{  
  "Element": "~*req.Account",  
  "Type": "*string",  
  "Values": ["Mobile_SIM_474a380a"]  
}
```

### Casos de Uso Comunes:

- Asegurarse de que los atributos solo se apliquen a cuentas específicas
- Filtrar por destino (nacional vs. internacional)
- Filtrado basado en tiempo (horas pico vs. fuera de pico)
- Filtrar por tipo o categoría de servicio

### Registro de Actividad:

Todas las modificaciones de filtros se registran en el feed de actividad del cliente.

## Sesiones Activas

Vea las sesiones activas en tiempo real para este servicio. Esto muestra llamadas en curso, sesiones de datos u otros eventos facturables que están actualmente en progreso.

## Visualizando Sesiones Activas:

1. Abra la vista de edición del servicio
2. Expanda la sección "**Sesiones Activas**"
3. Vea la lista de sesiones activas
4. Haga clic en "**Ver Detalles**" en cualquier sesión para ver los datos completos de la sesión
5. Haga clic en "**Actualizar**" para recargar la lista de sesiones

Vista de Sesiones Activas de CGRateS

## Información de Sesión Mostrada:

- **Hora de Configuración** - Cuando comenzó la sesión
- **Uso** - Duración actual de la sesión (en segundos)
- **Destino** - Número llamado o destino

## Modal de Detalles de Sesión:

Hacer clic en "Ver Detalles" abre un modal que muestra:

- **Información Básica:**
  - CGRID (ID de sesión)
  - Cuenta (UUID del servicio)
  - Hora de configuración
  - Uso/duración actual
  - Destino
  - Categoría
- **Datos Completos de la Sesión:**
  - Representación JSON completa de la sesión
  - Todos los campos de sesión de CGRateS
  - Estado de la sesión en tiempo real
  - Visor JSON desplazable para inspección

Modal de Detalles de Sesión de CGRateS

## Casos de Uso:

- Monitorear llamadas activas o sesiones de datos
- Resolver problemas de facturación
- Verificar que la sesión se esté calificando correctamente
- Comprobar atributos y valores de la sesión
- Auditarse el uso activo del servicio

## Tasa de Actualización:

Las sesiones se obtienen bajo demanda cuando expande la sección. Haga clic en "Actualizar" para obtener los últimos datos de la sesión.

Nota

Solo se muestran las sesiones que coinciden con la cuenta de este servicio (UUID del servicio). El filtro `*string:~*req.Account:{service_uuid}` se aplica automáticamente.

## Proxy de API de CGRateS

Todas las operaciones de CGRateS (atributos, filtros, sesiones) utilizan el endpoint proxy de la API de OmniCRM:

**Endpoint:** POST /crm/ocs/proxy

### Campos Requeridos:

- `method` - Método de la API de CGRateS (por ejemplo, `APIerSv1.GetAttributeProfile`)
- `params` - Array de parámetros para el método
- `customer_id` - ID del cliente (para registro de actividad)
- `service_id` - ID del servicio (para registro de actividad)

### Campos Opcionales:

- `tenant` - Inquilino de CGRateS (por defecto al valor de configuración)

### Ejemplo de Solicitud:

```
{  
  "method": "APIerSv1.GetAttributeProfile",  
  "params": [{"ID": "ATTR_ACCOUNT_Mobile_SIM_c2880638"}],  
  "customer_id": 123,  
  "service_id": 456  
}
```

### Configuración de Inquilino:

El inquilino se establece automáticamente desde el archivo de configuración de OmniCRM (`crm_config.yaml`) bajo `ocs.ocsTenant`. Esto asegura que todas las operaciones de CGRateS utilicen el inquilino correcto sin codificar valores en el frontend.

### Requisito de Permiso:

Se requiere el permiso `cgrates_api_access`. Este permiso se otorga al rol `admin` por defecto.

### Registro de Actividad:

Todas las operaciones de API de CGRateS que no son GET se registran automáticamente en el feed de actividad del cliente, incluyendo:

- Método de API llamado
- Inquilino utilizado
- Parámetros completos enviados
- ID del servicio en el que se realizó la operación
- Usuario que realizó la operación
- Marca de tiempo

Esto crea un rastro de auditoría completo de todos los cambios de configuración de CGRateS.

## Resolución de Problemas de Integración con CGRateS

### "Permiso Denegado" al acceder a funciones de CGRateS

- **Causa:** El usuario no tiene el permiso `cgrates_api_access`
- **Solución:** Conceder permiso al rol del usuario (típicamente función solo para administradores)

### Atributos o Filtros no cargando

- **Causa:** Problema de conectividad con CGRateS o el perfil no existe
- **Solución:**

- Verifique la conectividad del servidor CGRateS en la configuración
- Verifique que la configuración del inquilino sea correcta
- Revise la consola del navegador en busca de errores de API
- El perfil puede no existir aún (mostrará un formulario vacío)

## Cambios no guardados

- **Causa:** Error de validación o error de API de CGRateS
- **Solución:**
  - Verifique los campos requeridos (Ruta, Tipo, Elemento)
  - Verifique que el formato JSON sea correcto
  - Revise el registro de actividad para obtener detalles de errores
  - Revise los registros de CGRateS

## No se muestran sesiones activas

- **Causa:** No hay sesiones actualmente activas para este servicio
- **Solución:**
  - Esto es normal si el servicio no está en uso
  - Intente actualizar después de iniciar una sesión (llamada, datos, etc.)
  - Verifique que el UUID del servicio coincida con la cuenta en CGRateS

## Detalles de la sesión no se actualizan en tiempo real

- **Causa:** Los datos de la sesión se obtienen bajo demanda, no en vivo
- **Solución:** Haga clic en el botón "Actualizar" para obtener los últimos datos de la sesión

## El registro de actividad no muestra cambios de CGRateS

- **Causa:** Solo se registran operaciones que no son GET (las lecturas no se registran)
- **Solución:** Esto es por diseño - solo las escrituras/modificaciones crean entradas de actividad

# Flujos de Trabajo Comunes

## Flujo de Trabajo 1: Encontrar Servicios del Cliente

1. Haga clic en el **filtro de Tipo de Servicio** (opcional)
2. Haga clic en el **filtro de Cliente**
3. Escriba el nombre del cliente
4. Seleccione al cliente del desplegable
5. Revise los servicios del cliente

## Flujo de Trabajo 2: Identificar Servicios que Expiran

1. Haga clic en la pestaña "**Activos**"
2. Ordene por "**Fecha de Finalización**" (ascendente)
3. Los servicios que expiran primero aparecen primero
4. Contacte a los clientes para la renovación

## Flujo de Trabajo 3: Limpiar Servicios de Prueba

1. Busque "test" en el cuadro de búsqueda
2. Revise los resultados para confirmar que son datos de prueba
3. Seleccione todos los servicios de prueba
4. Haga clic en "**Eliminar Seleccionados**"
5. Confirme la eliminación

## Flujo de Trabajo 4: Suspender Cliente Moroso

1. Navegue a la cuenta del cliente
2. Vea la pestaña de servicios
3. Seleccione todos los servicios activos
4. Cambie el estado a "Suspendido"
5. Guarde los cambios

## Flujo de Trabajo 5: Ver Uso del Servicio Móvil

1. Filtre por **Tipo de Servicio: Móvil**
2. Haga clic en el nombre del servicio para abrir los detalles
3. Haga clic en la pestaña "**Uso**"
4. Revise el consumo de datos/voz/SMS
5. Identifique a los usuarios intensivos o sobrecargas

## Resolución de Problemas

### Servicios no aparecen en la lista

- **Causa:** El filtro de estado oculta resultados
- **Solución:** Haga clic en la pestaña "Todos los Resultados" para mostrar todos los estados

### No se puede encontrar el servicio por búsqueda

- **Causa:** El término de búsqueda no coincide con los datos almacenados
- **Solución:**
  - Intente una búsqueda parcial (por ejemplo, "0770" en lugar del número completo)
  - Utilice el filtro de cliente en su lugar
  - Verifique si hay errores tipográficos

### Botón de eliminación masiva deshabilitado

- **Causa:** No se han seleccionado servicios
- **Solución:** Marque las casillas junto a los servicios que desea eliminar

### Ordenar no funciona

- **Causa:** Columna no ordenable
- **Solución:** Solo las columnas con iconos **▲▼** son ordenables

### La página se carga lentamente

- **Causa:** Demasiados servicios para mostrar
- **Solución:**
  - Aplique filtros para reducir el conjunto de resultados
  - Reduzca los elementos por página
  - Utilice la búsqueda para reducir resultados

## Documentación Relacionada

- `csa_add_service` - Agregar nuevos servicios
- `csa_modify` - Modificar servicios y agregar complementos
- `concepts_products_and_services` - Conceptos de productos y servicios
- `basics_customers` - Gestión de clientes



# Seguimiento del Uso del Servicio y del Saldo

El sistema de Uso del Servicio proporciona **monitoreo en tiempo real** del consumo de los clientes para datos, voz, SMS y saldos monetarios. Esta función se integra con el OCS (Sistema de Carga en Línea) para mostrar el uso actual, las asignaciones restantes y la información de caducidad del saldo tanto a los clientes (a través del Portal de Autocuidado <self\_care\_portal>) como al personal.

## Descripción General

El seguimiento del uso permite:

- **Visualización del Saldo en Tiempo Real** - Ver el uso actual y las asignaciones restantes
- **Múltiples Tipos de Saldo** - Rastrear datos, voz, SMS y saldos monetarios simultáneamente
- **Monitoreo de Caducidad** - Ver cuándo caducan los saldos
- **Desglose del Saldo** - Vista detallada de los diferentes buckets de saldo
- **Actualización Automática** - Actualizaciones de uso cada 3 segundos automáticamente

## Accediendo al Uso del Servicio

**Desde la Lista de Servicios:**

1. Navega a **Servicios** → **Lista de Servicios**
2. Haz clic en el **menú de acciones** (◆◆) junto a un servicio
3. Selecciona "**Uso**"

**Desde los Detalles del Servicio:**

1. Abre la página de detalles de un servicio
2. Haz clic en la pestaña "**Uso**"

**Desde la Página del Cliente:**

1. Abre la vista general del cliente
2. Navega a la pestaña **Servicios**
3. Haz clic en "**Ver Uso**" junto a cualquier servicio

Se abre el modal o página de uso mostrando datos de consumo en tiempo real.

## Visualización del Uso

La interfaz de uso muestra tarjetas de resumen y barras de progreso detalladas para cada tipo de saldo.

Visualización del Uso del Servicio  
Modal de Saldo del Servicio

## Tarjetas de Resumen

La fila superior muestra tarjetas de vista rápida para cada tipo de saldo:

## Información de la Tarjeta:

- **Tipo de Saldo** - Icono y etiqueta (Datos, Voz, SMS, Monetario)
- **Cantidad Restante** - Saldo actual en unidades apropiadas
- **Tiempo de Caducidad** - Días/horas hasta que el saldo caduque
- **Botón de Más Información** - Haz clic para expandir el desglose detallado

## Barra de Progreso

Debajo de las tarjetas, las barras de progreso muestran el consumo visualmente con porciones llenas que indican el saldo restante.

### Características de la Barra de Progreso:

- **Indicador Visual** - La porción llena muestra el saldo restante
- **Porcentaje** - Porcentaje numérico del saldo restante
- **Valores Absolutos** - Muestra usado vs. total (por ejemplo, "12.5GB / 20GB")
- **Codificación de Color**:
  - Verde: >50% restante
  - Amarillo: 20-50% restante
  - Rojo: <20% restante
- **Clickable** - Haz clic para expandir el desglose detallado

## Tipos de Saldo

### Saldo de Datos

Rastrea el consumo de datos de internet.

**Unidades:** Gigabytes (GB) o Megabytes (MB)

### Formato de Visualización:

Progreso: 12.5GB / 20GB (62%)

### Escenarios Comunes:

- **Planes de datos móviles** - 5GB, 10GB, 20GB de asignaciones mensuales
- **Inalámbrico fijo** - Ilimitado o limitado a altas cantidades (500GB, 1TB)
- **Recargas** - Datos adicionales comprados a mitad de ciclo
- **Servicios de dongle** - Datos prepagados para dispositivos hotspot

### Múltiples Buckets:

Los servicios a menudo tienen múltiples saldos de datos:

- Asignación mensual (caduca mensualmente)
- Datos de bonificación (caduca después del período de campaña)
- Datos de recarga (caducidad más corta, consumidos primero)

## Saldo de Voz

Rastrea minutos de llamadas telefónicas.

**Unidades:** Minutos (min)

### Formato de Visualización:

Progreso: 125 min / 500 min (25%)

### Seguimiento de la Duración de las Llamadas:

- Llamadas entrantes (si se cobran)
- Llamadas salientes
- Llamadas internacionales (bucket separado si aplica)
- Números premium

### Cálculo:

El uso de voz se calcula por la duración de la llamada en nanosegundos internamente, convertido a minutos para la visualización.

## Saldo de SMS

Rastrea el uso de mensajes de texto.

**Unidades:** Mensajes (msgs)

### Formato de Visualización:

Progreso: 45 / 250 (18%)

### Tipos de Mensajes:

- SMS estándar (160 caracteres)
- SMS largos (múltiples segmentos)
- MMS (si se rastrean por separado)

## Saldo Monetario

Rastrea el crédito prepago o el saldo de la cuenta.

**Unidades:** Moneda (£, \$, €, etc.)

### Formato de Visualización:

Progreso: £15.50 / £20.00 (77%)

### Uso:

- Las cuentas prepagas utilizan el saldo monetario para pagar el uso
- El crédito se decrementa a medida que el cliente utiliza los servicios
- Puede recargarse a través de pago o vale
- Puede caducar si no se utiliza dentro del período de validez

## Desglose Detallado del Saldo

Haz clic en "**Más Información**" en cualquier tarjeta o haz clic en una barra de progreso para expandir el desglose detallado.

### Vista Expandida:

◇ Asignación Mensual 20GB

Restante: 12.5 GB Usado: 7.5 GB Caduca: 25 de enero de 2025 (15 días) Peso: 10

◊ Datos de Bonificación 5GB

Restante: 5.0 GB Usado: 0 GB Caduca: 31 de enero de 2025 (21 días) Peso: 20

◊ Datos de Recarga 3GB

Restante: 0 GB Usado: 3.0 GB Caduca: 18 de enero de 2025 (CADUCADO) Peso: 30

Total Restante: 17.5 GB

#### **Campos del Bucket de Saldo:**

- **ID/Nombre** - Identificador para el bucket de saldo
- **Restante** - Cantidad restante en este bucket específico
- **Usado** - Cantidad consumida de este bucket
- **Fecha de Caducidad** - Cuándo caduca este saldo
- **Peso** - Orden de prioridad (peso más alto consumido primero)

#### **Sistema de Peso**

Los saldos tienen un valor de **peso** que determina el orden de consumo:

- **Mayor peso = consumido primero**
- **Menor peso = consumido al final**

#### **Ejemplo de Pesos:**

- Datos de recarga: Peso 30 (consumido primero, caducidad más corta)
- Datos de bonificación: Peso 20 (consumido segundo)
- Asignación mensual: Peso 10 (consumido al final, caducidad más larga)

Esto asegura que los saldos que están a punto de caducar se utilicen antes que los que duran más.

#### **Actualizaciones en Tiempo Real**

Los datos de uso se actualizan automáticamente cada **3 segundos** mediante polling.

#### **Qué se Actualiza:**

- Cantidades actuales de saldo
- Barras de progreso de uso
- Temporizadores de caducidad
- Detalles de cada bucket

#### **Experiencia del Usuario:**

- No se requiere recarga de página
- Actualizaciones suaves sin parpadeo
- Superposición de carga durante la actualización
- Insignia de estado que muestra el estado actual del servicio

#### **Casos de Uso:**

- Monitorear el uso del cliente durante la llamada
- Ver la disminución del saldo en tiempo real a medida que el cliente utiliza el servicio
- Verificar la recarga inmediatamente después de la compra

# Uso en Diferentes Tipos de Servicio

## Servicios Móviles

Muestra los cuatro tipos de saldo:

- Datos (GB)
- Voz (minutos)
- SMS (mensajes)
- Monetario (moneda)

### Ejemplo:

DATOS: 12.5GB restantes VOZ: 125 min restantes SMS: 45 msgs restantes MONETARIO: £15.50 restantes

## Inalámbrico Fijo / Internet

Normalmente muestra solo:

- Datos (GB o TB)
- Monetario (si es prepago)

### Ejemplo:

DATOS: 450GB / 500GB restantes MONETARIO: £45.00 (crédito prepago)

## Servicios de Hotspot / Dongle

Muestra el seguimiento de datos específico del dongle:

- Datos (consumidos vs. prepagados)
- Monetario (saldo prepago)

## Modo de Visualización:

Cuando `dongle=true`, el componente oculta voz y SMS, mostrando solo los saldos de datos y monetarios relevantes.

# Solución de Problemas

## Uso mostrando como 0 / 0

- **Causa:** Servicio no integrado con OCS o CGRateS
- **Solución:**
  - Verifica que el servicio esté provisionado en OCS
  - Verifica la conectividad de la API de OCS
  - Revisa el mapeo del UUID del servicio

## Uso no actualizándose

- **Causa:** Polling detenido o OCS inalcanzable
- **Solución:**
  - Actualiza la página
  - Verifica la consola del navegador en busca de errores
  - Verifica que la API de OCS esté en línea

## Los saldos muestran cantidades incorrectas

- **Causa:** Desajuste de datos de OCS o problema de caché
- **Solución:**
  - Fuerza la actualización del saldo de OCS
  - Verifica si hay transacciones pendientes
  - Verifica la configuración de OCS

## Fechas de caducidad faltantes

- **Causa:** El saldo no tiene fecha de caducidad establecida
- **Solución:**
  - Algunos saldos están configurados para nunca caducar (validez ilimitada)
  - Verifica la configuración del saldo en OCS

## Múltiples saldos confusos

- **Causa:** Se han añadido múltiples recargas o datos de bonificación
- **Solución:**
  - Usa la vista de desglose detallado para ver todos los buckets
  - Ordena por peso para ver el orden de consumo
  - Revisa las fechas de caducidad individuales

## Integración con OCS/CGRateS

Los datos de uso provienen del **OCS (Sistema de Carga en Línea)**, típicamente CGRateS.

### Flujo de Datos:

1. El usuario abre la vista de uso
2. OmniCRM llama a GET /crm/service/{service\_id}
3. La API consulta OCS a través del UUID del servicio
4. OCS devuelve el mapa de saldo:

```
{  
  "BalanceMap": {  
    "*data": [  
      {  
        "ID": "monthly_data_20GB",  
        "Value": 13421772800,  
        "ExpiryTime": "2025-01-25T23:59:59Z",  
        "Weight": 10  
      }  
    ],  
    "*voice": [  
      {  
        "ID": "monthly_voice_500min",  
        "Value": 7500000000000,  
        "ExpiryTime": "2025-01-25T23:59:59Z",  
        "Weight": 10  
      }  
    ],  
    "*sms": [  
      {  
        "ID": "monthly_sms_250",  
        "Value": 250,  
        "ExpiryTime": "2025-01-25T23:59:59Z",  
        "Weight": 10  
      }  
    ]  
  }  
}
```

```

        "Weight": 10
    }
],
"*monetary": [
{
    "ID": "prepaid_credit",
    "Value": 1550,
    "ExpiryTime": "2025-02-25T23:59:59Z",
    "Weight": 10
}
]
}
}

```

5. La UI convierte los valores a unidades de visualización (bytes → GB, nanosegundos → minutos)
6. Se renderizan las barras de progreso y las tarjetas
7. El polling continúa cada 3 segundos

### **Mapeo de Tipos de Saldo de OCS:**

El OCS devuelve datos de saldo con prefijos de tipo que se mapean a la visualización de la UI:

- \*data → **TARJETA DE DATOS** (uso de internet)
- \*voice → **TARJETA DE VOZ** (minutos de llamada)
- \*sms → **TARJETA DE SMS** (mensajes de texto)
- \*monetary → **TARJETA MONETARIA** (crédito prepago)

Cada tipo de saldo puede tener múltiples buckets (por ejemplo, asignación mensual + datos de bonificación + datos de recarga), todos mostrados en la vista de desglose detallado.

### **Conversiones de Valores de Saldo:**

- **Datos:** Bytes → GB (dividir por 1024<sup>3</sup>)
- **Voz:** Nanosegundos → Minutos (dividir por  $60 \times 10^9$ )
- **SMS:** Conteo (sin conversión)
- **Monetario:** Céntimos → Moneda (dividir por 100)

## **Auto-Renovación y Planes de Acción**

Los servicios con auto-renovación habilitada tienen **Planes de Acción** programados en el OCS.

### **¿Qué son los Planes de Acción?**

Los Planes de Acción son tareas programadas en CGRateS que se ejecutan automáticamente en momentos específicos para:

- Agregar saldo a una cuenta (recarga automática)
- Renovar asignaciones mensuales
- Aplicar cargos recurrentes
- Expirar saldos antiguos

### **Cómo Funciona la Auto-Renovación:**

#### **1. Provisionamiento del Servicio:**

- Cuando se crea el servicio con `auto_renew = true`
- El libro de provisión crea un Plan de Acción en OCS

- El Plan de Acción está configurado para ejecutarse mensualmente (o por ciclo de facturación)

## 2. Configuración del Plan de Acción:

El Plan de Acción contiene:

- **ID de Cuenta** - UUID del servicio
- **Acciones** - Qué hacer (agregar datos, voz, SMS, saldo monetario)
- **Horario** - Cuándo ejecutar (por ejemplo, mensualmente el 1 a las 00:00 UTC)
- **Monto** - Cuánto saldo agregar

## 3. Ejecución Automática:

- OCS ejecuta el Plan de Acción a la hora programada
- Agrega saldo a la cuenta (por ejemplo, 20GB de datos, 500 minutos de voz)
- Establece la fecha de caducidad para el nuevo saldo (por ejemplo, 30 días)
- Se cobra al cliente a través del método de pago registrado

## 4. Visualización de Planes de Acción:

- Navega a los detalles del servicio en la vista de OCS
- Los Planes de Acción se enumeran con la próxima hora de ejecución
- Muestra: Nombre del plan, fecha de próxima ejecución, detalles de la acción

### Ejemplo de Plan de Acción:

```
{
  "ActionPlanId": "ProductID_MonthlyPlan__ProductName_20GB_Mobile__ActionPlan_Monthly_Renew",
  "NextExecTime": "2025-02-01T00:00:00+00:00",
  "ActionName_hr": "Renovación Mensual",
  "PlanName": "20GB Móvil",
  "ActionFrequency_hr": "Cada Plan Mensual",
  "custom_NextExecTime_hr": "en 22 días"
}
```

### Gestión de Auto-Renovación:

- **Habilitar** - Establecer durante la creación o modificación del servicio
- **Deshabilitar** - Eliminar el Plan de Acción de OCS (el servicio mantiene el saldo existente pero no se renovará automáticamente)
- **Modificar** - Cambiar el monto de renovación o la frecuencia mediante la modificación del servicio

### Renovación Manual:

Si la auto-renovación está deshabilitada, el cliente debe manualmente:

- Recargar antes de que el saldo caduque
- O el servicio se suspende cuando se agota el saldo

### Visualización en la UI:

La pestaña de servicios muestra el estado de auto-renovación:

Próxima Renovación: 1 de febrero de 2025 (en 22 días) Monto de Renovación: £15.00

# Mejores Prácticas

## Para el Personal de Soporte:

- Verifica el uso antes de responder a llamadas "¿Por qué mi servicio es lento?"
- Verifica el saldo después de las recargas para confirmar el éxito
- Usa el desglose detallado para identificar buckets caducados
- Monitorea a los clientes de alto uso para prevenir sobrecargas

## Para los Clientes (Autocuidado):

- Verifica el uso regularmente para evitar quedarte sin saldo
- Recarga antes de que el saldo caduque
- Entiende el sistema de peso para saber qué saldo se consume primero
- Contacta al soporte si el uso parece incorrecto

## Para los Administradores:

- Configura las caducidades de saldo apropiadas
- Establece valores de peso para priorizar los saldos que están por caducar
- Monitorea la conectividad de OCS para informes precisos
- Revisa que las configuraciones de saldo coincidan con las ofertas de productos

## Documentación Relacionada

- `features_topup_recharge` - Sistema de recarga para agregar saldo
- `csa_service_management` - Gestión de servicios
- `csa_modify` - Agregar complementos para aumentar las asignaciones
- `concepts_products_and_services` - Configuración de productos



# Atención al Cliente

## Suplantación de Usuario para Soporte y Solución de Problemas

La suplantación permite al personal autorizado iniciar sesión temporalmente como otro usuario para solucionar problemas, verificar configuraciones o ver exactamente lo que el usuario experimenta. Esta función es esencial para el soporte al cliente, pero requiere permisos apropiados y está completamente auditada.

Al suplantar a un cliente, el personal accede al Self-Care Portal `<self_care_portal>` exactamente como lo ve el cliente, lo que permite una solución de problemas y soporte precisos.

Véase también: RBAC `<rbac>` para la configuración de permisos, Customers `<basics_customers>` para la gestión de clientes, Self-Care Portal `<self_care_portal>` para las características del portal del cliente.

## Propósito

La suplantación de usuarios proporciona:

1. **Solución de Problemas** — Ver exactamente lo que el cliente ve para diagnosticar problemas
2. **Verificación** — Confirmar que las configuraciones y permisos del servicio funcionan correctamente
3. **Capacitación** — Demostrar características desde la perspectiva del cliente
4. **Soporte** — Ayudar a los clientes a navegar por el sistema sin requerir compartir pantalla
5. **Registro de Auditoría** — Todas las sesiones de suplantación se registran para seguridad y cumplimiento

## Permisos Requeridos

Para suplantar usuarios, debe tener uno de los siguientes permisos:

- `can_impersonate` — Permiso de suplantación dedicado para el personal de soporte
- `admin` — Acceso administrativo completo (incluye derechos de suplantación)

Los usuarios sin estos permisos no pueden acceder a la función de suplantación.

## Cómo Suplantar a un Usuario

### A través de la Interfaz Web:

1. **Navegar al Cliente** — Encontrar al cliente en el CRM
2. **Seleccionar Contactos** — Ver la lista de contactos del cliente
3. **Hacer clic en "Iniciar sesión como Usuario"** — El botón aparece junto a cada contacto que tiene una cuenta de usuario
4. **Confirmar Suplantación** — El sistema puede solicitar confirmación
5. **La Sesión Comienza** — Ahora ha iniciado sesión como ese usuario

Iniciando sesión como un Usuario en OmniCRM

### A través de la API:

Iniciar sesión de suplantación:

**Endpoint:** POST /auth/impersonate

**Permiso Requerido:** can\_impersonate o admin

### Solicitud:

```
{  
  "user_id": 42  
}
```

### Respuesta:

```
{  
  "success": true,  
  "impersonating_user_id": 1,  
  "target_user_id": 42,  
  "impersonation_start": "2025-01-04T15:30:00Z",  
  "access_token": "new_token_for_impersonated_user",  
  "refresh_token": "new_refresh_token"  
}
```

Los tokens devueltos son para la sesión del usuario suplantado.

## Qué Ocurre Durante la Suplantación

Cuando suplantas a un usuario:

- **Contexto Completo** — Ves el sistema exactamente como lo ve el usuario objetivo:

- Su panel de control y navegación
- Sus datos de cliente (si es un usuario cliente)
- Sus permisos y controles de acceso
- Sus servicios, facturas y uso
- **Seguimiento de Sesiones** — El sistema rastrea ambas identidades:
  - `impersonating_user_id` — Tu ID de usuario real
  - `target_user_id` — El usuario que estás suplantando
  - `impersonation_start` — Cuándo comenzó la suplantación
- **Indicador** — La interfaz muestra un banner que indica que estás suplantando a alguien:
  - "Actualmente has iniciado sesión como [Nombre de Usuario]"
  - "Haz clic aquí para detener la suplantación"
- **Registro de Auditoría** — Todas las acciones se registran con ambos IDs de usuario:
  - Las acciones aparecen como realizadas por el usuario objetivo
  - Los registros de auditoría registran quién las realizó realmente (usuario suplantador)
  - Se mantiene un registro completo de auditoría en la tabla `ImpersonationLog`

## Detener la Suplantación

### A través de la Interfaz Web:

1. **Hacer clic en el Banner** — Haz clic en el banner de suplantación en la parte superior de la página
2. **O Navegar** — Ir al menú de usuario y seleccionar "Detener Suplantación"
3. **Confirmación** — La sesión termina y regresas a tu propia cuenta

### A través de la API:

**Endpoint:** POST /auth/stop\_imPERSONATION

**Solicitud:** No se requiere cuerpo (solicitud autenticada)

### Respuesta:

```
{
  "success": true,
  "impersonation_end": "2025-01-04T15:45:00Z",
  "duration_seconds": 900,
  "access_token": "your_original_token",
  "refresh_token": "your_original_refresh_token"
}
```

Tu sesión original se restaura.

# Registro de Auditoría de Suplantación

Todas las sesiones de suplantación se registran en la tabla ImpersonationLog con:

- **impersonating\_user\_id** — Miembro del personal que realizó la suplantación
- **target\_user\_id** — Cliente o usuario que fue suplantado
- **impersonation\_start** — Marca de tiempo de inicio
- **impersonation\_end** — Marca de tiempo de finalización (cuando se detuvo la sesión)
- **impersonation\_duration** — Duración en segundos

Esto proporciona una responsabilidad completa por todas las sesiones de suplantación y permite:

- **Auditorías de seguridad** — Revisar quién suplantó a quién y cuándo
- **Informes de cumplimiento** — Demostrar el uso adecuado de acceso elevado
- **Investigación** — Rastrear acciones durante las sesiones de suplantación
- **Monitoreo** — Identificar patrones inusuales de suplantación

## Visualización de Registros de Suplantación:

Los administradores pueden consultar los registros de suplantación a través de la API:

```
GET /auth/  
impersonation_logs?user_id={user_id}&start_date={date}&end_date={date}
```

Filtrar por:

- Usuario suplantador (quién realizó la suplantación)
- Usuario objetivo (quién fue suplantado)
- Rango de fechas
- Duración

## Mejores Prácticas

1. **Minimizar la Duración** — Suplantar solo el tiempo necesario para resolver el problema
2. **Documentar el Propósito** — Anotar por qué se necesitó la suplantación en el registro de actividad del cliente
3. **Informar a los Clientes** — Hacer saber a los clientes que puede que necesite ver su cuenta (política de privacidad)
4. **Verificar la Identidad** — Confirmar la identidad del usuario antes de suplantar a través de su cuenta
5. **Revisar Registros** — Auditarse regularmente los registros de suplantación en busca de patrones inusuales

6. **Limitar Permisos** — Solo otorgar `can_impersonate` al personal de soporte que lo necesite
7. **Capacitación** — Asegurarse de que el personal entienda la responsabilidad y las implicaciones de auditoría

## Consideraciones de Seguridad

- **Acceso Completo** — La suplantación otorga acceso completo a la cuenta del usuario objetivo
- **No se Requiere Contraseña** — La suplantación elude la autenticación (solo basada en permisos)
- **Acciones Registradas** — Todas las acciones durante la suplantación se atribuyen al usuario objetivo en los registros de la aplicación (pero los registros de auditoría muestran al actor real)
- **Aislamiento de Sesiones** — La suplantación crea una nueva sesión; no afecta las sesiones activas del usuario objetivo
- **Límites de Tiempo** — Las sesiones de suplantación deben tener un límite de tiempo (configurable)
- **Eliminación de MFA** — La suplantación elude los requisitos de 2FA (utiliza la autenticación del suplantador)

## Restricciones

- **No se Puede Suplantar a Administradores** — Dependiendo de la configuración, puede que no se pueda suplantar a otros administradores
- **Mismos Permisos** — Obtienes los permisos del usuario objetivo, no la unión de los permisos de ambos usuarios
- **Límites de Sesión** — Solo se permite una sesión de suplantación por miembro del personal a la vez
- **Requisitos de Auditoría** — No se puede desactivar ni ocultar el registro de suplantación

## Solución de Problemas Comunes

### Problema: El botón "Iniciar sesión como Usuario" no aparece

- Solución: Verifica que tengas el permiso `can_impersonate` o `admin`
- Solución: Confirma que el contacto tenga una cuenta de usuario vinculada

### Problema: La suplantación falla con un error de permiso

- Solución: Verifica si el usuario objetivo es un administrador (puede estar restringido)
- Solución: Verifica que tu permiso de suplantación esté activo

### Problema: No se puede detener la suplantación

- Solución: Usa el endpoint de la API para detener la suplantación: POST `/auth/stop_ impersonation`
- Solución: Borra las cookies del navegador e inicia sesión nuevamente con tus credenciales

### **Problema: Las acciones no se registran correctamente**

- Solución: Verifica que la sesión de suplantación esté activa (verifica el banner)
- Solución: Revisa los registros de auditoría: las acciones se registran con ambos IDs de usuario



# Sistema de Difusión Celular

El Sistema de Difusión Celular en OmniCRM permite a los operadores de redes móviles enviar alertas de emergencia y advertencias públicas a dispositivos móviles dentro de áreas geográficas específicas. La Difusión Celular es una característica crítica de seguridad pública utilizada para alertas AMBER, advertencias meteorológicas, alertas de tsunamis y otras notificaciones de emergencia.

**Ventaja Clave:** A diferencia de los mensajes SMS estándar, los mensajes de Difusión Celular alertarán audiblemente en teléfonos que están en silencio, sin crédito o en roaming. Como se trata de un mensaje de difusión, es posible enviar una alerta a cada miembro de la población que tenga un teléfono móvil en cuestión de segundos.

## Enlace de Advertencia Omnitouch (OWL)

La plataforma Enlace de Advertencia Omnitouch (OWL) proporciona una solución integral para profesionales de gestión de desastres y operadores de redes móviles:

- **Entidad de Difusión Celular (CBE)** - Aplicación web segura para que usuarios autorizados creen y difundan mensajes de advertencia de emergencia.
- **Centro de Difusión Celular (CBC)** - Componente de integración de red conforme a estándares que se conecta a redes celulares (2G/3G/4G/5G) para distribuir mensajes.

OWL está diseñado para ser utilizado en cualquier dispositivo con un navegador web (Chrome/Firefox/Safari/Edge), como computadoras, laptops, tabletas o teléfonos móviles.

## Descripción General

La Difusión Celular (también conocida como Sistema de Advertencia Pública o PWS) permite a los operadores:

- **Enviar Alertas de Emergencia** - Distribuir información crítica de seguridad a todos los dispositivos en un área.
- **Dirigir Regiones Geográficas** - Difundir a áreas de seguimiento específicas o celdas de red controlando qué torres celulares difunden mensajes.
- **Soportar Múltiples Idiomas** - Proporcionar mensajes de alerta en

- múltiples idiomas simultáneamente (hasta 500 caracteres por idioma).
- **Gestionar el Ciclo de Vida de la Alerta** - Crear, actualizar, aprobar, monitorear y eliminar mensajes de difusión.
- **Integrarse con Sistemas Externos** - Conectar con la infraestructura del CBC (Centro de Difusión Celular) a través de múltiples interfaces de red celular.
- **Autenticación de Dos Factores** - Proceso de aprobación seguro utilizando Contraseñas de Un Solo Uso Basadas en Tiempo (TOTP).
- **Regla de Dos Personas** - Requisito opcional de aprobación de una segunda persona antes de la transmisión del mensaje.

A diferencia de los SMS, la Difusión Celular no requiere listas de suscriptores y puede alcanzar todos los dispositivos capaces en un área geográfica de inmediato, lo que la hace ideal para alertas de seguridad pública críticas en tiempo. En la mayoría de las redes, una difusión a todos los dispositivos toma menos de 10 segundos.

## Casos de Uso

La Difusión Celular se utiliza para:

- **Advertencias de Emergencia** - Desastres naturales (terremotos, tsunamis, inundaciones, incendios).
- **Alertas AMBER** - Notificaciones de secuestro de niños.
- **Alertas Meteorológicas** - Advertencias de clima severo, alertas de tornados.
- **Seguridad Pública** - Amenazas terroristas, derrames químicos, evacuaciones.
- **Mensajes de Prueba** - Pruebas del sistema y campañas de concienciación pública (los Mensajes de Prueba Mensuales no alertan a los usuarios, pero serán recibidos).

El direccionamiento geográfico es a menudo de suma importancia en situaciones de emergencia. Por ejemplo, las alertas de tsunamis que aconsejan a aquellos cerca de áreas costeras buscar terrenos más altos no deben enviarse a personas en áreas interiores lejos del peligro. Al controlar qué torres celulares difunden mensajes de advertencia de emergencia, el alcance de la transmisión puede limitarse al área geográfica apropiada.

## Ciclo de Vida del Mensaje de Advertencia de Emergencia

Transmitir un mensaje de Advertencia de Emergencia requiere acción rápida mientras se proporciona información precisa y autenticación para garantizar la validez del mensaje.

El ciclo de vida del mensaje consta de cuatro etapas:

1. **Definición del Mensaje** - Tipo de mensaje, contenido del mensaje, configuraciones de expiración.
2. **Dirección del Mensaje** - Áreas geográficas y torres celulares a las que se difundirán.
3. **Aprobación del Mensaje** - Autorización/verificación de la identidad del operador y aprobación de la segunda persona (si es necesario).
4. **Revisión del Mensaje** - Confirmación final antes de la transmisión.

Después de la difusión, los mensajes pueden ser monitoreados, actualizados a medida que evolucionan las situaciones y detenidos cuando el peligro inmediato disminuye.

## **Etapa 1: Definición del Mensaje**

Este paso define los parámetros básicos del mensaje de advertencia de emergencia que se va a difundir:

### **Identificador del Mensaje**

Diferentes tipos de mensajes tienen diferentes identificadores, que son tratados de manera diferente por los teléfonos receptores. Por ejemplo, un Mensaje de Prueba Mensual no debería alertar a los usuarios reales, pero aún así recibirán el mensaje.

**Nota:** Cada plantilla de mensaje disponible en el sistema OWL ya tiene el Identificador de Mensaje (MI) apropiado incrustado, por lo que los usuarios del sistema no necesitan seleccionarlo manualmente al crear un mensaje de alerta.

### **Texto del Mensaje**

Cuerpo de texto limitado a 500 caracteres que contiene el mensaje que se mostrará a los usuarios finales. Los mensajes pueden proporcionarse en **múltiples idiomas** agregando el texto del segundo idioma debajo del primero. Recuerde que el límite de 500 caracteres se aplica al mensaje total, incluidos todos los idiomas.

### **Plantillas de Mensaje**

Los mensajes de alerta de peligro pueden predefinirse con anticipación como "plantillas" para diferentes escenarios previstos, como:

- Inundaciones
- Tsunamis
- Terremotos
- Pruebas periódicas
- Otros escenarios de desastre

Las plantillas ahorran tiempo durante las emergencias. Estas plantillas pueden modificarse según sea necesario al definir el mensaje, o los mensajes pueden ser escritos desde cero.

## **Expiración y Repeticiones del Mensaje**

Los mensajes de emergencia tienen una vida útil finita para la cual son relevantes. Al definir el mensaje:

- **Expira (minutos)** - Cuánto tiempo continuará siendo difundido el mensaje.
- **Repeticiones del Mensaje** - Cuántas veces será retransmitido.

Cada teléfono solo mostrará el mensaje al usuario una vez. Sin embargo, los sitios celulares continuarán transmitiendo mensajes hasta que se alcance el tiempo de expiración para garantizar que las personas que ingresen al área de cobertura desde afuera reciban el mensaje.

## **Etapa 2: Direccionamiento**

Los mensajes de Difusión Celular se envían a nivel de torre celular, y el alcance geográfico puede limitarse seleccionando qué torres difunden el mensaje.

### **Direccionamiento Opcional**

Este paso es opcional. No ingresar ninguna información de direccionamiento significa que todas las torres celulares transmitirán el mensaje de advertencia de emergencia.

### **Áreas de Objetivo Predefinidas**

El sistema OWL tiene una base de datos de todas las torres celulares y puede definir áreas de objetivo en un mapa. Las áreas pueden ser dirigidas utilizando zonas predefinidas (determinadas con anticipación para una selección rápida) o dibujando áreas personalizadas en el mapa.

### **Herramientas de Dibujo en el Mapa**

Se pueden crear áreas de objetivo personalizadas utilizando:

- Herramienta de Polígono - Dibujar límites de cobertura precisos.
- Herramienta de Círculo - Alertas rápidas basadas en radio.
- Herramienta de Rectángulo - Cobertura alineada a la cuadrícula.

La función "Aregar Nueva Zona" permite definir áreas de objetivo personalizadas que pueden guardarse para uso futuro.

## **Etapa 3: Aprobación**

Un proceso de aprobación valida que la persona que emite el mensaje de Advertencia de Emergencia está autorizada para hacerlo.

### **Autenticación de Dos Factores**

Utiliza Contraseñas de Un Solo Uso Basadas en Tiempo (TOTP) a través de:

- **Token físico** (como un RSA SecurID).
- **Solución basada en aplicación** (Google Authenticator, Authy, Microsoft

Authenticator, u otras aplicaciones compatibles con TOTP).

Los usuarios que crearán o aprobarán borradores de alertas de mensajes deben tener una aplicación de autenticador en su teléfono inteligente para generar el código de autorización que el sistema solicitará.

## Configuración de 2FA

Al configurar por primera vez 2FA:

1. Instale una aplicación de autenticador en su teléfono inteligente (Google Authenticator, Authy, Microsoft Authenticator, etc.).
2. Navegue a la configuración de su cuenta OWL y escanee el código QR con su aplicación de autenticador.
3. Ingrese el código de verificación para confirmar la configuración.
4. Guarde los códigos de respaldo en un lugar seguro.
5. Pruebe la generación de códigos antes de las emergencias.

Para obtener instrucciones detalladas sobre la configuración de 2FA, consulte Autenticación de Dos Factores <2fa>.

Si cambia de dispositivo móvil o la aplicación deja de sincronizarse con su cuenta OWL, comuníquese con su Administrador del Sistema para obtener ayuda. Los administradores pueden restablecer los tokens de 2FA desde la página **Usuarios y Roles → Usuarios**.

## Regla de Dos Personas

Cuando el proceso requiere la aprobación de una segunda persona, la persona que emite la alerta debe ingresar la Contraseña de Un Solo Uso Basada en Tiempo de la otra persona antes de que se permita continuar con el proceso. Esto proporciona supervisión y minimiza el riesgo de abuso.

## Roles de Usuario Granulares

Los roles de usuario individuales pueden configurarse para:

- Permitir que solo ciertos usuarios envíen mensajes predefinidos.
- Restringir el direccionamiento de mensajes a regiones específicas.
- Requerir flujos de trabajo de aprobación adicionales.

## Etapa 4: Revisión

Una vez que se completan las etapas de Definición del Mensaje, Direccionamiento y Aprobación, el operador debe revisar el mensaje antes de la difusión final. Una vez satisfecho con los detalles del mensaje, puede transmitir el mensaje.

**Velocidad de Transmisión:** En la mayoría de las redes, una difusión a todos los dispositivos en la red toma menos de 10 segundos.

## **Etapa 5: Monitoreo y Actualizaciones**

Una vez que se inicia la difusión del mensaje, los operadores pueden monitorear y gestionar los mensajes transmitidos.

### **Retroalimentación de la Red**

Las redes celulares devuelven información sobre los sitios celulares que han difundido el mensaje. Si un sitio celular está fuera de línea o no disponible, esto se informará al operador.

### **Retransmisión Automática**

Si algún sitio celular fuera de línea vuelve a estar disponible mientras la Advertencia de Emergencia sigue activa, todos los teléfonos conectados a esa celda recibirán el mensaje.

### **Actualizaciones en Vuelo**

Una vez difundido, el mensaje puede ser:

- Actualizado a medida que evoluciona la situación.
- Modificado con nuevo contenido del cuerpo del mensaje.
- Recordado/detenido en cualquier momento.

### **Registros Históricos**

Toda la información sobre mensajes históricos puede ser vista y revisada para fines de auditoría.

## **Estructura del Mensaje**

Cada mensaje de Difusión Celular consta de:

### **Configuración del Mensaje**

- **Identificador del Mensaje** - Identificador único para el tipo de alerta (por ejemplo, 4370 para ETWS Terremoto, 4371 para ETWS Tsunami).
- **Categoría** - Categoría de alerta (normal, emergencia, alta, extrema).
- **Período de Repetición** - Segundos entre repeticiones de difusión.
- **Número de Difusiones** - Cuántas veces difundir el mensaje.
- **Período de Advertencia** - Duración en segundos durante la cual la advertencia es válida.
- **Indicador de Canal** - Tipo de canal utilizado para la difusión.

### **Mensajes Localizados**

Cada mensaje del CBC puede incluir múltiples variantes de idioma:

- **Idioma** - Código de idioma ISO (en, es, fr, zh, etc.).
- **Cuerpo del Mensaje** - Texto de alerta en ese idioma (hasta 1395 caracteres).

El sistema transmite automáticamente todas las variantes de idioma, permitiendo a los destinatarios ver alertas en su idioma preferido.

## Áreas de Seguimiento

Define el direccionamiento geográfico para la alerta:

- **Área de Seguimiento** - Identificador geográfico (ID de celda, código de área de seguimiento).
- **Operador** - Código del operador de red móvil (MCC-MNC).
- **Tipo de RAT** - Tecnología de Acceso Radio (LTE, 5G, UMTS, GSM).

Se pueden especificar múltiples áreas de seguimiento para cubrir regiones más grandes o múltiples operadores.

# Creando un Mensaje de Difusión Celular

## A través de la Interfaz Web:

1. **Navegar a Difusión Celular** - Acceda a la interfaz de gestión del CBC desde la navegación principal.
2. **Hacer clic en "Crear Alerta"** - Abre el formulario de creación de mensajes.
3. **Configurar Parámetros del Mensaje:**
  - Identificador del Mensaje (por ejemplo, 4370 para alertas de terremoto).
  - Categoría (normal, alta, extrema).
  - Período de Repetición (típicamente 5-60 segundos).
  - Número de Difusiones (999 para continuo, o conteo específico).
  - Período de Advertencia (duración en segundos).
  - Indicador de Canal (típicamente "básico").

Definición del Mensaje de Difusión Celular

4. **Agregar Mensajes Localizados:**
  - Hacer clic en "Agregar Idioma".
  - Seleccionar idioma del menú desplegable.
  - Ingresar texto del mensaje (máx. 1395 caracteres para GSM7, menos para Unicode).
  - Repetir para idiomas adicionales.
5. **Definir Áreas de Seguimiento:**
  - Hacer clic en "Agregar Área de Seguimiento".
  - Ingresar código de área de seguimiento.
  - Seleccionar operador (combinación MCC-MNC).
  - Elegir tipo de RAT (LTE, 5G, etc.).
  - Repetir para áreas geográficas adicionales.

Selección de Área de Seguimiento de Difusión Celular

**6. Revisar y Crear** - Verificar todos los detalles y hacer clic en "Crear Alerta".

Revisión del Mensaje de Difusión Celular

**A través de la API:**

**Endpoint:** PUT /crm/cbc/

**Permiso Requerido:** CREATE\_CBC\_MESSAGE

**Cuerpo de la Solicitud:**

```
{  
  "messageIdentifier": "4370",  
  "category": "emergency",  
  "repetitionPeriod": 10,  
  "numberOfBroadcasts": 999,  
  "warningPeriodSec": 3600,  
  "channelIndicator": "basic",  
  "localized_messages": [  
    {  
      "language": "en",  
      "messageBody": "EARTHQUAKE WARNING: Magnitude 6.5 earthquake  
detected. Take cover immediately. Drop, Cover, Hold On."  
    },  
    {  
      "language": "es",  
      "messageBody": "ADVERTENCIA DE TERREMOTO: Terremoto de magnitud  
6.5 detectado. Cúbrase inmediatamente. Agáchese, Cúbrase, Agárrese."  
    }  
  ],  
  "tracking_areas": [  
    {  
      "tracking_area": "12345",  
      "operator": "310-410",  
      "rat_type": "LTE"  
    },  
    {  
      "tracking_area": "12346",  
      "operator": "310-410",  
      "rat_type": "5G"  
    }  
  ]  
}
```

**Respuesta:**

```
{  
  "cbc_message_id": 123,
```

```
  "cbc_unique_id": "550e8400-e29b-41d4-a716-446655440000",
  "messageIdentifier": "4370",
  "category": "emergency",
  "repetitionPeriod": 10,
  "numberOfBroadcasts": 999,
  "warningPeriodSec": 3600,
  "channelIndicator": "basic",
  "initiating_user": 5,
  "approving_user": null,
  "created": "2025-01-10T14:30:00Z",
  "localized_messages": [...],
  "tracking_areas": [...]
}
```

El mensaje se envía inmediatamente al Centro de Difusión Celular para su transmisión.

## Gestión de Mensajes Existentes

### Ver Todos los Mensajes

**Endpoint:** GET /crm/cbc/

**Permiso Requerido:** VIEW\_CBC\_MESSAGE

Devuelve la lista de todos los mensajes del CBC con su estado, marcas de tiempo y configuración.

### Actualizar un Mensaje

**Endpoint:** PATCH /crm/cbc/{cbc\_message\_id}

**Permiso Requerido:** UPDATE\_CBC\_MESSAGE

Actualiza el contenido del mensaje, áreas de seguimiento o parámetros de difusión. Los mensajes actualizados se reenvían al CBC.

### Cuerpo de la Solicitud:

```
{
  "cbc_message_id": 123,
  "numberOfBroadcasts": 500,
  "localized_messages": [
    {
      "language": "en",
      "messageBody": "UPDATED: Earthquake warning still in effect.
Aftershocks possible."
    }
  ]
}
```

```
    ]  
}
```

## Eliminar un Mensaje

**Endpoint:** DELETE /crm/cbc/{cbc\_message\_id}

**Permiso Requerido:** DELETE\_CBC\_MESSAGE

Elimina el mensaje de la base de datos e intenta cancelarlo en el CBC.

## Flujo de Trabajo de Aprobación

Los mensajes de Difusión Celular soportan un flujo de trabajo de aprobación opcional para alertas de alto riesgo:

1. **Usuario Iniciador** - Miembro del personal que crea la alerta (`initiating_user` field).
2. **Usuario Aprobador** - Gerente que aprueba la alerta antes de la difusión (`approving_user` field).

Si se requiere aprobación:

- El mensaje se crea con `approving_user = null`.
- El mensaje se mantiene en estado de "aprobación pendiente".
- El usuario aprobador revisa el mensaje y lo aprueba o rechaza.
- Con la aprobación, se establece `approving_user` y se difunde el mensaje.

Este flujo de trabajo es configurable según la política de la organización.

## Identificadores de Mensaje

Los identificadores de mensaje estándar siguen 3GPP TS 23.041:

### Sistema de Advertencia de Terremoto y Tsunami (ETWS):

- **4370** - Advertencia de Terremoto ETWS.
- **4371** - Advertencia de Tsunami ETWS.
- **4372** - Advertencia Combinada de Terremoto y Tsunami ETWS.
- **4373-4378** - Otros Tipos de Emergencia ETWS.
- **4379** - Mensaje de Prueba ETWS.

### Sistema de Alerta Móvil Comercial (CMAS) / Alertas de Emergencia Inalámbricas (WEA):

- **4352** - Alerta Presidencial.
- **4353-4355** - Alertas Extremas.

- **4356-4359** - Alertas Severas.
- **4360-4363** - Alertas AMBER.
- **4364-4367** - Mensajes de Seguridad Pública.
- **4368-4369** - Pruebas Estatales/Locales.
- **4380-4381** - Mensajes de Prueba.

### **Rangos Personalizados:**

- **0-999** - Reservado para alertas específicas del operador.
- **1000-4095** - Tipos de mensajes personalizados.

## **Integración con el Centro de Difusión Celular**

La Entidad de Difusión Celular (CBE) necesita un mecanismo para entregar mensajes a redes celulares individuales. El Centro de Difusión Celular (CBC) de OWL se conecta a cada red celular para enviar mensajes de Advertencia de Emergencia al público.

### **Redundancia Multi-Network**

En caso de que un operador de red celular no esté disponible (corte o sin cobertura), si hay otra red celular operativa disponible, los usuarios seguirán recibiendo mensajes de Advertencia de Emergencia a través de las otras redes disponibles.

### **Instancias de CBC por Operador**

Para asegurar que no haya conexión entre redes competidoras, OWL ejecuta una instancia de CBC separada para cada MNO; no se comparte entre operadores.

## **Configuración de OWL CBC**

La URL del CBC se configura en `crm_config.yaml`:

```
cbc_url: "http://cbc.example.com:8080"
```

### **Transmisión de Mensajes:**

Cuando se crea o actualiza un mensaje del CBC:

1. OmniCRM CBE almacena el mensaje en su base de datos.
2. El mensaje se formatea para la API del CBC.
3. Se envía una solicitud HTTP POST a `{cbc_url}/alerts/send`.
4. El CBC se conecta a la red celular a través de la interfaz apropiada (CBSP, SBc-AP, o N50).
5. El CBC reconoce la recepción y comienza la difusión.
6. Los dispositivos en las áreas de seguimiento objetivo reciben la alerta.

### **Eliminación de Mensajes:**

Cuando se elimina un mensaje, el CBE intenta cancelarlo en el CBC para detener las difusiones en curso.

## **Puntos de Integración de Redes Celulares**

Diferentes generaciones de redes celulares (2G/3G/4G/5G) tienen interfaces únicas para conectarse desde/hacia el Centro de Difusión Celular. Según las tecnologías utilizadas por cada red celular, se debe configurar la interfaz correcta.

El Centro de Difusión Celular de OWL soporta interfaces de Difusión Celular de 2G, 3G, 4G y 5G y se ha integrado con numerosos componentes de red celular comúnmente utilizados.

### **CBSP - 2G/3G - Controlador de Estación Base (BSC)**

La interfaz del Protocolo de Servicio de Difusión Celular (CBSP) conecta el CBC al Controlador de Estación Base (BSC) que controla estaciones base de 2G (GSM).

- Utilizado para mensajería de Difusión Celular de 2G y 3G con despliegues combinados de Controlador de Red de Radio/Controlador de Estación Base.
- Puede configurarse como un cliente o un servidor dependiendo del proveedor del BSC.
- Se debe establecer una conexión entre todos los BSC en la red y el CBC de OWL.
- Las interfaces se monitorean continuamente con alertas para indicar si un enlace CBSP ha caído.

Nota: 3GPP definió el Protocolo de Difusión de Área de Servicio (SABP) para su uso en despliegues RNC independientes. Esto puede usarse si es necesario cuando CBSP no es compatible con celdas 3G en un RNC/BSC combinado, sin embargo, pueden ser necesarias pruebas adicionales y soporte del proveedor de RNC.

### **SBc-AP - 4G/5G No Independiente - MME/IWF**

La interfaz SBc-AP conecta el CBC de OWL al MME (Entidad de Gestión de Movilidad) que sirve a eNodeB/gNodeBs de 4G y 5G.

- Utilizado para redes LTE de 4G.
- También utilizado para 5G No Independiente (la mayoría de los despliegues a partir de 2025).
- Se debe establecer una conexión entre todos los MMEs en la red y el CBC de OWL.
- Las interfaces se monitorean continuamente con alertas para indicar si un enlace SBc-AP ha caído.

### **N50 - 5G Independiente - AMF**

Para redes 5G independientes, la interfaz N50 conecta el CBC de OWL al AMF (Función de Gestión de Acceso y Movilidad) que sirve a gNodeBs de 5G.

- La interfaz está presente en el CBC de OWL.
- No se ha probado extensamente con AMFs de terceros debido al pequeño número de redes 5G SA disponibles comercialmente en 2025.
- Se apoyará completamente a medida que los despliegues de 5G SA se vuelvan más comunes.

## **Requisitos de Redes de MNO**

Se debe establecer una red entre el CBC de OWL y la red del Operador de Red Móvil para alcanzar las interfaces descritas anteriormente.

Esto se maneja caso por caso, pero generalmente requiere:

- Conexión cruzada/fibra dedicada entre el CBC y la red MNO.
- Cada interfaz lógicamente separada.
- Conectividad a cada punto de integración (MME, RNC, BSC) en la red celular.

## **Equipos de Red Soportados**

El CBC de OWL ha sido probado e integrado con componentes de red celular comúnmente utilizados de proveedores importantes:

### **Integración de Datos de Sitios Celulares**

OWL soporta la recopilación automática de datos de:

- **Nokia NetAct**
- **Huawei U2000 / U2020**
- **ZTE NetNumen / ZXPOS**
- **Ericsson ENM**

Alternativamente, los datos del sitio celular pueden proporcionarse al equipo de operaciones de Omnitouch periódicamente por correo electrónico.

## **Gestión de Usuarios y Control de Acceso**

### **Control de Acceso Basado en Roles (RBAC)**

El sistema OWL utiliza control de acceso basado en roles (RBAC): las personas (Usuarios Registrados) se asignan uno o más Roles, y cada Rol es un conjunto de Permisos. Los Permisos son la unidad más pequeña de acceso (por ejemplo, crear alerta de mensaje borrador). El acceso efectivo de un Usuario Registrado es la unión de Permisos de todos los Roles asignados.

## Componentes de RBAC:

- **Usuarios** - Personas reales que inician sesión en el sistema OWL.
- **Permisos** - Micro capacidades (por ejemplo, aprobar borrador de mensaje, crear mensaje, ver informes).
- **Roles** - Conjuntos nombrados de permisos (por ejemplo, Aprobadores de Mensajes, Creadores de Mensajes).
- **Asignación** - Los usuarios reciben uno o más Roles; los permisos se agregan.

## Beneficios de RBAC:

1. **Protección de Datos** - Los usuarios solo ven y hacen lo que se les permite.
2. **Ajuste Operacional** - Los roles reflejan funciones laborales (Admin, Creador de Mensajes, Aprobador de Mensajes).
3. **Administración Simple** - Conceder acceso asignando roles; evitar la microgestión por usuario.

## Permisos del Sistema

Los permisos del sistema generalmente siguen patrones CRUD con cuatro opciones:

- **Ver** - Leer o navegar por mensajes e informes.
- **Crear** - Crear o agregar una alerta de mensaje.
- **Actualizar** - Editar o modificar una alerta de mensaje borrador.
- **Eliminar** - Eliminar o quitar una alerta de mensaje borrador.

## Permisos Básicos del CBC:

- CREATE\_CBC\_MESSAGE - Crear nuevos mensajes de difusión.
- VIEW\_CBC\_MESSAGE - Ver mensajes existentes y su estado.
- UPDATE\_CBC\_MESSAGE - Modificar el contenido del mensaje o parámetros de difusión.
- DELETE\_CBC\_MESSAGE - Eliminar mensajes y cancelar difusiones.

Asigne estos permisos a roles según las responsabilidades de seguridad pública de su organización.

Para obtener información completa sobre la gestión de roles y permisos de usuario, consulte **Usuarios y Roles <rbac>**.

## Gestión de Cuentas de Usuario

### Inicio de Sesión por Primera Vez

Cuando se configura un nuevo usuario en el sistema, se le proporcionan credenciales de inicio de sesión por parte del Administrador del Sistema:

1. Navegue a la URL de inicio de sesión de OWL proporcionada por su Administrador del Sistema.
2. Ingrese su nombre de usuario y contraseña.
3. Si 2FA está habilitado para su cuenta, se le pedirá que lo configure (consulte la sección de 2FA anterior).

**Mejor Práctica:** Los usuarios deben cambiar su contraseña inicial inmediatamente después del primer inicio de sesión por seguridad. Navegue a la configuración de su perfil de usuario para cambiar su contraseña.

Para obtener más detalles sobre los flujos de autenticación, consulte [Flujos de Autenticación y Controles de Administración <authentication\\_flows>](#).

## **Restablecimiento de Contraseña**

Los usuarios pueden restablecer sus propias contraseñas:

1. Desde la página de inicio de sesión del sistema, seleccione "Olvidé mi Contraseña".
2. Siga el proceso de verificación por correo electrónico.
3. Establezca una nueva contraseña.

## **Restablecimiento de Contraseña del Administrador:**

Los Administradores del Sistema pueden restablecer contraseñas para usuarios desde la página **Usuarios y Roles → Usuarios**:

1. Navegue a **Usuarios y Roles → Usuarios**.
2. Seleccione el usuario que requiere restablecimiento de contraseña.
3. Haga clic en **Restablecer Contraseña**.
4. El sistema genera una contraseña temporal segura y aleatoria (12 caracteres).
5. La contraseña temporal se muestra al administrador.
6. Proporcione la contraseña temporal al usuario a través de un canal seguro (teléfono o en persona, nunca por correo electrónico).
7. El usuario puede iniciar sesión con la contraseña temporal.
8. **Importante:** El usuario debe cambiar esta contraseña inmediatamente después de iniciar sesión.

Para obtener más detalles, consulte [Usuarios y Roles <rbac>](#).

## **Agregar Nuevos Usuarios al Sistema**

Los Administradores del Sistema pueden agregar nuevos usuarios navegando a **Usuarios y Roles → Usuarios**:

Información requerida para cada nuevo Usuario Registrado:

- Nombre

- Apellido
- Correo Electrónico
- Contraseña (temporal)
- Número de Móvil
- Rol(es)

**Mejor Práctica:** Implementar un Formulario de Solicitud de Nuevo Usuario del Sistema que se alinee con las políticas y procedimientos existentes de su organización. Solo agregue usuarios después de la aprobación adecuada.

## Editar Perfiles de Usuario

Los Administradores del Sistema pueden cambiar permisos y acceso para usuarios existentes:

1. Navegue a **Usuarios y Roles → Usuarios**.
2. Seleccione el usuario a editar.
3. Modifique los roles asignados según sea necesario.
4. Guarde los cambios.

## Eliminar Usuarios del Sistema

Los Administradores del Sistema pueden eliminar usuarios de **Usuarios y Roles → Usuarios**.

**Advertencia:** Una vez que un usuario ha sido eliminado del sistema OWL, la acción no puede deshacerse. Tenga mucho cuidado antes de ejecutar esta tarea.

## Ver Todos los Usuarios del Sistema

En cualquier momento, el Administrador del Sistema puede ver una lista de todos los Usuarios Registrados y ver qué permisos y roles tiene cada uno navegando a **Usuarios y Roles → Usuarios**.

## Historial de Mensajes de Alerta

El sistema OWL mantiene un registro de todos los mensajes de alerta enviados para fines de auditoría y para apoyar la revisión operativa.

### Accediendo al Historial de Mensajes:

Navegue a **Gestionar Difusiones Celulares** para ver:

- Todos los mensajes enviados anteriormente.
- Estado del mensaje (activo, expirado, cancelado).
- Marca de tiempo de transmisión.
- Usuarios iniciadores y aprobadores.
- Detalles de direccionamiento geográfico.

- Contenido y parámetros del mensaje.

Esto ayuda con:

- Auditoría del sistema.
- Revisión de procedimientos operativos utilizados por el NDMO y otras agencias de alerta.
- Comprender la efectividad del mensaje.
- Requisitos de cumplimiento e informes.

## Visualización del Mapa de Difusión Celular

El Mapa de Difusión Celular proporciona una **interfaz visual** para planificar áreas de cobertura geográfica e identificar qué torres celulares difundirán alertas.

### Accediendo al Mapa

Navegue a:

O directamente:

**Permiso Requerido:** VIEW\_CBC\_MESSAGE o CREATE\_CBC\_MESSAGE

### Características del Mapa

El mapa muestra:

#### Ubicaciones de Torres Celulares:

- **Íconos de Torres** - Posiciones de sitios celulares mostradas como íconos de torres celulares.
- **Indicadores de Sector** - Flechas direccionales que muestran la orientación del sector de la antena.
- **Nombres de Torres** - Identificadores y nombres de sitios.
- **Datos de Cobertura** - Cargados desde el archivo cellSites.txt.

#### Herramientas de Dibujo:

- **Herramienta de Polígono** - Dibujar áreas de cobertura personalizadas.
- **Herramienta de Círculo** - Crear zonas de difusión circulares.
- **Herramienta de Rectángulo** - Definir áreas de cobertura rectangulares.

#### Análisis de Cobertura:

- **Lista de Sitios** - Tabla que muestra todos los sitios celulares dentro del área dibujada.
- **Gestión de Zonas** - Guardar y cargar zonas de cobertura predefinidas.

- **Conteo de Sitios** - Conteo en tiempo real de torres en el área de cobertura.

## Usando el Mapa

### Paso 1: Cargar Datos de Sitios Celulares

El mapa carga automáticamente las ubicaciones de las torres celulares desde /cellSites.txt al cargar la página. Este archivo contiene:

- **lat/lng** - Coordenadas de la torre.
- **site\_name** - Identificador.
- **sectors** - Direcciones de antena en grados.

### Paso 2: Dibujar Área de Cobertura

Seleccione una herramienta de dibujo:

1. Haga clic en **Polígono** para dibujar formas personalizadas.
  - Haga clic para agregar puntos.
  - Haga doble clic para completar.
  - Crea límites de cobertura precisos.
2. Haga clic en **Círculo** para dibujar áreas circulares.
  - Haga clic en el punto central.
  - Arrastre para establecer el radio.
  - Útil para alertas rápidas basadas en radio.
3. Haga clic en **Rectángulo** para dibujar áreas rectangulares.
  - Haga clic en una esquina.
  - Arrastre hasta la esquina opuesta.
  - Eficiente para cobertura alineada a la cuadrícula.

### Paso 3: Revisar Sitios Cubiertos

Después de dibujar, el sistema:

- Calcula qué sitios celulares caen dentro del área.
- Muestra la lista en la tabla debajo del mapa.
- Muestra nombres de sitios y coordenadas.
- Proporciona un conteo de torres afectadas.

### Tabla de Cobertura de Ejemplo:

### Paso 4: Exportar Datos de Cobertura

Utilice los datos de cobertura para:

- **Identificar códigos de área de seguimiento** para la difusión.
- **Estimar el alcance de la alerta** basado en el conteo de torres.
- **Planificar la respuesta de emergencia** comprendiendo las áreas

afectadas.

- **Guardar definiciones de zona** para uso futuro.

## Integración del Flujo de Trabajo

La visualización del mapa se integra con la creación de difusión:

### Planificando una Alerta:

1. **Abrir Mapa del CBC** para visualizar el área afectada.
2. **Dibujar zona de cobertura** usando la herramienta de polígono.
3. **Revisar sitios cubiertos** y ajustar límites.
4. **Anotar códigos de área de seguimiento** para los sitios celulares afectados.
5. **Crear mensaje del CBC** con el direccionamiento apropiado.

### Ejemplo: Advertencia de Tsunami Costero

## Gestión de Zonas

### Guardando Zonas:

Las zonas de cobertura predefinidas pueden guardarse para escenarios comunes:

- **Región Costera** - Para advertencias de tsunami/surge.
- **Zonas de Incendios Forestales** - Para alertas de evacuación.
- **Centros Urbanos** - Para alertas AMBER en áreas pobladas.
- **Corredores de Autopistas** - Para alertas de tráfico/clima.

### Formato de Datos de Zona:

Las zonas se almacenan en `site_data.json`:

```
[  
  {  
    "area": "Región Costera",  
    "polygon": [  
      {  
        "coord": [  
          {"lat": -33.8688, "lng": 151.2093},  
          {"lat": -33.8650, "lng": 151.2070},  
          {"lat": -33.8600, "lng": 151.2150},  
          {"lat": -33.8688, "lng": 151.2093}  
        ]  
      }  
    ]  
  }  
]
```

## **Cargando Zonas Guardadas:**

1. Seleccione la zona del menú desplegable.
2. El polígono se muestra en el mapa automáticamente.
3. La lista de sitios se actualiza para mostrar torres cubiertas.
4. Modifique el polígono si es necesario.

## **Configuración del Mapa**

### **API de Google Maps:**

El mapa requiere una clave de API de Google Maps configurada en el entorno:

```
REACT_APP_GOOGLE_API_KEY=your_google_maps_api_key_here
```

### **Datos de Sitios Celulares:**

Actualice `public/cellSites.txt` con las ubicaciones de las torres celulares de su red:

```
# Formato: lat,lng,nombre,sector1,sector2,sector3
-33.8688,151.2093,SYD_CENTRAL,0,120,240
-33.8650,151.2070,SYD_HARBOUR,45,165,285
```

Obtenga coordenadas de sitios celulares de:

- Herramientas de planificación de redes.
- Registros de instalación de torres.
- Encuestas de campo con GPS.
- Configuración del controlador de red de radio (RNC).

### **Estilo del Mapa:**

Los estilos de mapa personalizados pueden configurarse en `mapStyles.js` para:

- Resaltar servicios de emergencia.
- Mostrar características topográficas.
- Enfatizar centros de población.
- Coincidir con la marca de la organización.

## **Casos de Uso**

### **Planificación de Emergencias:**

- Pre-planificar zonas de cobertura para áreas de peligro conocidas.
- Probar diferentes estrategias de direccionamiento.
- Estimar el alcance de la población.
- Coordinar con servicios de emergencia.

## **Verificación de Alertas:**

- Confirmar que el mensaje llegará al área prevista.
- Identificar brechas de cobertura.
- Evitar sobre-alertar regiones adyacentes.
- Validar códigos de área de seguimiento.

## **Análisis de Red:**

- Visualizar la distribución de torres celulares.
- Identificar superposiciones de cobertura.
- Planificar despliegues de torres para mejor cobertura de alertas.
- Optimizar la eficiencia de difusión.

# **Mejores Prácticas**

Basado en la experiencia con clientes de todo el mundo, Omnitouch recomienda las siguientes mejores prácticas para todas las implementaciones de OWL.

## **Contenido del Mensaje:**

- Mantenga los mensajes concisos y claros (menos de 360 caracteres para alertas de una sola página).
- Use letras mayúsculas para alertas de emergencia (mejora la legibilidad).
- Incluya elementos de acción específicos ("Evacuar inmediatamente", "Buscar refugio").
- Evite la jerga técnica.
- Pruebe los mensajes con dispositivos reales antes de las emergencias.

## **Soporte de Idiomas:**

- Siempre proporcione mensajes en el idioma principal de la región.
- Incluya idiomas adicionales para áreas multiculturales.
- Asegúrese de que las traducciones sean culturalmente apropiadas.
- Pruebe caracteres especiales y soporte Unicode.

## **Direccionamiento Geográfico:**

- Use las áreas de seguimiento más pequeñas necesarias para evitar la fatiga de alertas.
- Considere la densidad de población al establecer el período de repetición.
- Pruebe el direccionamiento geográfico antes de las emergencias.
- Mantenga documentación precisa de las áreas de seguimiento.

## **Pruebas:**

- Use identificadores de mensajes de prueba (4379, 4380-4381) para simulacros.

- Programe pruebas regulares del sistema.
- Verifique que la integración del CBC esté funcionando.
- Capacite al personal sobre procedimientos de emergencia.

### **Fatiga de Alertas:**

- Úselo solo para emergencias genuinas.
- Evite la sobre-difusión.
- Establezca períodos de repetición apropiados.
- Use niveles de severidad de manera apropiada.

## **Seguridad de la Cuenta de Usuario**

### **Seguridad de Tokens 2FA:**

- Proteja tokens 2FA físicos como tarjetas de acceso a edificios.
- Informe de inmediato sobre tokens perdidos o robados.
- Pruebe 2FA regularmente para asegurarse de que esté funcionando.
- Guarde los códigos de respaldo al configurar 2FA y almacénelos de forma segura fuera de línea.
- Comuníquese con el Administrador del Sistema si necesita un restablecimiento de 2FA.

Para obtener más información sobre la configuración y recuperación de 2FA, consulte Autenticación de Dos Factores <2fa>.

## **Mantenimiento de Datos**

### **Áreas de Objetivo Predefinidas:**

A medida que cambian los límites geográficos, se producen desarrollos y cambian las áreas de riesgo, hay una necesidad de revisar las Áreas de Objetivo Predefinidas. Omnitouch sugiere que estos datos se revisen **anualmente por el NDMO**, con el apoyo de otras agencias de alerta cuando sea aplicable.

Las actualizaciones de los límites pueden definirse utilizando varias plataformas GIS comunes o Google Earth, y luego proporcionarse por correo electrónico al equipo de operaciones de Omnitouch, que realizará los cambios en el sistema.

### **Plantillas de Mensaje Predefinidas:**

Las plantillas de mensajes predefinidas deben revisarse **al menos anualmente por el NDMO**, con el apoyo de otras agencias de alerta cuando sea aplicable, para asegurar:

- Que el contenido del mensaje siga reflejando con precisión el peligro y la llamada a la acción.
- Que la información de contacto y las instrucciones estén actualizadas.

- Que las traducciones de idiomas sigan siendo precisas.
- Que el tono y la urgencia del mensaje sean apropiados.
- Que las referencias a servicios de emergencia o procedimientos estén actualizadas.

Las actualizaciones pueden proporcionarse por correo electrónico al equipo de operaciones de Omnitouch, que realizará los cambios en el sistema.

### **Datos de Sitios Celulares:**

- Revise y actualice los datos de sitios celulares siempre que ocurran cambios en la red.
- Frecuencia típica de actualización: mensual o trimestral.
- Coordinar con los equipos de planificación de redes de MNO.
- Verifique la precisión después de actualizaciones o expansiones importantes de la red.

## **Arquitectura del Sistema**

Todos los productos de Omnitouch están diseñados para soportar implementaciones distribuidas geográficamente.

### **Opciones de Implementación**

Todos los componentes pueden ejecutarse como:

- **Contenedores (K8s)** - Despliegues de contenedores orquestados por Kubernetes.
- **Máquinas Virtuales** - VMware, Proxmox, HyperV.
- **Nube Privada** - Infraestructura de nube en las instalaciones.
- **Nube Pública** - AWS, GCP.
- **Hardware Dedicado** - Despliegue directo en hardware.

### **Arquitectura Distribuida**

La arquitectura distribuida permite:

- Que las Oficinas Locales de Gestión de Desastres accedan al CBE de OWL y distribuyan mensajes incluso si una región se aísla de la red nacional.
- Múltiples Entidades de Difusión Celular y múltiples Centros de Difusión Celular por operador/país.
- Que las agencias locales de respuesta a desastres (Gobierno Municipal, Policía, Bomberos, etc.) emitan alertas a su región incluso si la principal NDMO (Oficina Nacional de Gestión de Desastres) pierde acceso.

Esto es particularmente valioso cuando los MNO han distribuido su red celular con recursos locales de BSC/MME.

## **Redes CBE y CBC**

Para asegurar que no haya conexión entre redes competidoras, cada MNO tiene una instancia de CBC separada (no compartida).

Requisitos de red entre CBE y CBC:

- Todo el tráfico está cifrado entre CBE y CBC.
- Autenticación basada en certificados mutuos.
- Conectividad desde CBE a CBC a través de TLS en el puerto TCP 443.
- Coordinación requerida entre MNO y el NDMO o agencia que aloje el CBE.

## **Consideraciones de Acceso**

La agencia que aloja el CBE necesitará definir procedimientos de acceso para los usuarios finales (es decir, Citrix, VPN, etc.), teniendo en cuenta que el sistema debe ser accesible en escenarios no ideales como:

- Cortes a gran escala de redes de energía pública.
- Fallas en la red de telecomunicaciones.
- Desastres naturales que afectan la infraestructura.

# **Requisitos de Implementación**

## **Requisitos de VM de CBE (NDMO / Gobierno)**

### **3x Máquinas Virtuales:**

- **2x VMs de CBE** (idealmente en diferentes centros de datos/zonas de disponibilidad).
- **1x VM de Monitoreo.**

Cada VM requiere:

- **Almacenamiento:** 50GB.
- **CPU:** 2x CPU Virtual.
- **RAM:** 8GB.
- **SO:** SO Base proporcionado por Omnitouch.
- **Redes:** Permitir tráfico a las VMs de CBC en el puerto TCP 443 para tráfico TLS para controlar los CBC.

## **Requisitos de VM de CBC (MNO)**

### **3x Máquinas Virtuales:**

- **2x VMs de CBC** (idealmente en diferentes centros de datos/zonas de disponibilidad).
- **1x VM de Monitoreo.**

Cada VM requiere:

- **Almacenamiento:** 50GB.
- **CPU:** 2x CPU Virtual.
- **RAM:** 8GB.
- **SO:** SO Base proporcionado por Omnitouch.
- **Conectividad:** A cada uno de los Puntos de Integración en la red celular (MME, RNC, BSC).
- **Redes:** Permitir tráfico desde las VMs de CBE en el puerto TCP 443 para que el tráfico sea controlado por los CBE.

## Pasos de Integración

Desplegar OWL implica los siguientes pasos:

1. **Identificar nuevo(s) operador(es)** - Identificar operadores de red móvil participantes.
2. **Administración del proyecto (configuración)** - Establecer gobernanza y administración del proyecto.
3. **Seleccionar ubicación de alojamiento para CBE** - Determinar dónde se alojará el CBE.
4. **Definir Usuarios y Flujos/Procedimientos de Mensajes** - Establecer flujos de trabajo de aprobación y roles de usuario.
5. **Definir Polígonos para Áreas Objetivo y Plantillas de Mensajes** - Preconfigurar escenarios comunes.
6. **Asignaciones de Direcciones IP** - Asignar direcciones IP para el NDMO y los MNO.
7. **Configurar VPN de sitio a sitio** - Establecer conexión segura con el equipo de Omnitouch.
8. **Desplegar VMs de CBE al NDMO** - Instalar la Entidad de Difusión Celular.
9. **Desplegar VMs de CBC al MNO** - Instalar el Centro de Difusión Celular en cada operador.
10. **Configurar Elementos de Red** - Establecer conectividad al CBC desde el equipo de red celular.
11. **Redes entre VMs de CBE y CBC** - Establecer comunicación segura.
12. **Redes entre VMs de CBC y Elementos de Red** - Conectar a BSC/MME/AMF.
13. **Configurar acceso a API a NMS** - Configurar integración de datos de sitios celulares.
14. **Configuración de monitoreo y pruebas** - Verificar monitoreo y alertas.
15. **Verificación/Pruebas de alertas de prueba** - Realizar pruebas del sistema.
16. **Prueba de advertencia pública** - Realizar prueba pública de extremo a extremo.

# Límites de Caracteres

Los mensajes de Difusión Celular tienen límites de caracteres estrictos basados en la codificación:

## Codificación GSM de 7 bits (inglés, caracteres latinos básicos):

- Página única: 93 caracteres.
- Múltiples páginas:  $15 \text{ páginas} \times 93 = 1395$  caracteres como máximo.

## Codificación Unicode UCS-2 (escrituras no latinas, emojis):

- Página única: 41 caracteres.
- Múltiples páginas:  $15 \text{ páginas} \times 41 = 615$  caracteres como máximo.

## Plataforma OWL:

- Texto del mensaje limitado a 500 caracteres.
- La interfaz web muestra el conteo de caracteres restantes y advierte cuando se acerca a los límites.

# Monitoreo y Registros

La actividad de Difusión Celular se registra para fines de auditoría:

- Eventos de creación, actualización y eliminación de mensajes.
- Usuarios iniciadores y aprobadores.
- Marcas de tiempo e identificadores de mensajes.
- Respuestas y errores de la API del CBC.
- Detalles de direccionamiento geográfico.

Acceda a los registros a través del Registro de Actividad o consultas de base de datos:

```
SELECT * FROM cbc
WHERE created >= '2025-01-01'
ORDER BY created DESC;
```

# Integración con Dispositivos Móviles

Los mensajes de Difusión Celular son recibidos por dispositivos móviles compatibles:

## Soporte de Dispositivos:

- La mayoría de los teléfonos inteligentes de 2015 en adelante soportan Difusión Celular.

- Los teléfonos básicos pueden tener soporte limitado.
- El dispositivo debe estar conectado a la red (no se requieren créditos de datos/SMS).
- Funciona incluso durante congestiones de red cuando los SMS fallan.

### **Experiencia del Usuario:**

- La alerta se muestra como una notificación de pantalla completa.
- Se reproduce un tono de alerta único.
- La alerta persiste hasta que se reconoce.
- No se requiere suscripción de usuario.
- No puede ser bloqueada por los usuarios para alertas presidenciales/extremas.

### **Prueba de Recepción de Dispositivos:**

Para verificar que los dispositivos pueden recibir alertas:

1. Enviar mensaje de prueba (identificador 4379 o 4380-4381).
2. Asegurarse de que el dispositivo esté en el área de seguimiento objetivo.
3. Verificar que el dispositivo tenga habilitada la Difusión Celular en la configuración.
4. Verificar con múltiples modelos de dispositivos y versiones de SO.

## **Funcionalidad Adicional**

La plataforma OWL puede ampliarse con características opcionales para complementar la mensajería de Difusión Celular:

### **Mensaje de Texto Masivo / SMS**

Enviar SMS regulares a individuos para notificaciones suplementarias.

- A menudo utilizado para alertas degradadas para informar a las personas que la amenaza inmediata ha pasado, de una manera menos intrusiva.
- Mucho más lento que la Difusión Celular, pero puede incluir un mecanismo de confirmación para verificar que el mensaje fue recibido.
- Útil para comunicaciones de seguimiento dirigidas.

### **Llamadas de Voz para Línea Fija**

Llamar automáticamente a números de línea fija y reproducir mensajes de advertencia de emergencia.

- Reproducción de mensajes pregrabados.
- Rendición de texto a voz del mensaje de advertencia de emergencia.
- Llega a poblaciones sin teléfonos móviles.
- Puede verificar la entrega del mensaje a través de la finalización de la

llamada.

## **Publicación Cruzada en Redes Sociales**

Publicar automáticamente mensajes de Advertencia de Emergencia en canales oficiales de redes sociales.

- Extiende el alcance más allá de la red celular.
- Proporciona referencia para aquellos que se perdieron la alerta inicial.
- Permite mensajes extendidos más allá de los límites de caracteres.

## **Transmisión Automática de Radio/TV**

Transmitir automáticamente mensajes de emergencia a través de radio y televisión.

- Reproducción de mensajes pregrabados.
- Rendición de texto a voz del mensaje de advertencia de emergencia.
- Llega a poblaciones durante cortes de red.
- Complementa la Difusión Celular para una cobertura integral.

## **Dispositivos de Advertencia Externos**

El CBC de OWL puede conectarse a una variedad de fuentes externas:

- Plataformas de redes sociales.
- APIs públicas.
- Sistemas de llamadas de voz.
- Sistemas de transmisión de radio.
- Alarmas físicas y sirenas.
- Señalización electrónica.

Las opciones de personalización pueden explorarse como parte de la fase de diseño.

## **Personalización y Mantenimiento**

### **Pruebas Periódicas**

Se deben realizar pruebas periódicas de la solución a intervalos regulares para asegurar:

- Que la solución y todos los componentes estén funcionando correctamente.
- Que todo el personal esté familiarizado con los procesos y procedimientos requeridos para emitir Mensajes de Advertencia de Emergencia.
- Que los puntos de integración sigan operativos.
- Que las plantillas de mensajes estén actualizadas y sean efectivas.

## **Pruebas Recomendadas:**

- Mensajes de prueba mensuales utilizando identificadores 4379, 4380-4381.
- Pruebas completas del sistema trimestrales que incluyan flujos de trabajo de aprobación.
- Pruebas de advertencia pública anuales con notificación anticipada.
- Sesiones de capacitación regulares para usuarios autorizados.

## **Mantenimiento de Datos de Sitios Celulares**

Cuando los operadores agregan o eliminan sitios celulares, o cambian Áreas/Identificadores de Seguimiento de sitios celulares, esta información debe compartirse con el equipo de Omnitouch para asegurar que los datos de la herramienta de mapeo permanezcan precisos.

### **Integración Automática de Datos**

OWL soporta la recopilación automática de datos de:

- Nokia NetAct.
- Huawei U2000 / U2020.
- ZTE NetNumen / ZXPOS.
- Ericsson ENM.

### **Actualizaciones Manuales de Datos**

Alternativamente, los datos del sitio celular pueden proporcionarse al equipo de operaciones de Omnitouch periódicamente por correo electrónico en varios formatos.

**Frecuencia de Actualización:** Revise y actualice los datos del sitio celular siempre que ocurran cambios en la red, típicamente mensual o trimestral.

### **Áreas de Objetivo Predefinidas**

A medida que cambian los límites geográficos, se producen desarrollos y cambian las áreas de riesgo, hay una necesidad de revisar las Áreas de Objetivo Predefinidas utilizadas en la etapa de Direccionamiento.

**Revisión Anual:** Las áreas de objetivo predefinidas deben revisarse anualmente por el NDMO (Oficina Nacional de Gestión de Desastres).

**Proceso de Actualización:** Las actualizaciones de los límites pueden definirse utilizando varias plataformas GIS comunes o Google Earth, y luego proporcionarse al equipo de operaciones de Omnitouch.

## Plantillas de Mensaje Predefinidas

**Revisión Anual:** Las plantillas de mensajes predefinidas deben revisarse anualmente por el NDMO para asegurar:

- Que el contenido del mensaje refleje los procedimientos de emergencia actuales.
- Que las traducciones de idiomas sean precisas.
- Que los identificadores de mensajes sean apropiados.
- Que la información de contacto y las instrucciones estén actualizadas.

**Proceso de Actualización:** Las actualizaciones pueden proporcionarse por correo electrónico al equipo de operaciones de Omnitouch.

## Flujos de Aprobación de Mensajes

Diferentes regiones tienen diferentes requisitos respecto a la emisión de mensajes y flujos de aprobación.

**Regla de Dos Personas:** Se aboga por el uso de la regla de dos personas siempre que sea práctico para asegurar supervisión en la presentación de mensajes.

**Roles de Usuario Granulares:** Los roles de usuario individuales pueden configurarse para:

- Permitir que solo ciertos usuarios envíen mensajes predefinidos.
- Restringir el direccionamiento a regiones específicas.
- Requerir pasos de aprobación adicionales.
- Minimizar el riesgo de abuso.

## Mantenimiento de Seguridad / 2FA

El equipo de operaciones de Omnitouch puede apoyar con:

- Restablecimiento de tokens de 2FA.
- Reemisión de tokens perdidos/dañados/expirados.
- Auditoría de seguridad y gestión de tokens.

## Procedimientos de Seguridad de Tokens:

Cuando se emiten tokens, se detalla un procedimiento para los pasos a seguir si un token se pierde o no se puede localizar. La persona que acepta el token debe seguir estos procedimientos para asegurar que el sistema no sea abusado.

## Acciones Inmediatas para Tokens Perdidos:

1. Informe de inmediato sobre el token perdido al personal autorizado.

2. El token se desactiva en el sistema.
3. Se realiza una revisión de seguridad.
4. Se emite un nuevo token tras la verificación de seguridad.

## Integración de API Externas

Se dispone de una suite completa de APIs para la Entidad de Difusión Celular que permite a sistemas de terceros integrarse e interactuar con el CBE.

### Capacidades de API:

- **Informes/Monitoreo** - Verificar el estado y alcance de los mensajes transmitidos.
- **Creación de Mensajes** - Crear y difundir nuevos mensajes programáticamente.
- **Verificaciones de Salud** - Verificar periódicamente la salud del sistema con generación de tráfico de mensajes de prueba.
- **Consultas de Estado** - Recuperar el estado del mensaje, estadísticas de entrega y métricas del sistema.

Consulte la sección de Referencia de API a continuación para obtener documentación detallada de los endpoints.

## Referencia de API

Todos los endpoints del CBC requieren autenticación y permisos apropiados.

### Crear Mensaje:

PUT /crm/cbc/

### Obtener Todos los Mensajes:

GET /crm/cbc/

### Actualizar Mensaje:

PATCH /crm/cbc/{cbc\_message\_id}

### Eliminar Mensaje:

DELETE /crm/cbc/{cbc\_message\_id}

Consulte la documentación de Swagger en </crm/docs/> para especificaciones detalladas de la API.



# Búsqueda Global

La función de Búsqueda Global proporciona una **interfaz de búsqueda unificada** para encontrar rápidamente clientes, contactos, servicios, inventario y sitios en toda la base de datos de OmniCRM.

Ver también: *Customers <basics\_customers>, Inventory <administration\_inventory>, Service Management <csa\_service\_management>*.

## Acceder a la Búsqueda Global

### Desde cualquier lugar en el CRM:

Haz clic en el ícono de búsqueda en la barra de navegación superior o navega a:

La página de búsqueda global aparece con un gran cuadro de búsqueda y opciones de filtro.

## Cómo Funciona

La búsqueda global realiza una **búsqueda cruzada** entre cinco tipos de datos:

### Qué se Busca:

1. **Clientes** - Nombre del cliente
2. **Contactos** - Nombre, apellido, dirección de correo electrónico, número de teléfono
3. **Sitios** - Nombre del sitio
4. **Inventario** - Números de serie, ICCIDs, identificadores (itemtext1, itemtext2)
5. **Servicios** - Nombre del servicio, UUID del servicio

### Comportamiento de Búsqueda:

- **Coincidencia parcial** - Busca términos que contengan tu consulta (por ejemplo, "Smith" coincide con "John Smith" y "Smithson")
- **Sin distinción de mayúsculas** - "john" coincide con "John", "JOHN" y "john"
- **Múltiples entidades** - Una sola búsqueda devuelve resultados de todos los tipos de entidad
- **Resultados paginados** - Muestra 10 resultados por página por defecto

## Realizando una Búsqueda

### Búsqueda Básica

1. Ingresa tu término de búsqueda en el cuadro de búsqueda
2. Haz clic en **"Buscar"** o presiona Enter

Resultados de Búsqueda Global

### Ejemplos de términos de búsqueda:

- Nombre del cliente: "Acme Corp"
- Número de teléfono: "+1234567890" o "1234567890"
- Correo electrónico: "john@example.com" o "john"

- Número de serie: "ICCID8944" o solo "8944"
- UUID del servicio: "123e4567-e89b"

## Incluir Cuentas Cerradas

Por defecto, la búsqueda solo devuelve resultados de cuentas de clientes **Abiertas**.

Para buscar en todas las cuentas, incluidas las cerradas:

1. Marca la casilla "**Incluir Cuentas Cerradas**"
2. Haz clic en "**Buscar**" nuevamente

Esto buscará:

- Clientes con `customer_status = "Closed"`
- Contactos, servicios, sitios e inventario vinculados a clientes cerrados

### Casos de uso para la búsqueda de cuentas cerradas:

- Encontrar registros históricos de clientes
- Localizar equipos de servicios desprovisionados
- Buscar números de teléfono o servicios antiguos
- Recuperar datos de clientes para reactivación

## Entendiendo los Resultados de Búsqueda

### Formato de Visualización de Resultados

Los resultados se muestran en una lista desplazable que muestra:

John Smith ID de Cliente: 123 Tipo: cliente

John Smith (Contacto) ID de Cliente: 123 Tipo: contacto

Móvil - +44 7700 900123 ID de Cliente: 123 Tipo: servicio

### Cada resultado muestra:

- **Nombre/Título** - El identificador principal (enlace clicable)
- **ID de Cliente** - El cliente principal al que pertenece
- **Tipo** - El tipo de entidad (cliente, contacto, sitio, inventario, servicio)

## Tipos de Resultados Explicados

### Resultados de Clientes:

Hacer clic abre la página de resumen del cliente mostrando todos los detalles, servicios, contactos, etc.

### Resultados de Contactos:

Hacer clic abre la página del cliente con la pestaña de Contactos activa, desplazándose al contacto específico.

### Resultados de Sitios:

Hacer clic abre la página del cliente con la pestaña de Sitios activa.

## Resultados de Inventario:

Hacer clic abre la página del cliente con la pestaña de Inventario activa. Si el inventario no está asignado (sin customer\_id), se vincula a la lista principal de inventario.

## Resultados de Servicios:

Hacer clic abre la página del cliente con la pestaña de Servicios activa, destacando el servicio específico.

## Navegación desde Resultados

Todos los resultados de búsqueda son **enlaces clicables** que navegan directamente a la página relevante:

### Patrón de Enlace:

- `/customers/{customer_id}` - Registros de clientes
- `/customers/{customer_id}#4` - Contactos (pestaña 4)
- `/customers/{customer_id}#2` - Sitios (pestaña 2)
- `/customers/{customer_id}#8` - Inventario (pestaña 8)
- `/customers/{customer_id}#3` - Servicios (pestaña 3)
- `/inventory-items-list` - Inventario no asignado

El fragmento de hash (#) selecciona automáticamente la pestaña correcta cuando se carga la página del cliente.

## Paginación

Los resultados están paginados con 10 elementos por página:

Mostrando resultados 11-20 de 47

Navega a través de las páginas usando:

- Botones **Anterior/Siguiente**
- **Números de página** - Haz clic en una página específica
- **Teclado** - Flechas izquierda/derecha (si está implementado)

## Escenarios Comunes de Búsqueda

### Escenario 1: Encontrar Cliente por Teléfono

El usuario llama, proporciona el número de teléfono.

Resultados: • John Smith (Contacto) - ID de Cliente: 123 • Móvil - 555-0123 (Servicio) - ID de Cliente: 123

Haz clic en cualquiera de los resultados para acceder a la cuenta del cliente.

### Escenario 2: Localizar Tarjeta SIM

El técnico necesita encontrar qué cliente tiene una SIM específica.

Resultados: • 8944538000000001234 (Inventario) - ID de Cliente: 456

Haz clic en el resultado para ver la asignación de la SIM, detalles del cliente.

### **Escenario 3: Encontrar Cliente Inactivo**

Necesitas localizar un cliente que cerró su cuenta hace 6 meses.

- Incluir Cuentas Cerradas

Resultados: • Acme Corporation (cliente) - ID de Cliente: 789

### **Escenario 4: Buscar por Correo Electrónico**

El cliente envía un correo electrónico al soporte, el personal necesita encontrar su cuenta.

Resultados: • John Smith (Contacto) - ID de Cliente: 123

### **Escenario 5: Encontrar Servicio por UUID**

El registro de aprovisionamiento muestra el UUID del servicio, necesitas encontrar qué cliente.

Resultados: • Móvil - +44 7700 900123 (Servicio) - ID de Cliente: 456

## **Consejos de Búsqueda**

### **Para Mejores Resultados:**

- **Usa términos parciales** - "Smith" es mejor que "John Smith" para resultados más amplios
- **Prueba variaciones** - Si "John" no funciona, prueba con teléfono o correo electrónico
- **Incluye cuentas cerradas** - Al buscar datos históricos
- **Sé específico para equipos** - Usa números de serie completos para inventario
- **Busca UUID de servicio** - Cuando otros identificadores no son conocidos

### **Qué se Busca (por Entidad):**

#### **Clientes:**

- Solo el nombre del cliente (no dirección, notas u otros campos)

#### **Contactos:**

- Nombre
- Apellido
- Dirección de correo electrónico
- Número de teléfono

#### **Sitios:**

- Solo el nombre del sitio

#### **Inventario:**

- itemtext1 (típicamente ICCID, número de serie, dirección MAC)
- itemtext2 (típicamente IMSI, identificador secundario)
- *Nota: No busca itemtext3-20 o notas de inventario*

#### **Servicios:**

- Nombre del servicio
- UUID del servicio

## Qué No se Busca:

- Direcciones de clientes
- Notas de clientes
- Descripciones de transacciones
- Detalles de facturas
- Registros de aprovisionamiento
- Entradas de registro de actividades
- Notas de inventario (más allá de itemtext1/2)

# Referencia de API

## Endpoint de Búsqueda Global

```
GET /utilities/  
search_everything?search=Smith&page=1&per_page=10&search_closed_records=false  
Authorization: Bearer <token>
```

### Parámetros de Consulta:

- `search` (requerido) - El término de búsqueda
- `page` (opcional) - Número de página (por defecto: 1)
- `per_page` (opcional) - Resultados por página (por defecto: 10)
- `search_closed_records` (opcional) - Incluir cuentas cerradas (por defecto: false)

### Respuesta:

```
{  
  "data": [  
    {  
      "id": 123,  
      "name": "John Smith",  
      "customer_id": 123,  
      "type": "customer"  
    },  
    {  
      "id": 456,  
      "name": "John Smith",  
      "customer_id": 123,  
      "type": "contact"  
    },  
    {  
      "id": 789,  
      "name": "Mobile - +44 7700 900123",  
      "customer_id": 123,  
      "type": "service"  
    }  
,  
  "pagination": {  
    "current_page": 1,  
    "per_page": 10,  
    "total_pages": 5,  
    "total_items": 47  
  }  
}
```

## Lógica de Búsqueda (Backend):

El backend realiza una unión SQL entre todas las tablas de entidades:

```
-- Clientes
SELECT customer_id AS id,
       customer_name AS name,
       customer_id,
       'customer' AS type
FROM customer
WHERE customer_name LIKE '%Smith%'
  AND customer_status = 'Open'

UNION ALL

-- Contactos
SELECT contact_id AS id,
       CONCAT(contact_firstname, ' ', contact_lastname) AS name,
       customer_id,
       'contact' AS type
FROM customer_contact
WHERE (contact_firstname LIKE '%Smith%' OR
       contact_lastname LIKE '%Smith%' OR
       contact_email LIKE '%Smith%' OR
       contact_phone LIKE '%Smith%')

UNION ALL

-- Sistios
SELECT site_id AS id,
       site_name AS name,
       customer_id,
       'site' AS type
FROM customer_site
WHERE site_name LIKE '%Smith%'

UNION ALL

-- Inventario
SELECT inventory_id AS id,
       itemtext1 AS name,
       customer_id,
       'inventory' AS type
FROM inventory
WHERE itemtext1 LIKE '%Smith%' OR
      itemtext2 LIKE '%Smith%'

UNION ALL

-- Servicios
SELECT service_id AS id,
       service_name AS name,
       customer_id,
       'service' AS type
FROM customer_service
WHERE service_name LIKE '%Smith%' OR
      service_uuid LIKE '%Smith%'
```

Los resultados se paginan y se devuelven.

# Consideraciones de Rendimiento

## Rendimiento de Búsqueda:

- Las búsquedas utilizan consultas LIKE con comodines (%term%)
- No se ha implementado actualmente indexación de texto completo
- Bases de datos grandes (>100k clientes) pueden experimentar búsquedas más lentas
- Resultados limitados a 10 por página para mejorar el rendimiento

## Consejos de Optimización:

- Sé específico con los términos de búsqueda para reducir el conjunto de resultados
- Usa el filtro de cuentas cerradas para reducir el alcance de búsqueda
- Considera agregar índices de base de datos en campos que se buscan con frecuencia

# Solución de Problemas

## No se encontraron resultados (pero el registro existe)

- **Causa:** El término de búsqueda no coincide con el formato de datos almacenados
- **Ejemplos:**
  - Teléfono almacenado como "+44 7700 900123", buscar "07700900123" no coincidirá
  - Correo electrónico almacenado como "<[john.smith@example.com](mailto:john.smith@example.com)>", buscar "john" no coincidirá
- **Solución:** Prueba variaciones, usa coincidencias parciales que definitivamente existan

## Búsqueda demasiado lenta

- **Causa:** Base de datos grande, consulta compleja entre múltiples tablas
- **Solución:**
  - Usa términos de búsqueda más específicos
  - Limita solo a cuentas abiertas (desmarca cuentas cerradas)
  - Contacta al administrador sobre la indexación de la base de datos

## Resultados enlazan al cliente incorrecto

- **Causa:** Múltiples clientes/contactos con el mismo nombre
- **Solución:** Usa el ID de Cliente para diferenciar, o busca por identificador único (correo electrónico, teléfono)

## Cuentas cerradas no aparecen

- **Causa:** Casilla "Incluir Cuentas Cerradas" no marcada
- **Solución:** Marca la casilla y busca nuevamente

# Documentación Relacionada

- `basics_customers` - Gestión de clientes
- `basics_navigation` - Navegación general
- `administration_inventory` - Búsquedas de inventario



# Sistema de Recarga y Abono

El sistema de recarga de OmniCRM proporciona un **portal de recarga prepago de autoservicio** para que los clientes añadan crédito o extiendan la validez del servicio a través del Self-Care Portal <self\_care\_portal>. Esta función se utiliza comúnmente para:

- **Servicios de datos móviles** - Tarjetas SIM prepago y servicios solo de datos
- **Servicios de hotspot** - Dongles de hotspot WiFi y dispositivos de internet portátiles
- **Servicios inalámbricos fijos** - Acceso a internet prepago

## Descripción General

El sistema de recarga permite a los clientes comprar días adicionales de servicio a través de un proceso de pago simplificado y en múltiples pasos con procesamiento de pagos integrado de Stripe.

### Características Clave:

- Portal de autoservicio para clientes (no se requiere intervención del personal)
- Selección de duración flexible (1-30 días)
- Visualización de uso en tiempo real antes de la compra
- Procesamiento de pagos seguro impulsado por Stripe
- Reembolsos automáticos si la recarga falla
- Generación de facturas y transacciones
- Integración del sistema de aprovisionamiento para la activación del servicio

## Acceso al Portal de Recarga

El portal de recarga se accede a través de una **URL pública** que los clientes pueden visitar sin iniciar sesión en el CRM:

### Cómo Acceden los Clientes:

- Enlace directo enviado por SMS cuando el saldo es bajo
- Código QR en materiales impresos
- Enlace en el portal de autoservicio
- Compartido a través del soporte al cliente

El portal detecta automáticamente el servicio del cliente según su dirección IP de solicitud o IMSI.

## Proceso de Recarga

El flujo de recarga consta de **4 pasos**:

### Paso 1: Selección de Datos

Los clientes seleccionan cuántos días de servicio desean comprar.

**Interfaz:**

- **Control deslizante** - Seleccionar de 1 a 30 días
- **Cálculo de precio en vivo** - Muestra el costo total según la selección
- **Visualización de fecha de caducidad** - Calcula y muestra cuándo expirará el servicio
- **Visualización de uso actual** - Muestra el saldo restante/caducidad antes de la recarga

**Ejemplo de Visualización:**

**Configuración de Precios:**

- El precio por día se configura a través de la variable de entorno `REACT_APP_TOPUP_PRICE_PER_DAY`
- Predeterminado: \$10 USD por día
- La moneda se establece a través de `REACT_APP_CURRENCY_CODE`

### Paso 2: Información de Facturación

Los clientes proporcionan sus datos de contacto para la transacción:

- **Nombre**
- **Apellido**
- **Dirección de Correo Electrónico**

Esta información se utiliza para:

- Generación de facturas
- Correo electrónico de recibo de pago
- Registros de transacciones
- Procesamiento de reembolsos (si es necesario)

### Paso 3: Pago

Procesamiento de pagos seguro a través de **Stripe Elements**.

## **Métodos de Pago Soportados:**

- Tarjetas de crédito (Visa, Mastercard, Amex)
- Tarjetas de débito
- Monederos digitales (Apple Pay, Google Pay) *si están habilitados en Stripe*

## **Características de Seguridad:**

- Integración de Stripe conforme a PCI
- No se almacenan detalles de tarjetas en OmniCRM
- Soporte para autenticación 3D Secure
- Transmisión de pagos cifrada

## **Flujo de Pago:**

1. Se muestra el formulario de Stripe Elements con entrada de tarjeta
2. El cliente ingresa los detalles de pago
3. Se crea la Intención de Pago por el monto exacto
4. Se carga la tarjeta inmediatamente
5. Éxito/fallo del pago gestionado

### Nota

Si el pago tiene éxito pero el aprovisionamiento de la recarga falla (por ejemplo, error de red, OCS inalcanzable), el sistema inicia automáticamente un **reembolso completo** al método de pago del cliente.

## **Paso 4: Finalización**

### **Pantalla de Éxito:**

Su servicio ha sido extendido. Nueva fecha de caducidad: 17 de enero de 2025

Recibo enviado a: <[customer@example.com](mailto:customer@example.com)> ID de Transacción:  
TXN-123456

### **Pantalla de Fallo:**

Si la recarga falla, el sistema muestra un error y procesa automáticamente un reembolso:

No pudimos completar su recarga. Su pago ha sido reembolsado.

Error: No se puede conectar al sistema de facturación

Por favor, inténtelo de nuevo o contacte con el soporte.

# Procesamiento en el Backend

Cuando un cliente completa el pago, lo siguiente sucede automáticamente:

## 1. Validación del Pago

El sistema valida:

- El estado de la Intención de Pago es `succeeded`
- El monto del pago coincide con los días seleccionados (`days` × `price_per_day`)
- La Intención de Pago no ha sido procesada antes (previene recargas dobles)

## 2. Operación de Recarga

- [API endpoint](#): `POST /oam/topup_dongle`
- Valida `service_uuid` e `IMSI`
- Llama a `OCS/CGRateS` para añadir saldo
- Crea trabajo de aprovisionamiento (`play_topup_hotspot`)

## 3. Creación de Registros

El sistema crea múltiples registros en la base de datos:

- **Registro HotspotTopup** - Rastrea la transacción de recarga
  - `payment_intent_id`
  - `service_uuid`
  - `imsi`
  - días comprados
  - `topup_amount`
  - estado (Éxito/Fallido/Reembolsado)
- **Registro de Transacción** - Transacción financiera
  - Título: "Recarga de Hotspot - 7 Días"
  - Monto: `topup_amount` (positivo)
  - Vinculado a `service_id` y `customer_id`
- **Registro de Factura** - Factura de pago
  - Contiene la transacción de recarga
  - Marcado como pagado inmediatamente
  - Referencia de pago: Stripe `payment_intent_id`
- **Transacción de Pago** - Transacción de crédito compensatoria
  - Título: "Pago por [Título de la Factura]"
  - Monto: `topup_amount` (negativo - crédito)
  - Vincula el pago de la factura a la cuenta del cliente

## 4. Trabajo de Aprovisionamiento

Se crea un trabajo de aprovisionamiento con el playbook `play_topup_hotspot`

que:

- Conecta a la API de OCS/CGRateS
- Añade saldo a la cuenta
- Extiende la fecha de caducidad
- Crea una entrada de registro de actividad
- Envía una notificación de confirmación (si está configurado)

La API espera a que se complete el aprovisionamiento (polling con intervalos de 0.2s, máximo 25 iteraciones) antes de devolver el éxito al cliente.

## 5. Reembolso Automático en Caso de Fallo

Si algún paso falla después del pago:

```
if topup_provisioning_failed:
    refund = stripe.Refund.create(
        payment_intent=payment_intent_id,
        reason='requested_by_customer' # Reembolso automático del
sistema
    )
    status_message = "Recarga Fallida. Reembolsando pago..."
```

El reembolso se procesa automáticamente y se notifica al cliente en pantalla.

## Endpoints de la API

### Endpoint de Recarga

```
POST /oam/topup_dongle
Content-Type: application/json

{
  "service_uuid": "123e4567-e89b-12d3-a456-426614174000",
  "imsi": "310120123456789",
  "days": 7,
  "payment_intent_id": "pi_1234567890abcdef",
  "topup_amount": 70.00
}
```

### Respuesta (Éxito):

```
{
  "result": "OK",
  "status": 200,
  "provision_id": 456,
  "payment_intent_id": "pi_1234567890abcdef",
  "service_uuid": "123e4567-e89b-12d3-a456-426614174000",
```

```
    "invoice_id": 789
}
```

### Respuesta (Fallo):

```
{
  "result": "Failed",
  "Reason": "Tiempo de espera de conexión OCS",
  "status": 500
}
```

### Comprobaciones de Validación:

- Todos los campos requeridos presentes (service\_uuid, imsi, days, payment\_intent\_id, topup\_amount)
- topup\_amount coincide con días:  $\text{topup\_amount} \times 100 == \text{days} \times 1000$  (en centavos)
- La Intención de Pago existe en Stripe
- El monto de la Intención de Pago coincide:  $\text{payment\_intent.amount} == \text{topup\_amount} \times 100$
- El estado de la Intención de Pago es succeeded
- La Intención de Pago no ha sido procesada previamente (verifica la tabla HotspotTopup)

## Endpoint de Uso

Recupera el uso actual y la información del servicio para el cliente:

```
GET /oam/usage
```

### Respuesta:

```
{
  "imsi": "310120123456789",
  "service": {
    "service_uuid": "123e4567-e89b-12d3-a456-426614174000",
    "service_name": "Datos Móviles - 0412345678",
    "service_status": "Activo"
  },
  "balance": {
    "expiry": "2025-01-10T23:59:59Z",
    "unlimited": true
  },
  "requestingIp": "203.0.113.45"
}
```

Este endpoint utiliza la dirección IP de solicitud para identificar automáticamente el servicio del cliente.

# Configuración

## Variables de Entorno

Configura estas en el archivo .env de OmniCRM-UI:

```
# Configuración del Portal de Recarga
REACT_APP_TOPUP_PRICE_PER_DAY=10
REACT_APP_CURRENCY_CODE=AUD
REACT_APP_SELF_CARE_NAME="YourCompany"

# Configuración de Stripe
REACT_APP_STRIPE_PUBLISHABLE_KEY=pk_live_...
```

## Configuración de Stripe

El sistema de recarga utiliza Intenciones de Pago de Stripe:

1. **Habilitar Intenciones de Pago** en tu Panel de Control de Stripe
2. **Configurar Webhook** para recibir actualizaciones del estado del pago (opcional pero recomendado)
3. **Configurar métodos de pago** (tarjetas, monederos, etc.)
4. **Modo de prueba** - Usa claves de prueba para el desarrollo

```
# Desarrollo
REACT_APP_STRIPE_PUBLISHABLE_KEY=pk_test_...

# Producción
REACT_APP_STRIPE_PUBLISHABLE_KEY=pk_live_...
```

## Configuración del Playbook

El playbook de aprovisionamiento play\_topup\_hotspot.yaml debe configurarse para:

- Aceptar la variable days
- Conectar a la API de OCS/CGRateS
- Añadir saldo a la cuenta
- Actualizar la fecha de caducidad del servicio

Ejemplo de estructura de playbook:

```
- name: Recargar servicio de hotspot
  hosts: localhost
  tasks:
    - name: Añadir saldo a OCS
      uri:
        url: "{{ ocs_api_url }}/add_balance"
```

```
method: POST
body:
  imsi: "{{ imsi }}"
  days: "{{ days }}"
  service_uuid: "{{ service_uuid }}"
```

## Notificaciones de Saldo Bajo

El sistema puede enviar notificaciones automáticas cuando el saldo del cliente es bajo:

### Notificaciones por SMS:

Cuando se activan por eventos de OCS (Action\_Balance\_Low, Action\_Balance\_Out, Action\_Balance\_Expired):

### Notificaciones por Correo Electrónico:

Configuradas en los planes de acción de OCS/CGRateS para enviar alertas de saldo.

### Disparadores de Notificación:

- Action\_Balance\_Low - Saldo por debajo del umbral (por ejemplo, 2 días restantes)
- Action\_Balance\_Out - Saldo agotado
- Action\_Balance\_Expired - Servicio expirado

Cada notificación incluye el enlace al portal de recarga para un fácil acceso del cliente.

## Solución de Problemas

### Problemas Comunes

#### "Sistema de pago no disponible"

- **Causa:** La biblioteca de Stripe no se pudo cargar o clave publicable inválida
- **Solución:**
  - Verifica que REACT\_APP\_STRIPE\_PUBLISHABLE\_KEY esté configurado correctamente
  - Verifica que la cuenta de Stripe esté activa
  - Revisa la consola del navegador en busca de errores de JavaScript

#### "Recarga fallida. Reembolsando pago..."

- **Causa:** Fallo en el trabajo de aprovisionamiento (OCS inalcanzable, error en el playbook, etc.)
- **Solución:**
  - Verifica los registros de aprovisionamiento: GET /crm/provision/provision\_id/<id>
  - Verifica que la API de OCS/CGRateS sea accesible
  - Revisa el playbook play\_topup\_hotspot.yaml en busca de errores
  - Revisa los registros de Ansible

## "Intención de pago ya procesada"

- **Causa:** Cliente intentando reutilizar el mismo pago (por ejemplo, refrescar después del éxito)
- **Solución:** Esto funciona como se diseñó para prevenir la doble facturación. El cliente debe iniciar una nueva recarga si es necesario.

## "El monto de la intención de pago no coincide"

- **Causa:** Desajuste entre el cálculo de la UI y la validación en el backend
- **Solución:**
  - Verifica que REACT\_APP\_TOPUP\_PRICE\_PER\_DAY coincida con la expectativa del backend (predeterminado \$10)
  - Verifica que la configuración de la moneda sea consistente
  - Limpia la caché del navegador y vuelve a intentarlo

## Monitoreo de Recargas

### Ver Registros de Recarga:

Consulta la tabla HotspotTopup para ver todos los intentos de recarga:

```
SELECT
  hotspot_topup_id,
  service_uuid,
  days,
  topup_amount,
  status,
  payment_intent_id,
  created
FROM hotspot_topup
WHERE status = 'Failed'
ORDER BY created DESC;
```

### Verificar Estado de Aprovisionamiento:

```
GET /crm/provision/provision_id/<provision_id>
```

Muestra el estado detallado del trabajo de aprovisionamiento de la recarga.

## Panel de Control de Stripe:

Monitorea pagos, reembolsos y transacciones fallidas en tu Panel de Control de Stripe en <<https://dashboard.stripe.com>>

# Consideraciones de Seguridad

## Seguridad de Pagos:

- Todos los datos de tarjetas manejados por Stripe (conforme a PCI Nivel 1)
- No se almacenan datos sensibles de pago en la base de datos de OmniCRM
- Las Intenciones de Pago previenen cargos no autorizados
- Validación de montos tanto del lado del cliente como del servidor

## Prevención de Fraude:

- La detección de Intenciones de Pago duplicadas previene la doble facturación
- Seguimiento de direcciones IP para correlación de uso
- La validación de IMSI asegura que la recarga vaya al servicio correcto
- Los reembolsos automáticos limitan la exposición financiera

## Control de Acceso:

- El portal de recarga es público (por diseño - los clientes necesitan acceso)
- El endpoint de uso requiere identificación de servicio válida (IP o IMSI)
- La validación en el backend previene recargas arbitrarias de servicios
- El administrador puede ver todos los registros de recarga a través de la interfaz del CRM

# Mejores Prácticas

## Para Operadores:

1. **Probar el flujo de reembolso** - Probar regularmente escenarios de fallo para asegurar que los reembolsos funcionen
2. **Monitorear recargas fallidas** - Configurar alertas para altas tasas de fallo
3. **Mantener los playbooks simples** - Los playbooks de recarga deben ser rápidos y fiables
4. **Verificar conectividad de OCS** - Asegurarse de que la API de OCS esté siempre accesible
5. **Revisar precios** - Actualizar REACT\_APP\_TOPUP\_PRICE\_PER\_DAY según sea necesario

## Para Clientes:

1. **Agregar el URL de recarga a marcadores** - Acceso rápido cuando sea necesario
2. **Guardar notificaciones de saldo bajo** - El SMS contiene un enlace directo
3. **Mantener el correo electrónico actualizado** - Los recibos se envían al correo registrado
4. **Verificar caducidad antes de viajar** - Recargar antes de salir del área de cobertura

#### **Para Desarrolladores:**

1. **Manejar webhooks de Stripe** - Implementar controladores de webhook para actualizaciones de estado de pago
2. **Implementar idempotencia** - Siempre verificar payment\_intent\_id antes de procesar
3. **Registrar extensivamente** - Las fallas de recarga necesitan información detallada de solución de problemas
4. **Probar caminos de error** - Verificar que la automatización de reembolsos funcione correctamente
5. **Monitorear rendimiento** - El polling de aprovisionamiento debe completarse en <5 segundos

## **Documentación Relacionada**

- payments\_process - Procesamiento de pagos general
- concepts\_provisioning - Descripción general del sistema de aprovisionamiento
- integrations\_stripe - Detalles de integración de Stripe
- payments\_transaction - Gestión de transacciones
- payments\_invoices - Manejo de facturas



# Glosario de Términos

**OmniCRM** Solución CRM de Omnitouch para proveedores de servicios móviles y de línea fija

**OmniCharge**

Motor de carga utilizado en OmniCRM

**Servicio**

Una instancia de un producto provisionado para un cliente y vinculada a una cuenta de OCS para facturación y carga.

**Provisionamiento**

El proceso de convertir un producto en un servicio, incluyendo la asignación de inventario, la configuración de recursos de red y la generación de cargos a través de playbooks de Ansible.

**Addon**

Un servicio o característica adicional, como datos extra o minutos internacionales, que se puede agregar a un servicio existente.

**Bolt-On**

Un tipo de addon que extiende las capacidades de un servicio existente, a menudo de manera puntual.

**Elemento de inventario**

Recursos físicos o virtuales como tarjetas SIM, módems, números de teléfono o equipos de red requeridos para provisionar ciertos servicios.

**Ansible Playbook**

Un conjunto de instrucciones de automatización (archivos YAML) utilizados para provisionar servicios a partir de una definición de producto, configurar equipos de red e integrarse con sistemas externos.

**ARPU**

Ingresaos Promedio Por Usuario, una métrica clave para los proveedores de servicios que mide los ingresos promedio generados por cada cliente.

**OCS**

Sistema de Carga en Línea. El motor de carga en tiempo real (basado en CGRateS) que maneja cargos monetarios y recurrentes, gestiona saldos (monetarios y unitarios), provisiona ActionPlans para renovación automática y aplica límites de gasto a través de ThresholdS. Los servicios en el CRM se vinculan a cuentas de OCS a través de service\_uuid.

## CGrateS

La plataforma de sistema de carga en tiempo real de código abierto en la que se basa el OCS de OmniCRM. Proporciona APIs para la gestión de saldos, ejecución de acciones y decisiones de calificación/ruteo.

## Acción

Una tarea en OCS/CGrateS, como un cargo, restablecimiento de saldo o concesión de crédito, que se aplica a un servicio. Las acciones se pueden ejecutar de inmediato o programar a través de ActionPlans.

## ActionPlan

Un horario recurrente de acciones en OCS/CGrateS, como cargos mensuales o renovaciones automáticas, aplicadas a un servicio. Cada servicio puede tener múltiples ActionPlans para diferentes operaciones recurrentes.

## ThresholdS

Un límite establecido en una cuenta para prevenir gastos excesivos por parte de un cliente, como un límite de crédito en servicios prepagos.

## Cliente

Una empresa o individuo que tiene una relación con el proveedor de servicios.

## Contacto

Una persona asociada con un cliente, como un contacto de facturación o técnico, que influye en cómo se gestiona al cliente.

## Sitio

Una ubicación física donde se entregan servicios, asociada a un cliente.



# Guía Completa del Ciclo de Vida del Producto

Esta guía proporciona un recorrido completo del ciclo de vida del producto en OmniCRM, desde la creación de una definición de producto hasta la provisión de servicios, la adición de complementos y la desprovisión. Cubriremos la estrategia de precios, la integración de Ansible y proporcionaremos ejemplos del mundo real a lo largo del documento.

## Descripción general: El viaje del Producto al Servicio

El ciclo de vida de un producto en OmniCRM sigue estas etapas:

1. **Definición del Producto** - El administrador crea una plantilla de producto con reglas de precios y provisión.
2. **Creación del Servicio** - El cliente ordena el producto, el sistema provisióna la instancia del servicio.
3. **Ciclo de Vida del Servicio** - El cliente utiliza el servicio, agrega complementos/recargas, modifica el servicio.
4. **Desprovisión** - El servicio se termina, los recursos se liberan.

## Comprendiendo los Precios: Mayorista vs Minorista

Cada producto y servicio en OmniCRM tiene dos dimensiones de precios: **mayorista** y **minorista**.

### Costo Mayorista

El costo mayorista representa el costo real para entregar el servicio:

- Costos de infraestructura y ancho de banda
- Tarifas de licencia
- Costos de equipos
- Gastos operativos

### Costo Minorista

El costo minorista es el monto cobrado al cliente.

## Costos de Configuración

Tanto el mayorista como el minorista tienen variantes de costo de configuración para cargos de provisión únicos:

- `wholesale_setup_cost` - Su costo para provisionar
- `retail_setup_cost` - Monto cobrado al cliente por la activación

### Ejemplo:

```
{  
  "retail_cost": 15.00,  
  "wholesale_cost": 5.00,  
  "retail_setup_cost": 0.00,  
  "wholesale_setup_cost": 1.00  
}
```

## Etapa 1: Creando una Definición de Producto

Los productos son plantillas que definen qué se provisiona y cómo se cobra a los clientes.

### Creando un Producto de SIM Móvil

Vamos a crear un producto de SIM móvil prepago con 20GB de datos por mes.

#### Paso 1: Navegar a la Gestión de Productos

Desde la interfaz de administración, vaya a **Productos** → **Crear Producto**.

#### Paso 2: Definir Información Básica

```
{  
  "product_name": "Prepaid Mobile 20GB",  
  "product_slug": "prepaid-mobile-20gb",  
  "category": "standalone",  
  "service_type": "mobile",  
  "enabled": true,  
  "icon": "fa-solid fa-sim-card",  
  "comment": "Prepaid mobile SIM with 20GB data, unlimited calls &  
  texts"  
}
```

#### Explicaciones de Campos:

- `product_name` - Nombre visible para el cliente que se muestra en el catálogo.
- `product_slug` - Identificador seguro para URL utilizado en llamadas API y

enlaces.

- **category** - "standalone" significa que esto crea un nuevo servicio (vs complemento/paquete).
- **service\_type** - Agrupa productos relacionados, utilizado para filtrar complementos.
- **enabled** - Debe ser verdadero para que el producto sea ordenable.
- **icon** - Icono de FontAwesome mostrado en la interfaz de usuario.
- **comment** - Notas internas para referencia del personal.

### Paso 3: Establecer Precios

```
{  
  "retail_cost": 15.00,  
  "wholesale_cost": 5.00,  
  "retail_setup_cost": 0.00,  
  "wholesale_setup_cost": 1.00,  
  "contract_days": 30  
}
```

#### Desglose de Precios:

- Ingresos mensuales por cliente: £15.00
- Costo mensual para entregar: £5.00
- Margen de beneficio mensual: £10.00 (200% de margen, 67% de margen)
- Beneficio por configuración: -£1.00 (subvencionado para atraer clientes)
- Duración del contrato: 30 días (renovación mensual)

### Paso 4: Definir Elegibilidad del Cliente

```
{  
  "residential": true,  
  "business": false,  
  "customer_can_purchase": true,  
  "available_from": "2025-01-01T00:00:00Z",  
  "available_until": null  
}
```

- Los clientes residenciales pueden ordenar.
- Los clientes comerciales no pueden (línea de producto diferente).
- Compra de autoservicio habilitada.
- Disponible desde el 1 de enero de 2025 en adelante.
- Sin fecha de finalización (oferta continua).

### Paso 5: Configurar Renovación Automática

```
{  
  "auto_renew": "prompt",  
  "allow_auto_renew": true  
}
```

- "prompt" - Preguntar al cliente si desea la renovación automática al realizar la compra.
- "true" - Renovar automáticamente sin preguntar.
- "false" - Nunca renovar automáticamente (solo recarga manual).
- `allow_auto_renew: true` - El cliente puede habilitar/deshabilitar la renovación automática más tarde.

## Paso 6: Especificar Requisitos de Inventario

Los requisitos de inventario definen qué recursos físicos o virtuales deben ser asignados al provisionar este producto. Este es un paso crítico que conecta su catálogo de productos con su Sistema de Gestión de Inventario `<administration_inventory>`.

```
{
  "inventory_items_list": "[ 'SIM Card', 'Mobile Number' ]"
}
```

### ¿Qué Son los Elementos de Inventario?

Los elementos de inventario son recursos rastreables almacenados en el sistema de inventario de OmniCRM. Cada elemento tiene:

- **Tipo** - Definido por la Plantilla de Inventario (por ejemplo, "SIM Card", "Mobile Number", "Modem").
- **Atributos únicos** - Números de serie, direcciones MAC, números de teléfono, etc.
- **Estado** - En Stock, Asignado, Desmantelado, etc.
- **Ubicación** - Ubicación física o lógica.

### Cómo Funcionan los Requisitos de Inventario:

La `inventory_items_list` es una lista de Python (como una cadena) que contiene nombres de tipos de inventario. Cada nombre debe coincidir exactamente con un nombre de Plantilla de Inventario `<administration_inventory>` existente.

### Ejemplo de Requisitos de Inventario:

```
# Producto de SIM móvil
inventory_items_list: "[ 'SIM Card', 'Mobile Number' ]"

# Servicio de internet fijo
inventory_items_list: "[ 'Modem Router', 'Static IP Address' ]"

# Servicio digital (sin elementos físicos)
inventory_items_list: "[ ]"

# Inalámbrico fijo con CPE
inventory_items_list: "[ 'Fixed Wireless CPE', 'IPv4 Address', 'IPv6
```

Prefix']"

## El Proceso del Selector de Inventario

Cuando un usuario provisiona un producto con requisitos de inventario, el sistema impone un proceso de selección obligatorio:

### 1. Botón de Provisión Clicado

Después de seleccionar el producto, el usuario hace clic en "Provisionar". En lugar de provisionar inmediatamente, el sistema verifica `inventory_items_list`.

### 2. Modal del Selector de Inventario Aparece

Si se requiere inventario, aparece un cuadro de diálogo modal con un menú desplegable separado para cada tipo de inventario:

### 3. Filtrando Inventario Disponible

El menú desplegable para cada tipo de inventario solo muestra elementos que son:

- **Tipo Correcto** - Coincide exactamente con el nombre de la plantilla de inventario.
- **Estado Disponible** - `item_state` es "Nuevo" o "En Stock" (no "Asignado" o "Dañado").
- **No Asignado** - `service_id` y `customer_id` son NULL.
- **En Stock en Ubicación** - Opcionalmente filtrado por ubicación de almacén/tienda.

### Ejemplo de Opciones del Menú Desplegable:

Para el tipo de inventario "SIM Card", el menú desplegable podría mostrar:

Cada opción muestra:

- ID de inventario o número de referencia.
- Identificador principal (`itemtext1` - por ejemplo, ICCID para SIM, número para teléfono).
- Ubicación actual (`item_location`).

### 4. Selección Requerida para Proceder

**Regla Crítica:** La provisión NO puede proceder sin seleccionar todos los elementos de inventario requeridos.

- El botón "Continuar" está deshabilitado hasta que todos los menús desplegables tengan selecciones.
- El usuario debe seleccionar un elemento para cada tipo de inventario.

- El sistema valida las selecciones antes de proceder.

## 5. Inventario Seleccionado Pasado a Ansible

Una vez que el usuario hace clic en "Continuar", los IDs de inventario seleccionados se pasan al libro de jugadas de Ansible como variables:

```
# Usuario seleccionado:
# - SIM Card inventory_id: 5001
# - Mobile Number inventory_id: 5002

# Variables pasadas a Ansible:
{
  "product_id": 42,
  "customer_id": 123,
  "SIM Card": 5001,           # ID de Inventario
  "Mobile Number": 5002,      # ID de Inventario
  "access_token": "eyJ..."
}
```

**Nota:** El nombre de la variable coincide exactamente con el tipo de inventario. El libro de jugadas utiliza `hostvars[inventory_hostname]['SIM Card']` para acceder al ID de inventario.

## 6. El Libro de Jugadas Obtiene Detalles Completos del Inventario

El libro de jugadas de Ansible utiliza el ID de inventario para obtener detalles completos:

```
- name: Get SIM Card details from inventory
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ hostvars[inventory_hostname]['SIM Card'] }}"
    method: GET
    headers:
      Authorization: "Bearer {{ access_token }}"
  register: api_response_sim

- name: Extract ICCID and IMSI
  set_fact:
    iccid: "{{ api_response_sim.json.itemtext1 }}"
    imsi: "{{ api_response_sim.json.itemtext2 }}"
```

Ahora el libro de jugadas tiene todos los detalles de la SIM (ICCID, IMSI, etc.) para provisionar al suscriptor en el HSS.

## 7. Estado del Inventario Cambiado a "Asignado"

Después de que se crea el registro del servicio, el libro de jugadas actualiza el

inventario para vincularlo al servicio:

```
- name: Assign SIM Card to Service
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ hostvars[inventory_hostname]['SIM Card'] }}"
    method: PATCH
  body:
    {
      "service_id": "{{ service_creation_response.json.service_id }}"
    },
    "customer_id": "{{ customer_id }}",
    "item_state": "Assigned"
  }
```

**Importante:** La asignación de inventario ocurre **durante la ejecución del libro de jugadas** como una tarea específica, NO cuando se hace clic en el botón de provisión. Esto significa:

- **Riesgo de Doble Asignación:** Entre hacer clic en "Provisionar" y que el inventario sea asignado, otro usuario podría teóricamente seleccionar el mismo elemento de inventario.
- **Mejor Práctica:** Para operaciones de alto volumen, implemente bloqueo de inventario o use transacciones de base de datos.
- **Rollback en Caso de Fallo:** Si el libro de jugadas falla antes de la asignación de inventario, el inventario permanece no asignado y disponible para su reutilización.

### ¿Por Qué No Asignar Antes?

El inventario no se asigna cuando se hace clic en "Provisionar" porque:

1. **ID de Servicio Necesario:** El `service_id` no existe hasta que el servicio se crea en el libro de jugadas.
2. **Simplicidad de Rollback:** Si la provisión falla temprano (por ejemplo, falla en la creación de la cuenta OCS), no es necesario limpiar el inventario.
3. **Flexibilidad:** El libro de jugadas puede decidir no asignar inventario basado en lógica condicional.

### Manejo de Provisiones Fallidas:

Cuando una provisión falla después de que se ha asignado el inventario, el bloqueo de rescate debe liberar el inventario:

```
rescue:
  - name: Release inventory on failure
    uri:
      url: "{{ crm_config.crm.base_url }}/crm/inventory/
inventory_id/{{ hostvars[inventory_hostname]['SIM Card'] }}"
```

```

method: PATCH
body:
{
  "service_id": null,
  "customer_id": null,
  "item_state": "In Stock"
}
when: service_id is defined # Solo si se creó el servicio

```

Esto asegura que el inventario no quede en un estado "Asignado" para un servicio inexistente o fallido.

## Cuando la Lista de Inventario Está Vacía

Si `inventory_items_list: "[]"` (lista vacía), el selector de inventario se omite por completo y la provisión procede de inmediato. Esto es común para:

- **Productos digitales** - Licencias de software, cuentas VPN.
- **Complementos de servicio** - Recargas de datos que no necesitan nuevo hardware.
- **Servicios virtuales** - Que no consumen recursos rastreables.

**Ejemplo:** Un complemento "5GB Data Boost" tiene `inventory_items_list: "[]"` porque solo agrega saldo a un servicio existente sin necesidad de nuevo hardware.

## Configuración de Plantilla de Inventario

Antes de usar un tipo de inventario en `inventory_items_list`, debe crear la Plantilla de Inventario:

1. Navegue a **Administración → Inventario → Plantillas**.
2. Cree una plantilla con el nombre exacto (por ejemplo, "SIM Card").
3. Defina los campos:
  - `itemtext1_label: "ICCID"`.
  - `itemtext2_label: "IMSI"`.
  - `itemtext3_label: "Código PUK"`.
4. Agregue elementos de inventario de este tipo al stock.

Para obtener detalles completos sobre la creación y gestión de plantillas de inventario, consulte Gestión de Inventario `<administration_inventory>`.

## Múltiples Elementos del Mismo Tipo

Si bien `inventory_items_list` es un array, tener tipos duplicados (por ejemplo, `"['SIM Card', 'SIM Card']"`) **no se recomienda** ya que puede causar confusión en la interfaz de usuario y en la nomenclatura de variables del libro de jugadas.

Para escenarios que requieren múltiples elementos similares:

### Opción 1: Crear nombres de plantillas de inventario distintos

```
# Servicio de teléfono Dual-SIM
inventory_items_list: ["'Primary SIM Card', 'Secondary SIM Card',
'Mobile Number']"
```

Cree plantillas separadas: "Primary SIM Card" y "Secondary SIM Card" con los mismos campos pero diferentes nombres.

### Opción 2: Usar un solo elemento de inventario agrupado

```
# Kit Dual-SIM
inventory_items_list: ["'Dual SIM Kit', 'Mobile Number']"
```

Donde la plantilla de inventario "Dual SIM Kit" tiene campos para ambas SIMs (itemtext1: ICCID Primario, itemtext2: ICCID Secundario, etc.).

## Escenarios Comunes de Inventario

### Servicio Móvil:

```
inventory_items_list: ["'SIM Card', 'Mobile Number']"
```

- SIM Card: Física o eSIM con ICCID/IMSI.
- Mobile Number: Número de teléfono (MSISDN).

### Internet Fijo:

```
inventory_items_list: ["'Modem Router', 'Static IP Address']"
```

- Modem Router: Dispositivo CPE con dirección MAC.
- Static IP Address: IPv4 del grupo de direcciones.

### Inalámbrico Fijo:

```
inventory_items_list: ["'Fixed Wireless CPE', 'IPv4 Address', 'IPv6
Prefix']"
```

- CPE: Equipos de cliente (antena, módem).
- IPv4: Dirección IP pública.
- IPv6 Prefix: Prefijo /56 o /64.

**Nota:** Las citas y la programación **no** son elementos de inventario. Use sistemas de programación/calendario separados para citas de instalación.

### Servicio VoIP:

```
inventory_items_list: "[ 'DID Number' ]"
```

- DID Number: Número de teléfono de marcación directa.

**Nota:** Los nombres de usuario SIP, contraseñas y configuraciones de cuenta son **generados programáticamente** por el libro de jugadas de provisión, no seleccionados del inventario.

### **GPON/Fibra:**

```
inventory_items_list: "[ 'ONT Device', 'GPON Port', 'IPv4 Address', 'Fiber Drop Cable']"
```

- ONT Device: Terminal de Red Óptica con número de serie.
- GPON Port: Puerto específico en OLT con conexión de fibra.
- IPv4 Address: IP pública o privada.
- Fiber Drop Cable: Cable de fibra física de la calle a las instalaciones (rastreado para gestión de activos).

### **Alquiler de Equipos:**

```
inventory_items_list: "[ 'Rental Modem' ]"
```

- Rastrear qué módem está con qué cliente.
- Importante para recuperar equipos en caso de cancelación.

## **Por Qué Importan los Requisitos de Inventario**

### **1. Prevenir Doble Asignación**

Sin seguimiento de inventario, podría accidentalmente:

- Asignar la misma tarjeta SIM a dos clientes.
- Asignar la misma dirección IP a múltiples servicios.
- Enviar el mismo número de serie de equipo a diferentes ubicaciones.

El selector de inventario asegura que cada elemento esté asignado a exactamente un servicio.

### **2. Registro de Auditoría**

La asignación de inventario crea un registro de auditoría completo:

- Qué tarjeta SIM está con qué cliente.
- Cuándo fue asignada.
- Qué servicio está utilizando qué número de teléfono.
- Historial de equipos (quién lo tuvo, cuándo, para qué servicio).

### **3. Planificación de Recursos**

Rastrear niveles de inventario:

- Alertar cuando las tarjetas SIM estén bajas.
- Reordenar antes de que se agoten.
- Planificar horarios de técnicos según la disponibilidad de CPE.
- Gestionar la asignación de espacio de direcciones IP.

#### **4. Seguimiento de Costos**

Vincular el costo mayorista a un elemento específico:

- Rastrear el costo de cada tarjeta SIM.
- Calcular la depreciación del equipo.
- Identificar elementos perdidos o robados.
- COGS (Costo de Bienes Vendidos) preciso.

#### **5. Desprovisión**

Cuando se cancela el servicio, el inventario puede ser:

- Liberado de nuevo al stock (tarjetas SIM, módems).
- Retirado (equipos dañados).
- Devuelto al proveedor (equipos de alquiler).
- Conservado por un período de gracia (números de teléfono antes de la liberación).

### **Resolución de Problemas con el Selector de Inventario**

**Problema:** Aparece el mensaje "No hay inventario disponible".

**Causas:**

- No existen elementos de inventario del tipo requerido en la base de datos.
- Todos los elementos ya están "Asignados" a otros servicios.
- Los elementos están marcados como "Dañados" o "Fuera de Servicio".
- El nombre de la plantilla de inventario no coincide exactamente (sensible a mayúsculas).

**Solución:**

1. Verifique que la plantilla de inventario exista: **Administración → Inventario → Plantillas**.
2. Verifique que el nombre de la plantilla coincida exactamente (incluyendo espacios, mayúsculas).
3. Agregue elementos de inventario de este tipo: **Administración → Inventario → Agregar Elemento**.
4. Verifique que los elementos estén en estado "Nuevo" o "En Stock".
5. Verifique que los elementos no estén ya asignados (`service_id` debe ser `NULL`).

**Problema:** El selector de inventario no aparece.

**Causas:**

- `inventory_items_list` está vacío: "[]".
- `inventory_items_list` es NULL o no está configurado.
- La categoría del producto es "complemento" y hereda el inventario del servicio padre.

**Solución:**

- Si se necesita inventario, establezca `inventory_items_list: "['Type1', 'Type2']"`.
- Verifique que la definición del producto se haya guardado correctamente.
- Verifique la respuesta de la API para que el producto incluya `inventory_items_list`.

**Problema:** El libro de jugadas falla con "inventario no encontrado".

**Causas:**

- El libro de jugadas hace referencia al nombre de variable incorrecto.
- El ID de inventario no se pasó correctamente.
- El inventario fue eliminado entre la selección y la provisión.

**Solución:**

- Verifique que el libro de jugadas use la variable correcta: `hostvars[inventory_hostname]['SIM Card']`.
- Verifique que la variable sea un entero: `{{ hostvars[inventory_hostname]['SIM Card'] | int }}`.
- Agregue manejo de errores en el libro de jugadas para inventario faltante.

Consulte Gestión de Inventario `<administration_inventory>` para obtener detalles completos sobre la creación de plantillas, la adición de elementos y la gestión de niveles de stock.

## Paso 7: Definir Características y Términos

Las características y términos son contenido de marketing y legal visible para el cliente que ayuda a los clientes a entender qué están comprando y las obligaciones involucradas.

```
{  
  "features_list": "20GB High-Speed Data. Unlimited Calls & Texts. EU  
Roaming Included. No Contract. 30-Day Expiry",  
  "terms": "Credit expires after 30 days. Data, calls, and texts  
valid only within expiry period. Fair use policy applies. See website  
for full terms."  
}
```

}

## Propósito y Valor Comercial

### Lista de Características - Marketing y Ventas:

La lista de características cumple múltiples funciones comerciales críticas:

1. **Diferenciación de Productos** - Ayuda a los clientes a comparar rápidamente productos y elegir el adecuado.
  - "Prepaid Mobile 20GB" vs "Prepaid Mobile 50GB" - las características muestran claramente la diferencia.
  - Sin características, los clientes solo ven el precio, perdiendo la propuesta de valor.
2. **Comunicación de Marketing** - Puntos de venta clave mostrados de manera prominente.
  - "EU Roaming Included" atrae a viajeros internacionales.
  - "No Contract" atrae a clientes reacios al compromiso.
  - Las características impulsan las decisiones de compra.
3. **Expectativas del Cliente** - Establece expectativas claras sobre lo que está incluido.
  - Reduce las llamadas de soporte ("¿Esto incluye llamadas?" → claramente listado).
  - Previene malentendidos y solicitudes de reembolso.
  - Genera confianza a través de la transparencia.
4. **Autoservicio** - Permite a los clientes seleccionar por sí mismos los productos apropiados.
  - El cliente lee las características, entiende la oferta, toma una decisión informada.
  - Reduce la necesidad de explicación por parte del personal de ventas.
  - Acelera el proceso de compra.
5. **SEO y Descubridabilidad** - Las características pueden ser indexadas para búsqueda.
  - El cliente busca "plan móvil de llamadas ilimitadas" → el producto aparece.
  - Mejora la capacidad de búsqueda del catálogo de productos.

### Términos y Condiciones - Legal y Cumplimiento:

Los términos sirven a propósitos legales y operativos:

1. **Protección Legal** - Protege al negocio de disputas y responsabilidad.
  - "El crédito expira después de 30 días" - el cliente no puede exigir un reembolso a los 31 días.
  - "Se aplica política de uso justo" - previene abusos (usar toda la oficina en un plan móvil).
  - Crea un acuerdo vinculante.
2. **Gestión de Expectativas** - Previene la insatisfacción del cliente.
  - "Válido solo dentro del período de caducidad" - el cliente conoce la

- fecha límite de uso.
  - "No puede ser reembolsado" (para complementos) - previene compras fraudulentas.
  - Reduce devoluciones y quejas.
3. **Cumplimiento Regulatorio** - Cumple con requisitos legales.
- Las leyes de protección al consumidor requieren términos claros.
  - Las regulaciones de telecomunicaciones exigen divulgación.
  - Se pueden hacer referencia a términos de GDPR/privacidad.
4. **Límites Operativos** - Define el alcance y las limitaciones del servicio.
- "Sujeto a cobertura de red" - no es responsable por zonas muertas.
  - "La velocidad puede variar" - gestiona expectativas sobre velocidades "hasta".
  - "El equipo debe ser devuelto" - asegura la recuperación del equipo de alquiler.
5. **Registro de Auditoría** - Prueba que el cliente fue informado.
- El cliente aceptó los términos al realizar la compra.
  - El sistema registra la marca de tiempo de aceptación.
  - Defendible en disputas o procedimientos legales.

### **Ejemplo del Mundo Real:**

El cliente compra un plan de "Llamadas y Mensajes Ilimitados", luego lo usa para telemarketing (10,000 llamadas/día). Sin términos:

- Cliente: "¡Dijiste ilimitado!"
- Proveedor: "Nos referíamos al uso personal..."
- Cliente: "¡Eso no es lo que publicitaste!"
- Resultado: Disputa, posible queja ante el regulador, daño a la marca.

Con términos: "Se aplica política de uso justo. El servicio es solo para uso personal. El uso comercial está prohibido."

- Proveedor: Señala los términos que el cliente aceptó.
- El cliente no puede alegar ignorancia.
- Base legal para suspender el servicio.
- Disputa resuelta a favor del proveedor.

### **Formato de la Lista de Características:**

Entender el formato correcto es crítico porque **el formato incorrecto rompe la visualización de la interfaz de usuario**. Las características podrían aparecer como una larga cadena en lugar de puntos de viñeta, o no mostrarse en absoluto.

El campo `features_list` se puede formatear de dos maneras:

#### **Opción 1: Cadena Separada por Puntos (Recomendada)**

Las características se separan por un punto y un espacio ("." "."). La interfaz de usuario se divide en este delimitador y renderiza cada característica como un

punto de viñeta.

### ¿Por qué este formato?

- Simple de editar - solo escriba características con puntos entre ellas.
- Sin caracteres especiales que escapar.
- Funciona de manera confiable en todos los componentes de la interfaz de usuario.
- Fácil de actualizar sin romper la sintaxis JSON.

### Correcto vs Incorrecto:

#### Opción 2: Cadena de Array JSON

```
["20GB High-Speed Data", "Unlimited Calls & Texts", "EU Roaming Included"]
```

La interfaz de usuario también puede analizar arrays JSON. Tenga en cuenta que esto es una **cadena que contiene JSON**, no un array JSON real en la base de datos.

### ¿Por qué existe este formato?

- Permite características con puntos en ellas (por ejemplo, "Hasta 100Mbps. Sujeto a disponibilidad.").
- La generación programática desde scripts/API es más fácil.
- Importado de catálogos de productos externos que utilizan arrays.

**Importante:** Esto debe ser una sintaxis válida de lista de Python como una cadena. Comillas simples alrededor de cada elemento, comillas dobles alrededor de toda la cadena.

### ¿Qué Formato Usar?

- **Separado por puntos** - Para la creación manual de productos en la interfaz de usuario (más simple, menos propenso a errores).
- **Array JSON** - Para la creación de productos basada en API/script (más robusto para características complejas).

Ambos formatos producen una salida idéntica en la interfaz de usuario; solo afectan cómo se ingresan los datos.

### Dónde Aparecen las Características en la Interfaz de Usuario:

#### 1. Catálogo de Productos (Vista del Cliente)

Cuando los clientes navegan por los productos disponibles, las características se muestran como puntos de viñeta en cada tarjeta de producto:

## **2. Página de Detalles del Producto**

Al hacer clic en "Ver Detalles", se muestra la información completa del producto, incluyendo:

- Nombre del producto e icono.
- Precios (costo mensual, costo de configuración).
- Lista completa de características (puntos de viñeta).
- Términos y condiciones (ver abajo).
- Disponibilidad y elegibilidad.

## **3. Confirmación de Provisión**

Durante la provisión, las características se muestran para que el usuario las revise antes de confirmar:

Características: • 20GB High-Speed Data • Unlimited Calls & Texts • EU Roaming Included • No Contract • 30-Day Expiry

Costo: £15.00/mes Configuración: £0.00

[Cancelar] [Confirmar y Provisionar]

## **4. Detalles del Servicio (Después de la Provisión)**

Después de que el servicio esté activo, las características se muestran en la página de detalles del servicio para referencia del cliente.

### **Formato de Términos y Condiciones:**

El campo `terms` es texto plano que puede incluir saltos de línea:

### **Dónde Aparecen los Términos en la Interfaz de Usuario:**

#### **1. Página de Detalles del Producto**

Los términos se muestran en una sección colapsada que se expande al hacer clic:

#### **2. Confirmación de Pedido**

Durante la provisión, se requiere que el usuario acepte los términos mediante una casilla de verificación:

El botón [Provisionar] está deshabilitado hasta que se marque.

#### **3. Facturas**

Los términos del servicio pueden incluirse en las facturas como notas al pie para mayor claridad.

## Mejores Prácticas:

- **Características:** Mantener concisas (menos de 50 caracteres cada una), centrarse en beneficios clave.
  - **Términos:** Incluir requisitos legales críticos, políticas de caducidad, políticas de uso justo.
  - **Ambos:** Actualizar cuando el producto cambie para mantener informados a los clientes.

## Paso 8: Vincular el Libro de Jugadas de Provisión de Ansible

```
{  
  "provisioning_play": "play_local_mobile_sim",  
  "provisioning_json_vars": {"  
    \"days\": 30,  
    \"data_gb\": 20,  
    \"voice_minutes\": \"unlimited\",  
    \"sms_count\": \"unlimited\"  
  }  
}
```

- **provisioning\_play** - Nombre del libro de jugadas de Ansible (sin la extensión .yaml).
  - **provisioning\_json\_vars** - Variables predeterminadas pasadas al libro de jugadas.
  - El libro de jugadas debe existir en: OmniCRM-API/Provisioners/plays/**play\_local\_mobile\_sim.yaml**.

## Definición Completa del Producto

```

"available_from": "2025-01-01T00:00:00Z",
"available_until": null,

"auto_renew": "prompt",
"allow_auto_renew": true,

"inventory_items_list": "[ 'SIM Card', 'Mobile Number' ]",

"features_list": [
  '20GB High-Speed Data',
  'Unlimited Calls & Texts',
  'EU Roaming Included',
  'No Contract',
  '30-Day Expiry'
],
"terms": "Credit expires after 30 days. Data, calls, and texts valid only within expiry period. Fair use policy applies.",

"provisioning_play": "play_local_mobile_sim",
"provisioning_json_vars": "{\n  \"days\": 30,\n  \"data_gb\": 20,\n  \"voice_minutes\": \"unlimited\",\n  \"sms_count\": \"unlimited\"\n}"
}

```

## Creando un Producto Complementario

Los complementos mejoran o modifican los servicios existentes. Vienen en dos tipos: **complementos virtuales** (sin recursos físicos) y **complementos de hardware** (requieren inventario).

### Ejemplo 1: Complemento Virtual (5GB Data Boost)

Un complemento digital que agrega datos a un servicio móvil existente:

```

{
  "product_name": "5GB Data Boost",
  "product_slug": "5gb-data-boost",
  "category": "addon",
  "service_type": "mobile",
  "enabled": true,
  "icon": "fa-solid fa-plus",
  "comment": "Add 5GB extra data to existing mobile service",

  "retail_cost": 5.00,
  "wholesale_cost": 1.50,
}

```

```

"retail_setup_cost": 0.00,
"wholesale_setup_cost": 0.00,
"contract_days": 0,

"residential": true,
"business": true,
"customer_can_purchase": true,

"auto_renew": "false",
"allow_auto_renew": false,

"inventory_items_list": "[]",
"relies_on_list": "",

"features_list": "5GB High-Speed Data. Valid for 7 Days",
"terms": "Data expires after 7 days or when exhausted. Cannot be refunded.",

"provisioning_play": "play_topup_charge_then_action",
"provisioning_json_vars": "{\n    \"data_gb\": 5,\n    \"days\": 7\n}"
}
}

```

## Ejemplo 2: Complemento de Hardware (Alquiler de Módem)

Un complemento que proporciona equipo físico para un servicio de fibra existente:

```

{
  "product_name": "WiFi 6 Modem Rental",
  "product_slug": "wifi6-modem-rental",
  "category": "addon",
  "service_type": "internet",
  "enabled": true,
  "icon": "fa-solid fa-router",
  "comment": "Add WiFi 6 modem to fiber service - rental",

  "retail_cost": 10.00,
  "wholesale_cost": 3.00,
  "retail_setup_cost": 0.00,
  "wholesale_setup_cost": 45.00,
  "contract_days": 30,

  "residential": true,
  "business": true,
  "customer_can_purchase": true,
}

```

```

"auto_renew": "true",
"allow_auto_renew": true,

"inventory_items_list": "[ 'Rental Modem' ]",
"relies_on_list": "",

"features_list": "WiFi 6 (802.11ax). Dual-band 2.4GHz + 5GHz. Up to
40 devices. Parental controls",
"terms": "Equipment rental. Must be returned on service
cancellation or £150 replacement fee applies. Equipment remains
property of provider.",

"provisioning_play": "play_addon_assign_modem",
"provisioning_json_vars": "{\n    \"device_type\": \"modem_router\",\n    \"requires_configuration\": true\n}"
}
}

```

### Diferencias Clave para Complementos:

- **category**: "addon" - Aplicado a un servicio existente, no autónomo.
- **contract\_days**: 0 (virtual) o 30 (alquiler recurrente) - Frecuencia de facturación.
- **inventory\_items\_list**: "[]" (virtual) o "[ 'Rental Modem' ]" (hardware) - Recursos físicos.
- **auto\_renew**: "false" (una sola vez) o "true" (alquiler) - Comportamiento recurrente.
- **relies\_on\_list**: "" - Vacío significa que se aplica a cualquier servicio de tipo de servicio coincidente.

### Por Qué los Complementos de Hardware Necesitan inventario:

Los complementos de hardware requieren **inventory\_items\_list** porque:

1. **Rastrear Equipos** - Saber qué módem está con qué cliente.
2. **Prevenir Agotamientos de Stock** - No se puede provisionar el complemento si no hay módems en stock.
3. **Recuperación** - Cuando el cliente cancela, saber qué equipo recuperar.
4. **Seguimiento de Costos** - Vincular el costo mayorista a un número de serie específico.
5. **Depreciación** - Rastrear el valor del equipo durante el período de alquiler.
6. **Garantía** - Identificar unidades defectuosas por número de serie.

### Flujo de Provisión de Complementos con inventario:

Cuando un cliente agrega "Alquiler de Módem WiFi 6" a su servicio de fibra:

1. **Complemento Seleccionado** - El cliente hace clic en "Agregar al

- Servicio".
2. **Selector de Inventario Aparece** - Al igual que los servicios autónomos:
  3. **Pago Procesado** - Se cobra £10.00 de alquiler mensual.
  4. **Módem Asignado** - Inventario actualizado:
    - `service_id`: Vinculado al servicio de fibra.
    - `customer_id`: Vinculado al cliente.
    - `item_state`: "Asignado".
  5. **Envío Activado** - El sistema de cumplimiento es notificado para enviar el módem.
  6. **Instalación** - El cliente recibe el módem, lo conecta al ONT.
  7. **Facturación Recurrente** - Se cobra £10/mes hasta que se cancele el complemento.

### **Desprovisión de Complementos de Hardware:**

Cuando el cliente cancela el alquiler del módem:

1. **Cancelación Iniciada** - El cliente hace clic en "Eliminar Complemento".
2. **Proceso de Devolución Iniciado**:
  - Se envía un correo electrónico con instrucciones de devolución.
  - Se genera una etiqueta de envío prepagada.
  - Período de gracia de 14 días antes de la penalización.
3. **Equipo Devuelto**:
  - Inventario actualizado: `item_state` = "In Stock" (después de la renovación).
  - O `item_state` = "Dañado" (si es defectuoso).
  - Vinculado al siguiente cliente una vez renovado.
4. **Sin Devolución**:
  - Despues de 14 días, se cobra una tarifa de reemplazo de £150.
  - Inventario marcado: `item_state` = "Perdido".
  - Costo mayorista (£45) + valor de reemplazo recuperado.

### **Precios para Complementos:**

Los complementos pueden tener precios diferentes a los servicios autónomos:

- Los complementos virtuales típicamente no tienen costos de configuración.
- Los complementos de hardware pueden tener costos de configuración mayorista para el equipo.
- Los complementos de alquiler recurrentes utilizan `contract_days` para la frecuencia de facturación.

## **Etapa 2: El Proceso de Provisión**

Cuando un cliente ordena el producto "Prepaid Mobile 20GB", OmniCRM orquesta la provisión a través de Ansible.

# Diagrama de Flujo de Provisión

Cliente Ordena → Selección de Inventario → Trabajo de Provisión Creado ↓ ↓  
Pago Autorizado ← Variables Ensambladas ← Libro de Jugadas de Ansible  
Ejecutado ↓ ↓ Registro de Servicio Creado → Configuración de Cuenta OCS →  
Inventario Asignado → Servicio Activo

## Flujo de Provisión Paso a Paso

### 1. El Cliente Inicia el Pedido

Desde la página del cliente:

- El personal hace clic en "Agregar Servicio".
- Selecciona "Prepaid Mobile 20GB" del carrusel de productos.
- Se muestran los detalles del producto y los precios.

### 2. Selección de Inventario

El sistema solicita el inventario requerido:

- **SIM Card** - El menú desplegable muestra las SIM disponibles en stock.
  - Ejemplo: "SIM-00123 - ICCID: 8944...".
- **Mobile Number** - El menú desplegable muestra los números de teléfono disponibles.
  - Ejemplo: "+44 7700 900123".

El personal o el cliente seleccionan elementos del inventario disponible.

### 3. Confirmación de Precios

El sistema muestra el precio final:

- Costo de configuración: £0.00 (activación gratuita).
- Costo mensual: £15.00.
- Total a pagar hoy: £15.00 (primer mes).
- Fecha de renovación: 30 días a partir de hoy.

Si se habilita la solicitud de renovación automática, el cliente elige:

- Renovar automáticamente este servicio cada 30 días.

### 4. Botón de Provisión Clicado

Cuando se hace clic en "Provisionar", la API:

- Crea un registro de Provision con estado "En Ejecución" (status=1).
- Fusiona variables del producto + solicitud + selecciones de inventario.

- Genera un hilo en segundo plano para ejecutar el libro de jugadas de Ansible.
- Devuelve provision\_id a la interfaz de usuario para seguimiento de estado.

## 5. Variables Ensambladas

El sistema fusiona variables de múltiples fuentes:

### Del Producto:

```
{
  "days": 30,
  "data_gb": 20,
  "voice_minutes": "unlimited",
  "sms_count": "unlimited"
}
```

### De la Solicitud:

```
{
  "product_id": 42,
  "customer_id": 123,
  "SIM Card": 5001,
  "Mobile Number": 5002
}
```

### Agregado por el Sistema:

```
{
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
  "initiating_user": 7
}
```

### Variables Finales Pasadas a Ansible:

```
{
  "product_id": 42,
  "customer_id": 123,
  "SIM Card": 5001,
  "Mobile Number": 5002,
  "days": 30,
  "data_gb": 20,
  "voice_minutes": "unlimited",
  "sms_count": "unlimited",
  "access_token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
  "initiating_user": 7
}
```

## 6. Ejecución del Libro de Jugadas de Ansible

El libro de jugadas `play_local_mobile_sim.yaml` se ejecuta con estas variables.

## Comprendiendo el Libro de Jugadas de Provisión de Ansible

Examinemos un libro de jugadas de provisión real para entender lo que sucede detrás de escena.

### Ejemplo de Libro de Jugadas de Provisión de SIM Móvil

**Ubicación:** OmniCRM-API/Provisioners/plays/play\_local\_mobile\_sim.yaml

#### Estructura de Alto Nivel:

```
- name: Mobile SIM Provisioning
  hosts: localhost
  gather_facts: no
  become: False

  tasks:
    - name: Main block
      block:
        # 1. Cargar configuración
        # 2. Obtener detalles del producto de la API
        # 3. Obtener detalles del cliente de la API
        # 4. Obtener detalles del inventario de la API
        # 5. Crear cuenta en OCS (CGRateS)
        # 6. Agregar saldos y asignaciones a OCS
        # 7. Crear registro de servicio en CRM
        # 8. Asignar inventario al servicio
        # 9. Registrar transacciones
        # 10. Enviar notificaciones de bienvenida

  rescue:
    # Rollback en caso de fallo
    # - Eliminar cuenta OCS
    # - Liberar inventario
    # - Registrar error
```

#### Recorrido Detallado del Libro de Jugadas:

##### Tarea 1: Cargar Configuración

```
- name: Include vars of crm_config
  ansible.builtin.include_vars:
    file: "../../crm_config.yaml"
    name: crm_config
```

Carga la configuración del sistema, incluyendo:

- URL y credenciales de OCS/CGRateS.
- URL base de CRM.
- Configuración del inquilino.

## Tarea 2: Obtener Detalles del Producto

```
- name: Get Product information from CRM API
  uri:
    url: "{{ crm_config.crm.base_url }}}/crm/product/product_id/{{ product_id }}"
  method: GET
  headers:
    Authorization: "Bearer {{ access_token }}"
  return_content: yes
  register: api_response_product
```

### Qué Hace Esto:

- Llama a GET /crm/product/product\_id/42.
- Recupera la definición completa del producto.
- Almacena en la variable api\_response\_product.

**Por Qué:** Aunque tenemos provisioning\_json\_vars del producto, obtenemos el producto completo para obtener:

- Precios más recientes (pueden haber cambiado desde que comenzó el pedido).
- Nombre del producto para el nombramiento del servicio.
- Lista de características para la documentación.
- Costos mayoristas para el seguimiento del margen.

## Tarea 3: Establecer Hechos del Paquete

```
- name: Set package facts
  set_fact:
    package_name: "{{ api_response_product.json.product_name }}"
    monthly_cost: "{{ api_response_product.json.retail_cost }}"
    setup_cost: "{{ api_response_product.json.retail_setup_cost }}"
```

Extrae valores comúnmente utilizados en variables simples para mayor legibilidad.

## Tarea 4: Obtener Detalles del Inventario

```
- name: Get SIM information from CRM API
  uri:
    url: "{{ crm_config.crm.base_url }}}/crm/inventory/inventory_id/{{
```

```

hostvars[inventory_hostname]['SIM Card'] }}}"
  method: GET
  headers:
    Authorization: "Bearer {{ access_token }}"
register: api_response_sim

- name: Set IMSI from Inventory response
  set_fact:
    imsi: "{{ api_response_sim.json.itemtext2 }}"
    iccid: "{{ api_response_sim.json.itemtext1 }}"

```

### Qué Hace Esto:

- Busca el ID de inventario de la tarjeta SIM 5001.
- Recupera detalles de la SIM:
  - itemtext1 = ICCID (número de la tarjeta SIM).
  - itemtext2 = IMSI (identidad del suscriptor).
- Hace lo mismo para el número de teléfono móvil (recupera el número de teléfono).

### Por Qué Esto Importa:

- IMSI es necesario para provisionar al suscriptor en el HSS (Servidor de Suscriptores de Hogar).
- ICCID se registra en las notas del servicio para solucionar problemas.
- El número de teléfono (MSISDN) se muestra al cliente y se utiliza para el enrutamiento.

### Tarea 5: Generar UUID del Servicio

```

- name: Generate UUID Fact
  set_fact:
    uuid: "{{ 99999999 | random | to_uuid }}"

- name: Set Service UUID
  set_fact:
    service_uuid: "Local_Mobile_SIM_{{ uuid[0:8] }}"

```

### Qué Hace Esto:

- Genera un UUID aleatorio.
- Crea un service\_uuid como Local\_Mobile\_SIM\_a3f2c1d8.

### Por Qué:

- El UUID del servicio es el identificador único en OCS/CGRateS.
- Se utiliza para todas las operaciones de carga.
- Debe ser globalmente único entre todos los servicios.

## Tarea 6: Crear Cuenta en OCS

```
- name: Create account in OCS
uri:
  url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body_format: json
  headers:
    Content-Type: "application/json"
body:
  {
    "method": "ApierV2.SetAccount",
    "params": [
      "Tenant": "{{ crm_config.ocs.ocsTenant }}",
      "Account": "{{ service_uuid }}",
      "ActionPlanIds": [],
      "ExtraOptions": {
        "AllowNegative": false,
        "Disabled": false
      },
      "ReloadScheduler": true
    ]
  }
register: ocs_create_response
```

### Qué Hace Esto:

- Llama a la API JSON-RPC de CGRateS.
- Crea una nueva cuenta con service\_uuid.
- Establece la cuenta como activa (no deshabilitada).
- Previene saldo negativo (modo prepago).

### Por Qué:

- La cuenta de OCS es donde ocurre toda la carga.
- Los saldos (datos, voz, SMS, dinero) se almacenan aquí.
- El uso se rastrea y califica en tiempo real.

## Tarea 7: Agregar Saldo de Datos

```
- name: Add 20GB Data Balance
uri:
  url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body_format: json
  body:
    {
      "method": "ApierV1.AddBalance",
      "params": [{
```

```

    "Tenant": "{{ crm_config.ocs.ocsTenant }}",
    "Account": "{{ service_uuid }}",
    "BalanceType": "*data",
    "Balance": [
        {
            "ID": "DATA_20GB_Monthly",
            "Value": 21474836480,
            "ExpiryTime": "+720h",
            "Weight": 10,
            "DestinationIDs": "*any"
        }
    ]
}

```

### Qué Hace Esto:

- Agrega 20GB de saldo de datos a la cuenta.
- Valor: 21474836480 bytes ( $20 * 1024 * 1024 * 1024$ ).
- Expira en 720 horas (30 días).
- Peso 10 (peso más alto consumido primero).

### Tarea 8: Agregar Voz y SMS Ilimitados

```

- name: Add Unlimited Voice
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body_format: json
    body:
      {
        "method": "ApierV1.AddBalance",
        "params": [
          {
            "Account": "{{ service_uuid }}",
            "BalanceType": "*voice",
            "Balance": [
              {
                "ID": "VOICE_Unlimited",
                "Value": 999999999,
                "ExpiryTime": "+720h"
              }
            ]
          }
        ]
      }

```

- Agrega 999,999,999 segundos de voz (prácticamente ilimitado).
- Expira en 30 días.

### Tarea 9: Crear Registro de Servicio en CRM

```

- name: Add Service via API
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/service/"

```

```

method: PUT
body_format: json
headers:
  Authorization: "Bearer {{ access_token }}"
body:
{
  "customer_id": "{{ customer_id }}",
  "product_id": "{{ product_id }}",
  "service_name": "Mobile - {{ phone_number }}",
  "service_type": "mobile",
  "service_uuid": "{{ service_uuid }}",
  "service_status": "Active",
  "service_provisioned_date": "{{ provision_datetime }}",
  "retail_cost": "{{ monthly_cost }}",
  "wholesale_cost": "{{ api_response_product.json.wholesale_cost }}",
  "icon": "fa-solid fa-sim-card"
}
register: service_creation_response

```

### Qué Crea Esto:

- Registro de servicio vinculado al cliente.
- Vincula a OCS a través de `service_uuid`.
- Almacena costos minoristas y mayoristas.
- Establece el estado en "Activo".
- Devuelve `service_id` para operaciones posteriores.

### Tarea 10: Asignar Inventario al Servicio

```

- name: Assign SIM Card to Service
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/inventory/inventory_id/{{ hostvars[inventory_hostname]['SIM Card'] }}"
    method: PATCH
    body_format: json
    headers:
      Authorization: "Bearer {{ access_token }}"
    body:
      {
        "service_id": "{{ service_creation_response.json.service_id }}",
        "customer_id": "{{ customer_id }}",
        "item_state": "Assigned"
      }

```

### Qué Hace Esto:

- Actualiza el registro de inventario de la tarjeta SIM.

- Establece `service_id` para vincular la SIM al servicio.
- Cambia el estado de "En Stock" a "Asignado".
- Repite para el inventario del número de teléfono móvil.

### Por Qué:

- Rastrear qué SIM está asignada a qué cliente.
- Prevenir la doble asignación de inventario.
- Habilitar informes y auditorías de inventario.

### Tarea 11: Registrar Transacción de Costo de Configuración

```
- name: Add Setup Cost Transaction
  uri:
    url: "{{ crm_config.crm.base_url }}/crm/transaction/"
  method: PUT
  body_format: json
  headers:
    Authorization: "Bearer {{ access_token }}"
  body:
    {
      "customer_id": "{{ customer_id }}",
      "service_id": "{{ service_creation_response.json.service_id }}",
      "title": "{{ package_name }} - Setup",
      "description": "Activation fee",
      "retail_cost": "{{ setup_cost }}",
      "wholesale_cost": "{{ api_response_product.json.wholesale_setup_cost }}"
    }
  
```

### Qué Hace Esto:

- Registra un cargo de £0.00 de configuración al cliente (minorista).
- Registra un costo mayorista de £1.00.
- Crea un registro de transacción para facturación.

### Tarea 12: Bloque de Rescate (Manejo de Errores)

```
rescue:
  - name: Remove account in OCS on failure
    uri:
      url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body:
      {
        "method": "ApierV2.RemoveAccount",
        "params": [
          {
            "Account": "{{ service_uuid }}"
          }
        ]
      }
  
```

```
        }
    }

- name: Fail the provisioning
  fail:
    msg: "Provisioning failed, rolled back OCS account"
```

## Qué Hace Esto:

- Si alguna tarea falla, se ejecuta el bloque de rescate.
- Elimina la cuenta de OCS que fue creada parcialmente.
- Libera el inventario de nuevo a "En Stock".
- Falla el trabajo de provisión con un mensaje de error.

## Por Qué:

- Previene cuentas huérfanas en OCS.
- Asegura un rollback limpio en errores.
- Mantiene la consistencia de datos.

## Provisión Completa: Qué Se Creó

Después de una provisión exitosa, el sistema tiene:

### 1. Cuenta OCS (CGRateS):

- ID de Cuenta: Local\_Mobile\_SIM\_a3f2c1d8.
- Saldos:
  - 20GB de datos (expira en 30 días).
  - Voz ilimitada (999M segundos, expira en 30 días).
  - SMS ilimitados (999M mensajes, expira en 30 días).

### 2. Registro de Servicio en CRM:

- ID de Servicio: 1234.
- Cliente: John Doe (customer\_id: 123).
- Producto: Prepaid Mobile 20GB (product\_id: 42).
- Nombre del Servicio: "Mobile - +44 7700 900123".
- UUID del Servicio: Local\_Mobile\_SIM\_a3f2c1d8.
- Estado: Activo.
- Costo Mensual: £15.00 (minorista), £5.00 (mayorista).
- Beneficio: £10.00/mes.

### 3. Asignaciones de Inventario:

- SIM Card 5001: Asignada al servicio 1234, cliente 123.
- Mobile Number 5002: Asignado al servicio 1234, cliente 123.

### 4. Registros de Transacciones:

- Se creó una transacción de costo de configuración.
- Se registró el cargo del primer mes.

## 5. El Cliente Ahora Puede:

- Ver el servicio en el portal de autoservicio.
- Ver saldo de 20GB de datos.
- Realizar llamadas y enviar SMS.
- Recargar o agregar complementos.
- Ver uso en tiempo real.

# Etapa 3: Agregar Complementos y Recargas

Después de que un servicio esté activo, los clientes pueden comprar complementos para mejorar su servicio.

## Flujo de Provisión de Complementos

Supongamos que el cliente ha utilizado 18GB de su asignación de 20GB y desea comprar el complemento "5GB Data Boost".

### 1. El Cliente Navega al Servicio

- Abre la página del servicio "Mobile - +44 7700 900123".
- Ve el uso actual: 18GB de 20GB utilizados (90%).
- Hace clic en "Aregar Complemento" o "Recargar".

### 2. El Sistema Filtra los Complementos Disponibles

Solo muestra complementos donde:

- `category = "addon"`.
- `service_type = "mobile"` (coincide con el tipo de servicio).
- `residential = true` (si el cliente es residencial).
- `enabled = true`.

El cliente ve: "5GB Data Boost - £5.00".

### 3. El Cliente Selecciona el Complemento

- Hace clic en "5GB Data Boost".
- Confirma la compra por £5.00.
- El sistema captura la autorización de pago.

### 4. Se Inicia la Provisión del Complemento

El sistema llama a `play_topup_charge_then_action.yaml` con variables:

```
{
  "product_id": 43, # Producto 5GB Data Boost
  "customer_id": 123,
  "service_id": 1234, # Servicio existente
  "access_token": "eyJ...",
  "data_gb": 5, # Desde provisioning_json_vars
  "days": 7       # Desde provisioning_json_vars
}
```

### Diferencia Clave con Servicios Autónomos:

- `service_id` está incluido (servicio existente a modificar).
- No se requiere inventario.
- No se crea un servicio (modifica el existente).

## Recorrido del Libro de Jugadas de Provisión de Complementos

### Tarea 1: Obtener Detalles del Servicio

```
- name: Get Service information from CRM API
  uri:
    url: "http://localhost:5000/crm/service/service_id/{{ service_id }}"
  method: GET
  headers:
    Authorization: "Bearer {{ access_token }}"
  register: api_response_service

- name: Set service facts
  set_fact:
    service_uuid: "{{ api_response_service.json.service_uuid }}"
    customer_id: "{{ api_response_service.json.customer_id }}"
```

### Por Qué:

- Necesitamos `service_uuid` para agregar saldo a la cuenta OCS correcta.
- Verifica que el servicio exista y esté activo.
- Asegura que el servicio pertenezca al cliente.

### Tarea 2: Cargar al Cliente

```
- name: Get Customer's Default Payment Method
  uri:
    url: "http://localhost:5000/crm/stripe/customer_id/{{ customer_id }}"
  method: GET
  headers:
    Authorization: "Bearer {{ access_token }}"
  register: api_response_stripe
```

```

- name: Get default card ID
  set_fact:
    customer_stripe_id: "{{ api_response_stripe.json | json_query(query) }}"
  vars:
    query: "data[?default_payment_method=='true'].customer_stripe_id[0]"

- name: Charge card
  uri:
    url: "http://localhost:5000/crm/stripe/charge_card/{{customer_stripe_id}}"
    method: POST
    body_format: json
    headers:
      Authorization: "Bearer {{ access_token }}"
    body:
      {
        "retail_cost": 5.00,
        "description": "5GB Data Boost",
        "customer_id": "{{ customer_id }}",
        "service_id": "{{ service_id }}",
        "product_id": "{{ product_id }}",
        "wholesale_cost": 1.50,
        "invoice": true
      }
  register: charge_response

- name: Assert payment successful
  assert:
    that:
      - charge_response.status == 200

```

## Qué Hace Esto:

- Encuentra el método de pago predeterminado del cliente en Stripe.
- Carga £5.00 a la tarjeta.
- Registra un costo mayorista de £1.50 para el seguimiento del margen.
- Crea una transacción vinculada al servicio.
- Agrega a la próxima factura.
- Falla la provisión si el pago falla.

## Por Qué Cargar Primero:

- No se entrega crédito hasta que se confirme el pago.
- Previene fraudes.
- Coincide el pago con la provisión del complemento.

### Tarea 3: Agregar Saldo de Datos a OCS

```
- name: Add 5GB Data Balance
uri:
  url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body_format: json
  body:
    {
      "method": "ApierV1.AddBalance",
      "params": [
        {
          "Account": "{{ service_uuid }}",
          "BalanceType": "*data",
          "Balance": {
            "ID": "DATA_5GB_Boost_{{ uuid }}",
            "Value": 5368709120,
            "ExpiryTime": "+168h",
            "Weight": 20
          }
        }
      ]
    }
```

#### Qué Hace Esto:

- Agrega 5GB (5368709120 bytes) a la cuenta.
- Expira en 168 horas (7 días).
- Peso 20 (peso más alto consumido primero - impulso antes de la asignación mensual).

#### Saldo del Cliente Después del Complemento:

- Mensual original: 2GB restantes (expira en 25 días).
- Nuevo impulso: 5GB (expira en 7 días).
- Total disponible: 7GB.
- Orden de uso: impulso consumido primero, luego mensual.

### Tarea 4: Registrar Transacción

```
- name: Add Addon Transaction
uri:
  url: "http://localhost:5000/crm/transaction/"
  method: PUT
  body_format: json
  headers:
    Authorization: "Bearer {{ access_token }}"
  body:
    {
      "customer_id": "{{ customer_id }}",
      "service_id": "{{ service_id }}",
```

```

    "title": "5GB Data Boost",
    "description": "Additional 5GB data valid for 7 days",
    "retail_cost": 5.00,
    "wholesale_cost": 1.50
}

```

### Qué Hace Esto:

- Registra un cargo de £5.00 al cliente.
- Registra un costo mayorista de £1.50.
- Vincula la transacción al servicio para informes.

## Resumen Completo del Flujo de Complementos

1. El cliente selecciona el complemento de la lista filtrada.
2. Pago autorizado y cargado.
3. Saldo de datos agregado a la cuenta OCS.
4. Transacción registrada en CRM.
5. El cliente ve inmediatamente el saldo actualizado: 7GB disponibles.

### Seguimiento Financiero:

- Cargo mensual del servicio: £15 minorista, £5 mayorista.
- Compra del complemento: £5 minorista, £1.50 mayorista.

## Renovación Automática: Complementos Recurrentes

Algunos complementos pueden configurarse para renovarse automáticamente (planes de datos mensuales, suscripciones, etc.).

### Configuración del Producto:

```

{
  "product_name": "Monthly 10GB Data Plan",
  "category": "addon",
  "retail_cost": 10.00,
  "contract_days": 30,
  "auto_renew": "true",
  "provisioning_play": "play_topup_charge_then_action"
}

```

### La Provisión Crea un Plan de Acción:

```

- name: Create ActionPlan for Auto-Renewal
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
    body:

```

```

{
  "method": "ApierV1.SetActionPlan",
  "params": [
    {
      "Id": "ServiceID_{{ service_uuid }}_ProductID_{{ product_id }}_MonthlyRenewal",
      "ActionPlan": [
        {
          "ActionsId": "Action_{{ product_slug }}",
          "Years": "*any",
          "Months": "*any",
          "MonthDays": "*any",
          "WeekDays": "*any",
          "Time": "00:00:00",
          "Weight": 10
        },
        {
          "Overwrite": false,
          "ReloadScheduler": true
        }
      ]
    }
  ]
}

```

### Qué Hace Esto:

- Crea una tarea programada en OCS.
- Ejecuta Action\_{{ product\_slug }} cada 30 días.
- La acción carga al cliente y vuelve a aplicar el saldo de datos.
- Continúa hasta que el cliente cancele.

### Gestión del Cliente:

- El cliente ve "Próxima Renovación: 1 de febrero de 2025 - £10.00" en la vista del servicio.
- Puede hacer clic en "Cancelar Renovación Automática" para detener futuros cargos.
- Puede hacer clic en "Renovar Ahora" para aplicar inmediatamente la asignación del próximo mes.

## Etapa 4: Desprovisión de Servicios

Cuando un cliente cancela el servicio, el sistema debe eliminar todos los recursos de manera limpia.

### Disparadores de Desprovisión

La desprovisión puede ser desencadenada por:

1. **Cancelación del cliente** - El cliente hace clic en "Cancelar Servicio".
2. **Acción administrativa** - El personal marca el servicio para desactivación.
3. **No pago** - El servicio expira debido a la falta de renovación.
4. **Fin de contrato** - El contrato a plazo fijo llega a su fecha de finalización.

# Flujo de Desprovisión

## 1. El Cliente Inicia la Cancelación

- Navega al servicio.
- Hace clic en "Cancelar Servicio".
- El sistema pregunta: "¿Está seguro? Cualquier saldo restante se perderá."
- El cliente confirma.

## 2. Período de Gracia (Opcional)

Algunos operadores implementan un período de gracia:

- El servicio se marca como "Cancelación Pendiente".
- Permanece activo durante 7-30 días.
- El cliente puede revertir la cancelación durante el período de gracia.
- Desprovisión automática después del período de gracia.

## 3. Trabajo de Desprovisión Creado

El sistema crea un trabajo de provisión con:

```
{  
  "action": "deprovision",  
  "service_id": 1234,  
  "customer_id": 123,  
  "service_uuid": "Local_Mobile_SIM_a3f2c1d8"  
}
```

Llama al libro de jugadas especificado en `service.deprovisioning_play` o al bloque de rescate del libro de jugadas original.

## 4. Libro de Jugadas de Desprovisión de Ansible

```
- name: Deprovision Mobile Service  
  hosts: localhost  
  tasks:  
    - name: Disable OCS Account  
      uri:  
        url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"  
        method: POST  
        body:  
          {  
            "method": "ApierV2.SetAccount",  
            "params": [{  
              "Account": "{{ service_uuid }}",  
              "ExtraOptions": { "Disabled": true }  
            }]  
          }  
    }
```

```

- name: Remove ActionPlans (stop auto-renewals)
  uri:
    url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
    method: POST
  body:
    {
      "method": "ApierV1.RemoveActionPlan",
      "params": [
        {
          "Id": "ServiceID_{{ service_uuid }}_*"
        }
      ]
    }

- name: Update Service Status in CRM
  uri:
    url: "http://localhost:5000/crm/service/{{ service_id }}"
    method: PATCH
  body:
    {
      "service_status": "Deactivated",
      "service_deactivate_date": "{{ current_datetime }}"
    }

- name: Release Inventory to Stock
  uri:
    url: "http://localhost:5000/crm/inventory/inventory_id/{{ sim_card_id }}"
    method: PATCH
  body:
    {
      "service_id": null,
      "customer_id": null,
      "item_state": "Decommissioned"
    }

```

## Qué Hace Esto:

1. **Desactiva la cuenta de OCS** - Detiene toda la carga, el uso se bloquea.
2. **Elimina los Planes de Acción** - Cancela las renovaciones automáticas.
3. **Actualiza el servicio en CRM** - Estado "Desactivado", fecha registrada.
4. **Libera el inventario** - SIM marcada como "Desmantelada", disponible para reutilización (después de renovación).

## 5. Post-Desprovisión

El sistema realiza limpieza:

- El cliente ya no ve el servicio en el portal de autoservicio.
- El servicio permanece en CRM para informes históricos.

- Las transacciones y facturas se preservan para contabilidad.
- El inventario puede ser renovado y reutilizado.
- La cuenta de OCS puede ser archivada después del período de retención.

## Desprovisión Parcial vs Completa

### Desprovisión Parcial (Suspensión):

- Utilizada para no pago o suspensión temporal.
- Cuenta de OCS desactivada pero no eliminada.
- Saldos preservados.
- Puede reactivarse cuando se reciba el pago.

```
- name: Suspend Service
uri:
  url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body:
    {
      "method": "ApierV2.SetAccount",
      "params": [
        {
          "Account": "{{ service_uuid }}",
          "ExtraOptions": { "Disabled": true }
        }
      ]
    }
```

### Desprovisión Completa (Cancelación Permanente):

- Utilizada para cancelación permanente.
- Cuenta de OCS eliminada por completo.
- Saldos perdidos.
- No puede reactivarse.

```
- name: Remove OCS Account
uri:
  url: "http://{{ crm_config.ocs.cgrates }}/jsonrpc"
  method: POST
  body:
    {
      "method": "ApierV2.RemoveAccount",
      "params": [
        {
          "Account": "{{ service_uuid }}"
        }
      ]
    }
```

# Mejores Prácticas para la Gestión de Productos

## Gestión del Ciclo de Vida del Producto

### Estados del Producto:

- enabled: true - Producto disponible para nuevos pedidos.
- enabled: false - Producto deshabilitado, los servicios existentes continúan.

### Deshabilitar Productos:

- Marque el producto como enabled: false para prevenir nuevos pedidos.
- Los servicios existentes permanecen activos.
- Los clientes aún pueden renovar/modificar servicios existentes.
- Útil para descontinuar productos antiguos.

## Gestión de Inventario

### Estados del Inventario:

- Nuevo - Stock fresco, listo para asignar.
- En Stock - Disponible para provisión.
- Asignado - Vinculado al servicio del cliente.
- Desmantelado - Puede ser renovado y reutilizado.
- Dañado - Necesita reparación o eliminación.

### Reutilización del Inventario:

Después de la desprovisión:

- Tarjetas SIM: Renovar y marcar como "En Stock".
- Números de teléfono: Liberar después del período de portación (30 días).
- Equipos: Probar, renovar, marcar como "Usado".

## Métricas de Provisión

### Monitorear:

- Tasa de éxito de provisión.
- Tiempo promedio de provisión.
- Puntos de fallo comunes.
- Rotación de inventario.



# Integración de Mailjet con OmniCRM

OmniCRM se integra con **Mailjet** para gestionar toda la comunicación por correo electrónico con clientes y personal, asegurando una entrega de correos electrónicos profesional, con marca y confiable tanto para correos electrónicos transaccionales como para campañas de marketing.

## Descripción general

La integración de Mailjet proporciona:

- **Correos Electrónicos Transaccionales Automatizados** - Restablecimientos de contraseña, facturas, correos electrónicos de bienvenida, notificaciones
- **Sincronización de Contactos** - Contactos de clientes sincronizados automáticamente a Mailjet para campañas
- **Plantillas de Correo Electrónico** - Más de 10 tipos de correos electrónicos preconfigurados con plantillas personalizables de Mailjet
- **Campañas de Marketing** - Campañas de correo electrónico segmentadas basadas en datos de clientes
- **Entrega Confiable** - Infraestructura de correo electrónico profesional con seguimiento y análisis

## Configuración

Mailjet se configura en OmniCRM-API/crm\_config.yaml bajo la sección `mailjet`.

### Configuración Básica

```
mailjet:  
  api_key: your_mailjet_api_key  
  api_secret: your_mailjet_api_secret
```

### Obtención de Credenciales de API:

1. Crea una cuenta en <<https://www.mailjet.com>>
2. Navega a **Configuración de Cuenta → Claves API**
3. Copia **Clave API** y **Clave Secreta**
4. Pega en `crm_config.yaml`

# Configuración de Plantillas de Correo Electrónico

OmniCRM utiliza **10 tipos distintos de plantillas de correo electrónico** para comunicaciones automatizadas. Cada plantilla se configura con:

- **from\_email** - Dirección de correo electrónico del remitente
- **from\_name** - Nombre de visualización del remitente
- **template\_id** - ID de plantilla de Mailjet (numérico)
- **subject** - Línea de asunto del correo electrónico

## Tipos de Plantillas y Configuración

### Correo Electrónico de Bienvenida al Cliente

Enviado cuando se crea una nueva cuenta de cliente.

```
api_crmCommunicationCustomerWelcome:  
  from_email: "support@yourcompany.com"  
  from_name: "Soporte de Tu Empresa"  
  template_id: 5977509  
  subject: "Bienvenido a Tu Empresa"
```

#### Cuándo se Envía:

- Nuevo cliente se registra a través del portal de Autoayuda
- El personal crea una nueva cuenta de cliente
- El cliente activa el servicio por primera vez

#### Variables de Plantilla Disponibles:

- {{ var:customer\_name }} - Nombre completo del cliente
- {{ var:email }} - Dirección de correo electrónico del cliente
- {{ var:company\_name }} - Nombre de tu empresa
- {{ var:login\_url }} - Enlace al portal de Autoayuda
- {{ var:support\_url }} - Enlace a la página de soporte

### Correo Electrónico de Factura al Cliente

Enviado cuando se genera una factura y está lista para el pago.

```
api_crmCommunicationCustomerInvoice:  
  from_email: "billing@yourcompany.com"  
  from_name: "Facturación de Tu Empresa"  
  template_id: 6759851  
  subject: "Tu Factura - "
```

#### Cuándo se Envía:

- Factura generada automáticamente para el período de facturación
- Factura manual creada por el personal
- El cliente solicita una copia de la factura

### **Variables de Plantilla Disponibles:**

- {{ var:customer\_name }} - Nombre completo del cliente
- {{ var:invoice\_number }} - ID/número de factura
- {{ var:invoice\_date }} - Fecha de emisión de la factura
- {{ var:due\_date }} - Fecha de vencimiento del pago
- {{ var:total\_amount }} - Monto total a pagar
- {{ var:invoice\_url }} - Enlace para ver/descargar el PDF de la factura
- {{ var:pay\_url }} - Enlace para pagar la factura en línea

### **Adjunto de Factura:**

El PDF de la factura se adjunta automáticamente al correo electrónico.

### **Recordatorio de Factura al Cliente**

Enviado para recordar a los clientes sobre facturas vencidas.

```
api_crmCommunicationCustomerInvoiceReminder:
  from_email: "billing@yourcompany.com"
  from_name: "Facturación de Tu Empresa"
  template_id: 6759852
  subject: "Recordatorio de Pago - Factura Vencida"
```

### **Cuándo se Envía:**

- La factura está X días vencida (configurable)
- Recordatorio manual activado por el personal
- Flujo de trabajo de recordatorio automatizado (si está configurado)

### **Variables de Plantilla Disponibles:**

- {{ var:customer\_name }}
- {{ var:invoice\_number }}
- {{ var:due\_date }}
- {{ var:days\_overdue }}
- {{ var:total\_amount }}
- {{ var:pay\_url }}

### **Correo Electrónico de Bienvenida al Usuario del Personal**

Enviado cuando se crea una nueva cuenta de usuario del personal.

```
api_crmCommunicationUserWelcome:
```

```
from_email: "admin@yourcompany.com"
from_name: "Admin de Tu Empresa"
template_id: 5977510
subject: "Bienvenido al Equipo"
```

### **Cuándo se Envía:**

- El administrador crea un nuevo usuario del personal
- Se hace clic en el botón "Enviar Correo Electrónico de Bienvenida" en la gestión de usuarios

### **Variables de Plantilla Disponibles:**

- {{ var:user\_name }} - Nombre completo del usuario del personal
- {{ var:email }} - Correo electrónico del usuario del personal
- {{ var:role }} - Rol(es) asignado(s)
- {{ var:login\_url }} - Enlace para iniciar sesión en el portal de administración
- {{ var:temp\_password }} - Contraseña temporal (si corresponde)
- {{ var:support\_email }} - Contacto de soporte técnico

## **Restablecimiento de Contraseña del Usuario**

Enviado cuando un usuario solicita restablecer su contraseña.

```
api_crmCommunicationUserPasswordReset:
from_email: "noreply@yourcompany.com"
from_name: "Seguridad de Tu Empresa"
template_id: 5977511
subject: "Solicitud de Restablecimiento de Contraseña"
```

### **Cuándo se Envía:**

- El usuario hace clic en "Olvidé mi Contraseña" en la página de inicio de sesión
- El usuario envía una solicitud de restablecimiento de contraseña

### **Variables de Plantilla Disponibles:**

- {{ var:user\_name }}
- {{ var:reset\_url }} - Enlace de restablecimiento de contraseña con tiempo limitado (típicamente 1 hora)
- {{ var:expiry\_time }} - Cuándo expira el enlace de restablecimiento

### **Nota de Seguridad:**

Los enlaces de restablecimiento expiran después del período de tiempo configurado (por defecto 1 hora).

## Éxito en el Restablecimiento de Contraseña del Usuario

Enviado para confirmar que la contraseña se ha cambiado con éxito.

```
api.crmCommunicationUserPasswordResetSuccess:  
  from_email: "noreply@yourcompany.com"  
  from_name: "Seguridad de Tu Empresa"  
  template_id: 5977512  
  subject: "Contraseña Cambiada con Éxito"
```

### Cuándo se Envía:

- El usuario completa con éxito el restablecimiento de la contraseña
- Inmediatamente después de que se establece la nueva contraseña

### Variables de Plantilla Disponibles:

- {{ var:user\_name }}
- {{ var:change\_date }} - Fecha/hora en que se cambió la contraseña
- {{ var:ip\_address }} - Dirección IP del cambio (opcional)
- {{ var:support\_email }} - Contacto si el cambio fue no autorizado

## Cambio de Contraseña del Usuario

Enviado cuando un usuario cambia su contraseña desde la configuración.

```
api.crmCommunicationUserPasswordChange:  
  from_email: "noreply@yourcompany.com"  
  from_name: "Seguridad de Tu Empresa"  
  template_id: 5977513  
  subject: "Notificación de Cambio de Contraseña"
```

### Cuándo se Envía:

- El usuario cambia la contraseña desde el perfil/configuración
- El administrador restablece la contraseña del usuario

### Variables de Plantilla Disponibles:

- {{ var:user\_name }}
- {{ var:change\_date }}
- {{ var:changed\_by }} - "Yo mismo" o nombre del administrador
- {{ var:support\_email }}

## Verificación de Correo Electrónico

Enviado para verificar la dirección de correo electrónico de un usuario.

```
api_crmCommunicationEmailVerification:  
  from_email: "noreply@yourcompany.com"  
  from_name: "Tu Empresa"  
  template_id: 5977514  
  subject: "Verifica Tu Dirección de Correo Electrónico"
```

### **Cuándo se Envía:**

- Nueva cuenta creada (cliente o personal)
- El usuario cambia la dirección de correo electrónico
- Se requiere verificación de correo electrónico por razones de seguridad

### **Variables de Plantilla Disponibles:**

- {{ var:user\_name }}
- {{ var:verification\_url }} - Enlace para verificar el correo electrónico
- {{ var:verification\_code }} - Código para ingresar manualmente (alternativa al enlace)

### **Notificación de Balance Expirado**

Enviado cuando el saldo/permiso de servicio de un cliente ha expirado.

```
api_crmCommunicationsBalanceExpired:  
  from_email: "support@yourcompany.com"  
  from_name: "Soporte de Tu Empresa"  
  template_id: 5977515  
  subject: "Tu Saldo de Servicio Ha Expirado"
```

### **Cuándo se Envía:**

- El saldo prepago expira
- La fecha de renovación del permiso mensual ha pasado
- Se alcanzó la fecha de expiración del servicio

### **Variables de Plantilla Disponibles:**

- {{ var:customer\_name }}
- {{ var:service\_name }} - Nombre del servicio expirado
- {{ var:expiry\_date }}
- {{ var:balance\_type }} - "Datos", "Voz", "Monetario", etc.
- {{ var:renewal\_url }} - Enlace para renovar/recargar

### **Alerta de Bajo Saldo**

Enviado cuando el saldo de un cliente cae por debajo del umbral configurado.

```
api_crmCommunicationsBalanceLow:
```

```
from_email: "support@yourcompany.com"
from_name: "Soporte de Tu Empresa"
template_id: 5977516
subject: "Alerta de Bajo Saldo"
```

## Cuándo se Envía:

- El saldo cae por debajo del umbral (por ejemplo, 20% restante)
- Configurado en el plan de servicio o OCS
- Monitoreo en tiempo real activa la alerta

## Variables de Plantilla Disponibles:

- {{ var:customer\_name }}
- {{ var:service\_name }}
- {{ var:current\_balance }}
- {{ var:threshold }}
- {{ var:balance\_type }}
- {{ var:topup\_url }} - Enlace para agregar saldo

# Creando Plantillas de Correo Electrónico en Mailjet

Para cada tipo de correo electrónico, necesitas crear una plantilla correspondiente en Mailjet.

## Paso 1: Crear Plantilla en Mailjet

1. Inicia sesión en el panel de Mailjet
2. Navega a **Plantillas de Correo Electrónico** → **Plantillas Transaccionales**
3. Haz clic en **Crear una Nueva Plantilla**
4. Elige **Codifica tu propia plantilla** (para usuarios avanzados) o **Usa el creador de plantillas**

## Paso 2: Diseñar la Plantilla

Utiliza el creador de arrastrar y soltar de Mailjet o el editor HTML para diseñar tu correo electrónico.

## Elementos Esenciales:

- **Encabezado** - Logo y marca de la empresa
- **Saludo** - Personalizado con {{ var:customer\_name }} o {{ var:user\_name }}
- **Contenido** - Cuerpo del mensaje principal
- **Variables** - Inserta variables de plantilla de la lista anterior

- **Llamada a la Acción** - Botones/enlaces para acciones del usuario
- **Pie de Página** - Enlace para darse de baja, dirección de la empresa, contacto de soporte

### Ejemplo de Plantilla (Restablecimiento de Contraseña):

```

<!DOCTYPE html>
<html>
<head>
  <style>
    body { font-family: Arial, sans-serif; }
    .button { background-color: #4CAF50; color: white; padding:
14px 28px; }
  </style>
</head>
<body>
  

  <h2>Solicitud de Restablecimiento de Contraseña</h2>

  <p>Hola {{ var:user_name }},</p>

  <p>Hemos recibido una solicitud para restablecer tu contraseña.
  Haz clic en el botón de abajo para crear una nueva contraseña:</p>

  <a href="{{ var:reset_url }}" class="button">Restablecer
  Contraseña</a>

  <p>Este enlace expira en {{ var:expiry_time }}.</p>

  <p>Si no solicitaste esto, por favor ignora este correo
  electrónico.</p>

  <hr>
  <p style="font-size: 12px; color: #666;">
    Tu Empresa | support@yourcompany.com<br/>
    123 Calle de Negocios, Ciudad, País
  </p>
</body>
</html>

```

### Paso 3: Obtener el ID de la Plantilla

1. Guarda la plantilla en Mailjet
2. Toma nota del **ID de Plantilla** (numérico, por ejemplo, 5977509)
3. Copia este ID a `crm_config.yaml`

## Paso 4: Probar la Plantilla

1. En Mailjet, utiliza la función **Correo Electrónico de Prueba**
2. Proporciona valores de muestra para todas las variables
3. Envía un correo electrónico de prueba a ti mismo
4. Verifica el formato, los enlaces y la marca

## Paso 5: Configurar en OmniCRM

Agrega la configuración de la plantilla a `crm_config.yaml`:

```
mailjet:  
  api_key: your_api_key  
  api_secret: your_secret  
  
  api_crmCommunicationUserPasswordReset:  
    from_email: "noreply@yourcompany.com"  
    from_name: "Seguridad de Tu Empresa"  
    template_id: 5977511  
    subject: "Solicitud de Restablecimiento de Contraseña"
```

Reinicia la API de OmniCRM para que los cambios surtan efecto:

```
cd OmniCRM-API  
sudo systemctl restart omnicrm-api
```

## Sincronización de Contactos

Todos los contactos de clientes en OmniCRM se sincronizan automáticamente con Mailjet.

### Qué se Sincroniza:

- Nombre del contacto
- Dirección de correo electrónico
- Tipo de contacto (facturación, técnico, etc.)
- Ubicación del cliente
- Campos personalizados

### Frecuencia de Sincronización:

Los contactos se sincronizan en tiempo real cuando:

- Se crea un nuevo cliente
- Se agrega/actualiza un contacto
- Se modifican los detalles del cliente

# Solución de Problemas

## Correo electrónico no enviado

- **Causa:** Credenciales de API inválidas, cuenta de Mailjet suspendida o ID de plantilla incorrecto
- **Solución:**
  - Verifica `api_key` y `api_secret` en `crm_config.yaml`
  - Verifica el estado de la cuenta de Mailjet y la facturación
  - Verifica que el ID de plantilla exista en Mailjet
  - Revisa los registros de API en busca de errores

## Variables de plantilla no sustituidas

- **Causa:** Desajuste en el nombre de la variable o datos faltantes en OmniCRM
- **Solución:**
  - Verifica que los nombres de las variables coincidan exactamente (sensible a mayúsculas)
  - Usa el formato `{{ var:variable_name }}`
  - Verifica que OmniCRM esté pasando los datos de la variable en la llamada de API
  - Prueba con datos de muestra en Mailjet

## PDF de factura no adjunto

- **Causa:** Fallo en la generación del PDF o tamaño de archivo demasiado grande
- **Solución:**
  - Revisa los registros de generación de facturas
  - Verifica que la plantilla de factura se renderice correctamente
  - Asegúrate de que el PDF esté por debajo de 15MB (límite de Mailjet)
  - Prueba la generación del PDF de la factura por separado

## Contactos no sincronizados con Mailjet

- **Causa:** Límite de tasa de API excedido o servicio de sincronización no en ejecución
- **Solución:**
  - Verifica los límites de tasa de API de Mailjet (200 llamadas/minuto)
  - Verifica que el servicio OmniCRM-API esté en ejecución
  - Revisa los registros de sincronización en busca de errores
  - Activa manualmente la sincronización para pruebas

## Documentación Relacionada

- `administration_configuration` - Referencia completa de configuración

de Mailjet

- `payments_invoices` - Generación de facturas y entrega de correos electrónicos
- `authentication_flows` - Correos electrónicos de restablecimiento y verificación de contraseña
- `customer_care` - Correos electrónicos de bienvenida del portal de Autoayuda

## Lectura Adicional

- Documentación de Mailjet: <<https://dev.mailjet.com/>>
- Referencia de API de Mailjet: <<https://dev.mailjet.com/email/reference/>>

# Procesamiento de Pagos con Stripe

OmniCRM se integra con **Stripe** para manejar el procesamiento de pagos electrónicos, asegurando transacciones seguras y eficientes para los clientes.

Vea también: [Payment Methods](#) <payment\_methods>, [Billing Overview](#) <billing\_overview>, [Invoices](#) <payments\_invoices>, [Transactions](#) <payments\_transaction>.

## Gestión de Métodos de Pago

El sistema de Métodos de Pago de OmniCRM permite a los clientes y al personal gestionar de manera segura las tarjetas de pago utilizando la **integración de Stripe**. Los métodos de pago permiten la facturación automática de servicios, pagos únicos y cargos recurrentes sin almacenar datos sensibles de la tarjeta en OmniCRM.

Vea también: [Stripe Integration](#) <integrations\_stripe>, [Billing Overview](#) <billing\_overview>, [Payment Processing](#) <payments\_process>, [Invoices](#) <payments\_invoices>.

## Descripción General

El sistema de métodos de pago proporciona:

- **Almacenamiento Seguro de Tarjetas** - Tarjetas tokenizadas por Stripe, nunca almacenadas en OmniCRM
- **Múltiples Tarjetas** - Los clientes pueden almacenar múltiples métodos de pago
- **Selección Predeterminada** - Designar el método de pago preferido para cargos automáticos
- **Seguimiento de Vencimientos** - Monitorear y actualizar tarjetas que están por vencer
- **Autoservicio** - Los clientes pueden gestionar sus propias tarjetas a través del Self-Care Portal <self\_care\_portal>
- **Gestión del Personal** - El personal de soporte puede agregar/eliminar tarjetas en nombre de los clientes

### Métodos de Pago Soportados:

- Tarjetas de Crédito (Visa, Mastercard, American Express, Discover)
- Tarjetas de Débito
- Tarjetas Prepagadas (si son soportadas por la red de tarjetas)

### No Almacenado en OmniCRM:

Los detalles de la tarjeta son tokenizados por Stripe y almacenados de manera segura. OmniCRM solo almacena:

- Marca de la tarjeta (Visa, Mastercard, etc.)
- Últimos 4 dígitos
- Mes/año de vencimiento
- Nombre/apodo del titular de la tarjeta
- Referencia del token de Stripe

# Accediendo a los Métodos de Pago

## Desde la Página del Cliente:

1. Navegar a **Customers** → **[Select Customer]**
2. Hacer clic en la pestaña **Billing**
3. Desplazarse a la sección **Payment Methods**

O directamente:

## Desde el Dashboard de Tarjetas que Vencen:

Ver todos los clientes con tarjetas que están por vencer:

Esto muestra una lista a nivel de sistema de tarjetas que vencen dentro de los próximos 60 días.

# Lista de Métodos de Pago

La tabla de métodos de pago muestra todas las tarjetas almacenadas para un cliente:

Lista de Métodos de Pago

## Descripciones de Columnas:

- **Apodo** - Nombre amigable para la tarjeta (por ejemplo, "Tarjeta Personal", "Visa de Trabajo")
- **Emisor** - Marca de la tarjeta y últimos 4 dígitos
- **Vencimiento** - Mes/año de vencimiento (formato MM/YYYY)
- **Agregado** - Fecha en que se agregó la tarjeta a la cuenta
- **Predeterminado** - Marca de verificación indica el método de pago predeterminado para cargos automáticos

## Acciones por Tarjeta:

Cada fila tiene un menú de acciones ( : ) con opciones:

- **Establecer como Predeterminado** - Hacer de este el método de pago predeterminado
- **Eliminar** - Eliminar la tarjeta de la cuenta

# Agregando un Método de Pago

Haga clic en **"Add Payment Method"** para abrir el modal de pago de Stripe.

## Paso 1: Ingresar Detalles de la Tarjeta

Aparece el modal de Stripe Elements:

Modal para Agregar Método de Pago

## Campos Requeridos:

- **Información de la Tarjeta** - Número de tarjeta, vencimiento, CVC (validado por Stripe)
- **Nombre del Titular de la Tarjeta** - Nombre en la tarjeta
- **País/Región** - País de facturación

## Campos Opcionales:

- **Apodo de la Tarjeta** - Etiqueta amigable para distinguir entre tarjetas

### Seguridad de Stripe Elements:

- Los detalles de la tarjeta se ingresan directamente en el iframe seguro de Stripe
- OmniCRM nunca ve ni almacena números de tarjeta completos
- Cumplimiento de PCI DSS manejado por Stripe
- La validación en tiempo real previene números de tarjeta inválidos

## Paso 2: Enviar y Tokenizar

Cuando haga clic en "Add Payment Method":

1. **Validación del Lado del Cliente:**
  - Stripe valida el formato del número de tarjeta
  - Verifica que la fecha de vencimiento esté en el futuro
  - Verifica el formato del CVC
2. **Tokenización:**
  - Los detalles de la tarjeta se envían directamente a Stripe (no a OmniCRM)
  - Stripe crea un token seguro (por ejemplo, pm\_1A2B3C4D)
  - El token se devuelve a OmniCRM
3. **Procesamiento del Servidor:**
  - OmniCRM guarda el token en el registro del cliente
  - Almacena los últimos 4 dígitos, la marca, el vencimiento para mostrar
  - Ningún número de tarjeta completo toca los servidores de OmniCRM

## Paso 3: Confirmación

Aparece un mensaje de éxito:

Su Visa terminando en 1234 ha sido agregada a su cuenta.

La nueva tarjeta aparece en la tabla de métodos de pago.

### Selección Automática Predeterminada:

- Si esta es la primera tarjeta del cliente, se establece automáticamente como predeterminada
- Si el cliente ya tiene tarjetas, la nueva tarjeta se agrega como no predeterminada
- El cliente puede cambiar la predeterminada después de agregar

## Estableciendo el Método de Pago Predeterminado

El método de pago predeterminado se utiliza para:

- Cargos automáticos de servicios recurrentes
- Pagos de facturas
- Recargas y recargas
- Transacciones únicas (a menos que se especifique lo contrario)

### Para Cambiar el Predeterminado:

1. Localice la tarjeta que desea establecer como predeterminada en la tabla de métodos de pago
2. Haga clic en el **menú de acciones ( : )** junto a la tarjeta

3. Seleccione "**Set as Default**"

4. Aparece la confirmación

La Visa terminando en 5678 es ahora su método de pago predeterminado.

La marca de verificación se mueve a la tarjeta seleccionada recientemente.

#### **Indicador Visual:**

Las tarjetas predeterminadas muestran:

en la columna Predeterminado, típicamente con una insignia de marca de verificación verde.

## **Eliminando un Método de Pago**

Elimine tarjetas que estén vencidas, perdidas o que ya no se necesiten.

### **Paso 1: Iniciar Eliminación**

1. Encuentre la tarjeta a eliminar en la tabla de métodos de pago
2. Haga clic en el **menú de acciones ( : )**
3. Seleccione "**Delete**"

### **Paso 2: Confirmar Eliminación**

Aparece un modal de confirmación:

¿Está seguro de que desea eliminar este método de pago?

Tarjeta: Visa terminando en 1234 Vencimiento: 12/2026

⚠ Advertencia: Si este es su único método de pago, necesitará agregar uno nuevo para continuar utilizando los servicios que requieren facturación automática.

[Cancelar] [Eliminar Método de Pago]

Haga clic en "**Delete Payment Method**" para confirmar.

### **Paso 3: Eliminación Completa**

Mensaje de éxito:

La tarjeta se elimina de la tabla y se elimina de Stripe.

#### **Restricciones Importantes:**

- **No se puede eliminar el predeterminado si existen otras tarjetas** - Establezca una tarjeta diferente como predeterminada primero
- **Advertencia si se elimina la última tarjeta** - Los servicios que requieren pago pueden ser suspendidos
- **Sin deshacer** - La eliminación es permanente; el cliente debe volver a agregar la tarjeta si es necesario

# Gestión de Tarjetas que Vencen

OmniCRM rastrea las fechas de vencimiento de las tarjetas y proporciona herramientas para actualizar proactivamente las tarjetas que están por vencer.

## Dashboard de Tarjetas que Vencen

Navegue a **Billing** → **Expiring Cards** para ver una lista a nivel de sistema:

Cliente Tarjeta Vencimiento Días Hasta la Acción John Smith Visa **\*\*1234 02/2025 12 días** Actualizar Acme Corp MC5678 03/2025 45 días Actualizar Jane Doe Amex\*\*9012 01/2025 VENCIDA Actualizar

Dashboard de Tarjetas que Vencen

### Filtros:

- **Rango de Vencimiento** - Próximos 30/60/90 días o ya vencidos
- **Tipo de Cliente** - Individual vs Empresa
- **Tipo de Servicio** - Filtrar por servicio que requiere método de pago

### Acciones:

- **Actualizar** - Abre la página de métodos de pago del cliente para agregar una nueva tarjeta
- **Notificar** - Enviar recordatorio por correo electrónico al cliente (si Mailjet está configurado)

## Notificaciones de Vencimiento

Si Mailjet está configurado, se envían correos electrónicos automáticos:

- **60 días antes del vencimiento** - Primer recordatorio
- **30 días antes del vencimiento** - Segundo recordatorio
- **7 días antes del vencimiento** - Advertencia final
- **Al vencimiento** - Aviso de que la tarjeta ha vencido

Los clientes pueden hacer clic en un enlace en el correo electrónico para actualizar su método de pago a través del portal de autoservicio.

### Variables de Plantilla de Correo Electrónico:

Las plantillas de Mailjet reciben:

- Nombre del cliente
- Marca de la tarjeta y últimos 4 dígitos
- Fecha de vencimiento
- Enlace a la página de métodos de pago del Self-Care

Vea [integrations\\_mailjet](#) para la configuración de la plantilla de correo electrónico.

## Actualizando una Tarjeta que Vence

### Flujo de Trabajo Recomendado:

1. El cliente recibe un correo electrónico de notificación de vencimiento
2. El cliente inicia sesión en el portal de autoservicio
3. Navega a **Billing** → **Payment Methods**

4. Hace clic en "**Add Payment Method**"
5. Ingresa los nuevos detalles de la tarjeta (la misma tarjeta con vencimiento actualizado, o tarjeta de reemplazo)
6. Establece la nueva tarjeta como predeterminada
7. Elimina la tarjeta antigua/vencida

### **Flujo de Trabajo del Personal:**

Si el cliente llama al soporte:

1. El personal abre la cuenta del cliente
2. Navega a **Billing → Payment Methods**
3. Agrega una nueva tarjeta en nombre del cliente (el cliente proporciona los detalles por teléfono)
4. Establece la nueva tarjeta como predeterminada
5. Elimina la tarjeta vencida
6. Confirma con el cliente

Advertencia

Nunca pida a los clientes que envíen detalles de la tarjeta por correo electrónico o mensaje de texto. Siempre use:

- Portal de autoservicio seguro para autoservicio
- Teléfono con el personal ingresando detalles directamente en el sistema
- En persona en la ubicación minorista

### **Qué Sucedé Cuando las Tarjetas Vencen**

Cuando una tarjeta de pago alcanza su fecha de vencimiento y no se actualiza:

#### **Efectos Inmediatos:**

1. **Los Pagos Automáticos Fallan**
  - Stripe rechaza transacciones con tarjetas vencidas
  - Las renovaciones de servicios mensuales no se procesan
  - Las recargas automáticas fallan
  - Los pagos automáticos de facturas fallan
2. **Notificaciones al Cliente**
  - El sistema intenta cargar la tarjeta
  - Se envía una notificación de fallo de pago
  - Se envía un correo electrónico "Actualizar Método de Pago" con enlace al portal de autoservicio
3. **Cambios en el Estado del Servicio**
  - **Servicios Postpago** - Pueden continuar temporalmente con saldo pendiente
  - **Servicios Prepagos** - Suspensión del servicio cuando se agota el saldo
  - **Servicios de Renovación Automática** - La renovación falla, el servicio puede expirar

#### **Acciones Posteriores:**

##### **Día 1-3 (Período de Gracia):**

- El servicio continúa normalmente
- El cliente recibe el primer aviso de fallo de pago
- El sistema intenta reintentar (dependiendo de la configuración)

##### **Día 4-7:**

- Segundo intento de pago (si está configurado)
- Se envía un correo electrónico de advertencia
- El servicio al cliente puede contactar al cliente

#### Día 8-14:

- El servicio puede ser suspendido por falta de pago
- El estado suspendido previene el uso pero preserva la cuenta
- El cliente puede restaurar actualizando el método de pago y pagando el saldo pendiente

#### Día 15+:

- El servicio puede ser terminado por falta de pago
- Inventario (tarjetas SIM, equipos) marcado para devolución
- Se envía un aviso final
- La cuenta se refiere a cobranzas (si corresponde)

#### Previniendo la Interrupción del Servicio:

Para evitar la interrupción del servicio:

- Actualice las tarjetas **30 días antes del vencimiento**
- Agregue múltiples métodos de pago para redundancia
- Habilite alertas de fallo de pago
- Monitoree el dashboard de Tarjetas que Vencen semanalmente

#### Restaurando el Servicio Despues del Vencimiento:

Si el servicio se suspende debido a una tarjeta vencida:

1. Agregue un nuevo método de pago válido
2. Establezca como predeterminado
3. Pague el saldo pendiente (si lo hay)
4. Contacte al soporte para reactivar el servicio
5. Servicio restaurado en minutos a horas

## Seguridad del Método de Pago

### Tokenización

OmniCRM utiliza la tokenización de Stripe para garantizar la seguridad:

1. **El cliente ingresa la tarjeta** → Enviada directamente a los servidores de Stripe
2. **Stripe valida y tokeniza** → Crea un token único
3. **Token almacenado en OmniCRM** → Nunca se almacena el número completo de la tarjeta
4. **Procesamiento de pagos** → Token enviado a Stripe, Stripe carga la tarjeta

#### Lo que OmniCRM Almacena:

```
{
  "customer_stripe_id": "pm_1A2B3C4D5E6F",
  "brand": "visa",
  "last4": "1234",
  "exp_month": 12,
  "exp_year": 2026,
  "name": "John Smith",
  "nickname": "Personal Card",
```

```
        "is_default": true
    }
```

## Lo que OmniCRM NO Almacena:

- Número completo de la tarjeta
- Código CVV/CVC
- Datos de la banda magnética
- Números PIN

## Cumplimiento de PCI

Al usar Stripe Elements:

- **Ámbito de PCI reducido** - Los datos de la tarjeta nunca tocan los servidores de OmniCRM
- **Campos alojados por Stripe** - La entrada de la tarjeta ocurre en el iframe de Stripe
- **Sin almacenamiento de tarjetas** - Se utilizan tokens en lugar de datos de tarjeta en bruto
- **Transmisión segura** - Toda la comunicación a través de HTTPS/TLS

Vea `integrations_stripe` para detalles de seguridad de Stripe.

## Flujos de Trabajo Comunes

### Flujo de Trabajo 1: El Cliente Agrega el Primer Método de Pago

**Escenario:** Nuevo cliente que se registra para el servicio

1. El cliente crea una cuenta
2. Selecciona un plan de servicio
3. Se le solicita agregar un método de pago durante el proceso de pago
4. Ingresa los detalles de la tarjeta en el modal de Stripe
5. La tarjeta es tokenizada y guardada
6. Se establece automáticamente como predeterminada
7. Se provisiona el servicio
8. Se procesa el primer cargo

### Flujo de Trabajo 2: El Cliente Actualiza la Tarjeta que Vence

**Escenario:** Tarjeta de crédito a punto de vencer

1. El cliente recibe una notificación por correo electrónico (60 días antes del vencimiento)
2. Inicia sesión en el portal de autoservicio
3. Navega a **Billing → Payment Methods**
4. Revisa la tarjeta actual que vence en 12/2025
5. Hace clic en "**Add Payment Method**"
6. Ingresa la tarjeta de reemplazo con vencimiento 12/2028
7. Establece la nueva tarjeta como predeterminada
8. Elimina la tarjeta antigua
9. Se envía un correo electrónico de confirmación

### Flujo de Trabajo 3: El Personal Ayuda al Cliente por Teléfono

**Escenario:** El cliente llama: "Mi tarjeta fue rechazada"

1. El cliente llama al soporte

2. El personal verifica la identidad (preguntas de seguridad)
3. El personal revisa los métodos de pago: Tarjeta vencida 01/2025
4. El personal: "Su tarjeta ha vencido. ¿Tiene una nueva tarjeta?"
5. El cliente proporciona los nuevos detalles de la tarjeta por teléfono
6. El personal navega a **Customers** → **[Customer]** → **Billing**
7. Hace clic en "**Add Payment Method**"
8. Ingresa los detalles de la tarjeta mientras el cliente los lee
9. Establece la nueva tarjeta como predeterminada
10. Elimina la tarjeta vencida
11. Reintenta el pago fallido
12. Confirma con el cliente: "Pago exitoso, servicio restaurado"

## Flujo de Trabajo 4: Cliente Empresarial con Múltiples Tarjetas

**Escenario:** La empresa quiere diferentes tarjetas para diferentes propósitos

1. El cliente empresarial agrega la tarjeta principal (Visa terminando en 1111)
2. Se establece como predeterminada para cargos mensuales de servicio
3. Agrega tarjeta de respaldo (Mastercard terminando en 2222) para recargas
4. Agrega tarjeta de compras (Amex terminando en 3333) para compras de equipos
5. Al hacer una recarga, selecciona Mastercard manualmente en el proceso de pago
6. La Visa predeterminada aún se utiliza para la facturación automática mensual

## Flujo de Trabajo 5: Gestión de Tarjetas que Vencen (Administrador)

**Escenario:** Gestión proactiva de vencimientos

1. El administrador navega a **Billing** → **Expiring Cards**
2. Filtros: "Próximos 30 días"
3. Ve 15 clientes con tarjetas que están por vencer
4. Selecciona todos → "**Send Reminder Emails**"
5. Mailjet envía correos electrónicos personalizados a cada cliente
6. Los clientes actualizan tarjetas a través del Self-Care
7. El administrador revisa la lista 1 semana después
8. Llama a los clientes restantes que no han actualizado
9. Asiste con actualizaciones de tarjetas por teléfono

## Solución de Problemas

### "Tarjeta rechazada" al agregar método de pago

- **Causa:** Stripe rechazó la tarjeta (fondos insuficientes, prevención de fraude, rechazo del emisor)
- **Solución:**
  - Intente con una tarjeta diferente
  - Contacte al emisor de la tarjeta para autorizar la transacción
  - Asegúrese de que la tarjeta soporte compras en línea
  - Verifique que la dirección de facturación coincida con la tarjeta en el archivo

### "Error al agregar método de pago" (error genérico)

- **Causa:** Error de API de Stripe o problema de red
- **Solución:**
  - Actualice la página y vuelva a intentarlo
  - Verifique la conexión a Internet
  - Verifique que la clave publicable de Stripe sea correcta en .env
  - Verifique la consola del navegador para un mensaje de error específico

- Intente con un navegador diferente (desactive los bloqueadores de anuncios)

## **No se puede eliminar el método de pago (botón deshabilitado)**

- **Causa:** Intentando eliminar la tarjeta predeterminada, o es la única tarjeta
- **Solución:**
  - Establezca una tarjeta diferente como predeterminada primero
  - Si es la única tarjeta, agregue una nueva tarjeta antes de eliminar

## **La tarjeta aparece como vencida pero no en la lista de "Tarjetas que Vencen"**

- **Causa:** La tarjeta venció recientemente, caché no actualizado
- **Solución:**
  - Actualice la página
  - Verifique los filtros en el dashboard de Tarjetas que Vencen
  - Las tarjetas vencidas pueden moverse a una vista diferente

## **La nueva tarjeta no aparece inmediatamente**

- **Causa:** La página no se ha actualizado después de agregar la tarjeta
- **Solución:**
  - La tabla de métodos de pago debería actualizarse automáticamente
  - Si no, actualice manualmente el navegador
  - Verifique si ocurrió un error durante el proceso de adición

## **El modal de Stripe no se carga**

- **Causa:** La biblioteca Stripe.js no se carga, problema con la clave API, o extensión del navegador bloqueando
- **Solución:**
  - Verifique la consola del navegador en busca de errores
  - Desactive los bloqueadores de anuncios y la protección de seguimiento
  - Verifique REACT\_APP\_STRIPE\_PUBLISHABLE\_KEY en .env
  - Asegúrese de que el script de Stripe.js se cargue (verifique la pestaña de Red)
  - Intente en modo incógnito/navegación privada

## **El cliente no recibe notificaciones de vencimiento**

- **Causa:** Mailjet no configurado o falta la plantilla de correo electrónico
- **Solución:**
  - Verifique las credenciales de Mailjet en crm\_config.yaml
  - Verifique que exista una plantilla de correo electrónico para el vencimiento de la tarjeta
  - Confirme que la dirección de correo electrónico del cliente sea válida
  - Verifique los registros de Mailjet para fallos de entrega

# **Mejores Prácticas**

## **Para Clientes:**

- Agregue el método de pago antes de la activación del servicio para evitar retrasos
- Mantenga al menos 2 tarjetas en archivo para redundancia
- Actualice las tarjetas que están por vencer 30+ días antes del vencimiento
- Elimine tarjetas antiguas/vencidas para evitar confusiones
- Utilice apodos descriptivos ("Visa Personal", "Amex de Trabajo")
- Verifique que el método de pago predeterminado sea correcto para la facturación automática

## Para el Personal de Soporte:

- Verifique la identidad del cliente antes de acceder a los métodos de pago
- Nunca pida a los clientes que envíen detalles de la tarjeta por correo electrónico/SMS/chat
- Procese las adiciones de tarjetas inmediatamente durante las llamadas (no posponga)
- Confirme que la nueva tarjeta esté establecida como predeterminada después de agregar
- Elimine tarjetas antiguas solo después de confirmar que la nueva tarjeta funciona
- Pruebe el pago después de actualizar la tarjeta vencida (procese una autorización de £0.01)

## Para Administradores:

- Monitoree el dashboard de Tarjetas que Vencen semanalmente
- Envíe correos electrónicos de recordatorio 60/30/7 días antes del vencimiento
- Mantenga las claves de prueba/vivo de Stripe separadas para desarrollo vs producción
- Asegúrese de que las plantillas de Mailjet estén configuradas para notificaciones de vencimiento
- Revise los informes de pagos fallidos para identificar tarjetas vencidas
- Capacite al personal sobre procedimientos seguros para el manejo de tarjetas

## Mejores Prácticas de Seguridad:

- Solo use claves publicables de Stripe (nunca claves secretas en el frontend)
- Asegúrese de que todas las páginas de pago se carguen a través de HTTPS
- Revise regularmente el panel de Stripe en busca de actividad sospechosa
- Habilite las reglas de prevención de fraude de Stripe
- Requiera CVC para todas las transacciones sin tarjeta presente
- Registre los cambios en los métodos de pago en el registro de actividad

## Documentación Relacionada

- `integrations_stripe` - Configuración e integración de Stripe
- `payments_process` - Procesamiento de pagos con métodos de pago almacenados
- `payments_invoices` - Pago automático de facturas usando la tarjeta predeterminada
- `features_topup_recharge` - Sistema de recarga utilizando métodos de pago
- `basics_payment` - Conceptos generales de pago y facturación
- `customer_care` - Portal de autoservicio para que los clientes gestionen sus propias tarjetas

### Interfaz de Gestión de Métodos de Pago

Stripe se utiliza para gestionar tanto el almacenamiento de tarjetas como las compras, mientras que métricas financieras detalladas están disponibles directamente a través del panel de Stripe.

### Panel de Stripe

## Almacenamiento de Tarjetas y Pagos

En OmniCRM, las tarjetas de los clientes se almacenan como **Tokens de Stripe**. Estos tokens permiten el almacenamiento seguro y el manejo de información de pago sensible sin almacenar los detalles reales de la tarjeta dentro de OmniCRM. Esto proporciona una capa adicional de seguridad, ya que los datos sensibles de la tarjeta son gestionados completamente por Stripe.

Cuando un cliente realiza una compra, OmniCRM utiliza estos tokens de Stripe para procesar pagos sin problemas a través de la plataforma Stripe, asegurando que todas las transacciones sean seguras y cumplan con los estándares de la industria.

Al activar transacciones a través de la API `</concepts_api>` o la interfaz de OmniCRM, el

sistema procesará automáticamente los pagos utilizando los tokens de Stripe almacenados. Esto permite un procesamiento de pagos rápido y seguro, asegurando una experiencia fluida tanto para los operadores como para los clientes.

Panel de Stripe

## Métricas Financieras

Para análisis y reportes financieros detallados, los operadores pueden acceder al **Panel de Stripe**. Este panel proporciona información sobre transacciones, ingresos, tarifas y otras métricas importantes relacionadas con el procesamiento de pagos.

Panel de Stripe

Algunas de las métricas clave disponibles en el Panel de Stripe incluyen:

- **Ingresos y Pagos:** Ver ingresos totales, pagos exitosos y montos reembolsados.
- **Tarifas de Transacción:** Reporte detallado sobre tarifas aplicadas a cada transacción.
- **Disputas y Devoluciones:** Métricas sobre cualquier disputa o devolución, lo que le permite rastrear y gestionar riesgos financieros.

Los operadores pueden usar el panel de Stripe para obtener una visión integral de su rendimiento financiero, facilitando el monitoreo de ventas, la gestión de pagos y el seguimiento de la salud general del negocio.

## Accediendo al Panel de Stripe

Para reportes financieros más detallados, navegue al Panel de Stripe en:

<<https://dashboard.stripe.com/>>

Este panel proporciona información financiera en tiempo real, asegurando total transparencia sobre todas las transacciones procesadas a través de Stripe.



# Registro de cambios

Esto contiene los últimos 50 cambios realizados en la pila de software OmniCRM o en sus dependencias.

Nota: Esto no rastrea cambios en configuraciones individuales de clientes.



# Gestión de Métodos de Pago

El sistema de Métodos de Pago de OmniCRM permite a los clientes y al personal gestionar de forma segura las tarjetas de pago utilizando **integración de Stripe**. Los métodos de pago permiten la facturación automática de servicios, pagos únicos y cargos recurrentes sin almacenar datos sensibles de las tarjetas en OmniCRM.

Véase también: [Stripe Integration <integrations\\_stripe>](#), [Billing Overview <billing\\_overview>](#), [Payment Processing <payments\\_process>](#), [Invoices <payments\\_invoices>](#).

## Descripción General

El sistema de métodos de pago proporciona:

- **Almacenamiento Seguro de Tarjetas** - Tarjetas tokenizadas por Stripe, nunca almacenadas en OmniCRM
- **Múltiples Tarjetas** - Los clientes pueden almacenar múltiples métodos de pago
- **Selección Predeterminada** - Designar el método de pago preferido para cargos automáticos
- **Seguimiento de Vencimientos** - Monitorear y actualizar tarjetas que están por vencer
- **Autoservicio** - Los clientes pueden gestionar sus propias tarjetas a través del Self-Care Portal [<self\\_care\\_portal>](#)
- **Gestión del Personal** - El personal de soporte puede añadir/eliminar tarjetas en nombre de los clientes

### Métodos de Pago Soportados:

- Tarjetas de Crédito (Visa, Mastercard, American Express, Discover)
- Tarjetas de Débito
- Tarjetas Prepagadas (si son soportadas por la red de tarjetas)

### No Almacenado en OmniCRM:

Los detalles de la tarjeta son tokenizados por Stripe y almacenados de forma segura. OmniCRM solo almacena:

- Marca de la tarjeta (Visa, Mastercard, etc.)
- Últimos 4 dígitos
- Mes/año de vencimiento
- Nombre/apodo del titular de la tarjeta
- Referencia del token de Stripe

## Accediendo a Métodos de Pago

### Desde la Página del Cliente:

1. Navegar a **Clients** → **[Seleccionar Cliente]**
2. Hacer clic en la pestaña **Facturación**
3. Desplazarse a la sección **Métodos de Pago**

O directamente:

## Desde el Panel de Tarjetas que Vencen:

Ver todos los clientes con tarjetas que están por vencer:

Esto muestra una lista a nivel de sistema de tarjetas que vencen dentro de los próximos 60 días.

## Lista de Métodos de Pago

La tabla de métodos de pago muestra todas las tarjetas almacenadas para un cliente:

Lista de Métodos de Pago

### Descripciones de Columnas:

- **Apodo** - Nombre amigable para la tarjeta (por ejemplo, "Tarjeta Personal", "Visa de Trabajo")
- **Emisor** - Marca de la tarjeta y últimos 4 dígitos
- **Vencimiento** - Mes/año de expiración (formato MM/YYYY)
- **Añadido** - Fecha en que se añadió la tarjeta a la cuenta
- **Predeterminado** - Marca de verificación indica el método de pago predeterminado para cargos automáticos

### Acciones por Tarjeta:

Cada fila tiene un menú de acciones ( : ) con opciones:

- **Establecer como Predeterminado** - Hacer de este el método de pago predeterminado
- **Eliminar** - Eliminar la tarjeta de la cuenta

## Añadiendo un Método de Pago

Hacer clic en "Añadir Método de Pago" para abrir el modal de pago de Stripe.

### Paso 1: Ingresar Detalles de la Tarjeta

Aparece el modal de Stripe Elements:

Modal de Añadir Método de Pago

### Campos Requeridos:

- **Información de la Tarjeta** - Número de tarjeta, vencimiento, CVC (validado por Stripe)
- **Nombre del Titular de la Tarjeta** - Nombre en la tarjeta
- **País/Región** - País de facturación

### Campos Opcionales:

- **Apodo de la Tarjeta** - Etiqueta amigable para distinguir entre tarjetas

### Seguridad de Stripe Elements:

- Los detalles de la tarjeta se ingresan directamente en el iframe seguro de Stripe
- OmniCRM nunca ve ni almacena números de tarjeta completos
- Cumplimiento de PCI DSS manejado por Stripe
- La validación en tiempo real previene números de tarjeta inválidos

## Paso 2: Enviar y Tokenizar

Cuando haces clic en "Añadir Método de Pago":

1. **Validación del Lado del Cliente:**
  - Stripe valida el formato del número de tarjeta
  - Verifica que la fecha de vencimiento esté en el futuro
  - Verifica el formato del CVC
2. **Tokenización:**
  - Los detalles de la tarjeta se envían directamente a Stripe (no a OmniCRM)
  - Stripe crea un token seguro (por ejemplo, pm\_1A2B3C4D)
  - El token se devuelve a OmniCRM
3. **Procesamiento en el Servidor:**
  - OmniCRM guarda el token en el registro del cliente
  - Almacena los últimos 4 dígitos, la marca y el vencimiento para su visualización
  - Ningún número de tarjeta completo toca los servidores de OmniCRM

## Paso 3: Confirmación

Aparece un mensaje de éxito:

Tu Visa terminada en 1234 ha sido añadida a tu cuenta.

La nueva tarjeta aparece en la tabla de métodos de pago.

### Selección Automática Predeterminada:

- Si esta es la primera tarjeta del cliente, se establece automáticamente como predeterminada
- Si el cliente ya tiene tarjetas, la nueva tarjeta se añade como no predeterminada
- El cliente puede cambiar la predeterminada después de añadir

## Estableciendo el Método de Pago Predeterminado

El método de pago predeterminado se utiliza para:

- Cargos automáticos recurrentes de servicios
- Pagos de facturas
- Recargas y recargas
- Transacciones únicas (a menos que se especifique lo contrario)

### Para Cambiar el Predeterminado:

1. Localiza la tarjeta que deseas establecer como predeterminada en la tabla de métodos de pago
2. Haz clic en el **menú de acciones ( : )** junto a la tarjeta
3. Selecciona "**Establecer como Predeterminado**"
4. Aparece una confirmación

La Visa terminada en 5678 ahora es tu método de pago predeterminado.

La marca de verificación se mueve a la tarjeta recién seleccionada.

### Indicador Visual:

Las tarjetas predeterminadas muestran:

en la columna Predeterminado, típicamente con una insignia de marca de verificación verde.

## Eliminando un Método de Pago

Eliminar tarjetas que están vencidas, perdidas o que ya no se necesitan.

### Paso 1: Iniciar Eliminación

1. Encuentra la tarjeta a eliminar en la tabla de métodos de pago
2. Haz clic en el **menú de acciones ( : )**
3. Selecciona "**Eliminar**"

### Paso 2: Confirmar Eliminación

Aparece un modal de confirmación:

¿Estás seguro de que deseas eliminar este método de pago?

Tarjeta: Visa terminada en 1234 Vencimiento: 12/2026

⚠ Advertencia: Si este es tu único método de pago, necesitarás añadir uno nuevo para continuar utilizando servicios que requieren facturación automática.

[Cancelar] [Eliminar Método de Pago]

Haz clic en "**Eliminar Método de Pago**" para confirmar.

### Paso 3: Eliminación Completa

Mensaje de éxito:

La tarjeta se elimina de la tabla y se elimina de Stripe.

#### Restricciones Importantes:

- **No se puede eliminar el predeterminado si existen otras tarjetas** - Establece una tarjeta diferente como predeterminada primero
- **Advertencia si se elimina la última tarjeta** - Los servicios que requieren pago pueden ser suspendidos
- **Sin deshacer** - La eliminación es permanente; el cliente debe volver a añadir la tarjeta si es necesario

## Gestión de Tarjetas que Vencen

OmniCRM rastrea las fechas de vencimiento de las tarjetas y proporciona herramientas para actualizar proactivamente las tarjetas que están por vencer.

### Panel de Tarjetas que Vencen

Navega a **Facturación → Tarjetas que Vencen** para ver una lista a nivel de sistema:

Cliente Tarjeta Vencimiento Días Hasta la Acción John Smith Visa **\*\*1234 02/2025 12 días Actualizar Acme Corp MC5678 03/2025 45 días Actualizar Jane Doe Amex\*\*9012 01/2025 VENCIDA Actualizar**

## Panel de Tarjetas que Vencen

### Filtros:

- **Rango de Vencimiento** - Próximos 30/60/90 días o ya vencidos
- **Tipo de Cliente** - Individual vs Empresa
- **Tipo de Servicio** - Filtrar por servicio que requiere método de pago

### Acciones:

- **Actualizar** - Abre la página de métodos de pago del cliente para añadir una nueva tarjeta
- **Notificar** - Enviar recordatorio por correo electrónico al cliente (si Mailjet está configurado)

## Notificaciones de Vencimiento

Si Mailjet está configurado, se envían correos electrónicos automáticos:

- **60 días antes del vencimiento** - Primer recordatorio
- **30 días antes del vencimiento** - Segundo recordatorio
- **7 días antes del vencimiento** - Advertencia final
- **Al vencimiento** - Aviso de que la tarjeta ha expirado

Los clientes pueden hacer clic en un enlace en el correo electrónico para actualizar su método de pago a través del portal de autoservicio.

### Variables de Plantilla de Correo Electrónico:

Las plantillas de Mailjet reciben:

- Nombre del cliente
- Marca de la tarjeta y últimos 4 dígitos
- Fecha de vencimiento
- Enlace a la página de métodos de pago del Self-Care

Consulta `integrations_mailjet` para la configuración de la plantilla de correo electrónico.

## Actualizando una Tarjeta que Vence

### Flujo de Trabajo Recomendado:

1. El cliente recibe un correo electrónico de notificación de vencimiento
2. El cliente inicia sesión en el portal de autoservicio
3. Navega a **Facturación → Métodos de Pago**
4. Hace clic en "**Añadir Método de Pago**"
5. Ingresa los nuevos detalles de la tarjeta (la misma tarjeta con vencimiento actualizado, o tarjeta de reemplazo)
6. Establece la nueva tarjeta como predeterminada
7. Elimina la tarjeta antigua/vencida

### Flujo de Trabajo del Personal:

Si el cliente llama al soporte:

1. El personal abre la cuenta del cliente
2. Navega a **Facturación → Métodos de Pago**
3. Añade una nueva tarjeta en nombre del cliente (el cliente proporciona detalles por teléfono)

4. Establece la nueva tarjeta como predeterminada
5. Elimina la tarjeta vencida
6. Confirma con el cliente

#### Advertencia

Nunca pidas a los clientes que envíen detalles de la tarjeta por correo electrónico o mensaje de texto. Siempre utiliza:

- Portal de autoservicio seguro para autoservicio
- Teléfono con el personal ingresando detalles directamente en el sistema
- En persona en la ubicación comercial

## Qué Sucedé Cuando las Tarjetas Vencen

Cuando una tarjeta de pago alcanza su fecha de vencimiento y no se actualiza:

#### Efectos Inmediatos:

1. **Los Pagos Automáticos Fallan**
  - Stripe rechaza transacciones con tarjetas vencidas
  - Las renovaciones de servicios mensuales no se procesan
  - Las recargas automáticas fallan
  - Los pagos automáticos de facturas fallan
2. **Notificaciones al Cliente**
  - El sistema intenta cobrar a la tarjeta
  - Se envía notificación de fallo de pago
  - Se envía un correo electrónico "Actualizar Método de Pago" con un enlace al portal de autoservicio
3. **Cambios en el Estado del Servicio**
  - **Servicios Postpago** - Pueden continuar temporalmente con saldo pendiente
  - **Servicios Prepago** - Suspensión del servicio cuando se agota el saldo
  - **Servicios de Renovación Automática** - La renovación falla, el servicio puede expirar

#### Acciones Posteriores:

##### Día 1-3 (Período de Gracia):

- El servicio continúa normalmente
- El cliente recibe el primer aviso de fallo de pago
- El sistema intenta reintentar (dependiendo de la configuración)

##### Día 4-7:

- Segundo intento de pago (si está configurado)
- Correo electrónico de advertencia enviado
- El servicio al cliente puede contactar al cliente

##### Día 8-14:

- El servicio puede ser suspendido por falta de pago
- El estado suspendido previene el uso pero preserva la cuenta
- El cliente puede restaurar actualizando el método de pago y pagando el saldo pendiente

##### Día 15+:

- El servicio puede ser terminado por falta de pago

- Inventario (tarjetas SIM, equipo) marcado para devolución
- Se envía un aviso final
- La cuenta se refiere a cobranzas (si corresponde)

## Previniendo la Interrupción del Servicio:

Para evitar la interrupción del servicio:

- Actualiza las tarjetas **30 días antes del vencimiento**
- Añade múltiples métodos de pago para redundancia
- Habilita alertas de fallo de pago
- Monitorea el panel de Tarjetas que Vencen semanalmente

## Restaurando el Servicio Después del Vencimiento:

Si el servicio se suspende debido a una tarjeta vencida:

1. Añade un nuevo método de pago válido
2. Establece como predeterminado
3. Paga el saldo pendiente (si lo hay)
4. Contacta al soporte para reactivar el servicio
5. Servicio restaurado en minutos a horas

# Seguridad del Método de Pago

## Tokenización

OmniCRM utiliza la tokenización de Stripe para garantizar la seguridad:

1. **El cliente ingresa la tarjeta** → Se envía directamente a los servidores de Stripe
2. **Stripe valida y tokeniza** → Crea un token único
3. **Token almacenado en OmniCRM** → El número completo de la tarjeta nunca se almacena
4. **Procesamiento de pagos** → Token enviado a Stripe, Stripe cobra a la tarjeta

## Lo que OmniCRM Almacena:

```
{
  "customer_stripe_id": "pm_1A2B3C4D5E6F",
  "brand": "visa",
  "last4": "1234",
  "exp_month": 12,
  "exp_year": 2026,
  "name": "John Smith",
  "nickname": "Personal Card",
  "is_default": true
}
```

## Lo que OmniCRM NO Almacena:

- Número completo de la tarjeta
- Código CVV/CVC
- Datos de la banda magnética
- Números de PIN

## Cumplimiento de PCI

Al utilizar Stripe Elements:

- **Alcance PCI reducido** - Los datos de la tarjeta nunca tocan los servidores de OmniCRM
- **Campos alojados por Stripe** - La entrada de la tarjeta ocurre en el iframe de Stripe
- **Sin almacenamiento de tarjetas** - Se utilizan tokens en lugar de datos de tarjeta en bruto
- **Transmisión segura** - Toda la comunicación a través de HTTPS/TLS

Consulta `integrations_stripe` para detalles de seguridad de Stripe.

## Flujos de Trabajo Comunes

### Flujo de Trabajo 1: El Cliente Añade su Primer Método de Pago

**Escenario:** Nuevo cliente que se inscribe para un servicio

1. El cliente crea una cuenta
2. Selecciona un plan de servicio
3. Se le solicita añadir un método de pago durante la compra
4. Ingresa los detalles de la tarjeta en el modal de Stripe
5. La tarjeta es tokenizada y guardada
6. Se establece automáticamente como predeterminada
7. Servicio provisionado
8. Primer cargo procesado

### Flujo de Trabajo 2: El Cliente Actualiza una Tarjeta que Vence

**Escenario:** Tarjeta de crédito a punto de vencer

1. El cliente recibe una notificación por correo electrónico (60 días antes del vencimiento)
2. Inicia sesión en el portal de autoservicio
3. Navega a **Facturación → Métodos de Pago**
4. Revisa la tarjeta actual que vence en 12/2025
5. Hace clic en "**Añadir Método de Pago**"
6. Ingresa la tarjeta de reemplazo con vencimiento 12/2028
7. Establece la nueva tarjeta como predeterminada
8. Elimina la tarjeta antigua
9. Se envía un correo electrónico de confirmación

### Flujo de Trabajo 3: El Personal Ayuda al Cliente por Teléfono

**Escenario:** El cliente llama: "Mi tarjeta fue rechazada"

1. El cliente llama al soporte
2. El personal verifica la identidad (preguntas de seguridad)
3. El personal revisa los métodos de pago: Tarjeta vencida 01/2025
4. El personal: "Tu tarjeta ha vencido. ¿Tienes una nueva tarjeta?"
5. El cliente proporciona los nuevos detalles de la tarjeta por teléfono
6. El personal navega a **Clientes → [Cliente] → Facturación**
7. Hace clic en "**Añadir Método de Pago**"
8. Ingresa los detalles de la tarjeta mientras el cliente los lee
9. Establece la nueva tarjeta como predeterminada
10. Elimina la tarjeta vencida
11. Reintenta el pago fallido
12. Confirma con el cliente: "Pago exitoso, servicio restaurado"

## Flujo de Trabajo 4: Cliente Empresarial con Múltiples Tarjetas

**Escenario:** La empresa quiere diferentes tarjetas para diferentes propósitos

1. El cliente empresarial añade la tarjeta principal (Visa terminada en 1111)
2. La establece como predeterminada para los cargos mensuales de servicio
3. Añade una tarjeta de respaldo (Mastercard terminada en 2222) para recargas
4. Añade una tarjeta de compra (Amex terminada en 3333) para compras de equipo
5. Al hacer una recarga, selecciona Mastercard manualmente en la compra
6. La Visa predeterminada sigue utilizándose para la facturación automática mensual

## Flujo de Trabajo 5: Gestión de Tarjetas que Vencen (Administrador)

**Escenario:** Gestión proactiva de vencimientos

1. El administrador navega a **Facturación → Tarjetas que Vencen**
2. Filtros: "Próximos 30 días"
3. Ve 15 clientes con tarjetas que vencen
4. Selecciona todos → "**Enviar Correos Electrónicos de Recordatorio**"
5. Mailjet envía correos electrónicos personalizados a cada cliente
6. Los clientes actualizan las tarjetas a través del portal de autoservicio
7. El administrador revisa la lista una semana después
8. Llama a los clientes restantes que no han actualizado
9. Asiste con las actualizaciones de tarjetas por teléfono

## Resolución de Problemas

### "Tarjeta rechazada" al añadir un método de pago

- **Causa:** Stripe rechazó la tarjeta (fondos insuficientes, prevención de fraude, rechazo del emisor)
- **Solución:**
  - Intenta con una tarjeta diferente
  - Contacta al emisor de la tarjeta para autorizar la transacción
  - Asegúrate de que la tarjeta soporte compras en línea
  - Verifica que la dirección de facturación coincida con la tarjeta en archivo

### "Error al añadir método de pago" (error genérico)

- **Causa:** Error de API de Stripe o problema de red
- **Solución:**
  - Actualiza la página e intenta de nuevo
  - Verifica la conexión a internet
  - Verifica que la clave publicable de Stripe sea correcta en .env
  - Revisa la consola del navegador para un mensaje de error específico
  - Intenta con un navegador diferente (desactiva bloqueadores de anuncios)

### No se puede eliminar el método de pago (botón deshabilitado)

- **Causa:** Intentando eliminar la tarjeta predeterminada, o es la única tarjeta
- **Solución:**
  - Establece una tarjeta diferente como predeterminada primero
  - Si es la única tarjeta, añade una nueva tarjeta antes de eliminar

### La tarjeta aparece como vencida pero no en la lista de "Tarjetas que Vencen"

- **Causa:** La tarjeta venció recientemente, caché no actualizado

- **Solución:**
  - Actualiza la página
  - Verifica los filtros en el panel de Tarjetas que Vencen
  - Las tarjetas vencidas pueden moverse a una vista diferente

### **La nueva tarjeta no aparece de inmediato**

- **Causa:** La página no se ha actualizado después de añadir la tarjeta
- **Solución:**
  - La tabla de métodos de pago debería actualizarse automáticamente
  - Si no, actualiza manualmente el navegador
  - Verifica si ocurrió un error durante el proceso de adición

### **El modal de Stripe no se carga**

- **Causa:** La biblioteca Stripe.js no se carga, problema con la clave API, o extensión del navegador bloqueando
- **Solución:**
  - Revisa la consola del navegador para errores
  - Desactiva bloqueadores de anuncios y protección de seguimiento
  - Verifica REACT\_APP\_STRIPE\_PUBLISHABLE\_KEY en .env
  - Asegúrate de que el script de Stripe.js se cargue (verifica la pestaña de Red)
  - Intenta en modo incógnito/navegación privada

### **El cliente no recibe notificaciones de vencimiento**

- **Causa:** Mailjet no está configurado o falta la plantilla de correo electrónico
- **Solución:**
  - Verifica las credenciales de Mailjet en crm\_config.yaml
  - Verifica que exista una plantilla de correo electrónico para el vencimiento de la tarjeta
  - Confirma que la dirección de correo electrónico del cliente sea válida
  - Revisa los registros de Mailjet para fallos de entrega

## **Mejores Prácticas**

### **Para Clientes:**

- Añade un método de pago antes de la activación del servicio para evitar retrasos
- Mantén al menos 2 tarjetas en archivo para redundancia
- Actualiza tarjetas que están por vencer 30+ días antes del vencimiento
- Elimina tarjetas antiguas/vencidas para evitar confusiones
- Usa apodos descriptivos ("Visa Personal", "Amex de Trabajo")
- Verifica que el método de pago predeterminado sea correcto para la facturación automática

### **Para el Personal de Soporte:**

- Verifica la identidad del cliente antes de acceder a los métodos de pago
- Nunca pidas a los clientes que envíen detalles de la tarjeta por correo electrónico/SMS/chat
- Procesa las adiciones de tarjetas inmediatamente durante las llamadas (no lo pospongas)
- Confirma que la nueva tarjeta esté establecida como predeterminada después de añadir
- Elimina tarjetas antiguas solo después de confirmar que la nueva tarjeta funciona
- Prueba el pago después de actualizar una tarjeta vencida (procesa una autorización de £0.01)

### **Para Administradores:**

- Monitorea el panel de Tarjetas que Vencen semanalmente
- Envía correos electrónicos de recordatorio 60/30/7 días antes del vencimiento
- Mantén las claves de prueba/producción de Stripe separadas para desarrollo vs producción
- Asegúrate de que las plantillas de Mailjet estén configuradas para notificaciones de vencimiento
- Revisa los informes de fallos de pago para identificar tarjetas vencidas
- Capacita al personal sobre procedimientos seguros de manejo de tarjetas

### **Mejores Prácticas de Seguridad:**

- Utiliza solo claves publicables de Stripe (nunca claves secretas en el frontend)
- Asegúrate de que todas las páginas de pago se carguen a través de HTTPS
- Revisa regularmente el panel de Stripe en busca de actividad sospechosa
- Habilita las reglas de prevención de fraude de Stripe
- Requiere CVC para todas las transacciones sin tarjeta presente
- Registra los cambios en el método de pago en el registro de actividad

## **Documentación Relacionada**

- [integrations\\_stripe](#) - Configuración e instalación de la integración de Stripe
- [payments\\_process](#) - Procesamiento de pagos con métodos de pago almacenados
- [payments\\_invoices](#) - Pago automático de facturas utilizando la tarjeta predeterminada
- [features\\_topup\\_recharge](#) - Sistema de recarga utilizando métodos de pago
- [basics\\_payment](#) - Conceptos generales de pago y facturación
- [customer\\_care](#) - Portal de autoservicio para que los clientes gestionen sus propias tarjetas

Interfaz de Gestión de Métodos de Pago



# Facturas de Clientes

Transactions </payments\_transaction> se agrupan para formar una factura, que se envía al cliente para su pago.

Las facturas tienen una fecha de inicio y una fecha de finalización, que es el período que cubre la factura, y una fecha de vencimiento, que es la fecha en que la factura debe ser pagada.

Generar una Factura Proforma

Las facturas pueden ser generadas automáticamente por el sistema, por ejemplo, cuando se factura un servicio, se crea una factura por el costo minorista, o pueden ser creadas manualmente, por ejemplo, si un cliente solicita una copia de una factura, o si se factura a un cliente por un cargo único.

Las facturas de clientes están completamente plantilladas con Mailjet <integrations\_mailjet> y pueden ser personalizadas para incluir el logotipo de la empresa, la dirección y los detalles de pago, y pueden ser enviadas al cliente por correo electrónico, o descargadas como un PDF.

Registro de Actividad

## Personalizando Plantillas de Facturas

OmniCRM utiliza plantillas HTML con Jinja2 para generar facturas. Puedes personalizar completamente el diseño de la factura, la marca, los colores y el diseño.

### Ubicación de la Plantilla de Factura

Las plantillas de factura se almacenan en OmniCRM-API/invoice\_templates/

#### Plantillas Predeterminadas:

- norfone\_invoice\_template.html - Plantilla de factura de muestra
- cifi\_invoice\_template.html - Ejemplo de plantilla alternativa

#### Configuración:

La plantilla de factura activa se especifica en OmniCRM-API/crm\_config.yaml:

```
invoice:  
  template_filename: 'norfone_invoice_template.html'
```

### Variables de Plantilla Disponibles

Las plantillas de factura tienen acceso a las siguientes variables de Jinja2:

#### Información de la Factura:

- {{ invoice\_number }} - ID único de la factura (por ejemplo, INV-2025-001234)
- {{ date }} - Fecha de emisión de la factura (formato ISO: 2025-01-10T12:00:00)
- {{ due\_date }} - Fecha de vencimiento del pago (por ejemplo, 2025-02-10)
- {{ start\_date }} - Fecha de inicio del período de facturación

- {{ end\_date }} - Fecha de finalización del período de facturación
- {{ total\_amount }} - Monto total de la factura antes de impuestos (numérico)
- {{ total\_tax }} - Monto total de impuestos calculado a partir de todas las transacciones (numérico)

### Información del Cliente:

- {{ client.name }} - Nombre completo del cliente o nombre de la empresa
- {{ client.address.address\_line\_1 }} - Línea de dirección 1
- {{ client.address.address\_line\_2 }} - Línea de dirección 2
- {{ client.address.city }} - Ciudad
- {{ client.address.state }} - Estado/provincia
- {{ client.address.zip\_code }} - Código postal
- {{ client.address.country }} - País

### Artículos de Línea de Transacción:

Recorre las transacciones usando:

```
{% for sub_transaction in transactions %}
  <tr>
    <td>{{ sub_transaction.transaction_id }}</td>
    <td>{{ sub_transaction.created.split("T")[0] }}</td>
    <td>{{ sub_transaction.title }}</td>
    <td>{{ sub_transaction.description }}</td>
    <td>${{ "%.2f" | format(sub_transaction.retail_cost) }}</td>
  </tr>
{% endfor %}
```

### Campos de Transacción:

- sub\_transaction.transaction\_id - ID de la transacción
- sub\_transaction.created - Fecha/hora de la transacción
- sub\_transaction.title - Título de la transacción
- sub\_transaction.description - Descripción detallada
- sub\_transaction.retail\_cost - Monto del artículo
- sub\_transaction.tax\_percentage - Porcentaje de impuestos aplicado (por ejemplo, 10 para 10%)
- sub\_transaction.tax\_amount - Monto de impuestos calculado en dólares

### Mostrando Impuestos en Plantillas:

```
<td>
  {% if sub_transaction.tax_amount and sub_transaction.tax_amount > 0 %}
    ${{ "%.2f" | format(sub_transaction.tax_amount) }} ({{ sub_transaction.tax_percentage }}%)
  {% else %}
    -
  {% endif %}
</td>
```

## Creando una Plantilla de Factura Personalizada

### Paso 1: Copiar Plantilla Existente

```
cd OmniCRM-API/invoice_templates/
cp norfone_invoice_template.html your_company_invoice_template.html
```

## Paso 2: Personalizar HTML/CSS

Edita `your_company_invoice_template.html` para que coincida con tu marca:

### Áreas Clave de Personalización:

#### 1. Logotipo y Marca de la Empresa

```
<!-- Reemplaza con la URL de tu logotipo -->


<!-- Actualiza el nombre de la empresa -->
<h1>Nombre de Tu Empresa</h1>
```

#### 2. Esquema de Color

```
<style>
    /* Color de marca principal */
    .navbar {
        background: linear-gradient(to bottom right, #your-color-1, #your-color-2);
    }

    /* Encabezados de tabla */
    .table thead th {
        background-color: #your-brand-color !important;
        color: white !important;
    }

    /* Botones y enlaces */
    .btn-primary {
        background-color: #your-brand-color;
    }
</style>
```

#### 3. Información de la Empresa en el Pie de Página

```
<footer>
    <p>Nombre de Tu Empresa</p>
    <p>Calle 123, Ciudad, País</p>
    <p>Teléfono: +1-555-123-4567 | Correo: billing@yourcompany.com</p>
    <p>ABN/ID Fiscal: 12345678900</p>
</footer>
```

#### 4. Instrucciones de Pago

```
<div class="payment-info">
    <h3>Métodos de Pago</h3>
    <p><strong>En Línea:</strong> Paga en https://yourcompany.com/pay</p>
    <p><strong>Transferencia Bancaria:</strong></p>
    <ul>
        <li>Nombre de la Cuenta: Your Company Ltd</li>
        <li>BSB: 123-456</li>
        <li>Número de Cuenta: 987654321</li>
        <li>Referencia: {{ invoice_number }}</li>
    </ul>
</div>
```

## 5. Términos y Condiciones

```
<div class="terms">
    <h4>Términos de Pago</h4>
    <p>Pago debido dentro de los 30 días a partir de la fecha de la
factura.</p>
    <p>Recargos por pago tardío: 2% por mes sobre saldos vencidos.</p>
    <p>Para consultas de facturación: billing@yourcompany.com</p>
</div>
```

### Paso 3: Actualizar Configuración

Edita OmniCRM-API/crm\_config.yaml:

```
invoice:
  template_filename: 'your_company_invoice_template.html'
```

### Paso 4: Reiniciar API

```
cd OmniCRM-API
sudo systemctl restart omnicrm-api
```

### Paso 5: Probar la Generación de Facturas

1. Navega a un cliente con transacciones
2. Genera una factura de prueba
3. Descarga el PDF para verificar el formato
4. Envía la factura por correo electrónico a ti mismo para probar la entrega del correo

## Personalización Avanzada

### Contenido Condicional:

Usa condicionales de Jinja2 para mostrar/ocultar contenido:

```
{% if total_amount > 1000 %}
  <div class="high-value-notice">
    <p><strong>Nota:</strong> Saldo alto - Plan de pago disponible a
solicitud.</p>
  </div>
{% endif %}

{% if client.address.country == "Australia" %}
  <p>GST Incluido: ${{ "%.2f" | format(total_amount * 0.10) }}</p>
{% endif %}
```

### Soporte Multilingüe:

Crea plantillas específicas para cada idioma:

```
invoice_template_en.html
invoice_template_es.html
invoice_template_fr.html
```

Configura según la preferencia de idioma del cliente.

### Cálculos Personalizados:

```

<!-- Mostrar subtotal y desglose de impuestos -->
<tr>
  <td colspan="4" class="text-right"><strong>Subtotal:</strong></td>
  <td>${{ "%.2f" | format(total_amount) }}</td>
</tr>
<tr>
  <td colspan="4" class="text-right"><strong>Impuesto:</strong></td>
  <td>${{ "%.2f" | format(total_tax) }}</td>
</tr>
<tr>
  <td colspan="4" class="text-right"><strong>Total:</strong></td>
  <td>${{ "%.2f" | format(total_amount + total_tax) }}</td>
</tr>

```

**Nota:** La variable `total_tax` se calcula automáticamente sumando el `tax_amount` de todas las transacciones en la factura. El impuesto de cada transacción se calcula en función de su campo `tax_percentage`, que por defecto es el `tax_percentage` del producto o 0% si no se especifica.

### Código QR para Pago:

Genera un código QR para el pago móvil:

```

<div class="qr-payment">
  
  <p>Escanea con tu teléfono para pagar al instante</p>
</div>

```

## Mejores Prácticas de Estilo para PDF

OmniCRM utiliza **WeasyPrint** para convertir HTML a PDF. Sigue estas directrices:

### CSS Soportado:

- La mayoría de las propiedades CSS 2.1
- CSS3 limitado (flexbox, algunas transformaciones)
- Fuentes web a través de `@font-face`

### No Soportado:

- JavaScript
- CSS Grid (usa tablas en su lugar)
- Animaciones complejas
- Algunas propiedades CSS modernas

## Tamaño de Página y Márgenes:

```

@page {
  size: A4;
  margin: 1cm;
}

body {
  font-family: Arial, sans-serif;
  font-size: 10pt;
}

```

## Estilo Específico para Impresión:

```
@media print {  
  .no-print {  
    display: none;  
  }  
  
  .page-break {  
    page-break-after: always;  
  }  
}
```

## Diseño de Tabla:

```
.table {  
  table-layout: fixed;  
  width: 100%;  
}  
  
.table th, .table td {  
  word-wrap: break-word;  
  padding: 4px;  
}
```

## Incorporación de Fuentes:

Para fuentes personalizadas, usa fuentes seguras para la web o incrusta:

```
@font-face {  
  font-family: 'YourFont';  
  src: url('https://yourcompany.com/fonts/yourfont.woff2') format('woff2');  
}  
  
body {  
  font-family: 'YourFont', Arial, sans-serif;  
}
```

## Pruebas de Plantillas de Facturas

### Lista de Verificación de Pruebas:

1. **Inspección Visual:**
  - El logotipo se muestra correctamente
  - Los colores coinciden con las pautas de la marca
  - El texto es legible (no demasiado pequeño)
  - Las tablas están alineadas correctamente
  - Todas las secciones están presentes
2. **Precisión de Datos:**
  - Detalles del cliente correctos
  - Los montos de las transacciones suman correctamente
  - Fechas formateadas correctamente
  - Todas las variables se sustituyen correctamente
3. **Calidad del PDF:**
  - Tamaño de archivo razonable (<5MB)
  - Imágenes nítidas y claras
  - Sin corte de texto ni desbordamiento
  - Saltos de página en lugares apropiados
4. **Facturas de Múltiples Páginas:**

- Los encabezados se repiten en cada página
- Los números de página se muestran
- Las listas de transacciones largas se paginan correctamente

##### 5. Entrega de Correo Electrónico:

- El PDF se adjunta al correo electrónico
- Tamaño de archivo por debajo del límite de Mailjet (15MB)
- Se renderiza en Gmail, Outlook, Apple Mail

#### Comando de Prueba (Generación Manual):

Puedes probar la generación de facturas a través de la API:

```
curl -X GET "http://localhost:5000/crm/invoice/{invoice_id}/pdf" \
-H "Authorization: Bearer YOUR_TOKEN" \
--output test_invoice.pdf
```

## Problemas Comunes de Plantillas

#### Variables no sustituidas:

- **Causa:** Error tipográfico en el nombre de la variable o datos faltantes
- **Solución:** Verifica la ortografía exactamente (sensible a mayúsculas y minúsculas), verifica que los datos existan en la base de datos

#### Estilo del PDF roto:

- **Causa:** Propiedad CSS no soportada
- **Solución:** Usa propiedades CSS 2.1, prueba con CSS compatible con WeasyPrint

#### Imágenes no se muestran:

- **Causa:** URLs relativas o recursos externos bloqueados
- **Solución:** Usa URLs HTTPS absolutas, asegúrate de que las imágenes sean accesibles públicamente

#### Tablas desbordando la página:

- **Causa:** Anchos de columna fijos demasiado anchos
- **Solución:** Usa anchos porcentuales, table-layout: fixed

#### Fuentes no se renderizan:

- **Causa:** Fuente no incrustada o no disponible
- **Solución:** Usa fuentes seguras para la web (Arial, Times New Roman, etc.) o incrusta correctamente fuentes personalizadas

#### La generación de PDF falla:

- **Causa:** Errores de sintaxis HTML o fallo de WeasyPrint
- **Solución:** Valida el HTML, verifica los registros de WeasyPrint, simplifica diseños complejos

## Caché de PDF de Facturas

Para mejorar el rendimiento y reducir la generación redundante de PDF, OmniCRM incluye un sistema de caché de PDF de facturas. Cuando se genera por primera vez un PDF de factura, se almacena en la base de datos para solicitudes posteriores.

## Cómo Funciona la Caché de PDF:

- 1. Primera Solicitud** - Cuando se solicita un PDF de factura (descarga o correo electrónico), el sistema:
  - Genera el PDF a partir de la plantilla de factura
  - Codifica el PDF como Base64
  - Calcula un hash SHA256 del contenido del PDF
  - Almacena en la tabla `Invoice_PDF_Cache` con:
    - Referencia de ID de factura
    - Datos del PDF (codificados en Base64)
    - Nombre del archivo
    - Hash de contenido (para verificación de integridad)
    - Marca de tiempo de creación
- 2. Solicitudes Posteriores** - Cuando se solicita la misma factura nuevamente:
  - El sistema verifica si hay un PDF en caché por `invoice_id`
  - Si la caché existe y es válida, devuelve el PDF en caché de inmediato
  - Actualiza la marca de tiempo `last_accessed` para rastrear el uso de la caché
- 3. Invalidación de Caché** - Los PDFs en caché se invalidan cuando:
  - La factura se modifica (transacciones añadidas/eliminadas, detalles cambiados)
  - La plantilla de factura se actualiza
  - Se activa la limpieza manual de caché

## Beneficios:

- Rendimiento** - Entrega instantánea de PDF para solicitudes repetidas (sin retraso de regeneración)
- Consistencia** - El mismo PDF para todas las descargas de una factura (a menos que la factura se modifique)
- Carga del Servidor** - Reduce el uso de CPU por la generación de PDF
- Experiencia del Usuario** - Un indicador de carga aparece durante la generación inicial, las solicitudes posteriores son instantáneas

## Gestión de Caché:

La Caché de PDF de Factura es gestionada automáticamente por el sistema. Las entradas de caché antiguas o no utilizadas pueden ser eliminadas periódicamente en función de:

- Edad (por ejemplo, eliminar entradas de caché de más de 90 días)
- Patrones de acceso (eliminar entradas no accedidas en 30 días)
- Límites de almacenamiento (implementar límites de tamaño de caché si es necesario)

## Comportamiento de la API:

Al descargar una factura a través de la API o la interfaz de usuario:

- Primera solicitud: Muestra un indicador de carga mientras se genera el PDF, luego se almacena en caché
- Solicitudes posteriores: Descarga inmediata desde la caché
- El acierto/fallo de caché es transparente para el usuario

**Importante:** Cuando actualices tu plantilla de factura, limpia la caché para asegurar que las nuevas facturas utilicen el diseño actualizado:

```
-- Limpiar todos los PDFs de factura en caché (ejecutar en MySQL)
DELETE FROM Invoice_PDF_Cache;
```

O actualiza `crm_config.yaml` para invalidar automáticamente la caché en cambio de plantilla.

# Accediendo a las Facturas

Las facturas se pueden ver a nivel del sistema o por cliente:

## Vista por Cliente:

1. Navega a **Clientes** → **[Seleccionar Cliente]**
2. Haz clic en la pestaña **Facturación**
3. Ver lista de facturas en la tercera tarjeta

## Vista a Nivel del Sistema:

1. Navega a **Facturación** → **Facturas** (desde el menú principal)
2. Ver todas las facturas de todos los clientes

# Widgets de Estadísticas de Facturas

En la parte superior de la página de facturas, cuatro tarjetas de estadísticas muestran resúmenes financieros.

Estadísticas y Lista de Facturas

## Descripciones de Widgets:

- **Total de Facturas** - Suma de todos los costos minoristas de las facturas (de todos los tiempos) y conteo de facturas enviadas
- **Facturas No Pagadas** - Suma de facturas aún no pagadas y conteo de facturas no pagadas
- **Facturas Este Mes** - Suma de facturas creadas este mes calendario con conteo
- **Facturas del Mes Pasado** - Suma de facturas creadas el mes pasado con conteo

## Formato de Valores:

- Valores superiores a 1,000: Se muestran con sufijo "k" (por ejemplo, \$1.5k)
- Valores superiores a 1,000,000: Se muestran con sufijo "M" (por ejemplo, \$2.3M)
- Valores superiores a 1,000,000,000: Se muestran con sufijo "B" (por ejemplo, \$1.1B)

## Indicadores de Tendencia:

- Widgets para "Este Mes" y "Mes Pasado" muestran cambio porcentual
- Flecha verde hacia arriba: Aumento respecto al período anterior
- Flecha roja hacia abajo: Disminución respecto al período anterior
- Flecha gris hacia la derecha: Sin cambio

# Lista de Facturas

La tabla de facturas muestra todas las facturas con las siguientes columnas:

Lista Global de Facturas

## Descripciones de Columnas:

- **ID** - ID único de la factura
- **Título** - Título/descripción de la factura
- **Período** - Período de facturación (fecha de inicio - fecha de finalización) o "N/A" para facturas únicas
- **Fecha de Vencimiento** - Fecha de vencimiento del pago

- **Creado** - Fecha de creación de la factura
- **Monto** - Monto total de la factura (costo minorista)
- **Estado** - Pagado, No Pagado o Reembolsado
- **Acciones** - Acciones disponibles (varía según el estado)

#### Iconos de Acción:

-  **(Descargar)** - Descargar PDF de la factura
-  **(Eliminar)** - Anular factura (solo si no está pagada)
-  **(Pagar)** - Pagar factura en línea (solo si no está pagada)
-  **(Correo)** - Enviar correo electrónico de factura al cliente
-  **(Reembolsar)** - Reembolsar pago de Stripe (solo para facturas pagadas de Stripe)

## Generando una Factura

Haz clic en "**+** Generar Factura Proforma" para crear una nueva factura.

Modal de Generación de Factura con Vista Previa de Transacciones

#### Descripciones de Campos:

- **Buscar Clientes** - Seleccionar cliente (solo se muestra en vista a nivel del sistema, prellenado en vista de cliente)
- **Título** - Título/nombre de la factura (opcional, predeterminado a "Factura para [Período]")
- **Fecha de Inicio** - Inicio del período de facturación (predeterminado a 14 días atrás)
- **Fecha de Finalización** - Fin del período de facturación (predeterminado a hoy)
- **Fecha de Vencimiento** - Fecha límite de pago (predeterminado a hoy)
- **Vista Previa de Transacciones** - Muestra todas las transacciones no facturadas en el rango de fechas con la capacidad de incluir/excluir transacciones específicas

#### Selección de Transacciones:

-  **(Más Verde)** - Haz clic para excluir una transacción de la factura
-  **(X Roja)** - Haz clic para incluir una transacción previamente excluida
- **Seleccionar Todo** - Incluir todas las transacciones mostradas
- **Limpiar Todo** - Excluir todas las transacciones
- Las transacciones excluidas aparecen atenuadas con texto tachado
- Los totales se actualizan en tiempo real a medida que seleccionas/deseleccionas transacciones

#### Qué Sucede:

1. El sistema encuentra todas las transacciones no facturadas para el cliente dentro del rango de fechas
2. Muestra la vista previa de transacciones con la capacidad de incluir/excluir transacciones individuales
3. Muestra el cálculo en tiempo real de subtotal, impuestos y total basado en las transacciones seleccionadas
4. Solo las transacciones seleccionadas (incluidas) se agregan a la factura
5. Genera el PDF de la factura y lo almacena en caché
6. Marca las transacciones seleccionadas como facturadas (`invoice_id` campo poblado)
7. Las transacciones excluidas permanecen no facturadas y disponibles para futuras facturas
8. La factura aparece en la lista con estado "No Pagada"

#### Ejemplos de Casos de Uso:

**Facturación Mensual:** Establece la fecha de inicio al primer día del mes, la fecha de finalización

al último día del mes, la vista previa muestra todas las transacciones no facturadas de ese período. Selecciona todas o excluye manualmente las específicas.

**Factura Específica de Servicio:** Usa el mismo rango de fechas, luego excluye manualmente transacciones no deseadas (por ejemplo, excluye transacciones no móviles para crear una factura solo para móviles).

**Factura Única:** Establece tanto la fecha de inicio como la de finalización en el mismo día, la vista previa muestra solo transacciones de esa fecha. Excluye cualquier cargo no relevante para esta factura específica.

## Visualizando Detalles de la Factura

Haz clic en cualquier fila de factura en la tabla para ver los detalles completos de la factura incluyendo todas las transacciones, totales y acciones disponibles.

Vista de Detalles de la Factura

### Modal de Detalles de la Factura:

- **Información de la Factura** - Muestra el ID de la factura, título, fechas, estado de pago y estado de anulación
- **Lista de Transacciones** - Muestra todas las transacciones incluidas en la factura con:
  - Fecha de transacción
  - Título y descripción
  - Costo minorista
  - Monto de impuestos y porcentaje (formateado como \$10.00 (10%))
  - Las transacciones exentas de impuestos muestran "-" en la columna de Impuestos
- **Resumen de Totales** - Cálculo en tiempo real que muestra:
  - Conteo de transacciones
  - Subtotal (suma de todos los costos minoristas)
  - Impuesto (suma de todos los montos de impuestos)
  - Total de la Factura (subtotal + impuestos)
- **Botones de Acción** - Las mismas acciones disponibles que en la tabla:
  - **Descargar PDF** - Descargar PDF de la factura (siempre disponible)
  - **Enviar Correo** - Enviar factura al cliente (facturas no anuladas)
  - **Pagar Factura** - Procesar pago (solo facturas no pagadas y no anuladas)
  - **Reembolsar** - Reembolsar pago de Stripe (solo facturas pagadas de Stripe)
  - **Eliminar** - Anular factura (solo facturas no pagadas y no anuladas)

## Descargando PDFs de Facturas

Haz clic en el **ícono de descarga (↓)** en la tabla o en el botón "**Descargar PDF**" en el modal de detalles de la factura para descargar una factura como PDF.

### Proceso de Descarga:

1. Haz clic en el ícono de descarga junto a la factura
2. Aparece un spinner de carga durante la generación (solo la primera vez)
3. El navegador solicita guardar el archivo: *Invoice\_01234.pdf*
4. El PDF se abre o se guarda en la carpeta de Descargas

### Comportamiento de Caché de PDF:

- **Primera Descarga** - PDF generado a partir de la plantilla, almacenado en la base de datos (puede tardar 2-3 segundos)
- **Descargas Posteriores** - Descarga instantánea desde la caché

- **Invalidación de Caché** - Caché limpiada si la factura se modifica o la plantilla se actualiza

## Resolviendo Problemas de Descarga:

- **El spinner nunca se detiene** - Verifica la consola del navegador, la API puede estar caída
- **PDF en blanco o dañado** - Verifica la plantilla de factura por errores de sintaxis
- **La descarga falla** - Verifica la configuración del bloqueador de ventanas emergentes, prueba con otro navegador

## Pagando Facturas

Haz clic en el **ícono de pago** (◊) para pagar una factura en línea.

Modal de Pago de Factura

### Proceso de Pago:

1. Haz clic en el ícono de pago en la factura no pagada
2. Se abre el modal de pago mostrando los detalles de la factura
3. Selecciona el método de pago:
  - **Transacción de Stripe** - Cargar tarjeta de crédito guardada (disponible para todos los usuarios)
  - **Efectivo** - Pago en efectivo manual (solo personal)
  - **Reembolso** - Aplicar reembolso como pago (solo personal)
  - **Transacción POS** - Terminal de punto de venta (solo personal)
  - **Transferencia Bancaria** - Transferencia bancaria manual (solo personal)
4. Si se selecciona Stripe:
  - Selecciona tarjeta de los métodos de pago guardados
  - La tarjeta predeterminada se preselecciona
  - Haz clic para seleccionar una tarjeta diferente
5. Si se selecciona otro método:
  - Ingresa el número de referencia (opcional)
6. Haz clic en "**Pagar Factura**" para procesar
7. El sistema procesa el pago:
  - **Stripe** - Carga la tarjeta a través de la API de Stripe
  - **Otros métodos** - Crea una transacción negativa por el monto del pago
8. El estado de la factura cambia a "Pagada"
9. Se muestra una notificación de éxito

### Autocuidado vs Pago del Personal:

:doc:`Portal de Autocuidado <self\_care\_portal>` (Clientes):

- Solo disponible el pago por Stripe
- Debe tener un método de pago guardado
- Se muestra una advertencia si no existen métodos de pago
- Se proporciona un enlace para agregar un método de pago

### Portal del Personal (Administradores):

- Todos los métodos de pago disponibles
- Puede marcar la factura como pagada manualmente (efectivo, POS, transferencia bancaria)
- Puede ingresar números de referencia para seguimiento

### Advertencia de Método de Pago:

Si el cliente no tiene métodos de pago guardados, se muestra una advertencia sugiriendo que

agregue un método de pago antes de poder pagar facturas.

Advertencia de Método de Pago Faltante

## Enviando Facturas por Correo Electrónico

Haz clic en el **ícono de correo** (✉) para enviar la factura al cliente.

### Qué Sucede:

1. Haz clic en el ícono de correo junto a la factura
2. El sistema recupera el PDF de la factura de la caché (o genera si no está en caché)
3. Envía el correo electrónico a través de Mailjet <integrations\_mailjet> usando la plantilla `api_crmCommunicationCustomerInvoice`
4. El correo electrónico incluye:
  - PDF de la factura como adjunto
  - Nombre del cliente
  - Número de factura y fecha de vencimiento
  - Monto total debido
  - Enlace para pagar la factura en línea
  - Enlace para ver/descargar la factura
5. Notificación de éxito: "Correo electrónico de la factura enviado con éxito"

### Destinatarios del Correo Electrónico:

El correo se envía a todos los contactos del cliente con tipo "facturación" o contacto principal si no existe contacto de facturación.

### Variables de Plantilla de Correo Electrónico:

- `{{ var:customer_name }}` - Nombre completo del cliente
- `{{ var:invoice_number }}` - ID de la factura
- `{{ var:invoice_date }}` - Fecha de emisión de la factura
- `{{ var:due_date }}` - Fecha de vencimiento del pago
- `{{ var:total_amount }}` - Monto total debido
- `{{ var:invoice_url }}` - Enlace para ver/descargar PDF
- `{{ var:pay_url }}` - Enlace para pagar la factura en línea

### Resolviendo Problemas de Correo Electrónico:

- **Correo no enviado** - Verifica las credenciales de la API de Mailjet en `crm_config.yaml`
- **Cliente no recibe** - Verifica las direcciones de correo electrónico de contacto del cliente
- **PDF no adjunto** - Verifica que la generación de PDF haya sido exitosa (intenta descargar primero)

## Anulando Facturas

Haz clic en el **ícono de eliminar** (✖) para anular una factura.

### Requisitos:

- La factura debe estar **No Pagada**
- Las facturas pagadas no pueden ser anuladas (deben ser reembolsadas en su lugar)

### Cómo Anular:

1. Localiza la factura no pagada en la lista

2. Haz clic en el ícono de eliminar (⊖)
3. Confirma en el modal:

Modal de Confirmación de Anulación de Factura

#### Qué Sucede:

- La factura se marca como `void = true`
- Todas las transacciones se desvinculan de la factura (`invoice_id` se establece en null)
- Las transacciones se convierten nuevamente en "no facturadas"
- Las transacciones pueden incluirse en una nueva factura
- La factura aparece en la lista con el prefijo "Anulado:" en el título
- Las acciones de la factura están deshabilitadas (sin descarga, pago o correo)
- Puede ser vista filtrando por facturas "Anuladas"

#### Notas Importantes:

- Anular NO es lo mismo que reembolsar
- **Anular** = "Esta factura nunca debió existir" (error de facturación, duplicado)
- **Reembolsar** = "Revertir una factura válida pagada" (devolver dinero al cliente)

## Reembolsando Facturas

Haz clic en el **ícono de reembolso** (⊖) para reembolsar una factura pagada.

#### Requisitos:

- La factura debe estar **Pagada**
- La factura debe haber sido pagada a través de **Stripe**
- La factura debe tener un `payment_reference` válido (ID de intención de pago de Stripe)
- Solo disponible para usuarios del personal (no para Autocuidado)

#### Cómo Reembolsar:

1. Localiza la factura pagada de Stripe
2. Haz clic en el ícono de reembolso (⊖)
3. Se abre el modal de confirmación de reembolso:

Modal de Confirmación de Reembolso de Factura

4. Haz clic en "**Confirmar Reembolso**"
5. El sistema procesa el reembolso de Stripe:
  - Llama a la API de Stripe para reembolsar el pago
  - Crea una transacción de reembolso en Stripe
  - Actualiza la factura con `refund_reference`
6. El estado de la factura cambia a "Reembolsada"
7. Se muestra una notificación de éxito

#### Qué Sucede Después del Reembolso:

- La factura permanece en el sistema (no se anula)
- El estado muestra "Reembolsada"
- Las transacciones permanecen vinculadas a la factura
- El cliente recibe el reembolso en el método de pago original (3-7 días hábiles)
- El panel de Stripe muestra la transacción de reembolso

#### Restricciones de Reembolso:

- No se pueden reembolsar facturas pagadas en efectivo, POS o transferencia bancaria (se requiere reversión manual)
- No se puede reembolsar parcialmente (solo el monto total de la factura)
- No se puede reembolsar dos veces

## Buscando y Filtrando Facturas

### Búsqueda

Usa la barra de búsqueda para encontrar facturas. Busca en:

- ID de factura
- Título de factura
- Nombre del cliente (solo vista a nivel del sistema)

### Filtros

Aplica filtros para reducir la lista de facturas:

#### Filtros Disponibles:

- **Estado de Anulación** - Todos, Anulado, No Anulado
- **Estado de Pago** - Todos, Pagado, No Pagado

#### Acciones de Filtro:

- **Aplicar Filtros** - Aplicar filtros seleccionados a la lista
- **Restablecer Filtros** - Limpiar todos los filtros y mostrar todas las facturas

### Ordenamiento

Haz clic en cualquier encabezado de columna para ordenar:

- **ID** - Ordenar por ID de factura (más nuevo/más antiguo)
- **Título** - Ordenar alfabéticamente
- **Fecha de Vencimiento** - Ordenar por fecha de vencimiento
- **Creado** - Ordenar por fecha de creación
- **Monto** - Ordenar por costo minorista (más alto/más bajo)
- **Estado** - Ordenar por estado de pago (pagado primero o no pagado primero)

Haz clic nuevamente para invertir la dirección de ordenamiento (ascendente ↔ descendente).

### Paginación

Navega a través de listas grandes de facturas con controles de página que muestran la página actual, el total de páginas y un selector de elementos por página (10, 25, 50 o 100 elementos).

## Flujos de Trabajo Comunes de Facturas

### Flujo de Trabajo 1: Facturación Mensual con Vista Previa de Transacciones

1. Llega el final del mes (por ejemplo, 31 de enero)
2. Navega a **Facturación → Facturas**
3. Haz clic en "**+ Generar Factura Proforma**"
4. Selecciona cliente (o hazlo por cliente si facturas a muchos clientes)
5. Establece fechas:

- Fecha de Inicio: 2025-01-01
  - Fecha de Finalización: 2025-01-31
  - Fecha de Vencimiento: 2025-02-15 (15 días a partir de ahora)
  - Título: "Servicios de Enero 2025" (opcional)
6. La sección de **Vista Previa de Transacciones** aparece mostrando todas las transacciones no facturadas de enero
  7. Revisa la vista previa:
    - Todas las transacciones están incluidas por defecto
    - Verifica totales: Subtotal, Impuestos y Total de la Factura
    - Verifica que todos los cargos sean correctos
  8. Haz clic en "**Generar Factura**" (el botón muestra el conteo de transacciones, por ejemplo, "Generar Factura (15)")
  9. Factura creada con todas las transacciones seleccionadas
  10. Haz clic en la fila de la factura para ver detalles y verificar
  11. Haz clic en el botón "**Enviar Correo**" en el modal de detalles o en el ícono de correo en la tabla
  12. El cliente recibe un correo electrónico de la factura con PDF y enlace de pago

## Flujo de Trabajo 2: Facturación Selectiva de Transacciones

1. El cliente tiene múltiples servicios (Móvil + Internet) y cargos misceláneos
2. Quiere facturas separadas para cada servicio
3. **Generar primera factura (Servicios Móviles):**
  - Haz clic en "**+ Generar Factura Proforma**"
  - Título: "Servicios Móviles - Enero 2025"
  - Inicio/Fin: 1-31 de enero
  - Fecha de Vencimiento: 15 de febrero
  - En la vista previa de transacciones, **excluir** todas las transacciones no móviles:
    - Haz clic en el botón **X** junto a las transacciones de Internet
    - Haz clic en el botón **X** junto a los cargos misceláneos
    - Solo permanecen seleccionadas las transacciones de servicio móvil
  - Verifica que los totales reflejen solo los servicios móviles
  - Haz clic en "**Generar Factura**" (muestra el conteo de transacciones móviles)
4. **Generar segunda factura (Servicios de Internet):**
  - Haz clic en "**+ Generar Factura Proforma**" nuevamente
  - Título: "Servicios de Internet - Enero 2025"
  - Inicio/Fin: 1-31 de enero (mismo período)
  - En la vista previa de transacciones:
    - Las transacciones móviles ya facturadas (no aparecen)
    - Excluye los cargos misceláneos usando el botón **X**
    - Solo permanecen las transacciones de servicio de Internet
  - Haz clic en "**Generar Factura**"
5. **Generar tercera factura (Cargos Adicionales):**
  - Haz clic en "**+ Generar Factura Proforma**" nuevamente
  - Título: "Cargos Adicionales - Enero 2025"
  - Solo aparecen en la vista previa los cargos misceláneos no facturados
  - Haz clic en "**Seleccionar Todo**" para incluir todos
  - Haz clic en "**Generar Factura**"
6. Envía las tres facturas al cliente

## Flujo de Trabajo 3: Excluyendo Transacciones Disputadas o Pendientes

1. Llega el final del período de facturación
2. Navega a la pestaña de **Facturación** del cliente
3. Haz clic en "**+ Generar Factura Proforma**"
4. Establece las fechas del período de facturación
5. La vista previa de transacciones muestra 20 transacciones

6. El cliente ha disputado un cargo y otro está pendiente de investigación
7. En la vista previa de transacciones:
  - Localiza la transacción disputada (por ejemplo, "Cargo por exceso de datos")
  - Haz clic en el botón **X** para excluirla
  - Localiza la transacción pendiente (por ejemplo, "Cargo por instalación")
  - Haz clic en el botón **X** para excluirla
  - El conteo de transacciones se actualiza: "18 Transacciones seleccionadas"
  - Los totales se recalculan automáticamente
8. Revisa los totales actualizados (excluye los montos disputados)
9. Haz clic en "**Generar Factura (18)**"
10. Factura generada con solo las transacciones aprobadas
11. Las transacciones disputadas/pendientes permanecen no facturadas para el próximo ciclo de facturación

## Flujo de Trabajo 4: Revisión Rápida de Factura y Ajuste

1. El personal genera la factura mensual
2. La vista previa de transacciones muestra un total inesperadamente alto
3. Revisa cada transacción en la vista previa:
  - Nota un cargo duplicado por el mismo servicio
  - Haz clic en **X** para excluir el duplicado
  - Nota una transacción de prueba que no debería ser facturada
  - Haz clic en **X** para excluir la transacción de prueba
4. Los totales se actualizan en tiempo real
5. Verifica que el nuevo total coincida con el monto esperado
6. Haz clic en "**Generar Factura**" con las transacciones corregidas
7. Regresa y anula/elimina las transacciones excluidas si es necesario
8. Envía la factura al cliente con confianza

## Flujo de Trabajo 5: Factura de Instalación Única

1. El técnico de campo completa la instalación
2. El personal agrega manualmente la transacción de instalación
3. Navega a la pestaña de **Facturación** del cliente
4. Haz clic en "**+ Generar Factura Proforma**"
5. Establece las fechas:
  - Fecha de Inicio: hoy
  - Fecha de Finalización: hoy
  - Fecha de Vencimiento: hoy + 7 días
  - Título: "Servicios de Instalación"
6. La vista previa de transacciones muestra solo las transacciones de hoy
7. Verifica que el cargo de instalación aparezca
8. Excluye cualquier cargo recurrente usando el botón **X** (si está presente)
9. Haz clic en "**Generar Factura**"
10. Envía por correo al cliente de inmediato
11. El cliente paga en línea a través de Stripe

## Flujo de Trabajo 6: Revisando la Factura Antes del Contacto con el Cliente

1. El cliente llama con una pregunta de facturación
2. El personal navega a la lista de facturas del cliente
3. **Haz clic en la fila de la factura** para abrir el modal de Detalles de la Factura
4. Revisa la información de la factura:
  - ID de factura, fechas, estado
  - Todas las transacciones incluidas con descripciones
  - Desglose de impuestos por transacción
  - Subtotal, Impuestos y Totales

5. Responde a las preguntas del cliente con detalles exactos
6. Si el cliente solicita el PDF, haz clic en el botón "**Descargar PDF**" en el modal
7. Si el cliente solicita reenvío del correo, haz clic en el botón "**Enviar Correo**"
8. Cierra el modal cuando termines

## Flujo de Trabajo 7: Corrigiendo un Error de Facturación

1. El cliente informa un cargo incorrecto
2. El personal hace clic en la fila de la factura para ver los detalles
3. Revisa la lista de transacciones en el modal de Detalles de la Factura
4. Identifica la transacción incorrecta
5. La factura está sin pagar, por lo que puede ser anulada
6. Haz clic en el botón "**Eliminar**" en el pie del modal
7. Confirma la anulación
8. Las transacciones se convierten nuevamente en no facturadas
9. El personal modifica o elimina la transacción incorrecta de la lista de transacciones
10. Genera una nueva factura con las transacciones corregidas:
  - Usa la vista previa de transacciones para excluir la transacción corregida si es necesario
  - Incluye solo los cargos válidos
11. Envía la factura corregida al cliente

## Flujo de Trabajo 8: Procesando Múltiples Pagos

1. El cliente trae efectivo para pagar múltiples facturas
2. Navega a la pestaña de **Facturación** del cliente
3. Ver facturas no pagadas
4. Haz clic en la primera fila de factura para ver detalles
5. Verifica el monto y las transacciones
6. Haz clic en el botón "**Pagar Factura**" en el pie del modal
7. Selecciona el método de pago "**Efectivo**"
8. Ingresa la referencia: "Efectivo pagado 2025-01-15"
9. Haz clic en "**Pagar Factura**"
10. El modal se cierra, la factura se marca como "Pagada"
11. Repite para las facturas restantes
12. Todas las facturas ahora están marcadas como "Pagadas"

## Flujo de Trabajo 9: Manejo de Solicitud de Reembolso

1. El cliente solicita un reembolso por sobrepago
2. El personal verifica que la factura fue pagada a través de Stripe
3. Navega a la factura en la lista
4. Haz clic en la fila de la factura para ver detalles
5. Verifica la información de pago y el monto
6. Haz clic en el botón "**Reembolsar**" en el pie del modal (solo aparece para facturas de Stripe)
7. Confirma el reembolso
8. El sistema procesa el reembolso de Stripe
9. El estado de la factura cambia a "Reembolsada"
10. El cliente recibe el reembolso en 3-7 días hábiles
11. El personal sigue con el cliente para confirmar la recepción

# Resolución de Problemas

## No se puede generar factura - No se encontraron transacciones

- **Causa:** No hay transacciones no facturadas en el rango de fechas especificado

- **Solución:** Verifica la lista de transacciones, verifica que existan transacciones y que no estén ya facturadas. Ajusta el rango de fechas o elimina el filtro.

### La generación de PDF de factura falla

- **Causa:** Error de sintaxis en la plantilla, fallo de WeasyPrint o falta de datos del cliente
- **Solución:** Verifica la plantilla HTML de la factura en busca de errores, verifica que los campos de dirección del cliente estén poblados, revisa los registros de la API.

### El pago falla con error de Stripe

- **Causa:** Tarjeta rechazada, fondos insuficientes, tarjeta expirada o problema con la API de Stripe
- **Solución:** Prueba con un método de pago diferente, verifica que la tarjeta sea válida, revisa el panel de Stripe para conocer la razón del rechazo.

### No se puede anular la factura

- **Causa:** La factura ya está pagada
- **Solución:** Las facturas pagadas no pueden ser anuladas. Si se necesita un reembolso, utiliza la función de reembolso para facturas de Stripe o crea una transacción de crédito manualmente.

### El correo electrónico de la factura no se envía

- **Causa:** Credenciales de API de Mailjet inválidas, el cliente no tiene contacto de facturación o falta la plantilla de correo electrónico
- **Solución:** Verifica la configuración de Mailjet en `crm_config.yaml`, verifica los contactos del cliente, verifica que exista la plantilla de correo electrónico de la factura.

### El botón de reembolso no aparece

- **Causa:** La factura fue pagada en efectivo/POS/transferencia bancaria (no Stripe), o la factura no está pagada
- **Solución:** El botón de reembolso solo aparece para pagos de Stripe. Para otros métodos de pago, crea una transacción de crédito manual.

### La descarga de PDF muestra el diseño antiguo de la plantilla

- **Causa:** PDF almacenado en caché antes de la actualización de la plantilla
- **Solución:** Limpia la caché de PDF de la factura: `DELETE FROM Invoice_PDF_Cache WHERE invoice_id = X;`

### El cliente no puede pagar la factura (sin métodos de pago)

- **Causa:** No hay métodos de pago guardados en el portal de Autocuidado
- **Solución:** El cliente debe agregar una tarjeta de crédito en la página de **Métodos de Pago** antes de pagar facturas.

### Se generaron múltiples facturas para el mismo período

- **Causa:** El personal generó la factura dos veces, o los rangos de fechas se superponen
- **Solución:** Anula la factura duplicada. Ajusta los rangos de fechas para evitar superposiciones. Usa la vista previa de transacciones para asegurar conjuntos de transacciones únicos.

### La vista previa de transacciones no muestra transacciones

- **Causa:** Todas las transacciones en el rango de fechas ya están facturadas o no existen transacciones
- **Solución:** Verifica que el rango de fechas sea correcto. Verifica la lista de transacciones para confirmar que existan transacciones. Filtra las facturas para ver qué factura contiene las transacciones.

### No se puede excluir la transacción de la generación de la factura

- **Causa:** La transacción ya está facturada o problema del navegador
- **Solución:** Verifica que la transacción se muestre en la vista previa con una marca de verificación. Actualiza la página y vuelve a intentarlo. Limpia la caché del navegador si el problema persiste.

### El total de la factura no coincide con el monto esperado

- **Causa:** Se incluyeron transacciones inesperadas, impuestos no calculados o las transacciones excluidas aún se cuentan
- **Solución:** Revisa la vista previa de transacciones cuidadosamente. Verifica cada costo minorista y el impuesto de la transacción. Verifica que las transacciones excluidas estén atenuadas. Verifica el distintivo de conteo de transacciones en el botón Generar Factura.

### El botón Generar Factura está deshabilitado

- **Causa:** No se seleccionó ninguna transacción o rango de fechas inválido
- **Solución:** Asegúrate de que al menos una transacción esté incluida (no excluida). Verifica que la Fecha de Inicio sea anterior a la Fecha de Finalización. Verifica que la Fecha de Vencimiento esté establecida.

### El modal de Detalles de la Factura no se abre

- **Causa:** Error de JavaScript o la página no se ha cargado completamente
- **Solución:** Actualiza la página. Verifica la consola del navegador en busca de errores. Prueba con otro navegador. Verifica la conexión a Internet.

### El impuesto de la transacción no se muestra en los Detalles de la Factura

- **Causa:** La transacción tiene un impuesto del 0% o tax\_amount es nulo
- **Solución:** Verifica que la transacción tenga el tax\_percentage establecido. Verifica que tax\_amount se haya calculado cuando se creó la transacción. Actualiza la transacción si es necesario.

### Faltan botones de acción en el modal de Detalles de la Factura

- **Causa:** La factura está anulada o el usuario no tiene permisos
- **Solución:** Las facturas anuladas solo muestran el botón Descargar PDF. Verifica el estado de la factura. Verifica el rol y los permisos del usuario.

## Documentación Relacionada

- [integrations\\_mailjet](#) - Entrega de correo electrónico de factura y plantillas
- [administration\\_configuration](#) - Configuración de plantilla de factura
- [payments\\_transaction](#) - Creación de transacciones que aparecen en facturas
- [payments\\_process](#) - Procesamiento de pagos de facturas
- [basics\\_payment](#) - Métodos de pago e integración con Stripe



# Procesar Pagos

La mayoría de los pagos serán procesados automáticamente por el sistema, pero hay ocasiones en las que puede que necesite procesar un pago manualmente.

Para pagar una factura, seleccione la factura no pagada y haga clic en el botón "Pagar".

Esto abrirá un formulario de pago, donde puede ingresar el método de pago y hacer clic en "Enviar" para procesar el pago.

El cliente recibirá automáticamente un recibo por el pago y la factura se marcará como pagada.

Para transferencias bancarias, puede ingresar la referencia de pago y la fecha en que se realizó el pago (si es diferente de la fecha actual).

[Pagar una factura](#) [Pagar una factura](#)



# Transacciones de Clientes

Cualquier cosa que cueste dinero en el sistema se registra como una transacción bajo el cliente.

Cada transacción tiene un monto monetario para el costo mayorista y el costo minorista, y una descripción de para qué es la transacción.

Las transacciones pueden ser generadas automáticamente por el sistema, por ejemplo, cuando se provisiona un servicio, se crea una transacción para el costo de configuración, y cuando se factura un servicio, se crea una transacción para el costo minorista.

Las transacciones también pueden ser creadas manualmente, por ejemplo, si se le otorga un crédito a un cliente, se crea una transacción por el monto del crédito, o si se cobra una tarifa de instalación, se crea una transacción por la tarifa de instalación.

Las transacciones se agrupan para formar Facturas `<payments_invoices>`, que se envían al cliente para su pago.

Gestión de Transacciones

## Accediendo a las Transacciones

Las transacciones se pueden ver a nivel del sistema o por cliente:

### Vista por Cliente:

1. Navegar a **Clientes** → **[Seleccionar Cliente]**
2. Hacer clic en la pestaña **Facturación**
3. Ver la lista de transacciones en la primera tarjeta

### Vista a Nivel del Sistema:

1. Navegar a **Facturación** → **Transacciones** (desde el menú principal)
2. Ver todas las transacciones de todos los clientes

## Widgets de Estadísticas de Transacciones

En la parte superior de la página de transacciones, cuatro tarjetas de estadísticas muestran resúmenes financieros:

Estadísticas y Lista de Transacciones

### Descripciones de Widgets:

- **Total de Transacciones** - Suma de todos los costos minoristas de transacciones (todo el tiempo)
- **Total de Transacciones No Facturadas** - Suma de transacciones que aún no se han incluido en una factura
- **Total de Transacciones Este Mes** - Suma de transacciones creadas este mes calendario
- **Total de Transacciones El Mes Pasado** - Suma de transacciones creadas el mes pasado

### Formato de Valores:

- Valores superiores a 1,000: Mostrar como sufijo "k" (por ejemplo, \$1.5k)
- Valores superiores a 1,000,000: Mostrar como sufijo "M" (por ejemplo, \$2.3M)
- Valores superiores a 1,000,000,000: Mostrar como sufijo "B" (por ejemplo, \$1.1B)

## Lista de Transacciones

La tabla de transacciones muestra todas las transacciones con las siguientes columnas:

### Descripciones de Columnas:

- **ID** - ID único de la transacción
- **Fecha** - Fecha de creación de la transacción
- **Título** - Nombre corto de la transacción
- **Descripción** - Descripción detallada de para qué es la transacción
- **Monto** - Costo minorista (positivo para cargos, negativo para créditos)
- **Factura** - ID de la factura si la transacción ha sido facturada (enlace clicable)
- **Estado** - Marca de verificación si está facturada, guion si aún no ha sido facturada

### Acciones Por Fila:

Cada fila tiene un menú de acciones (:) con opciones:

- **Ver Detalles** - Abre el modal de detalles de la transacción
- **Descargar PDF de Factura** - Descargar PDF (solo si está facturada)
- **Anular Transacción** - Marcar la transacción como anulada (solo si no está facturada)

## Tipos de Transacciones

Las transacciones se dividen en dos categorías principales:

### Transacciones de Débito (Cargos)

Montos positivos que aumentan el saldo adeudado por el cliente:

- **Tarifas de Configuración de Servicio** - Cargos únicos cuando se provisiona el servicio
- **Tarifas Mensuales de Servicio** - Cargos recurrentes por servicios
- **Tarifas de Instalación** - Cargos por visitas de técnicos de campo
- **Cargos por Equipos** - Cargos por módems, enruteadores, tarjetas SIM
- **Cargos por Pago Tardío** - Penalizaciones por facturas vencidas
- **Cargos Manuales** - Cargos personalizados añadidos por el personal

### Transacciones de Crédito (Pagos/Reembolsos)

Montos negativos que disminuyen el saldo adeudado por el cliente:

- **Pagos en Efectivo** - Cliente pagó en efectivo
- **Pagos con Tarjeta** - Cliente pagó con tarjeta de crédito/débito
- **Pagos por Transferencia Bancaria** - Cliente pagó mediante transferencia bancaria
- **Créditos de Cuenta** - Créditos de buena voluntad, compensación
- **Reembolsos** - Dinero devuelto al cliente
- **Descuentos** - Descuentos promocionales o de lealtad

## Agregar una Transacción Manualmente

Haga clic en "**+ Agregar Transacción**" para abrir el modal de agregar transacción.

### **Transacción de Débito (Cargo):**

Agregar Modal de Transacción de Débito

### **Transacción de Crédito (Pago/Reembolso):**

Agregar Modal de Transacción de Crédito

### **Descripciones de Campos:**

- **Tipo de Transacción** - Seleccionar Débito (cargo) o Crédito (pago/reembolso)
- **Tipo de Crédito** - Si se selecciona Crédito, elegir método de pago (Efectivo, Tarjeta, Transferencia Bancaria)
- **Título** - Nombre corto para la transacción (requerido)
- **Descripción** - Explicación detallada (opcional)
- **Costo Minorista** - Monto que paga el cliente (requerido, número positivo)
- **Costo Mayorista** - Su costo (opcional, para seguimiento de márgenes)
- **Porcentaje de Impuesto** - Tasa de impuesto aplicada a esta transacción (opcional, por defecto al impuesto del producto o 0%)
- **Servicio** - Vincular la transacción a un servicio específico (opcional)
- **Sitio** - Vincular la transacción a un sitio específico (opcional)
- **Fecha de Transacción** - Fecha de la transacción (por defecto hoy)

### **Validación:**

- El título y el costo minorista son obligatorios
- El costo minorista debe ser un número positivo
- Si se selecciona el tipo de Crédito, se debe elegir un tipo de crédito

### **Qué Sucede:**

1. Transacción creada en la base de datos
2. Aparece en la lista de transacciones del cliente
3. Incluida en el conteo de "Transacciones No Facturadas"
4. Disponible para inclusión en la próxima generación de facturas
5. Entrada de registro de actividad creada

## **Buscar y Filtrar Transacciones**

### **Buscar**

Utilice la barra de búsqueda para encontrar transacciones. Busca en:

- ID de Transacción
- Título
- Descripción
- ID de Factura

### **Filtros**

Aplique filtros para reducir la lista de transacciones:

### **Filtros Disponibles:**

- **Estado de Anulación** - Todos, Anulados, No Anulados
- **Estado de Factura** - Todos, Facturados, No Facturados

## Acciones de Filtro:

- **Aplicar Filtros** - Aplicar filtros seleccionados a la lista
- **Restablecer Filtros** - Limpiar todos los filtros y mostrar todas las transacciones

## Ordenar

Haga clic en cualquier encabezado de columna para ordenar:

- **ID** - Ordenar por ID de transacción (más nuevo/más antiguo)
- **Fecha** - Ordenar por fecha de transacción
- **Título** - Ordenar alfabéticamente
- **Monto** - Ordenar por costo minorista (más alto/más bajo)
- **Factura** - Ordenar por ID de factura

Haga clic nuevamente para invertir la dirección de orden (ascendente ↔ descendente).

## Anulando Transacciones

Las transacciones añadidas por error pueden ser **anuladas** (marcadas como eliminadas).

### Requisitos:

- La transacción NO debe estar facturada
- Una vez facturada, las transacciones no pueden ser anuladas (deben ser reembolsadas en su lugar)

### Cómo Anular:

1. Localizar la transacción en la lista
2. Hacer clic en el menú de acciones ( : )
3. Seleccionar "**Anular Transacción**"
4. Confirmar en el modal

Modal de Confirmación de Anulación de Transacción

### Qué Sucede:

- Transacción marcada como `void = true`
- Ya no aparece en la lista de transacciones por defecto
- Excluida de la generación de facturas
- Puede ser vista filtrando por transacciones "Anuladas"
- Deducida del total de "Transacciones No Facturadas"

**Nota:** Anular NO es lo mismo que reembolsar. Anular significa "esta transacción nunca debió haber existido." Reembolsar significa "revertir una transacción válida."

## Impuesto sobre Transacciones

Las transacciones pueden incluir impuestos, que se calculan automáticamente en función de la configuración de impuestos del producto o especificados manualmente por transacción.

Visualización de Impuestos de Transacción

### Comportamiento del Impuesto:

- **Transacciones de Débito (Cargos)** - Se aplica impuesto a los cargos basado en:

- **Porcentaje de Impuesto del Producto** - Si la transacción está vinculada a un producto, se aplica automáticamente el porcentaje de impuesto del producto
- **Anulación Manual** - El personal puede anular el porcentaje de impuesto al crear una transacción
- **Monto del Impuesto** - Calculado como: costo\_minorista × (porcentaje\_impuesto / 100)
- **Formato de Visualización** - Mostrado como: \$10.00 (10%) en listas de transacciones
- **Transacciones de Crédito (Pagos/Reembolsos)** - No se aplica impuesto a los créditos
  - El campo de porcentaje de impuesto está oculto para transacciones de crédito
  - El impuesto se establece automáticamente en 0% para todos los pagos y reembolsos
  - Los créditos reducen el saldo pendiente del cliente sin implicaciones fiscales

#### **Ejemplo de Cálculo de Impuestos:**

- Producto: Plan Móvil con 10% de impuesto, \$50.00 costo minorista
- Cálculo Automático de Impuestos:  $\$50.00 \times 0.10 = \$5.00$
- Visualización: \$5.00 (10%)

#### **Impuesto Cero (NIL/Exento):**

- Los productos pueden estar exentos de impuestos configurando el porcentaje de impuesto a 0
- El impuesto por defecto es 0% si no se especifica
- Las transacciones exentas de impuestos muestran "-" en la columna de Impuestos

Transacción de Impuesto Cero

## **Vista de Detalles de Transacción**

Haga clic en una transacción para ver todos los detalles:

## **Transacciones Facturadas vs No Facturadas**

#### **Transacciones No Facturadas:**

- Aún no incluidas en ninguna factura
- Disponibles para la próxima generación de facturas
- Pueden ser anuladas
- Cuentan para el total de "Transacciones No Facturadas"
- El estado muestra un guion (-)

#### **Transacciones Facturadas:**

- Incluidas en una factura
- No pueden ser anuladas (deben reembolsarse si es necesario)
- ID de factura clicable (enlaces a detalles de la factura)
- El estado muestra una marca de verificación (✓)
- No pueden ser modificadas

#### **Generación de Factura:**

Cuando genera una factura para un cliente:

1. El sistema encuentra todas las transacciones no facturadas para ese cliente
2. Opcionalmente filtrar por rango de fechas
3. Transacciones incluidas en la nueva factura

4. Campo `invoice_id` de la transacción poblado
5. Transacción ahora marcada como "facturada"

Consulte `payments_invoices` para detalles sobre la generación de facturas.

## Flujos de Trabajo Comunes

### Flujo de Trabajo 1: Crédito Manual por Interrupción de Servicio

1. Cliente llama: "El servicio estuvo inactivo durante 2 días"
2. El personal decide acrediitar £10
3. Navegar a la pestaña **Facturación** del cliente
4. Hacer clic en "**+ Agregar Transacción**"
5. Seleccionar tipo de transacción **Crédito**
6. Seleccionar tipo de crédito **Pago en Efectivo**
7. Ingresar título: "Crédito por Interrupción de Servicio"
8. Ingresar descripción: "Compensación por interrupción de 2 días 8-9 de enero"
9. Ingresar costo minorista: 10.00
10. Seleccionar servicio afectado del desplegable
11. Hacer clic en "**Agregar Transacción**"
12. La transacción aparece con un monto de -£10.00
13. Se incluirá en la próxima factura como crédito

### Flujo de Trabajo 2: Tarifa de Instalación Manual

1. El técnico de campo instala el servicio
2. El personal necesita cobrar una tarifa de instalación de £75
3. Navegar a la pestaña **Facturación** del cliente
4. Hacer clic en "**+ Agregar Transacción**"
5. Seleccionar tipo de transacción **Débito**
6. Ingresar título: "Tarifa de Instalación"
7. Ingresar descripción: "Visita del técnico de campo para instalación de fibra"
8. Ingresar costo minorista: 75.00
9. Ingresar costo mayorista: 45.00 (opcional, para seguimiento de márgenes)
10. Seleccionar servicio instalado
11. Seleccionar sitio donde se instaló
12. Hacer clic en "**Agregar Transacción**"
13. La transacción aparece en la lista de no facturadas
14. Se incluirá en la próxima factura

### Flujo de Trabajo 3: Anulando Transacción Duplicada

1. El personal nota una transacción duplicada
2. Verificar que la transacción NO esté facturada aún
3. Hacer clic en el menú de acciones ( : ) en la transacción duplicada
4. Seleccionar "**Anular Transacción**"
5. Confirmar en el modal
6. Transacción eliminada de la lista
7. El total no facturado disminuye en consecuencia

### Flujo de Trabajo 4: Encontrar Transacciones para Factura

1. Necesita generar una factura mensual
2. Hacer clic en **Filtro de Factura: "No Facturado"**
3. Hacer clic en **Aplicar Filtros**
4. Ver todas las transacciones no facturadas

5. Anotar el monto total de los widgets
6. Navegar para generar factura
7. Seleccionar rango de fechas (por ejemplo, 1-31 de enero)
8. Transacciones en el rango incluidas en la factura

## Solución de Problemas

### No se puede anular la transacción

- **Causa:** La transacción ya fue facturada
- **Solución:** La transacción es parte del historial de facturas. Si se necesita un reembolso, crear una transacción de Crédito en su lugar.

### Aparecen transacciones duplicadas

- **Causa:** Servicio cobrado múltiples veces o error de aprovisionamiento
- **Solución:** Anular la(s) transacción(es) duplicada(s) si no están facturadas. Si están facturadas, emitir un crédito.

### La transacción no aparece en la lista

- **Causa:** Filtros aplicados o transacción anulada
- **Solución:** Hacer clic en "Restablecer Filtros" para mostrar todas las transacciones. Para ver transacciones anuladas, filtrar por "Anular: Anulado".

### El total no facturado no coincide con lo esperado

- **Causa:** Algunas transacciones ya están facturadas, o transacciones anuladas excluidas
- **Solución:** Aplicar el filtro "Factura: No Facturada" para ver solo las no facturadas. Verificar las transacciones anuladas por separado.

### No se puede agregar transacción (campo de cliente deshabilitado)

- **Causa:** Viendo la página de transacciones específicas del cliente
- **Solución:** El cliente está preseleccionado. Si necesita agregar una transacción para un cliente diferente, ir a la página de Transacciones a Nivel del Sistema.

## Documentación Relacionada

- `payments_invoices` - Generación y gestión de facturas
- `payments_process` - Procesamiento de pagos contra facturas
- `basics_payment` - Resumen de métodos de pago
- `cса_activity_log` - Visualización del historial de transacciones en el registro de actividad



# Control de Acceso Basado en Reglas

## Roles, Permisos y Usuarios en OmniCRM

OmniCRM utiliza **control de acceso basado en roles (RBAC)**: las personas (Usuarios) son asignadas a uno o más Roles, y cada Rol es un conjunto de Permisos. Los Permisos son la unidad más pequeña de acceso (por ejemplo, `view_customer`, `create_inventory`). El acceso efectivo de un usuario es la **unión** de los permisos de todos los roles asignados.

## Propósito

RBAC permite:

1. **Protección de Datos** — Los usuarios solo ven y hacen lo que se les permite.
2. **Ajuste Operativo** — Los roles reflejan funciones laborales (Admin, Soporte, Finanzas, Administrador de Clientes).
3. **Administración Sencilla** — Conceder acceso asignando roles; evitar la microgestión por usuario.
4. **Aislamiento de Inquilinos** — Los permisos de “ver propio ...” limitan la visibilidad a los datos de clientes/inquilinos propios de un usuario.

Permisos RBAC

## Cómo se Integran Usuarios, Roles y Permisos

- **Usuarios** — Personas reales que inician sesión en OmniCRM.
- **Permisos** — Capacidades atómicas (por ejemplo, `view_customer`, `delete_product`).
- **Roles** — Conjuntos nombrados de permisos (por ejemplo, `Admin`, `Soporte`, `Finanzas`).
- **Asignación** — Los usuarios reciben uno o más roles; los permisos se agregan.

Permisos RBAC

La autenticación prueba *quién eres* (JWT, clave API o IP en lista blanca). La autorización (roles/permisos) decide *qué puedes hacer*.

## Gestión de Usuarios

El sistema de gestión de usuarios de OmniCRM permite a los administradores

crear y gestionar usuarios del personal (administradores, agentes de servicio al cliente), ver y modificar roles de usuario, restablecer contraseñas, gestionar la autenticación de dos factores y controlar el acceso de los usuarios.

## Tipos de Usuario

**Usuarios Clientes** - Creado a través de auto-registro o por administradores. Asignado automáticamente el rol "Cliente". Estos usuarios acceden al portal de autoservicio para gestionar sus servicios, ver uso, pagar facturas, etc.

**Usuarios del Personal** - Creado por administradores con permisos apropiados. Pueden ser asignados roles como Admin, Soporte, Finanzas, etc. Estos usuarios acceden a la interfaz de CRM para gestionar clientes, aprovisionar servicios, manejar facturación, etc.

**Usuarios Administrativos** - Usuarios con el permiso `admin`. Tienen acceso completo al sistema, incluyendo gestión de usuarios, gestión de roles y todos los puntos finales protegidos.

El usuario administrativo inicial es creado por el equipo de Omnitouch cuando se despliega el sistema.

## Agregar Nuevos Usuarios (Administradores y CSAs)

Los administradores pueden crear nuevos usuarios del personal a través de la interfaz web o API.

### A través de la Interfaz Web

1. **Navegar a Usuarios y Roles** - Acceder a la interfaz de gestión de usuarios desde el menú de administración.
2. **Hacer clic en "Agregar Usuario"** - Abre el formulario de creación de usuario.

Agregando un Usuario

3. **Completar los Detalles del Usuario:**
  - **Nombre de Usuario** - Nombre de usuario único para iniciar sesión (requerido).
  - **Correo Electrónico** - Dirección de correo electrónico del usuario (requerido, debe ser único).
  - **Contraseña** - Contraseña temporal (requerido, el usuario debe cambiarla en el primer inicio de sesión).
  - **Nombre** - Primer nombre del usuario (requerido).
  - **Segundo Nombre** - Segundo nombre del usuario (opcional).
  - **Apellido** - Apellido del usuario (requerido).
  - **Número de Teléfono** - Número de teléfono de contacto (opcional).

- **Rol** - Seleccionar uno o más roles para asignar (requerido).
  - **Contacto del Cliente** - Opcionalmente vincular usuario a un registro de contacto de cliente (para usuarios clientes).
4. **Hacer clic en "Crear Usuario"** - El usuario es creado y puede iniciar sesión inmediatamente con las credenciales proporcionadas.
  5. **El usuario recibe una notificación** - Opcionalmente enviar un correo electrónico de bienvenida con instrucciones de inicio de sesión.

## Mejores Prácticas:

- Usar una contraseña temporal como TempP@ssw0rd! y requerir que el usuario la cambie en el primer inicio de sesión.
- Asignar roles apropiados según la función laboral (ver Diseños de Roles Típicos a continuación).
- Habilitar 2FA para todo el personal administrativo y de soporte.
- Vincular usuarios clientes a su registro de contacto de cliente para un adecuado alcance de datos.

## A través de la API

Crear un usuario programáticamente:

**Endpoint:** POST /auth/users

**Permiso Requerido:** admin

### Cuerpo de la Solicitud:

```
{
  "username": "john.smith",
  "email": "john.smith@company.com",
  "password": "TempP@ssw0rd!",
  "first_name": "John",
  "middle_name": "D",
  "last_name": "Smith",
  "phone_number": "+61412345678",
  "role": "Support"
}
```

### Respuesta:

```
{
  "id": 123,
  "username": "john.smith",
  "email": "john.smith@company.com",
  "first_name": "John",
  "last_name": "Smith",
  "roles": ["Support"],
  "created": "2025-01-04T10:30:00Z"
```

```
}
```

## Asignando Múltiples Roles:

Los usuarios pueden tener múltiples roles. Los permisos son aditivos (unión de todos los permisos de rol asignados).

Para asignar múltiples roles, inclúyelos en la solicitud:

```
{
  "username": "jane.doe",
  "email": "jane.doe@company.com",
  "password": "TempP@ssw0rd!",
  "first_name": "Jane",
  "last_name": "Doe",
  "role": "Support,Finance"
}
```

O utiliza el endpoint de asignación de roles después de la creación del usuario:

```
POST /auth/roles/{role_id}/users/{user_id}
```

## Ver y Buscar Usuarios

### Listar Todos los Usuarios (Admin):

```
GET /auth/users
```

Devuelve una lista paginada de todos los usuarios con sus roles e información básica.

### Buscar Usuarios:

```
GET /auth/users/
search?search={query}&filters={"role": ["Support"]}&page=1&per_page=50
```

Filtrar por:

- Nombre del rol
- Dominio del correo electrónico
- Estado activo/eliminado
- Estado de 2FA habilitado
- Fecha del último inicio de sesión

### Obtener Usuario Específico:

```
GET /auth/users/{user_id}
```

Devuelve detalles completos del usuario incluyendo:

- Información personal
- Roles asignados y permisos efectivos
- Estado de 2FA
- Último inicio de sesión e información de sesión
- Contacto de cliente vinculado (si aplica)

## Crear y Gestionar Roles

Los roles son colecciones de permisos que pueden ser asignados a los usuarios. En lugar de asignar permisos individualmente a cada usuario, creas roles que agrupan permisos relacionados y asignas esos roles a los usuarios.

### Crear un Nuevo Rol

#### A través de la Interfaz Web:

1. Navegar a **Usuarios y Roles** → pestaña **Roles**.
2. Hacer clic en "**Crear Rol**".
3. Ingresar detalles del rol:
  - **Nombre** - Nombre corto y descriptivo (por ejemplo, "Tier2\_Support").
  - **Descripción** - Explicar el propósito y responsabilidades del rol.
4. Hacer clic en "**Crear**".
5. El rol es creado sin permisos; agregar permisos en el siguiente paso.

#### A través de la API:

**Endpoint:** POST /auth/roles

**Permiso Requerido:** admin

#### Solicitud:

```
{  
  "name": "Tier2_Support",  
  "description": "Equipo de soporte de nivel 2 con acceso elevado a  
  aprovisionamiento"  
}
```

#### Respuesta:

```
{  
  "id": 45,  
  "name": "Tier2_Support",  
  "description": "Equipo de soporte de nivel 2 con acceso elevado a  
  aprovisionamiento",  
  "permissions": [],  
  "users": []  
}
```

## Agregar Permisos a un Rol

Una vez creado un rol, asigna permisos para definir lo que los usuarios con ese rol pueden hacer.

### A través de la Interfaz Web:

1. Navegar a **Usuarios y Roles** → pestaña **Roles**.
2. Hacer clic en el nombre del rol para ver detalles.
3. En la sección **Permisos**, hacer clic en "**Agregar Permiso**".
4. Seleccionar uno o más permisos de la lista.
5. Hacer clic en "**Agregar**" - Los permisos se asignan inmediatamente.

Gestionando Permisos de Rol Asignando Permisos a Rol

### A través de la API:

**Endpoint:** POST /auth/roles/{role\_id}/permissions

#### Solicitud:

```
{  
  "permission_id": 123  
}
```

O agregar múltiples permisos:

```
{  
  "permission_ids": [123, 124, 125]  
}
```

### Ejemplo: Creando un Rol de "Especialista en Aprovisionamiento"

Este rol puede ver clientes, gestionar servicios y aprovisionar:

1. Crear el rol:

```
POST /auth/roles  
{  
  "name": "Provisioning_Specialist",  
  "description": "Puede aprovisionar servicios y gestionar servicios de clientes"  
}
```

2. Agregar permisos:

```
POST /auth/roles/45/permissions  
{  
  "permission_ids": [  
    123, 124, 125  
  ]  
}
```

```
1,    # view_customer
20,   # view_customer_service
21,   # create_customer_service
22,   # update_customer_service
30,   # view_provision
31,   # create_provision
40,   # view_inventory
50,   # view_product
]
}
```

## Eliminar Permisos de un Rol

### A través de la Interfaz Web:

1. Navegar a los detalles del rol.
2. En la lista de **Permisos**, hacer clic en el "X" o en el botón "**Eliminar**" junto al permiso.
3. Confirmar la eliminación.

### A través de la API:

**Endpoint:** `DELETE /auth/roles/{role_id}/permissions/{permission_id}`

#### Ejemplo:

```
DELETE /auth/roles/45/permissions/31
```

Esto elimina el permiso `create_provision` del rol.

## Editar Detalles del Rol

Actualizar el nombre o la descripción del rol:

### A través de la Interfaz Web:

1. Navegar a **Usuarios y Roles** → pestaña **Roles**.
2. Hacer clic en el rol para editar.
3. Modificar el nombre o la descripción del rol.
4. Hacer clic en "**Guardar**".

Editando Detalles del Rol

### A través de la API:

**Endpoint:** `PUT /auth/roles/{role_id}`

```
{
  "name": "Senior_Support",
```

```
  "description": "Equipo de soporte senior con acceso completo a
clientes"
}
```

## Eliminar un Rol

**Advertencia:** Eliminar un rol lo elimina de todos los usuarios asignados. Asegúrate de que los usuarios tengan roles alternativos o perderán el acceso.

### A través de la API:

`DELETE /auth/roles/{role_id}`

**Mejor Práctica:** En lugar de eliminar, considera archivar o renombrar roles que ya no son necesarios.

## Asignar Roles a Usuarios

### Durante la Creación de Usuarios:

Incluir rol en la solicitud de creación de usuario (ver "Agregando Nuevos Usuarios" arriba).

### Para Usuarios Existentes:

#### A través de la Interfaz Web:

1. Navegar a **Usuarios y Roles** → pestaña **Usuarios**.
2. Hacer clic en el usuario para editar.
3. En la sección **Roles**, seleccionar/deseleccionar roles.
4. Hacer clic en "**Guardar**".

Editando Roles de Usuario

### A través de la API:

Actualizar los roles del usuario:

**Endpoint:** `PUT /auth/users/{user_id}`

```
{
  "role": "Support,Finance"
}
```

O asignar un solo rol a un usuario a través del endpoint de rol:

**Endpoint:** `POST /auth/roles/{role_id}/users/{user_id}`

## Ver Asignaciones de Roles

### Todos los usuarios en un rol:

GET /auth/roles/{role\_id}/users

Devuelve una lista de todos los usuarios asignados a ese rol.

### Todos los roles para un usuario:

GET /auth/users/{user\_id}

La respuesta incluye un array roles con todos los roles asignados.

## Gestión de Contraseñas de Usuario

OmniCRM proporciona múltiples métodos para la gestión de contraseñas dependiendo del contexto.

### Restablecimiento de Contraseña por Auto-Servicio del Usuario

Los usuarios que olvidaron su contraseña pueden restablecerla ellos mismos a través de la página de inicio de sesión:

1. **Hacer clic en "Olvidé mi Contraseña"** en la página de inicio de sesión.
2. **Ingresar dirección de correo electrónico** - El sistema envía un correo electrónico de restablecimiento de contraseña.
3. **Revisar correo electrónico** - El correo contiene un enlace de restablecimiento seguro con un token (válido por 1 hora).
4. **Hacer clic en el enlace** - Abre el formulario de restablecimiento de contraseña.
5. **Ingresar nueva contraseña** - Debe cumplir con los requisitos de complejidad de la contraseña:
  - Mínimo 8 caracteres.
  - Al menos una letra mayúscula.
  - Al menos una letra minúscula.
  - Al menos un número.
  - Al menos un carácter especial.
6. **Enviar** - La contraseña se actualiza inmediatamente; el usuario puede iniciar sesión con la nueva contraseña.

### Flujo de API:

1. **Solicitar restablecimiento:**

**Endpoint:** POST /auth/forgot\_password

{

```
        "email": "user@example.com"  
    }
```

El sistema genera un token de restablecimiento y envía un correo electrónico.

## 2. Restablecer con token:

**Endpoint:** POST /auth/reset\_password

```
{  
    "token": "abc123... ",  
    "new_password": "NewSecureP@ssw0rd!"  
}
```

## Restablecimiento de Contraseña por Administrador

Los administradores pueden restablecer la contraseña de un usuario sin requerir verificación por correo electrónico. Esto establece una contraseña temporal que el usuario debe cambiar en el siguiente inicio de sesión.

### A través de la Interfaz Web:

1. Navegar a **Usuarios y Roles** → **Usuarios**.
2. Encontrar al usuario y hacer clic en el botón **"Restablecer Contraseña"**.
3. Ingresar una contraseña temporal.
4. Hacer clic en **"Restablecer"**.
5. Notificar al usuario sobre su contraseña temporal (a través de un canal seguro).
6. El usuario debe cambiar la contraseña en el siguiente inicio de sesión.

### A través de la API:

**Endpoint:** POST /auth/users/{user\_id}/admin\_reset\_password

**Permiso Requerido:** admin

### Solicitud:

```
{  
    "new_password": "TempP@ssw0rd!",  
    "force_change": true  
}
```

### Parámetros:

- **new\_password** - Contraseña temporal a establecer.
- **force\_change** (opcional) - Si es verdadero, el usuario debe cambiar la contraseña en el siguiente inicio de sesión.

## Cambio de Contraseña por el Usuario

Los usuarios autenticados pueden cambiar su propia contraseña desde su perfil:

**Endpoint:** POST /auth/change\_password

**Solicitud:**

```
{  
  "current_password": "OldP@ssw0rd!",  
  "new_password": "NewSecureP@ssw0rd!"  
}
```

El sistema valida la contraseña actual antes de permitir el cambio.

## Seguridad de Contraseña

- Las contraseñas se almacenan utilizando bcrypt (seguridad de werkzeug).
- Nunca se almacenan en texto plano.
- Los tokens de restablecimiento exirpan después de 1 hora.
- Los intentos fallidos de inicio de sesión pueden activar el bloqueo de la cuenta (configurable).
- El seguimiento del historial de contraseñas previene la reutilización (configurable).
- Se imponen requisitos de complejidad.

## Gestión de Autenticación de Dos Factores (2FA)

OmniCRM soporta autenticación de dos factores basada en TOTP para una mayor seguridad. Los administradores pueden habilitar, deshabilitar y restablecer 2FA para los usuarios.

Popup de advertencia de 2FA de OmniCRM mostrado a usuarios administradores

### Habilitar 2FA para un Usuario

**A través de la Interfaz Web:**

1. Navegar a **Usuarios y Roles** → **Usuarios**.
2. Hacer clic en el usuario para ver detalles.
3. En la sección **Seguridad**, hacer clic en "**Habilitar 2FA**".
4. El sistema genera:
  - Secreto TOTP (código QR mostrado).
  - 10 códigos de respaldo (uso único).
5. El usuario escanea el código QR con la aplicación de autenticador (Google Authenticator, Authy, etc.).
6. El usuario ingresa el código de verificación de la aplicación para confirmar la configuración.

7. El usuario guarda los códigos de respaldo en un lugar seguro.
8. 2FA ahora está habilitado; requerido para todos los futuros inicios de sesión.

Página de inscripción de 2FA de OmniCRM

### A través de la API:

1. Generar secreto TOTP:

**Endpoint:** POST /2fa/enable/user/{user\_id}

**Respuesta:**

```
{  
  "totp_secret": "JBSWY3DPEHPK3PXP",  
  "qr_code_url": "otpauth://totp/  
OmniCRM:user@example.com?secret=JBSWY3DPEHPK3PXP&issuer=OmniCRM",  
  "backup_codes": [  
    "12345678",  
    "23456789",  
    "34567890",  
    ...  
  ]  
}
```

2. Verificar configuración:

**Endpoint:** POST /2fa/verify-setup/user/{user\_id}

```
{  
  "code": "123456"  
}
```

Devuelve {"verified": true} en caso de éxito.

## Flujo de Inicio de Sesión 2FA

Una vez habilitado 2FA, el proceso de inicio de sesión cambia:

1. El usuario ingresa nombre de usuario y contraseña.
2. El sistema valida las credenciales.
3. Si son válidas, solicita el código 2FA.
4. El usuario ingresa el código de la aplicación de autenticador O el código de respaldo.
5. El sistema verifica el código.
6. En caso de éxito, el usuario inicia sesión.

Inicio de sesión 2FA de OmniCRM

## Códigos de Respaldo:

- 10 códigos generados durante la configuración de 2FA.
- Solo de un solo uso (consumidos después de su uso).
- Usados si la aplicación de autenticador no está disponible.
- Pueden ser regenerados por el usuario o el administrador.

## Verificando el Código 2FA

**Endpoint:** POST /2fa/verify/user/{user\_id}

```
{  
  "code": "123456"  
}
```

Acepta ambos:

- **Código TOTP** (6 dígitos de la aplicación de autenticador).
- **Código de respaldo** (8 dígitos de la lista de códigos de respaldo).

## Regenerando Códigos de Respaldo

Si un usuario agota los códigos de respaldo o los pierde, genera nuevos:

### A través de la Interfaz Web:

1. Navegar a los detalles del usuario.
2. Hacer clic en **"Regenerar Códigos de Respaldo"**.
3. Mostrar o enviar nuevos códigos al usuario.
4. Los códigos antiguos se invalidan.

### A través de la API:

**Endpoint:** POST /2fa/backup-codes/regenerate/user/{user\_id}

### Respuesta:

```
{  
  "backup_codes": [  
    "98765432",  
    "87654321",  
    "76543210",  
    ...  
  ]  
}
```

## Restablecimiento 2FA por Administrador

Si un usuario pierde el acceso a su aplicación de autenticador y todos los códigos

de respaldo, un administrador puede deshabilitar y volver a habilitar 2FA.

### **A través de la Interfaz Web:**

1. Navegar a **Usuarios y Roles** → **Usuarios**.
2. Hacer clic en el usuario.
3. Hacer clic en el botón "**Restablecer 2FA**".
4. Confirmar el restablecimiento.
5. 2FA está deshabilitado; el usuario puede iniciar sesión solo con la contraseña.
6. Guiar al usuario para que configure 2FA nuevamente con un nuevo secreto.

Restablecimiento 2FA de OmniCRM

### **A través de la API:**

**Endpoint:** POST /2fa/admin/disable/user/{user\_id}

**Permiso Requerido:** admin

Esto deshabilita completamente 2FA para el usuario:

- Secreto TOTP borrado.
- Códigos de respaldo borrados.
- La bandera `is_2fa_enabled` se establece en falso.

El usuario puede volver a habilitar 2FA para obtener un nuevo secreto y códigos de respaldo.

### **Restablecimiento 2FA por Auto-Servicio del Usuario (Nuevo Dispositivo)**

Si un usuario obtiene un nuevo dispositivo pero aún tiene acceso a los códigos de respaldo:

**Endpoint:** POST /2fa/reset-for-new-device/user/{user\_id}

```
{  
  "backup_code": "12345678"  
}
```

El sistema valida el código de respaldo, luego genera un nuevo secreto TOTP y códigos de respaldo. El usuario puede configurar la aplicación de autenticador en el nuevo dispositivo.

### **Mejores Prácticas para 2FA**

- **Requerir 2FA para todo el personal administrativo y de soporte.**
- **Almacenar códigos de respaldo de manera segura** (gestor de

contraseñas o nota segura).

- **Regenerar códigos de respaldo** después de usar varios.
- **Usar aplicaciones de autenticador de buena reputación** (Google Authenticator, Authy, Microsoft Authenticator).
- **Documentar procedimientos de restablecimiento de 2FA** para el personal de soporte.
- **Auditar el uso de 2FA** - monitorear qué usuarios tienen 2FA habilitado.

## Actualizando Información del Usuario

Los administradores pueden actualizar los detalles del usuario en cualquier momento.

### A través de la Interfaz Web:

1. Navegar a **Usuarios y Roles** → **Usuarios**.
2. Hacer clic en el usuario para editar.
3. Modificar cualquier campo editable:
  - Primer nombre, segundo nombre, apellido.
  - Dirección de correo electrónico (requiere verificación).
  - Número de teléfono.
  - Roles.
  - Vinculación de contacto de cliente.
4. Hacer clic en "**Guardar**".

### A través de la API:

**Endpoint:** PUT /auth/users/{user\_id}

```
{  
  "first_name": "Jane",  
  "last_name": "Doe-Smith",  
  "email": "jane.doesmith@newcompany.com",  
  "phone_number": "+61498765432",  
  "role": "Support,Finance"  
}
```

### Cambios de Correo Electrónico:

Cuando se cambia el correo electrónico, el nuevo correo se marca como pendiente hasta que se verifique:

- El campo `pending_email` almacena el nuevo correo.
- Se envía un correo electrónico de verificación a la nueva dirección.
- El usuario hace clic en el enlace para verificar.
- El campo `email` se actualiza al nuevo valor.
- La bandera `email_verified` se establece en verdadero.

# Eliminando Usuarios

OmniCRM utiliza **eliminaciones suaves** para los usuarios: se marcan como eliminados pero no se eliminan de la base de datos. Esto preserva las auditorías y los datos históricos.

## Eliminando un Usuario

### A través de la Interfaz Web:

1. Navegar a **Usuarios y Roles** → **Usuarios**.
2. Encontrar al usuario a eliminar.
3. Hacer clic en el botón "**Eliminar**".
4. Confirmar la eliminación.
5. El usuario es inmediatamente desconectado y no puede iniciar sesión nuevamente.

### A través de la API:

**Endpoint:** `DELETE /auth/users/{user_id}`

**Permiso Requerido:** `admin`

### Qué Sucede:

- La bandera `deleted` se establece en `True`.
- Se registra la marca de tiempo `deleted_at`.
- El usuario no puede iniciar sesión.
- Todas las sesiones activas se invalidan.
- El usuario aún aparece en los registros de auditoría y registros históricos.
- Los datos vinculados (contactos de clientes, actividades) se preservan.

## Ver Usuarios Eliminados

### Filtrar por usuarios eliminados:

`GET /auth/users/search?filters={"deleted": [true]}`

## Restaurar un Usuario Eliminado

Si un usuario fue eliminado por error, los administradores pueden restaurarlo:

**Endpoint:** `PUT /auth/users/{user_id}`

```
{  
  "deleted": false  
}
```

Esto borra la bandera `deleted` y permite al usuario iniciar sesión nuevamente.

**Nota:** La contraseña del usuario permanece sin cambios, por lo que puede usar su contraseña anterior.

## Eliminando Permanentemente un Usuario

**Advertencia:** Esto es irreversible y elimina todos los datos del usuario de la base de datos.

No se expone a través de la interfaz de usuario. Solo está disponible a través del acceso directo a la base de datos por razones de cumplimiento (por ejemplo, solicitudes de eliminación de datos GDPR).

## Mejores Prácticas para la Eliminación de Usuarios

- **Eliminar suavemente por defecto** - Preserva las auditorías.
- **Documentar razones de eliminación** - Agregar nota en el registro de actividades antes de eliminar.
- **Transferir propiedad** - Reasignar tickets abiertos y tareas del usuario antes de eliminar.
- **Revisar acceso** - Asegurarse de que no haya procesos críticos que dependan del usuario.
- **Archivar datos** - Exportar el historial de trabajo del usuario si es necesario.
- **Notificar a los equipos relevantes** - Informar a gerentes/compañeros sobre la eliminación.

## Catálogo de Permisos

Los permisos generalmente siguen patrones CRUD:

- `view_*` — leer/navegar.
- `create_*` — crear/agregar.
- `update_*` — editar/modificar.
- `delete_*` — eliminar/remover.

Algunas entidades también incluyen variantes de “**ver propio ...**” que restringen la visibilidad a los datos del cliente/inquilino **propio** del usuario.

Permisos RBAC

## Global / Administrativo

- `admin` — Acceso administrativo completo (gestionar usuarios, roles y permisos; acceder a todos los puntos finales protegidos).
- `can_impersonate` — Actuar temporalmente como otro usuario (auditado; para soporte/troubleshooting).

# Clientes y Registros Relacionados

- **Cliente**
  - view\_customer, create\_customer, update\_customer, delete\_customer
  - **Alcance propio:** *ver propio cliente*
- **Sitio del Cliente**
  - view\_customer\_site, create\_customer\_site, update\_customer\_site, delete\_customer\_site
  - **Alcance propio:** *ver propio sitio del cliente*
- **Contacto del Cliente**
  - view\_customer\_contact, create\_customer\_contact, update\_customer\_contact, delete\_customer\_contact
  - **Alcance propio:** *ver propio contacto del cliente*
- **Atributo del Cliente** (ver Atributos del Cliente  
<administration\_attributes>)
  - view\_customer\_attribute, create\_customer\_attribute, update\_customer\_attribute, delete\_customer\_attribute
  - **Alcance propio:** *ver propio atributo del cliente*
- **Etiqueta del Cliente** (ver Etiquetas del Cliente  
<administration\_tags>)
  - view\_customer\_tag, create\_customer\_tag, update\_customer\_tag, delete\_customer\_tag
  - **Alcance propio:** *ver propia etiqueta del cliente*
- **Servicio del Cliente**
  - view\_customer\_service, create\_customer\_service, update\_customer\_service, delete\_customer\_service
  - **Alcance propio:** *ver propio servicio del cliente*
- **Actividad del Cliente**
  - view\_customer\_activity, create\_customer\_activity, update\_customer\_activity, delete\_customer\_activity
  - **Alcance propio:** *ver propia actividad del cliente*

# Facturación

- **Tarjeta de Stripe**
  - view\_customer\_stripe\_card, create\_customer\_stripe\_card, update\_customer\_stripe\_card, delete\_customer\_stripe\_card
  - **Alcance propio:** *ver propia tarjeta de stripe del cliente*
- **Transacciones**
  - view\_customer\_transaction, create\_customer\_transaction, update\_customer\_transaction, delete\_customer\_transaction
  - **Alcance propio:** *ver propia transacción del cliente*
- **Facturas**
  - view\_customer\_invoice, create\_customer\_invoice, update\_customer\_invoice, delete\_customer\_invoice
  - **Alcance propio:** *ver propia factura del cliente*

# Comunicaciones

- view\_communication, create\_communication, update\_communication, delete\_communication
- **Alcance propio:** *ver propia comunicación*

# Inventario y Plantillas

- **Inventario**
  - view\_inventory, create\_inventory, update\_inventory, delete\_inventory
  - **Alcance propio:** *ver propio inventario*
- **Plantilla de Inventario**
  - view\_inventory\_template, create\_inventory\_template, update\_inventory\_template, delete\_inventory\_template
  - **Alcance propio:** *ver propia plantilla de inventario*

# Productos

- view\_product, create\_product, update\_product, delete\_product

# Difusión Celular (CBC)

- view\_cbc\_message, create\_cbc\_message, update\_cbc\_message, delete\_cbc\_message

# Aprovisionamiento

- **Aprovisionar**
  - view\_provision, create\_provision, update\_provision, delete\_provision
  - **Alcance propio:** *ver propio aprovisionamiento*
- **Evento de Aprovisionamiento**
  - view\_provision\_event, create\_provision\_event, update\_provision\_event, delete\_provision\_event

# Acceso "Ver Propio"

Los permisos de “ver propio ...” limitan las lecturas (y opcionalmente las ediciones, donde se implementen) a los datos asociados con el **propio cliente/inquilino** del usuario. Por ejemplo, un rol de *Administrador de Clientes* puede gestionar los contactos, sitios, facturas y servicios de *su inquilino*, pero no puede ver otros inquilinos.

# Diseños de Roles Típicos

Rol	Permisos Típicos	Notas
Administrador del Sistema	Administrador <code>admin</code> , opcionalmente <code>can_impersonate</code> ; más CRUD amplio según sea necesario	Control total sobre usuarios/roles/permisos
Soporte	<code>view_customer</code> , <code>view_customer_service</code> , <code>view_communication</code> , <code>view_provision</code> ; actualizaciones opcionales	Agregar <code>can_impersonate</code> si se permite
Finanzas	<code>view_customer_invoice</code> , <code>view_customer_transaction</code> , <code>view_product</code> ; opcional <code>create_customer_invoice</code>	Enfoque en lectura; escritura limitada
Administrador de Clientes (inquilino)	"ver propio ..." en clientes, sitios, contactos, servicios, inventario, facturas, transacciones, comunicaciones, aprovisionamiento	Gestión con alcance de inquilino
Auditor Solo de Lectura	Amplio <code>view_*</code> solo	Sin crear/actualizar/eliminar

Ejemplo de Roles y Permisos Incluidos (resumen)

## Gestión de Roles y Permisos a través de la API

Todos los endpoints requieren permiso `admin`.

### Listar permisos

**Endpoint:** GET /auth/permissions

### Crear un permiso (raro)

**Endpoint:** POST /auth/permissions

### Cuerpo de la Solicitud:

```
{  
  "name": "view_example",  
  "description": "Acceso solo de lectura a objetos de ejemplo"  
}
```

### Listar roles

**Endpoint:** GET /auth/roles

## Crear un rol

**Endpoint:** POST /auth/roles

### Cuerpo de la Solicitud:

```
{  
  "name": "Support",  
  "description": "Equipo de soporte de nivel 1"  
}
```

## Agregar un permiso a un rol

**Endpoint:** POST /auth/roles/{role\_id}/permissions

### Cuerpo de la Solicitud:

```
{  
  "permission_id": 123  
}
```

## Eliminar un permiso de un rol

**Endpoint:** DELETE /auth/roles/{role\_id}/permissions/{permission\_id}

## Asignar Roles a Usuarios

### Crear un usuario con rol

**Endpoint:** POST /auth/users

### Cuerpo de la Solicitud:

```
{  
  "username": "sara",  
  "email": "sara@example.com",  
  "password": "TempP@ssw0rd!",  
  "first_name": "Sara",  
  "last_name": "Ng",  
  "phone_number": "+61...",  
  "role": "Support"  
}
```

## Actualizar el rol de un usuario

**Endpoint:** PUT /auth/users/{user\_id}

## Cuerpo de la Solicitud:

```
{  
  "role": "Finance"  
}
```

## Listar usuarios (solo Admin)

**Endpoint:** GET /auth/users

## Suplantación (Controlada)

- **Requerido:** can\_impersonate o admin

## Iniciar suplantación

**Endpoint:** POST /auth/impersonate

## Cuerpo de la Solicitud:

```
{ "user_id": 42 }
```

## Detener suplantación

**Endpoint:** POST /auth/stop\_impersonation

## Mejores Prácticas

- **Menor privilegio primero.** Comienza con roles mínimos; agrega permisos según sea necesario.
- **Preferir “ver propio ...”.** Utiliza permisos con alcance de inquilino para roles orientados al cliente.
- **Mantener roles estables.** Actualiza los permisos de rol cuando cambian las características—no edites cada usuario.
- **Auditar regularmente.** Revisa quién tiene admin o can\_impersonate.

## Preguntas Frecuentes

**¿Puede un usuario tener múltiples roles?** Sí. Los permisos son aditivos.

**¿Necesito permisos personalizados?** Generalmente no. El catálogo integrado cubre la mayoría de las necesidades.

**¿Cómo saben las reglas de “ver propio ...” qué es mío?** Evalúan el enlace entre tu usuario/contacto y tu cliente (inquilino).



# Portal de Autocuidado

El Portal de Autocuidado es una interfaz orientada al cliente que permite a los usuarios finales gestionar sus propias cuentas, ver el uso, pagar facturas y modificar servicios sin necesidad de asistencia del personal de atención al cliente.

## Métodos de Acceso:

- Inicio de sesión directo a través de credenciales del cliente
- **Impersonación de Personal** <customer\_care> para resolución de problemas (desde Clientes → Contactos → "Iniciar sesión como Usuario")

Consulta también: Atención al Cliente <customer\_care> para detalles de impersonación, Flujos de Autenticación <authentication\_flows> para el proceso de inicio de sesión.

## Propósito

El Portal de Autocuidado proporciona a los clientes:

1. **Gestión de Cuentas** - Ver y actualizar información personal, contactos y direcciones
2. **Resumen de Servicios** - Ver todos los servicios activos, uso y fechas de caducidad
3. **Seguimiento de Uso** - Monitorear datos, voz, SMS y saldos monetarios
4. **Acceso a Facturación** - Ver y pagar facturas, gestionar métodos de pago
5. **Modificaciones de Servicios** - Agregar recargas, comprar complementos, modificar servicios
6. **Disponibilidad 24/7** - Acceder a la información de la cuenta en cualquier momento sin llamar al soporte

## Resumen del Portal

Cuando los clientes inician sesión en el Portal de Autocuidado, ven un panel de control personalizado con:

### Secciones de Navegación:

- **Panel de Control** - Visión rápida de servicios y actividad reciente
- **Cuenta** - Detalles personales, contactos, sitios
- **Servicios** - Lista de todos los servicios con estado y detalles
- **Uso** - Información sobre el consumo de saldo y fechas de caducidad
- **Facturación** - Facturas, transacciones, métodos de pago

- **Recarga** - Comprar crédito de datos, voz, SMS o crédito monetario

## Detalles de la Cuenta

La sección de Cuenta muestra la información del cliente y permite actualizaciones de autoservicio limitadas.

Portal de Autocuidado - Detalles de la Cuenta

### **Campos Editables:**

Los clientes pueden actualizar:

- Dirección de correo electrónico (requiere verificación)
- Número de teléfono
- Contraseña
- Preferencias de notificación

### **Información Solo de Visualización:**

- ID del cliente
- Fecha de creación de la cuenta
- Tipo de cliente (Individual/Negocio)
- Sitios (direcciones)
- Contactos vinculados

### **Actualización de la Información de la Cuenta:**

1. Navegar a **Cuenta** → **Detalles**
2. Hacer clic en "**Editar**" junto al campo a actualizar
3. Ingresar nueva información
4. Hacer clic en "**Guardar Cambios**"
5. Para cambios de correo electrónico, verificar a través del enlace enviado a la nueva dirección

### **Características de Seguridad:**

- Los cambios de contraseña requieren la contraseña actual
- Los cambios de correo electrónico requieren verificación
- Actividad registrada para auditoría
- Configuraciones de 2FA (si están habilitadas)

## Resumen de Servicios

La sección de Servicios muestra todos los servicios activos e inactivos para el cliente.

## Visualización de Tarjetas de Servicio:

Cada servicio muestra:

- **Nombre del Servicio** - Identificador legible por humanos (por ejemplo, "Móvil - +44 7700 900123")
- **Producto** - Nombre del plan o producto
- **Estado** - Activo, Suspendedo, Expirado, Cancelado
- **Fecha de Creación** - Fecha de activación del servicio
- **Fecha de Expiración** - Cuando el servicio expira (si aplica)
- **Costo Mensual** - Cargo recurrente
- **Renovación Automática** - Indicador Habilitado/Deshabilitado

## Acciones del Servicio:

- **Ver Uso** - Ver el consumo de saldo (Datos, Voz, SMS, Monetario)
- **Recarga** - Agregar crédito o datos
- **Complementos** - Comprar características adicionales
- **Modificar** - Cambiar parámetros del servicio (si está permitido)
- **Ver Detalles** - Ver la configuración completa del servicio

## Indicadores de Estado del Servicio:

- ◊ **Activo** - El servicio está operativo
- ◊ **Próximo a Expirar** - Renueva o expira en <7 días
- ◊ **Suspendido** - Servicio temporalmente deshabilitado (problema de pago, suspensión manual)
- • **Expirado** - Servicio ya no activo

# Seguimiento de Uso

Los clientes pueden monitorear su uso a través de todos los tipos de saldo en tiempo real.

## Uso de Datos

Ver el consumo de datos con desgloses detallados por bucket y expiración.

Portal de Autocuidado - Uso de Datos

## Visualización del Uso de Datos:

- **Saldo Total** - Todos los buckets de datos combinados
- **Usado en Este Período** - Consumo desde la última renovación
- **Barra de Progreso** - Representación visual del consumo
- **Información de Expiración** - Cuando expira cada bucket

- **Desglose de Buckets** - Múltiples buckets de datos con orden de prioridad

### **Prioridad de Buckets:**

Cuando existen múltiples buckets de datos (por ejemplo, plan base + recargas), se consumen en orden de peso:

- **Peso 10** - Consumido primero (típicamente datos promocionales/bonus)
- **Peso 20** - Consumido segundo (típicamente datos de recarga)
- **Peso 30** - Consumido último (típicamente datos del plan base)

### **Uso de Voz**

Rastrear minutos de llamadas restantes a través de todos los buckets de voz.

Portal de Autocuidado - Uso de Voz

### **Visualización del Uso de Voz:**

- **Minutos Restantes** - Saldo total de voz
- **Minutos Usados** - Consumo en este período
- **Historial de Llamadas** - Llamadas recientes (si está habilitado)
- **Fechas de Expiración** - Cuando expiran los buckets de voz
- **Minutos Internacionales** - Seguimiento separado (si aplica)

### **Desglose de Uso:**

- Llamadas On-Net - Llamadas dentro de la misma red
- Llamadas Off-Net - Llamadas a otras redes
- Llamadas Internacionales - Llamadas fuera del país
- Números Premium - Números de tarifa especial

### **Uso de SMS**

Monitorear las asignaciones y el consumo de mensajes SMS.

### **Visualización del Uso de SMS:**

- **Mensajes Restantes** - Saldo de SMS
- **Usados Este Mes** - Mensajes enviados
- **MMS Incluidos** - Si los MMS cuentan para el saldo
- **SMS Internacionales** - Seguimiento separado (si aplica)

### **Saldo Monetario**

Ver el saldo de crédito prepago para servicios de pago por uso.

### **Visualización Monetaria:**

- **Saldo Actual** - Crédito disponible
- **Última Recarga** - Monto y fecha de recarga más reciente
- **Fecha de Expiración** - Cuando expira el saldo (si aplica)
- **Recarga Automática** - Estado Habilitado/Deshabilitado

## Gestión de Facturación

Los clientes pueden ver facturas, transacciones y gestionar métodos de pago.

### Facturas

Ver y pagar facturas pendientes directamente desde el portal.

Portal de Autocuidado - Facturas

#### Visualización de la Lista de Facturas:

- **Número de Factura** - Identificador único
- **Fecha** - Fecha de creación de la factura
- **Fecha de Vencimiento** - Plazo de pago
- **Monto** - Monto total de la factura
- **Estado** - Pagada, No Pagada, Vencida
- **Acciones** - Descargar PDF, Pagar en Línea

#### Pago de una Factura:

1. Navegar a **Facturación** → **Facturas**
2. Encontrar la factura no pagada en la lista
3. Hacer clic en el botón "**Pagar Ahora**"
4. Seleccionar método de pago (tarjeta guardada o nueva tarjeta)
5. Confirmar el pago
6. Recibir correo electrónico de confirmación

#### Descargando Facturas:

1. Hacer clic en el ícono "**Descargar**" junto a la factura
2. PDF se descarga con todos los detalles de la factura
3. Guardado para registros fiscales y documentación

## Transacciones

Ver el historial completo de transacciones, incluyendo cargos, créditos y pagos.

#### Visualización de Transacciones:

- **Fecha** - Fecha de creación de la transacción
- **Descripción** - Para qué es el cargo/crédito
- **Monto** - Cargo (positivo) o crédito (negativo)

- **Factura** - Qué factura incluye esta transacción
- **Estado** - Facturado o No Facturado

## Métodos de Pago

Gestionar tarjetas de crédito guardadas para facturación automática y pagos en línea.

### Gestión de Métodos de Pago:

- **Agregar Tarjeta** - Agregar de forma segura una nueva tarjeta de crédito/débito a través de Stripe
- **Establecer Predeterminado** - Elegir método de pago principal
- **Eliminar Tarjeta** - Eliminar tarjetas caducadas o no utilizadas
- **Detalles de la Tarjeta** - Ver últimos 4 dígitos, fecha de caducidad, marca de la tarjeta

Consulta Métodos de Pago <payment\_methods> para documentación detallada sobre la gestión de pagos.

## Recarga / Recarga

Comprar crédito adicional de datos, voz, SMS o crédito monetario al instante.

### Proceso de Recarga:

1. Navegar a **Servicios** → Seleccionar servicio → "**Recarga**"
2. Elegir producto de recarga del catálogo
3. Seleccionar monto (opciones preestablecidas o personalizadas)
4. Revisar información de costo y expiración
5. Seleccionar método de pago
6. Confirmar compra
7. Saldo actualizado inmediatamente

### Recargas Disponibles:

- **Recargas de Datos** - Opciones de 1GB, 5GB, 10GB, 20GB, 50GB
- **Recargas de Voz** - Minutos de llamada adicionales
- **Paquetes de SMS** - Paquetes de mensajes
- **Crédito Monetario** - Saldo prepago (£5, £10, £20, £50, £100)

Consulta Recarga y Recarga <features\_topup\_recharge> para flujos de trabajo de recarga detallados.

## Complementos de Servicio

Comprar características y mejoras adicionales para servicios existentes.

## **Complementos Disponibles:**

- **Roaming Internacional** - Habilitar servicio en el extranjero
- **Dirección IP Estática** - IP fija para internet en casa
- **Contenido Premium** - Canales IPTV, servicios de streaming
- **Mejoras de Hardware** - Alquiler de módem, cajas de televisión
- **Aumentos de Velocidad** - Aumentos temporales de ancho de banda

## **Compra de Complementos:**

1. Navegar a **Servicios** → Seleccionar servicio → "**Complementos**"
2. Navegar por los complementos disponibles para este tipo de servicio
3. Hacer clic en "**Agregar al Servicio**" en el complemento deseado
4. Revisar costo (único + recurrente)
5. Confirmar compra
6. Complemento provisionado automáticamente

Consulta Modificación de Servicios - Complementos <csa\_modify> para detalles sobre la gestión de complementos.

## **Notificaciones y Alertas**

Los clientes reciben notificaciones automatizadas para eventos importantes:

### **Notificaciones por Correo Electrónico:**

- Factura generada y lista para pago
- Confirmación de pago recibido
- Advertencias de expiración del servicio (7 días, 3 días, 1 día)
- Alertas de saldo bajo (datos, voz, monetario)
- Activación/desactivación del servicio
- Solicitudes de restablecimiento de contraseña
- Alertas de seguridad (inicio de sesión desde un nuevo dispositivo)

### **Alertas en el Portal:**

- Facturas no pagadas
- Servicios que expiran
- Advertencias de datos bajos (10% restantes)
- Método de pago que expira
- Acciones requeridas (verificar correo electrónico, actualizar método de pago)

### **Preferencias de Notificación:**

Los clientes pueden configurar:

- Frecuencia de notificación por correo electrónico

- Alertas por SMS (si están habilitadas)
- Umbrales de alerta (por ejemplo, notificar cuando <20% de datos restantes)
- Categorías de notificación (facturación, uso, servicio)

## Limitaciones de Autoservicio

Algunas operaciones requieren asistencia del personal:

### Requiere Atención al Cliente:

- Cambiar tipo de cliente (Individual ↔ Negocio)
- Transferir servicios entre clientes
- Cancelar servicios (puede ser autoservicio si está habilitado)
- Disputar facturas
- Solicitar reembolsos
- Cambiar contacto principal
- Problemas complejos de aprovisionamiento

### Restricciones de Seguridad:

- No se puede ver ni modificar cuentas de otros usuarios
- Limitado a los propios datos del cliente (aislamiento de inquilinos)
- No se puede acceder a funciones administrativas
- No se pueden anular facturas o transacciones
- No se puede modificar la configuración del servicio (solo complementos/recargas)

## Acceso de Impersonación de Personal

El personal de soporte puede acceder al Portal de Autocuidado como cliente para resolución de problemas.

### Proceso de Impersonación:

1. Navegar a **Clientes** → Seleccionar cliente
2. Ir a la pestaña **Contactos**
3. Encontrar el contacto vinculado a la cuenta de usuario
4. Hacer clic en el botón "**Iniciar sesión como Usuario**"
5. Se abre una nueva pestaña con la vista del Portal de Autocuidado del cliente
6. Todas las acciones se registran y se atribuyen al usuario impersonado
7. El personal ve exactamente lo que ve el cliente

Iniciar sesión como Usuario - Impersonación

### Casos de Uso:

- **Resolución de Problemas** - Reproducir problemas informados por el cliente
- **Verificación** - Confirmar que las configuraciones de servicio aparecen correctamente
- **Capacitación** - Demostrar características del portal
- **Soporte** - Guiar al cliente a través del portal mientras se ve su pantalla

### **Seguridad y Auditoría:**

- Requiere permiso `can_impersonate` o `admin`
- Todas las acciones se registran en el historial de auditoría
- El cliente ve la impersonación en el registro de actividad
- Tiempo de espera de sesión después de inactividad
- No se puede cambiar la contraseña del cliente mientras se está impersonando

Consulta Atención al Cliente - Impersonación de Usuario <customer\_care> para la documentación completa de impersonación.

## **Responsividad Móvil**

El Portal de Autocuidado es completamente responsive y está optimizado para dispositivos móviles.

### **Características Móviles:**

- Navegación optimizada para toque
- Diseños simplificados para pantallas pequeñas
- Gestos de deslizamiento para navegación en carrusel
- Formularios e inputs amigables para móviles
- Escaneo de códigos QR para aprovisionamiento de eSIM
- Llamadas telefónicas con un toque
- Integración GPS para autocompletar direcciones

### **Aplicación Web Progresiva (PWA):**

- Instalar como aplicación en la pantalla de inicio
- Visualización sin conexión de datos recientes
- Notificaciones push (si están habilitadas)
- Carga rápida con trabajadores de servicio

## **Restablecimiento de Contraseña y Recuperación de Cuenta**

Los clientes pueden restablecer contraseñas olvidadas sin llamar al soporte.

### **Restablecimiento de Contraseña de Autoservicio:**

1. Hacer clic en "**Olvidé mi Contraseña**" en la página de inicio de sesión
2. Ingresar dirección de correo electrónico
3. Recibir correo electrónico de restablecimiento de contraseña (válido por 1 hora)
4. Hacer clic en el enlace en el correo electrónico
5. Ingresar nueva contraseña (debe cumplir con los requisitos de complejidad)
6. Enviar e iniciar sesión con la nueva contraseña

#### **Requisitos de Contraseña:**

- Mínimo 8 caracteres
- Al menos una letra mayúscula
- Al menos una letra minúscula
- Al menos un número
- Al menos un carácter especial (!@#\$%^&\*)

#### **Bloqueo de Cuenta:**

Después de 5 intentos fallidos de inicio de sesión:

- Cuenta bloqueada durante 30 minutos
- Correo electrónico de restablecimiento de contraseña enviado automáticamente
- Notificación de seguridad enviada al correo electrónico registrado

## **Mejores Prácticas para Clientes**

#### **Recomendaciones de Seguridad:**

1. Habilitar 2FA para mayor seguridad
2. Usar una contraseña única y fuerte
3. Mantener la dirección de correo electrónico actual para notificaciones
4. Configurar método de pago predeterminado para renovaciones automáticas
5. Monitorear el uso regularmente para evitar cargos por exceso
6. Guardar códigos de respaldo en un lugar seguro (si 2FA está habilitado)
7. Cerrar sesión después de usar computadoras compartidas/públicas

#### **Gestión de Uso:**

1. Habilitar alertas de saldo bajo (10-20% restantes)
2. Recargar antes de que los saldos expiren para evitar interrupciones en el servicio
3. Revisar facturas mensuales para cargos inesperados
4. Actualizar métodos de pago antes de que las tarjetas expiren
5. Monitorear el uso de datos a lo largo del mes para evitar estrangulamiento

#### **Escalación de Soporte:**

Si el autoservicio no resuelve el problema:

1. Consultar la base de conocimientos / artículos de ayuda (si están disponibles)
2. Revisar el registro de actividad para cambios recientes
3. Contactar al servicio al cliente por teléfono, correo electrónico o chat
4. Proporcionar ID de cliente y detalles del servicio para una resolución más rápida

## **Acceso a la API**

Los clientes con requisitos técnicos pueden usar la API directamente.

### **Generación de Clave API:**

Disponible para clientes comerciales o a solicitud:

1. Navegar a **Cuenta → Acceso a la API**
2. Hacer clic en "**Generar Clave API**"
3. Establecer permisos (solo lectura o lectura-escritura)
4. Establecer fecha de expiración
5. Guardar la clave API de forma segura (se muestra solo una vez)

### **Casos de Uso de la API:**

- Monitoreo automatizado de uso
- Integración con sistemas de facturación internos
- Recargas programáticas
- Aprovisionamiento de servicios a través de scripts
- Exportaciones de datos para análisis

Consulta Documentación de la API <concepts\_api> para detalles de los endpoints y ejemplos.

## **Preguntas Frecuentes**

### **P: ¿Por qué no puedo ver todos mis servicios?**

R: Asegúrate de haber iniciado sesión con la cuenta correcta. Si tienes múltiples cuentas de cliente, cada una tiene servicios separados. Contacta al soporte para fusionar cuentas si es necesario.

### **P: Mi pago falló pero me cobraron. ¿Qué debo hacer?**

R: Revisa tu estado de cuenta bancario en busca de cargos pendientes. Si el cargo aparece pero la factura aún muestra impaga, contacta al soporte con el número de referencia de la transacción.

**P: ¿Cómo cancelo un servicio?**

R: Navega a los detalles del servicio y haz clic en "Cancelar Servicio" (si el autoservicio está habilitado). De lo contrario, contacta al servicio al cliente para procesar la cancelación.

**P: ¿Puedo transferir un servicio a otra persona?**

R: No, las transferencias de servicios requieren asistencia del servicio al cliente por razones de seguridad y cumplimiento.

**P: ¿Por qué mi saldo de datos disminuye más rápido de lo esperado?**

R: Revisa las actualizaciones de aplicaciones en segundo plano, la calidad de streaming de video y las copias de seguridad automáticas en la nube. Revisa el desglose de uso en el portal para un consumo detallado.

**P: Perdí mi dispositivo de 2FA. ¿Cómo recupero el acceso?**

R: Usa los códigos de respaldo para iniciar sesión, luego desactiva y vuelve a activar 2FA. Si no tienes códigos de respaldo, contacta al soporte para un restablecimiento de 2FA (requiere verificación de identidad).

**P: ¿Puedo pagar una factura sin iniciar sesión?**

R: El pago de factura como invitado puede estar disponible a través de un enlace directo de factura. De lo contrario, se requiere inicio de sesión por seguridad.

**P: ¿Cómo descargo todas mis facturas a la vez?**

R: Actualmente se requiere descargar cada factura individualmente. Para descargas masivas, contacta al soporte o usa la API si está disponible.

## Documentación Relacionada

- Atención al Cliente - Impersonación de Usuario <customer\_care> - Acceso de personal para resolución de problemas
- Métodos de Pago <payment\_methods> - Gestión de tarjetas y pagos
- Recarga y Recarga <features\_topup\_recharge> - Compra de crédito adicional
- Uso de Servicio <csa\_service\_usage> - Comprender el seguimiento de saldo
- Resumen de Facturación <billing\_overview> - Conceptos de facturación y facturas
- Flujos de Autenticación <authentication\_flows> - Inicio de sesión y seguridad
- 2FA &lt;2fa> - Configuración de autenticación de dos factores
- Modificaciones de Servicio <csa\_modify> - Agregar características y

complementos