

Integración Pasarela de Pagos Bancard VPOS

Novasis — Módulo de Cobros Electrónicos

Especificación funcional, API reference, modelo de datos y mockups

VPOS · Zimple · Tokenización · QR SIPAP · Webhook · Conciliación

PRODUCTO	Novasis — ERP localizado Paraguay
MÓDULO	Pasarela de Pagos Bancard (Pay-01)
PROVEEDOR	Bancard S.A. — VPOS
VERSIÓN	1.0 — Documento de desarrollo
FECHA	Abril 2026
AUDIENCIA	Equipo de desarrollo backend, frontend y QA
ESTADO	Spec listo para sprint
PRERREQUISITO	Credenciales VPOS (public_key + private_key) habilitadas

Contenido

1. Resumen ejecutivo
2. Contexto · Ubicación del módulo dentro de Novasis
3. Alcance funcional y casos de uso
4. Arquitectura técnica
5. Servicios Bancard VPOS a integrar
6. Generación del token de autenticación (HMAC SHA256)
7. API Reference — Endpoints Bancard
8. Webhooks de confirmación
9. Modelo de datos — Tablas Novasis
10. Configuración del módulo · Parámetros y credenciales
11. Mockups de pantallas y especificación UI
12. Flujos de secuencia
13. Integración con Tesorería, Bancos y Contabilidad
14. Integración con Facturación y SIFEN
15. Seguridad · PCI-DSS · Tokenización
16. Manejo de errores y códigos de respuesta
17. Plan de pruebas y tarjetas de prueba
18. Cronograma de implementación
19. Anexos

1. Resumen ejecutivo

Este documento especifica la integración de Novasis con la pasarela de pagos Bancard VPOS. El objetivo es habilitar el cobro electrónico con tarjeta de crédito y débito (Visa, Mastercard, Cabal, Panal) y con billetera Zimple, de forma online y presencial, desde los módulos de Facturación, Cobros, Créditos, Suscripciones, POS y Panel Cobrador de Novasis. La integración cubre el flujo completo: generación del pago, redirección al portal seguro de Bancard, recepción del resultado vía callback/webhook, registro en Novasis, conciliación automática contra la liquidación bancaria y asiento contable.

El documento está dirigido al equipo de desarrollo. Incluye: (i) casos de uso por módulo, (ii) referencia de endpoints de Bancard con request/response en JSON, (iii) modelo de datos propuesto para Novasis (tablas, índices, campos clave), (iv) diez mockups de pantallas con especificaciones de campos, validaciones y comportamiento, (v) flujos de secuencia end-to-end, (vi) tratamiento de errores y reintentos, y (vii) plan de pruebas con tarjetas de prueba oficiales de Bancard.

OBJETIVO DE LA INTEGRACIÓN

Que el cliente final de una empresa que usa Novasis pueda pagar con tarjeta o Zimple desde un link, un QR, el POS o una suscripción recurrente, y que el pago quede registrado de forma automática en Novasis con el asiento contable correspondiente y conciliado contra la liquidación semanal de Bancard.

1.1 Alcance resumido

- Pago único con tarjeta (single buy) desde link público o iframe embebido.
- Pago con Zimple (billetera de Bancard) por confirmación en celular.
- Tokenización de tarjetas (catastro / card alias) para cobros recurrentes en Suscripciones.
- Cobro con tarjeta presencial en POS y Panel Cobrador vía iframe o redirect.
- QR dinámico SIPAP para cobro presencial sin tarjeta física.
- Rollback (anulación en el mismo día) y refund (devolución).
- Webhook de confirmación autenticado para registrar el resultado aún si el usuario no retorna al comercio.
- Conciliación automática contra el archivo de liquidación bancaria (TXT/CSV Bancard).
- Asiento contable automático con cuentas parametrizables por empresa.
- Reportes de transacciones, comisiones y devoluciones.

2. Contexto — Ubicación del módulo dentro de Novasis

Novasis es un ERP localizado para Paraguay con facturación electrónica SIFEN, contabilidad local, libros fiscales e IPS/MTESS. Los módulos principales hoy son: Dashboard, IA Dashboard, POS, Facturación, Mayoristas, Créditos, Cobros, Panel Cobrador, Productos, Contactos, Compras, Suscripciones, Finanzas, Tesorería, Contabilidad, Reportes y Configuración. La pasarela Bancard VPOS se integra como una capa transversal de cobros, no como un módulo visible autónomo: aparece dentro de los módulos que ya cobran dinero al cliente.

2.1 Módulos que consumen la integración

Módulo Novasis	Caso de uso
Facturación	Link de pago en factura emitida · botón "Pagar con tarjeta" en la vista pública del DTE · iframe embebido al final del flujo de emisión.
Cobros	Cobro de facturas pendientes del cliente · selección de una o más facturas y cobro único con tarjeta · aplicación automática a cuenta corriente.
Créditos	Pago de cuotas de un plan de financiación · pago puntual o agenda de débitos automáticos (con tarjeta tokenizada).
Suscripciones	Cobro recurrente mensual con tarjeta tokenizada · reintentos automáticos ante falla · notificación al cliente si el pago falla.
POS	Cobro presencial con tarjeta en el frente de caja · iframe VPOS en ventana modal · ticket impreso con código de autorización.
Panel Cobrador	Cobrador en campo genera link de pago o QR Zimple para el cliente visitado, confirma el cobro en su dispositivo móvil.
Tesorería	Conciliación del archivo de liquidación Bancard contra los pagos registrados · ingreso del neto al banco recaudador.
Contabilidad	Asiento automático de cobro: Débito Banco Bancard por liquidar / Crédito cliente · luego compensación al ingreso real al banco y gasto por comisión.
Bancos	Movimientos de acreditación de la liquidación bancaria en la cuenta corriente del comercio.
Configuración	Alta de credenciales VPOS por empresa · comisiones por tipo de tarjeta · URLs callback · cuentas contables · medios de pago habilitados.
Reportes	Transacciones por período, por tipo de tarjeta, por módulo origen · tasas de aprobación/rechazo · comisiones totales · devoluciones.

2.2 Principios de diseño de la integración

- Single Source of Truth: el estado real de un pago lo dicta Bancard, no el comercio. Novasis confirma el pago únicamente cuando el webhook lo confirma o una consulta `get_confirmation` lo devuelve como aprobado.
- Idempotencia: cada intento de cobro tiene un `shop_process_id` único. Un webhook repetido no duplica el registro.
- Desacople: la UI nunca toca directamente la API de Bancard. Todo va por el backend de Novasis, que firma el request y mantiene las credenciales.
- Reversibilidad: todo cobro tiene botón de rollback (mismo día) y refund (posterior). El módulo nunca debe dejar un cobro "huérfano" sin estado.
- Multi-empresa: Novasis es multi-tenant. Cada empresa tiene sus propias credenciales VPOS, comisiones, cuentas contables y medios habilitados.

3. Alcance funcional y casos de uso

3.1 Casos de uso principales

CU-01 — Link de pago desde una factura emitida

Tras emitir una factura electrónica, el sistema ofrece al usuario (o al cliente por email/WhatsApp) un link público único que abre una página de pago con los datos de la factura y un botón "Pagar con tarjeta" / "Pagar con Zimple". El cliente completa en el formulario de Bancard. El resultado se registra en Cobros y se aplica automáticamente a la factura.

CU-02 — Cobro desde el módulo de Cobros

Desde la cuenta corriente de un cliente, el cobrador administrativo selecciona una o varias facturas pendientes, elige "Cobrar con tarjeta", el sistema genera el pago, lanza el iframe VPOS, y al aprobar registra el cobro, imputa por FIFO o por selección manual a las facturas, y genera el recibo electrónico.

CU-03 — Cobro recurrente en Suscripciones con tarjeta tokenizada

Al dar de alta una suscripción, el cliente ingresa sus datos de tarjeta una única vez mediante el formulario de catastro (Card Registration). Bancard devuelve un alias y un card_id que Novasis guarda cifrado. En cada vencimiento mensual, Novasis dispara un charge_aliased automático contra ese alias, sin intervención del cliente, y genera la factura electrónica correspondiente.

CU-04 — Cobro presencial en POS

En el frente de caja, el cajero cierra la venta eligiendo "Tarjeta". Se abre un modal con el iframe VPOS y el monto pre-cargado. El cliente ingresa su tarjeta, firma con OTP si corresponde, y al aprobar se imprime el ticket con código de autorización. Si el cliente abandona, el modal permite reintentar o cancelar.

CU-05 — Panel Cobrador en campo con QR

El cobrador visita al cliente, abre Novasis en su celular, selecciona la factura y elige "Generar QR de pago". Se muestra el QR dinámico SIPAP o un link Zimple. El cliente paga desde su app bancaria. Cuando Bancard confirma, el cobrador ve el OK en pantalla en ≤ 5 segundos y emite el recibo.

CU-06 — Rollback (anulación mismo día)

Si un cobro fue erróneo y aún no se liquidó, el usuario con permiso puede anular el cobro. Novasis llama al endpoint rollback de Bancard, y si responde OK, marca el cobro como anulado, revierte el asiento contable y notifica al cliente.

CU-07 — Refund (devolución posterior)

Una devolución total o parcial que se procesa después del día del cobro. Se permite únicamente contra cobros con estado "Confirmado" o "Liquidado" y genera un asiento contable de nota de crédito. Requiere motivo obligatorio y permiso específico.

CU-08 — Conciliación de liquidación bancaria

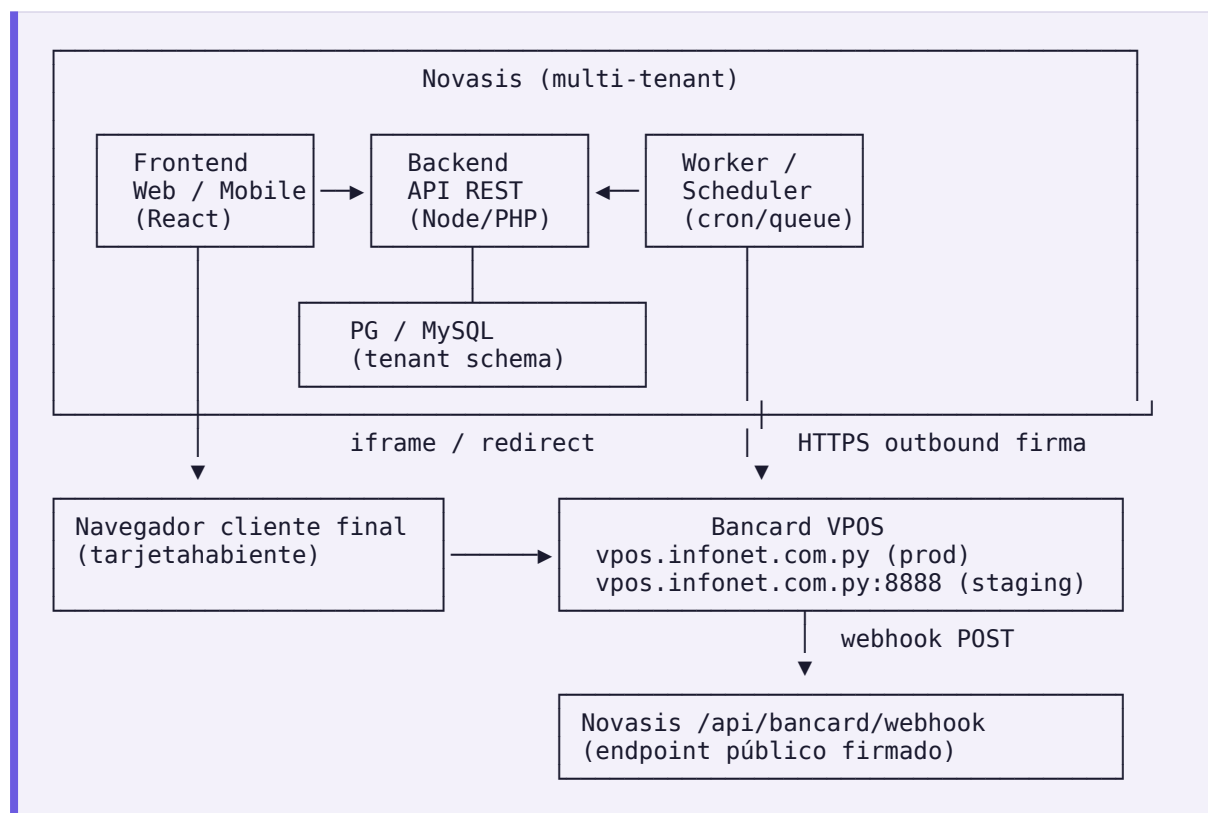
Bancard liquida al comercio el neto (bruto menos comisión) en la cuenta corriente bancaria. El módulo de Tesorería importa el archivo de liquidación (TXT/CSV), hace match por shop_process_id o ticket_number, y cuadra cada transacción. Las diferencias se listan y permiten ajuste manual.

3.2 Fuera de alcance (v1)

- Pagos QR Bancard 2.0 con integración a la red SIPAP extendida más allá del QR dinámico del VPOS (queda para fase 2).
- Split de pagos a múltiples beneficiarios (marketplace). Requiere contrato específico con Bancard.
- Pagos en cuotas sin interés con tarjeta de crédito (requiere acuerdo comercio-emisor).
- Débito automático vía SIPAP tradicional (no VPOS); queda en el roadmap del módulo Bancos.

4. Arquitectura técnica

4.1 Diagrama de componentes (lógico)



4.2 Responsabilidades

Componente	Responsabilidades
Frontend Novasis	Renderizar el iframe o hacer redirect a la URL de Bancard · mostrar resultado al usuario · nunca conoce las credenciales VPOS.
Backend Novasis	Armar request, firmar con HMAC SHA256, llamar a Bancard, persistir la transacción, procesar webhook, encolar reintentos, generar asientos contables.
Worker / Scheduler	Ejecutar cobros recurrentes programados de Suscripciones, reintentos exponenciales en fallas transitorias, conciliación diaria de liquidaciones.
DB (tenant)	Tablas pago_bancard, bancard_log, bancard_card_alias, bancard_conciliacion, parámetros por empresa. Cifrado en reposo para credenciales y alias.
Bancard VPOS	Procesa el pago en su portal seguro · devuelve resultado síncrono al browser y asíncrono al webhook · emite liquidación diaria al banco del comercio.

4.3 Ambientes

Ambiente	URL VPOS	Credenciales
Staging	https://vpos.infonet.com.py:8888/vpos/api/0.3/	Staging keys de Bancard (obtener con Alta de Comercio)
Producción	https://vpos.infonet.com.py/vpos/api/0.3/	Production keys emitidas tras certificación

CERTIFICACIÓN PREVIA A PRODUCCIÓN

Antes de activar credenciales de producción, Bancard exige que el comercio pase por su proceso de certificación (homologación). El equipo técnico de Bancard valida: generación correcta del token, manejo del webhook, idempotencia, rollback, refund y pantallas de resultado. Sin el sello de certificación las keys de producción no se emiten.

4.4 Zonas de red y firewall

- Outbound HTTPS del backend de Novasis hacia los dominios VPOS (443).
- Inbound HTTPS al endpoint /api/bancard/webhook desde los rangos IP de Bancard (documentados; configurar whitelist).
- Sin tráfico directo del navegador del cliente hacia endpoints autenticados — solo al portal seguro VPOS vía redirect/iframe.

5. Servicios Bancard VPOS a integrar

Bancard expone su pasarela como una API REST con JSON. Cada operación sigue el mismo patrón: POST a un endpoint, body con { public_key, operation: { ... } }, el campo operation incluye un token HMAC SHA256 que firma los datos críticos. La respuesta es JSON con status (success/error), messages y datos específicos de la operación.

5.1 Lista de servicios usados por Novasis

Servicio	Endpoint	Uso en Novasis
Single Buy	POST /vpos/api/0.3/single_buy	Pago único con tarjeta (link, iframe, POS).
Zimple Payment	POST /vpos/api/0.3/single_buy/zimple	Pago con billetera Zimple.
Card Registration (Catastro)	POST /vpos/api/0.3/cards/new	Registrar tarjeta y obtener alias_token para cobros futuros.
Charge with Alias	POST /vpos/api/0.3/charge	Cobrar contra tarjeta tokenizada (Suscripciones, Créditos).
Rollback	POST /vpos/api/0.3/single_buy/rollback	Anular pago del mismo día.
Refund	POST /vpos/api/0.3/single_buy/refund	Devolución total o parcial post-liquidación.
Get Confirmation	POST /vpos/api/0.3/single_buy/confirmations	Consultar estado de una transacción por shop_process_id.
List Cards	POST /vpos/api/0.3/users/{user_id}/cards	Listar tarjetas tokenizadas de un usuario.
Delete Card	DELETE /vpos/api/0.3/users/{user_id}/cards	Dar de baja un alias.
QR Dynamic (SIPAP)	POST /vpos/api/0.3/qr	Generar QR SIPAP para cobro presencial.

5.2 Identificadores que Novasis genera y que viajan en la API

Identificador	Generado por	Regla de formato
shop_process_id	Novasis (BIGINT)	Entero único por empresa. Sugerencia:

Identificador	Generado por	Regla de formato
		secuencia por empresa con prefijo de año (p. ej. 26 + 000001234 → 26000001234). Una vez enviado a Bancard, inmutable.
user_id (Bancard)	Novasis	ID interno del cliente/tarjetahabiente. Sugerencia: mismo ID del contacto en Novasis para 1:1.
ticket_number	Bancard	Comprobante devuelto por Bancard al aprobar. Se muestra al usuario y se usa para conciliación.
authorization_number	Bancard	Código de autorización del emisor (6 dígitos). Obligatorio en el ticket.
alias_token	Bancard	Token que representa una tarjeta catastrada. Opaco. Se guarda cifrado.

6. Generación del token de autenticación (HMAC SHA256)

Cada operación crítica se autentica con un campo token dentro del objeto operation. El token es un HMAC SHA256 sobre una cadena que concatena los campos sensibles de la operación, usando la private_key del comercio como clave secreta. La private_key nunca viaja al cliente ni se guarda en el frontend.

6.1 Fórmulas de token por operación

Operación	Cadena a firmar
single_buy	private_key + shop_process_id + amount + currency
rollback	private_key + shop_process_id + "rollback" + "0.00"
refund	private_key + shop_process_id + amount + "PYG"
confirmation	private_key + shop_process_id + "get_confirmation"
card_new	private_key + card_id + user_id + "request_new_card"
charge	private_key + shop_process_id + amount + currency + alias_token
cards_list	private_key + user_id + "request_cards"
card_delete	private_key + card_id + user_id + "delete_card"

6.2 Ejemplo de generación en Node.js

```
const crypto = require("crypto");

function tokenSingleBuy({ privateKey, shopProcessId, amount, currency }) {
  const amountStr = Number(amount).toFixed(2); // siempre 2 decimales
  const raw = `${privateKey}${shopProcessId}${amountStr}${currency}`;
  return crypto.createHash("md5").update(raw).digest("hex");
  // NOTA: Bancard usa MD5 en /vpos/api/0.3 (heredado). Para ambientes
  //        futuros con /vpos/api/1.0 consultar documento vigente de Bancard.
}

// Uso:
const token = tokenSingleBuy({
  privateKey: process.env.BANCARD_PRIVATE_KEY,
  shopProcessId: 26000001234,
  amount: 150000, // Gs.
  currency: "PYG",
});
```

ATENCIÓN: FORMATO DEL AMOUNT

El monto siempre debe firmarse y enviarse con dos decimales exactos y punto como separador (ej. 150000.00). Firmar con 150000 o con coma rompe el token y Bancard responde 401/invalid_token. Implementar una utilería centralizada y no duplicar lógica.

6.3 Custodia de credenciales

- public_key y private_key se guardan en la tabla empresa_config, cifradas con AES-256-GCM usando la llave maestra del entorno (KMS en producción).
- Nunca loguear la private_key, ni en texto plano ni parcial. En logs escribir {private_key: "***"}.
- Rotación: Bancard permite regenerar keys. Al rotar, invalidar sesiones activas y recalcular tokens de alias si aplica.

7. API Reference — Endpoints Bancard

A continuación se detalla cada endpoint con request, response esperado y mapeo a Novasis. Todos los requests son POST con Content-Type: application/json.

7.1 Single Buy (pago único con tarjeta)

Request

```
POST https://vpos.infonet.com.py/vpos/api/0.3/single_buy
Content-Type: application/json

{
  "public_key": "YOUR_PUBLIC_KEY",
  "operation": {
    "token": "HMAC(private_key + shop_process_id + amount + currency)",
    "shop_process_id": 26000001234,
    "amount": "150000.00",
    "currency": "PYG",
    "additional_data": "factura-00001-00000123",
    "description": "Factura 001-001-123 - Farmacia Central SA",
    "return_url": "https://novasis.io/pay/result?tx=26000001234",
    "cancel_url": "https://novasis.io/pay/cancel?tx=26000001234"
  }
}
```

Response (éxito — genera sesión de pago)

```
{
  "status": "success",
  "process_id": "abcd1234-ef56-7890-abcd-1234567890ab" // id de sesión VPOS
}
```

Luego de recibir process_id

Novasis redirige al navegador del cliente (o embebe iframe) a: https://vpos.infonet.com.py/vpos/#/pagos?process_id={process_id}. El cliente completa su tarjeta en Bancard. Al finalizar, Bancard redirige a return_url (éxito) o cancel_url (cancelación) y envía webhook en paralelo.

Mapeo a Novasis

Campo Bancard	Tabla.columna Novasis
shop_process_id	pago_bancard.shop_process_id
amount	pago_bancard.monto
currency	pago_bancard.moneda
additional_data	pago_bancard.referencia_origen (factura, cobro, suscripción)
process_id (session)	pago_bancard.process_id

Campo Bancard	Tabla.columna Novasis
return_url / cancel_url	generadas por backend con el shop_process_id como parámetro firmado

7.2 Zimple Payment

Request

```
POST /vpos/api/0.3/single_buy/zimple

{
  "public_key": "YOUR_PUBLIC_KEY",
  "operation": {
    "token": "HMAC(private_key + shop_process_id + amount + \"PYG\")",
    "shop_process_id": 26000001234,
    "amount": "150000.00",
    "number_cell_phone": "595981123456",
    "description": "Factura 001-001-123"
  }
}
```

Bancard envía una notificación push al celular del usuario. El usuario aprueba desde la app Zimple. Novasis recibe la confirmación por webhook. Si el usuario no responde en 5 minutos, la operación expira.

7.3 Card Registration (Catastro de tarjeta)

Request

```
POST /vpos/api/0.3/cards/new

{
  "public_key": "YOUR_PUBLIC_KEY",
  "operation": {
    "token": "HMAC(private_key + card_id + user_id + \"request_new_card\")",
    "card_id": 987654321,           // id generado por Novasis
    "user_id": 12345,              // id del contacto Novasis
    "user_cell_phone": "595981123456",
    "user_mail": "cliente@ejemplo.com",
    "return_url": "https://novasis.io/cards/result?ref=987654321"
  }
}
```

Response

```
{
  "status": "success",
  "process_id": "reg-abcd-1234"
}
```

Igual que Single Buy, se redirige al usuario al formulario de Bancard. Al finalizar, la tarjeta queda tokenizada y Bancard devuelve por webhook un alias_token asociado al (user_id, card_id).

7.4 Charge con alias (cobro recurrente)

Request

```
POST /vpos/api/0.3/charge

{
  "public_key": "YOUR_PUBLIC_KEY",
  "operation": {
    "token": "HMAC(private_key + shop_process_id + amount + currency +
alias_token)",
    "shop_process_id": 26000001300,
    "amount": "99000.00",
    "currency": "PYG",
    "alias_token": "alias-abcd-xyz-9988",
    "number_of_payments": 1,
    "description": "Suscripción Plan Pro - Abril 2026"
  }
}
```

Response (síncrona)

```
{
  "status": "success",
  "confirmation": {
    "response": "S",
    "response_details": "Transacción aprobada",
    "authorization_number": "123456",
    "ticket_number": "000987654",
    "response_code": "00"
  }
}
```

CHARGE ES SÍNCRONO

A diferencia de single_buy, charge devuelve el resultado en la misma respuesta HTTP. No hay redirección al cliente, ni webhook como único canal. Aún así, el worker debe estar preparado para reintentos y para consultar get_confirmation si la respuesta se pierde por timeout.

7.5 Rollback (anulación mismo día)

```
POST /vpos/api/0.3/single_buy/rollback

{
  "public_key": "YOUR_PUBLIC_KEY",
  "operation": {
```

```
{
  "token": "HMAC(private_key + shop_process_id + \"rollback\" + \"0.00\")",
  "shop_process_id": 26000001234
}
```

Rollback solo aplica antes del cierre del batch (generalmente 23:59 hora Paraguay). Pasada esa ventana, corresponde Refund. El backend de Novasis debe inferir qué operación llamar según fecha del cobro y responder al usuario en consecuencia.

7.6 Refund (devolución)

POST /vpos/api/0.3/single_buy/refund

```
{
  "public_key": "YOUR_PUBLIC_KEY",
  "operation": {
    "token": "HMAC(private_key + shop_process_id + amount + \"PYG\")",
    "shop_process_id": 26000001234,
    "amount": "50000.00",
    "currency": "PYG"
  }
}
```

Refund permite devoluciones parciales. Novasis valida que el acumulado de refunds no supere el monto original. Al aprobarse, genera nota de crédito electrónica SIFEN automáticamente y el asiento contable correspondiente.

7.7 Get Confirmation (consulta de estado)

POST /vpos/api/0.3/single_buy/confirmations

```
{
  "public_key": "YOUR_PUBLIC_KEY",
  "operation": {
    "token": "HMAC(private_key + shop_process_id + \"get_confirmation\")",
    "shop_process_id": 26000001234
  }
}
```

Se usa ante cualquier duda sobre el estado real de una transacción: webhook perdido, timeout en el browser, reintento sospechoso. Debe invocarse antes de reintentar un cobro, para no duplicar.

8. Webhooks de confirmación

Bancard envía un POST al endpoint configurado en el comercio al finalizar una operación asíncrona. Es el canal autoritativo de estado. El endpoint de Novasis es:

```
POST https://{tenant}.novasis.io/api/bancard/webhook
```

8.1 Payload típico

```
{
  "operation": {
    "token": "HMAC(private_key + shop_process_id + amount + currency)",
    "shop_process_id": 26000001234,
    "response": "S", // S = success, N = rejected
    "response_details": "Transacción aprobada",
    "amount": "150000.00",
    "currency": "PYG",
    "authorization_number": "123456",
    "ticket_number": "000987654",
    "response_code": "00",
    "response_description": "APROBADA",
    "extended_response_description": "..."
  }
}
```

8.2 Procesamiento en Novasis

1. Validar el token recibido recalculándolo con la private_key. Si no coincide: responder 401 y no procesar.
2. Buscar pago_bancard por shop_process_id. Si no existe: responder 404 y loguear alerta.
3. Verificar idempotencia: si ya está en estado final (approved/rejected/refunded), responder 200 y no reprocesar.
4. Actualizar estado, ticket_number, authorization_number, response_code, response_description, fecha_confirmacion.
5. Si response = S → disparar post-hooks: aplicar el cobro a la factura, generar recibo electrónico, enviar email/WhatsApp al cliente, publicar evento en bus para el módulo de contabilidad.
6. Si response = N → marcar rechazado, registrar motivo, notificar al usuario para reintento (y al worker si es recurrente).
7. Responder HTTP 200 con body {status:"ok"}.

IDEMPOTENCIA Y REINTENTOS DEL WEBHOOK

Bancard reintenta el webhook si no recibe 200 en < 30 segundos. Hacer el

procesamiento dentro de una transacción de base de datos y responder 200 antes de 30 s. Si hay trabajo pesado (facturación, email), encolarlo y responder 200 primero.

9. Modelo de datos — Tablas Novasis

Se propone crear las siguientes tablas en el schema por tenant. Se usan nombres en singular con prefijo bancard_ para agrupar.

9.1 pago_bancard

Campo	Tipo	Notas
id	BIGSERIAL PK	Identidad interna
empresa_id	INT FK empresa	Multi-tenant
shop_process_id	BIGINT UNIQUE	Enviado a Bancard. Inmutable.
process_id	VARCHAR(64)	ID de sesión VPOS devuelto por Bancard
tipo_operacion	ENUM	single_buy, zimple, charge_alias, refund, rollback, qr
origen_modulo	ENUM	factura, cobro, credito, suscripcion, pos, panel_cobrador
origen_id	BIGINT	FK al documento que originó el cobro
cliente_id	BIGINT FK contacto	Quien paga
monto	DECIMAL(14, 2)	Monto en Gs.
moneda	CHAR(3)	PYG USD
estado	ENUM	pendiente, procesando, aprobado, rechazado, anulado, devuelto_total, devuelto_parcial, expirado
response_code	VARCHAR(8)	Código Bancard. 00 = aprobado
response_description	VARCHAR(128)	Texto Bancard
authorization_number	VARCHAR(16)	Autorización del emisor
ticket_number	VARCHAR(32)	Ticket Bancard para conciliación
alias_token	VARCHAR(128)	Si se usó tarjeta tokenizada
usuario_id	INT FK usuario	Operador Novasis que inició
fecha_creacion	TIMESTAMP	UTC

Campo	Tipo	Notas
fecha_confirmacion	TIMESTAMP	Cuando llegó webhook/respuesta final
monto_devuelto	DECIMAL(14, 2)	Acumulado por refunds parciales
meta_json	JSONB	Request y response completos para auditoría

Índices: (empresa_id, shop_process_id), (empresa_id, estado, fecha_creacion), (ticket_number), (cliente_id, fecha_creacion).

9.2 bancard_card_alias (tarjetas tokenizadas)

Campo	Tipo	Notas
id	BIGSERIAL PK	
empresa_id	INT FK	
cliente_id	BIGINT FK contacto	Dueño de la tarjeta
card_id	BIGINT	ID generado por Novasis y enviado en el request de catastro
alias_token	VARCHAR(128)	CIFRADO en reposo (AES-256-GCM)
brand	ENUM	VISA, MASTERCARD, CABAL, PANAL, BANCARD_CHECK
last4	CHAR(4)	Últimos 4 dígitos (devueltos por Bancard)
expiry_month	SMALLINT	
expiry_year	SMALLINT	
titular	VARCHAR(64)	Nombre del titular
activo	BOOLEAN	false al borrar alias en Bancard
fecha_alta	TIMESTAMP	
fecha_baja	TIMESTAMP	NULL si activo

9.3 bancard_log

Campo	Tipo	Notas
id	BIGSERIAL PK	
pago_bancard_id	BIGINT FK	Puede ser NULL para eventos generales
timestamp	TIMESTAMP	UTC
direccion	ENUM	OUT (Novasis→Bancard), IN (webhook o response), INTERNAL (eventos)
endpoint	VARCHAR(128)	Ruta invocada
http_status	INT	
request_body	JSONB	Con private_key redactada
response_body	JSONB	
latencia_ms	INT	
ip_origen	INET	Para webhooks recibidos

9.4 bancard_conciliacion

Campo	Tipo	Notas
id	BIGSERIAL PK	
empresa_id	INT FK	
fecha_liquidacion	DATE	La que liquida Bancard al comercio
archivo_origen	VARCHAR(256)	Nombre del archivo importado
bruto_total	DECIMAL(14, 2)	
comision_total	DECIMAL(14, 2)	
iva_comision_total	DECIMAL(14, 2)	
neto_total	DECIMAL(14, 2)	
estado	ENUM	importado, cuadrado, con_diferencias, cerrado

Tabla hija bancard_conciliacion_item con el detalle de cada transacción liquidada y su match al pago_bancard.

9.5 bancard_config (por empresa)

Campo	Tipo
ambiente	ENUM: staging produccion
public_key	VARCHAR cifrado
private_key	VARCHAR cifrado
url_vpos	URL base
return_url_base	Ej. https://{tenant}.novasis.io/pay/result
webhook_secret	String adicional para firma custom interna
cuenta_contable_por_liquidar	FK plan_cuentas
cuenta_contable_banco_recaudador	FK plan_cuentas
cuenta_contable_comision	FK plan_cuentas
cuenta_contable_iva_comision	FK plan_cuentas
porcentaje_comision_credito	DECIMAL(5,2) — informativo, el real lo da la liquidación
porcentaje_comision_debito	DECIMAL(5,2) — informativo
medios_habilitados	SET: credit,debit,zimple,q,alias
timeout_segundos	INT — tiempo máximo esperando confirmación

10. Configuración del módulo

Se agrega una entrada nueva al módulo Configuración de Novasis: Configuración → Pasarelas de pago → Bancard. Es una pantalla por empresa.

10.1 Parámetros requeridos

Campo / Control	Tipo	Regla / Validación
Ambiente	select	Staging / Producción. El cambio a producción exige confirmación y queda auditado.
Public Key	input	Pegado desde el panel de Bancard. Validación: 64 caracteres hex.
Private Key	input password	Se ingresa una vez. No se muestra luego (solo último 4). Cifrado AES-256-GCM.
URL Callback Webhook	readonly	La genera Novasis: <code>https://{tenant}.novasis.io/api/bancard/webhook</code> . Debe cargarse en el panel de Bancard.
URL Return	readonly	<code>https://{tenant}.novasis.io/pay/result</code>
URL Cancel	readonly	<code>https://{tenant}.novasis.io/pay/cancel</code>
Medios habilitados	checkboxes	Crédito · Débito · Zimple · QR · Tarjeta tokenizada. Desactivar uno suspende su uso en todas las pantallas.
Cuentas contables	4 selects	Banco por liquidar · Banco recaudador · Comisión · IVA comisión. Validación: no vacías al guardar.
Banco recaudador	select	Cuenta corriente de la tabla bancos Novasis donde Bancard liquida.
Timeout del cliente	number	Minutos que el cliente tiene para completar el pago (default 30).
Email notificación resultados	email list	Para alertas de rechazos atípicos o webhooks fallidos.
Test de conexión	botón	Envía un request dummy de <code>get_confirmation</code> inválido y valida que la respuesta de Bancard tenga la estructura esperada (indica keys bien configuradas).

10.2 Permisos

- PERM_BANCARD_CONFIG — ver y editar configuración.
- PERM_BANCARD_COBRAR — iniciar un cobro desde cualquier módulo.
- PERM_BANCARD_ROLLBACK — anular cobro del día.

- PERM_BANCARD_REFUND — devolver cobro posterior.
- PERM_BANCARD_CONCILIAR — importar y procesar el archivo de liquidación.
- PERM_BANCARD_VER_REPORTES — acceso a la sección de reportes Bancard.

11. Mockups de pantallas y especificación UI

Se presentan diez pantallas, cada una con: (1) el wireframe en formato texto, (2) la tabla de campos, (3) reglas de comportamiento y (4) notas de estilo consistentes con la identidad Novasis (paleta lavanda / púrpura #6B5BE0, botones pill, tarjetas con borde suave).

11.1 Pantalla 1 — Configuración de Bancard

MOCKUP · Configuración > Pasarelas de pago > Bancard

Configuración

● Ambiente [STAGING ▼]

Bancard VPOS

○ Inactivo ● Activo

Credenciales

Public Key
[64a3f8.....]

Private Key
[.....] Rotar

URL Webhook
https://novasis.io/api/bancard/webhook
[Copiar]

URL Return
https://novasis.io/pay/result
[Copiar]

Medios habilitados

☒ Tarjeta Crédito
☒ Tarjeta Débito
☒ Zimple
☒ QR SIPAP
☒ Tarjeta tokenizada (suscripciones)

Cuentas contables

Banco por liquidar
[1.1.02.003 Bancard ... ▼]

Banco recaudador
[1.1.01.008 Itaú CC ... ▼]

Comisión bancaria
[5.2.04.001 ... ▼]

IVA comisión
[1.4.01.002 ... ▼]

[Probar conexión] [Guardar cambios]

Nota para el desarrollador · Al cambiar de Staging a Producción exigir 2FA y registrar evento en auditoría. La Private Key se ingresa una sola vez; luego solo se permite rotar (borrar + reingresar).

Campo / Control	Tipo	Regla / Validación
Ambiente	Select	Solo dos valores. Cambio auditado. Deshabilita el resto hasta confirmar.
Public Key	Text	Hex 64. Validar regex. Permite copiar.
Private Key	Password	Input con máscara. Mostrar solo last-4 tras guardar.
Webhook / Return / Cancel URL	Readonly + Copy	Se generan automáticamente con el subdominio del tenant.

Campo / Control	Tipo	Regla / Validación
Medios habilitados	Multi-check	Deshabilitar un medio oculta el botón en todas las pantallas de pago.
Cuentas contables	4 selects	Autocomplete sobre plan de cuentas. Validar no vacías al guardar.
Probar conexión	Botón	Llama a un endpoint dummy de Bancard, muestra ✓ verde o ✗ rojo con el error.

11.2 Pantalla 2 — Generar link de pago desde Factura

◆ MOCKUP · Facturación › Detalle de Factura › Enviar link de pago

Factura 001-001-000123

Emitida · Aprobada SIFEN ✓

Cliente: Farmacia Central SA

Total: Gs. 1.250.000

[Imprimir] [Enviar PDF] [Anular]

[Cobrar ▼]

Efectivo

Transferencia

● Tarjeta / Zimple (Bancard)

Enviar link de pago

Monto a cobrar

Gs. [1.250.000]

Concepto

[Factura 001-001-000123]

Medios habilitados

☒ Tarjeta
☒ Zimple
☒ QR

Enviar por

☒ Email
☒ WhatsApp
☐ SMS

Destinatario

[ventas@farmaciacentral.com.py]

Caduca en

[24 h ▼]

[Cancelar] [Generar y enviar]

Link público generado:

https://pay.novasis.io/4f8a-9c2d

[Copiar]

Nota para el desarrollador · El link corto mapea internamente al shop_process_id. Una vez pagado, el link retorna a una página de confirmación y no se puede reutilizar.

Campo / Control	Tipo	Regla / Validación
Monto a cobrar	Decimal	Default = saldo pendiente de la factura. Si se modifica a un monto menor, genera cobro parcial.
Concepto	Text	Default = número de factura + cliente. Se envía a Bancard como description.
Medios habilitados	Multi-check	Limita los medios disponibles para este link específico.
Enviar por	Multi-check	Email, WhatsApp (requiere integración WhatsApp Business), SMS.
Destinatario	Input	Prellenado con el email/teléfono del cliente. Editable.

◆ MOCKUP · Facturación › Detalle de Factura › Enviar link de pago		
Caduca en	Select	1h · 24h · 7d · Sin caducidad. Default 24h.
Generar y enviar	Botón	POST /api/pagos/link → crea pago_bancard estado=pendiente y encola envío.

11.3 Pantalla 3 — Vista del cliente final (landing de pago)

◆ MOCKUP · pay.novasis.io/4f8a-9c2d (vista pública)

[Logo empresa]

Pago a ComercioEjemplo S.A.

Factura 001-001-000123

Total a pagar Gs. 1.250.000

Elegí cómo pagar:

Tarjeta crédito/déb

Zimple billetera

QR SIPAP escaneá desde tu banco

Transacción segura procesada por Bancard VPOS
 Tus datos no son almacenados por el comercio

Caduca en 23h 45m Problemas? contacto@novasis.io

Nota para el desarrollador · Página pública accesible sin login. Branding del comercio (logo, color primario) configurado en Configuración › Branding. Al elegir medio, redirect o iframe a Bancard.

11.4 Pantalla 4 — Resultado del pago (return_url)

◆ MOCKUP · novasis.io/pay/result?tx=26000001234 · Éxito

✓

Pago aprobado

Monto	Gs. 1.250.000
Ticket Bancard	000987654
Autorización	123456
Tarjeta	VISA **** 4321
Fecha	22/04/2026 14:32

Factura imputada 001-001-000123

[Descargar comprobante] [Descargar factura]

Enviamos un email con el detalle a ventas@farmaciacentral.com.py

◆ MOCKUP · novasis.io/pay/result?tx=26000001234 · Éxito

Nota para el desarrollador · La pantalla NO decide el estado: pinta lo que el backend ya confirmó vía webhook. Si aún está pendiente al llegar, mostrar "Procesando..." con polling cada 3 s durante 60 s; si expira, mostrar "Estamos confirmando, te avisamos por email" y no bloquear.

11.5 Pantalla 5 — POS, cobro con tarjeta

◆ MOCKUP · POS > Frente de caja > Cerrar venta

Venta #4823		Cajero: María P.		14:37:12	
Producto	Qty	\$	Total	Gs. 450.000	
Remera algodón negra	2	180k	IVA (10%)	Gs. 40.909	
Jean slim azul	1	250k			
Medias blancas	1	20k	Forma de pago		
			[Efectivo]	[Tarjeta ●]	
			[Transf.]	[QR Pagopar]	
			Pasá la tarjeta en el formulario emergente  Esperando confirmación de Bancard... [Cancelar cobro]		
			[Cerrar venta e imprimir]		

Nota para el desarrollador · Al aprobar: cerrar venta, emitir DTE, imprimir ticket con autorización Bancard. Al rechazar: mensaje rojo claro y opción de reintentar o cambiar medio.

11.6 Pantalla 6 — Panel Cobrador con QR

◆ MOCKUP · Panel Cobrador (mobile) > Cobrar con QR Zimple

◀ Cliente

Farmacia San Roque

Saldo: Gs. 1.250.000

Factura 001-001-000123

Monto Gs. 1.250.000

[ Cobrar con QR]



QR Bancard

Válido por 05:00

◆ MOCKUP · Panel Cobrador (mobile) › Cobrar con QR Zimble

● Esperando pago...

[Compartir por WhatsApp
Copiar link]
[Cancelar cobro]

Nota para el desarrollador · Al recibir webhook: vibración + sonido + pantalla verde "Pago recibido". El cobrador confirma e imprime recibo con impresora térmica Bluetooth. Offline-first: si no hay red cuando se genera QR, caer a flujo Zimble por número de celular.

11.7 Pantalla 7 — Catastro de tarjeta para Suscripción

◆ MOCKUP · Suscripciones › Nueva suscripción › Registrar tarjeta

Nueva suscripción · Plan Pro · Gs. 99.000 / mes
Cliente: Clínica Dental Asunción

Paso 3 de 4 · Método de cobro recurrente

☐ Cobrar contra tarjeta existente
☒ Registrar nueva tarjeta

Al continuar, el cliente será redirigido al formulario seguro de Bancard para ingresar los datos de su tarjeta. Novasis NO almacena el número ni el código de seguridad. Solo guarda un token cifrado.

Titular (informativo) [Clínica Dental Asunción]
Celular del titular [595981 123 456]
Email [admin@clinicadental.com.py]

[◀ Volver] [Registrar tarjeta con Bancard ▶]

Nota para el desarrollador · Al continuar: POST /cards/new → redirect a Bancard → tras registrar, webhook con alias_token → guardar en bancard_card_alias (cifrado) → pasar al paso 4 (confirmación).

11.8 Pantalla 8 — Detalle de un pago (vista interna)

◆ MOCKUP · Cobros › Detalle de pago Bancard

Pago Bancard #26000001234 Estado: ● Aprobado

Monto	Gs. 1.250.000	Origen	Factura 001-001-123
Cliente	Farmacia Central SA	Módulo	Cobros
Medio	VISA crédito **** 4321	Creado	22/04/26 14:32
Autorización	123456	Confirmado	22/04/26 14:33
Ticket	000987654	Ambiente	Producción

Timeline

- 14:32:01 Cobro iniciado por María P.
- 14:32:03 Request enviado a Bancard
- 14:32:45 Cliente completó formulario
- 14:33:02 Webhook recibido · response_code=00

◆ MOCKUP · Cobros › Detalle de pago Bancard

- 14:33:02 Factura imputada · recibo 001-001-000987 emitido
- 14:33:02 Asiento contable #12345 generado

Acciones

[Ver recibo] [Ver asiento] [Reenviar comprobante]
 [Anular (rollback)] [Devolver (refund)] [Descargar JSON]

Nota para el desarrollador · Los botones Anular/Devolver se muestran según estado y ventana temporal. Rollback solo si fecha_confirmacion = hoy y antes del cierre de batch. Refund solo si estado=aprobado o liquidado.

11.9 Pantalla 9 — Conciliación en Tesorería

◆ MOCKUP · Tesorería › Conciliaciones › Bancard · 20/04/2026

Archivo liquidación: liquidacion_20260420.txt [Re-importar]
 Bruto Gs. 18.450.000 · Comisión Gs. 553.500 · Neto Gs. 17.896.500
 Estado: ● Con diferencias (2 transacciones sin match)

Transacciones liquidadas

Ticket	Fecha	Monto	Match Novasis	Estado
000987654	22/04/26	Gs. 1.250.000	pago #...1234	✓ Cuadrado
000987655	22/04/26	Gs. 450.000	pago #...1235	✓ Cuadrado
000987656	22/04/26	Gs. 899.000	pago #...1236	✓ Cuadrado
000987657	22/04/26	Gs. 150.000	—	△ Sin match
000987658	22/04/26	Gs. 89.000	—	△ Sin match

Transacciones Novasis aprobadas sin entrada en liquidación 0

[Buscar match manual] [Generar asiento neto] [Cerrar día]

Nota para el desarrollador · El match automático se hace por ticket_number. Las diferencias se listan y admiten match manual. El asiento neto registra ingreso al banco recaudador + gasto comisión + IVA.

11.10 Pantalla 10 — Reporte de transacciones Bancard

◆ MOCKUP · Reportes › Bancard › Transacciones

Filtros: [Desde 01/04] [Hasta 22/04] [Estado: Todos ▼]
 [Medio: Todos ▼] [Módulo: Todos ▼] [Aplicar] [Excel]

KPIs

Transacciones	1.284	
Aprobadas	1.213	(94.5%)
Rechazadas	53	(4.1%)
Expiradas	18	(1.4%)
Monto bruto	Gs. 234.500.000	
Comisión	Gs. 7.035.000	(3.0%)
Refunds	Gs. 1.200.000	

Tabla

Fecha	SPID	Cliente	Medio	Monto	Estado
22/04	26-...234	Farmacia Central	VISA	1.250.000	● Aprob.
22/04	26-...235	Clínica Dental	MASTER	99.000	● Aprob.

◆ MOCKUP · Reportes › Bancard › Transacciones

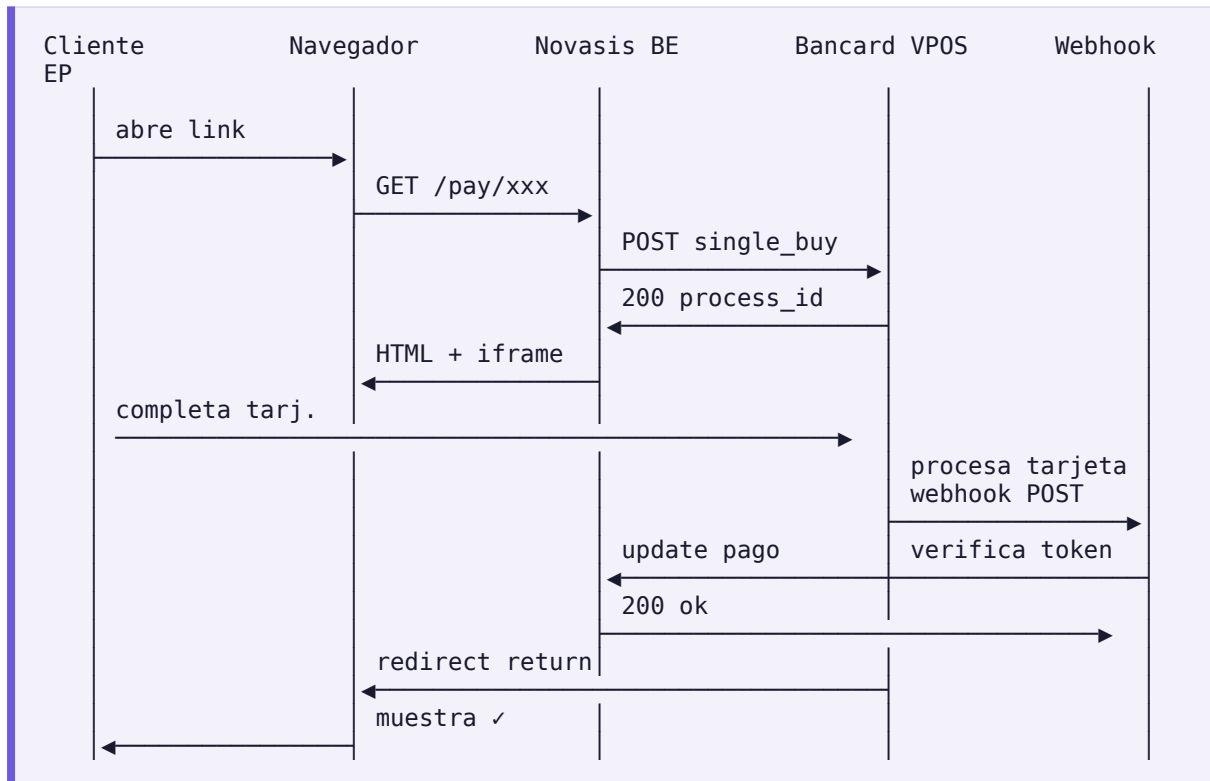
22/04	26-...236	Ferretería Sur	ZIMPLE	450.000	● Aprob.
22/04	26-...237	Librería Luna	VISA	30.000	x Rech.

Mostrando 50 de 1.284 · [◀] [1] 2 3 ... 26 [▶]

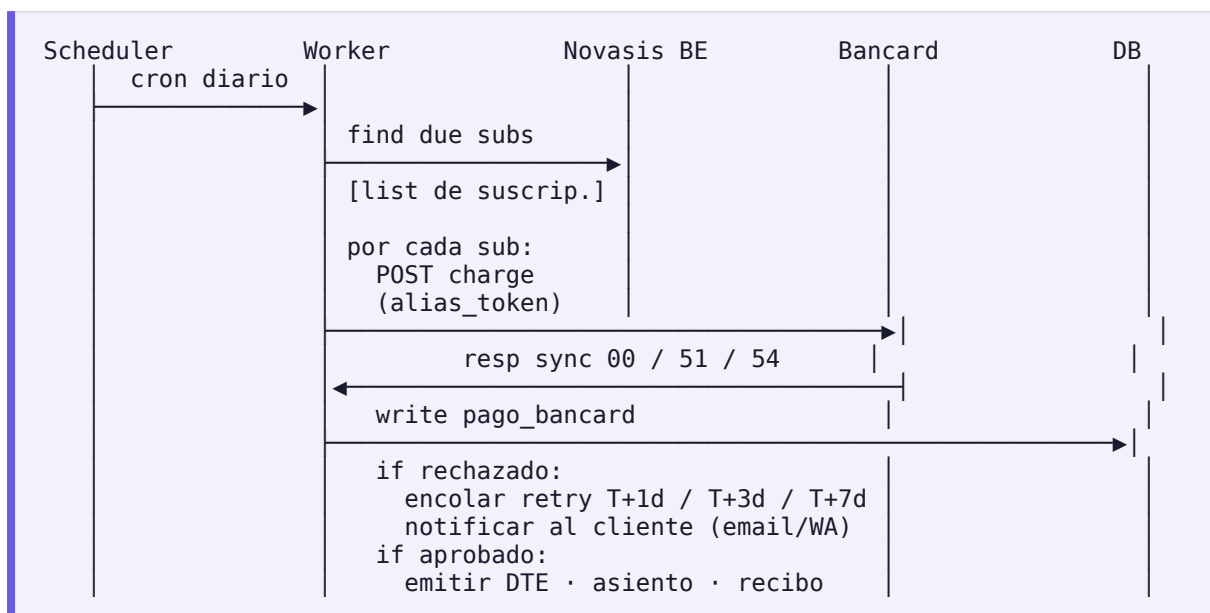
Nota para el desarrollador · Exportable a Excel con las columnas visibles y las ocultas (response_code, response_description, authorization, IP). Drilldown por clic en SPID abre pantalla 8.

12. Flujos de secuencia

12.1 Pago único desde link (CU-01)



12.2 Cobro recurrente de Suscripción (CU-03)



12.3 Anulación (Rollback) vs Devolución (Refund)

Criterio	Rollback	Refund
Ventana temporal	Mismo día hábil, antes del cierre de batch (aprox. 23:59)	Cualquier momento posterior a la liquidación
Efecto contable	Revierde el asiento original	Genera asiento de nota de crédito
Efecto SIFEN	No genera DTE nuevo · anula el recibo electrónico	Genera nota de crédito electrónica
Costo para el comercio	Sin comisión (operación no llegó a liquidarse)	La comisión original no se devuelve
Monto	Siempre total	Total o parcial
Endpoint	POST /single_buy/rollback	POST /single_buy/refund
Permiso	PERM_BANCARD_ROLLBACK	PERM_BANCARD_REFUND

13. Integración con Tesorería, Bancos y Contabilidad

13.1 Asiento contable al confirmar un cobro

Al recibir el webhook de aprobación, Novasis genera el siguiente asiento automático:

Cuenta	Debe	Haber
1.1.02.003 Bancard por liquidar	1.250.000	
1.1.03.001 Clientes - Farmacia Central SA		1.250.000

13.2 Asiento al liquidar la cuenta en el banco

Al conciliar el archivo de liquidación (bruto – comisión – IVA comisión = neto acreditado al banco):

Cuenta	Debe	Haber
1.1.01.008 Banco Itaú CC	17.896.500	
5.2.04.001 Comisión bancaria Bancard	503.182	
1.4.01.002 IVA Crédito s/ comisión	50.318	
1.1.02.003 Bancard por liquidar		18.450.000

13.3 Asiento al refund

Nota de crédito electrónica + asiento de reversión:

Cuenta	Debe	Haber
4.1.01.001 Ventas (o cuenta original)	Monto neto	
2.1.01.004 IVA Débito fiscal	IVA 10%	
1.1.01.008 Banco Itaú CC (o Bancard por liquidar)		Total devuelto

13.4 Reglas contables del módulo

- Cada asiento referencia al shop_process_id y al documento origen (factura, suscripción, contrato de crédito).

- Las cuentas de comisión, IVA comisión, banco por liquidar y banco recaudador son parametrizables por empresa.
- El asiento de liquidación se puede hacer detallado (uno por transacción) o neteado (uno total por día), configurable.
- En multi-moneda, cobros en USD (si aplica) registran diferencia de cambio al momento del neteo con el cotización del día.

14. Integración con Facturación y SIFEN

Los cobros con Bancard tienen efectos documentales sobre facturación electrónica SIFEN. Se documentan los mapeos.

14.1 Medio de pago en el DTE

- En la factura electrónica, el elemento <gPagCred> (crédito) o <gPagTarCD> (tarjeta) se llena según el medio efectivo usado.
- Para tarjeta: tipo=1(Visa), 2(Mastercard), 3(Cabal), 4(Amex—no aplica), etc. Número últimos 4, código de autorización.
- Para Zimple: medio="Billetera electrónica", referencia=ticket_number Bancard.

14.2 Recibo electrónico

Al aprobarse el cobro desde Facturación o Cobros, el sistema emite automáticamente un Recibo Electrónico (si la empresa lo usa) con el mismo medio y autorización, vinculado al shop_process_id.

14.3 Nota de crédito electrónica por refund

Cada refund (total o parcial) dispara la emisión de una Nota de Crédito Electrónica que referencia el DTE original. El motivo es obligatorio y se copia al campo motivo de la NCE.

14.4 Reimpresión de comprobantes

El comprobante de Bancard (ticket de pago) se almacena en formato PDF y se puede reimprimir desde el detalle del pago. Queda como anexo al DTE.

15. Seguridad · PCI-DSS · Tokenización

15.1 Alcance PCI del comercio

Al usar iframe/redirect de Bancard, Novasis queda en alcance SAQ A (el más simple). Nunca se tocan datos PAN en el backend. Si en alguna fase se decidiera capturar el PAN en el frontend de Novasis, el alcance sube a SAQ A-EP y requiere certificación. Se recomienda mantener SAQ A.

15.2 Checklist de seguridad

- TLS 1.2+ en todos los endpoints expuestos.
- Certificate pinning en el cliente móvil para el endpoint VPOS.
- Rate limiting en /api/bancard/webhook (máx 50 req/s por IP).
- IP whitelist para el webhook con los rangos de Bancard.
- Validación de firma HMAC en cada webhook recibido.
- Cifrado en reposo de public_key, private_key y alias_token con AES-256-GCM.
- Llaves de cifrado en KMS (AWS KMS, GCP KMS o HashiCorp Vault). Nunca en .env texto plano en producción.
- Logs no incluyen la private_key, el PAN ni el CVV (no aplican) ni el alias_token completo — solo last-4 del alias.
- Auditoría: cada cambio de credenciales o cambio de ambiente queda registrado con usuario, IP y timestamp.
- 2FA obligatorio para usuarios con PERM_BANCARD_CONFIG, PERM_BANCARD_REFUND y PERM_BANCARD_CONCILIAR.

15.3 Ley 1682/01 y Ley 6534/20 (protección de datos Paraguay)

El titular del dato debe estar informado de que sus datos de tarjeta se comunican a Bancard para procesar el pago. El aviso se muestra en la landing pública de pago (pantalla 3). El comercio es responsable de tener política de privacidad accesible.

16. Manejo de errores y códigos de respuesta

16.1 Códigos response_code más frecuentes

Código	Significado	Acción en Novasis
00	Aprobada	Estado=aprobado. Ejecutar post-hooks.
01	Referir al emisor	Estado=rechazado. Sugerir al cliente contactar al banco. Reintento manual.
05	No autorizar	Estado=rechazado. No reintentar automático.
12	Transacción inválida	Log severidad alta. Normalmente issue de config o token.
14	Tarjeta inexistente	Rechazado. No reintentar.
41	Tarjeta perdida	Rechazado. Marcar alias como inválido si aplica.
43	Tarjeta robada	Rechazado. Desactivar alias.
51	Fondos insuficientes	Rechazado. Reintento programado T+3d en suscripciones.
54	Tarjeta vencida	Rechazado. Marcar alias como expirado. Solicitar recatastro.
57	Transacción no permitida	Rechazado. No reintentar auto. Revisar si la tarjeta es crédito vs débito y medios habilitados.
61	Excede límite del titular	Rechazado. Reintento manual / menor monto.
65	Excede frecuencia	Rechazado. Reintento T+24h.
91	Emisor no disponible	Timeout en red. Reintento inmediato (exponencial).
96	Error del sistema	Timeout. Reintento exponencial. Si persiste > 15 min: alerta a equipo.

16.2 Errores HTTP de Bancard

HTTP	Causa típica
400	JSON inválido, campo faltante, monto con formato incorrecto.

HTTP	Causa típica
401	Token HMAC no coincide. Revisar concatenación y orden de campos.
403	Public key inválida o ambiente no habilitado.
404	Endpoint mal escrito o shop_process_id inexistente (en get_confirmation).
409	shop_process_id duplicado (ya procesado).
500/502/504	Indisponibilidad de Bancard. Reintento exponencial.

16.3 Política de reintentos

- Reintentos de red (timeout, 5xx): backoff exponencial 2s · 4s · 8s · 16s · 32s · abandonar y marcar error_transitorio.
- Reintentos de negocio (51 fondos, 65 frecuencia, 91 emisor): programados a T+1d / T+3d / T+7d. Al 4º fallo se suspende.
- Antes de reintentar, siempre llamar a get_confirmation para evitar doble cobro por respuesta perdida.
- El usuario ve en la UI el estado textual: "Pendiente de confirmación", "Aprobado", "Rechazado (motivo)", "En reintento automático".

17. Plan de pruebas y tarjetas de prueba

17.1 Tarjetas de prueba oficiales Bancard (staging)

Tarjeta	Número	Resultado
VISA aprobación	4012 0010 3844 0005	APROBADA (00)
MASTERCARD aprobación	5204 7300 0200 0003	APROBADA (00)
VISA rechazo fondos	4012 0010 3700 0005	FONDOS INSUFICIENTES (51)
VISA tarjeta vencida	4012 0010 3800 0005	TARJETA VENCIDA (54)
MASTERCARD no autorizar	5204 7300 0300 0003	NO AUTORIZAR (05)
Cabal aprobación	6042 0330 0200 0004	APROBADA (00)

CVV: 123 · Fecha exp: cualquiera futura. Los nombres y direcciones son libres.

17.2 Casos de prueba mínimos

ID	Caso	Resultado esperado
T01	Pago único con VISA aprobación · link desde factura	Webhook 200 · estado aprobado · recibo emitido · asiento generado
T02	Pago con VISA fondos insuficientes	Estado rechazado (51) · factura queda impaga · UI muestra motivo
T03	Pago Zimple con número inexistente	Error en el envío · UI muestra advertencia
T04	Webhook con token inválido	Responde 401 · no actualiza estado · alerta en logs
T05	Webhook duplicado (mismo shop_process_id)	Segunda llamada responde 200 sin reprocesar (idempotente)
T06	Rollback el mismo día · antes del cierre	Estado anulado · asiento revertido
T07	Rollback al día siguiente	Error · UI ofrece Refund
T08	Refund parcial de 30%	Estado devuelto_parcial · NCE emitida ·

ID	Caso	Resultado esperado
		monto_devuelto actualizado
T09	Refund acumulado supera original	Error · no permite · UI muestra monto máximo
T10	Catastro de tarjeta · luego charge con alias	Alias guardado cifrado · charge devuelve 00 · pago registrado
T11	Charge con alias de tarjeta vencida	Devuelve 54 · alias marcado como expirado · notificación al cliente
T12	Scheduler de suscripción falla por fondos · reintento T+3d OK	Dos pagos_bancard registrados · el segundo aprueba
T13	Conciliación con 2 transacciones sin match	Estado con_diferencias · UI permite match manual
T14	Cambio de Staging a Producción sin credenciales válidas	Bloqueado · UI exige credenciales de producción antes de guardar
T15	Timeout de red al llamar single_buy	Reintento exponencial · si persiste, estado error_transitorio · UI permite reintentar
T16	Monto firmado sin 2 decimales	Bancard responde 401 · se captura y se reporta como bug de implementación

17.3 Pruebas de carga

- Pico simulado: 200 cobros concurrentes en POS. Verificar que backend no cuellos en el endpoint firmador.
- Worker de suscripciones: 5.000 charges encolados. Verificar no saturar la API de Bancard (respetar rate limits).
- Webhook: 500 POST simultáneos con payload válido. Verificar respuesta < 2s promedio.

18. Cronograma de implementación

Estimación basada en equipo de 2 backend + 1 frontend + 1 QA dedicados, con el lead ya certificado en el stack Novasis.

Fase	Alcance	Duración	Entregable
F1	Setup · Credenciales staging · tabla pago_bancard · Configuración UI · test conexión	5 días	Pantalla 1 viva
F2	Single Buy · link de pago · landing pública · pantalla resultado · webhook con firma	10 días	CU-01 end-to-end en staging
F3	Cobro desde módulo Cobros · imputación a facturas · recibo electrónico	6 días	CU-02
F4	POS con iframe · ticket con autorización · cancelación	6 días	CU-04
F5	Panel Cobrador móvil con QR Zimple · offline-first	7 días	CU-05
F6	Catastro + Charge alias · Suscripciones recurrentes · reintentos	10 días	CU-03
F7	Rollback y Refund · notas de crédito SIFEN · permisos	5 días	CU-06, CU-07
F8	Conciliación · importador liquidación · asientos automáticos · reportes	7 días	CU-08
F9	Hardening · seguridad · pruebas · certificación Bancard · go-live producción	10 días	Producción

Total: aproximadamente 66 días hábiles (13–14 semanas). La certificación con Bancard puede tardar 2–3 semanas adicionales, se solapa con F8–F9.

18.2 Riesgos

- Demora en la entrega de credenciales por Bancard — impacta F1. Mitigación: solicitar alta del comercio antes de arrancar desarrollo.
- Cambio de versión de la API VPOS durante el desarrollo — mitigación: implementar un adapter por endpoint para aislar versiones.
- Incompatibilidad del formato de archivo de liquidación con el importador — mitigación: obtener muestra real en F1 y testear.
- Concurrencia en webhook y worker — mitigación: uso de locks a nivel de shop_process_id y jobs con exactly-once.

19. Anexos

19.1 Glosario

Término	Definición
VPOS	Virtual Point of Sale · pasarela de pagos online de Bancard.
shop_process_id	Identificador único del proceso de pago generado por el comercio (Novasis).
process_id	Identificador de sesión del formulario VPOS, devuelto por Bancard.
ticket_number	Comprobante interno Bancard, usado para conciliación.
authorization_number	Código de autorización del emisor de la tarjeta.
alias_token	Token opaco que representa una tarjeta tokenizada.
Rollback	Anulación del pago antes del cierre del batch diario.
Refund	Devolución posterior a la liquidación, total o parcial.
Catastro	Proceso de registro de una tarjeta para futuros cobros sin reingresar datos.
Webhook	Notificación HTTP asíncrona de Bancard al comercio para informar resultado de una operación.
SIPAP	Sistema de Pagos del Paraguay · red de transferencias interbancarias.
DTE	Documento Tributario Electrónico — factura, nota, etc. bajo SIFEN.
SIFEN	Sistema Integrado de Facturación Electrónica Nacional (SET Paraguay).
Tenant	Empresa cliente de Novasis en el multi-tenant.

19.2 URLs y recursos

- Portal de desarrolladores Bancard: <https://vpos.infonet.com.py/docs>
- Panel del comercio (alta credenciales): provisto por Bancard tras firma del contrato
- Ambiente staging: <https://vpos.infonet.com.py:8888/vpos/api/0.3/>
- Soporte técnico Bancard: soporte.vpos@bancard.com.py (ejemplo — confirmar en contrato)

19.3 Checklist de entrega

-
- Configuración por empresa completa y funcional.
 - 10 pantallas implementadas según mockups y aprobadas por producto.
 - Todos los 16 casos de prueba T01–T16 verdes en QA.
 - Documentación operativa (runbook) para soporte N1.
 - Alerta en Grafana/Datadog para webhooks fallidos > 5 en 10 minutos.
 - Dashboard con KPIs Bancard en el módulo de Reportes.
 - Sello de certificación Bancard obtenido antes del pase a producción.
 - Política de retención de logs: 7 años para pagos, 1 año para bancard_log técnicos.