



Guía de Integración — API de Pagos Recurrentes

Esta guía describe cómo integrarse con la API de Pagos Recurrentes (Recurring Payments) para gestionar planes de suscripción, suscripciones, cobros, clientes y métodos de pago.

Contenido

Guía de Integración — API de Pagos Recurrentes	1
1. Introducción	1
2. Autenticación	1
3. Convenciones generales	3
4. Plans	3
5. Subscriptions	8
6. Charges	9
7. Customers	11
8. Payment Methods	12
9. Manejo de errores	13
10. Resumen de endpoints	13

1. Introducción

La API expone cinco recursos principales:

- **Plans:** planes de suscripción (precio, frecuencia de facturación, ciclos).
- **Subscriptions:** suscripciones de un cliente a un plan.
- **Charges:** cobros generados por una suscripción en cada ciclo de facturación.
- **Customers:** clientes y sus métodos de pago asociados.
- **Payment Methods:** métodos de pago individuales.

La URL base de todos los endpoints es:

```
BASE_URL = https://api.geopagos.io
```

```
https://api.geopagos.io/api/v1.0/plans
https://api.geopagos.io/api/v1.0/subscriptions
https://api.geopagos.io/api/v1.0/charges
https://api.geopagos.io/api/v1.0/customers
https://api.geopagos.io/api/v1.0/payment-methods
```

2. Autenticación

La API utiliza el flujo OAuth 2.0 `client_credentials` contra el Identity Provider (Keycloak) de GeoPagos.

2.1 Credenciales necesarias

Credencial	Descripción
client_id	Identificador OAuth del cliente/integrador. Se solicita al equipo de Sipago.
client_secret	Secreto OAuth del cliente/integrador. Se solicita al equipo de Sipago.
TENANT_IDENTIFIER	UUID del realm, necesario para el path del endpoint de token. Valor fijo y compartido por todos los integradores: 5662e892-59f2-4a54-aea8-cfbf77942843.

Importante: las credenciales son sensibles. Deben almacenarse en un vault/secret manager y nunca versionarse en repositorios ni exponerse en clientes front-end.

2.2 Obtener el access token

```
POST https://idp.prd.geopagos.io/auth/realms/{TENANT_IDENTIFIER}/protocol/openid-connect/token
Content-Type: application/x-www-form-urlencoded
```

```
curl --location 'https://idp.prd.geopagos.io/auth/realms/{TENANT_IDENTIFIER}/protocol/openid-connect/token' \
--header 'Content-Type: application/x-www-form-urlencoded' \
--data-urlencode 'client_id={CLIENT_ID}' \
--data-urlencode 'client_secret={CLIENT_SECRET}' \
--data-urlencode 'grant_type=client_credentials' \
--data-urlencode 'scope=openid'
```

Response 200 OK

```
{
  "access_token": "eyJhbGciOiJIUzI1NiIs... ",
  "expires_in": 900,
  "refresh_expires_in": 0,
  "token_type": "Bearer",
  "id_token": "eyJhbGciOiJIUzI1NiIs... ",
  "not-before-policy": 0,
  "scope": "openid platform merchant",
  "tid": "{TENANT_IDENTIFIER}"
}
```

2.3 Uso del token

El access_token obtenido se envía en cada request contra <https://api.geopagos.io> mediante el header:

```
Authorization: Bearer <access_token>
```

En las peticiones que envían body (POST, PUT, DELETE con body) se debe agregar además:

```
Content-Type: application/json
```

Si el header Authorization no está presente o el token es inválido, la API responde con un error de autenticación (ver [sección 9](#)).

3. Convenciones generales

Paginación

Los endpoints de listado (GET /plans, GET /subscriptions, GET /charges, GET /customers) aceptan limit y offset como parámetros de query y devuelven la información de paginación junto con los datos.

El valor máximo de limit es 100. Valores mayores deben rechazarse o serán truncados por la API.

Según el recurso, la respuesta puede venir como:

```
{ "data": [ ... ], "total": 0, "limit": 20, "offset": 0 }
```

o

```
{ "data": [ ... ], "pagination": { "total": 0, "limit": 20, "offset": 0 } }
```

Casing de parámetros y campos

Importante: la casing de query params y campos de body no es uniforme entre recursos. Verificar la tabla de cada endpoint antes de integrar:

Recurso	Query params de filtro	Campos de body
Plans	camelCase (pricingType, billingFrequency, externalReference)	snake_case (pricing_type, billing_frequency, merchant_id)
Subscriptions	camelCase (planId, customerId, createdAfter, createdBefore)	snake_case (pause_until, cancel_at_period_end, payment_method_id)
Charges	snake_case (subscription_id, created_after, created_before)	—
Customers	camelCase (createdAfter)	—

Manejo de errores

Ver [sección 9](#).

4. Plans

4.1 Conceptos de planes

Antes de crear o actualizar planes, tené en cuenta el siguiente comportamiento del recurso.

Información del comercio

Para crear planes es necesario contar con merchant_name y merchant_id. Estos valores se obtienen ingresando a <https://portal.sipago.coop/> con las credenciales del comercio y consultando la sección **Perfil**, donde se muestra la información del comercio:

Información del Comercio

Comercio:

Comercio Demo SRL

ID Sucursal:

10001

ID Merchant:

999

Nombre del Merchant:

comercio demo

merchant_name corresponde al campo **Nombre del Merchant** del portal; merchant_id corresponde al campo **ID Merchant**. Los demás campos (**Comercio, ID Sucursal**) no se usan en el body de creación de planes.

Tipos de precio (pricing_type)

Tipo	Comportamiento	amount
fixed	Todos los suscriptores pagan la misma cuota.	Obligatorio al crear el plan. Se puede modificar después (ver 4.4 Actualizar plan).
dynamic	No hay un monto inicial: cada suscripción se cobra por un monto que se define individualmente.	No aplica al crear el plan; se define por suscripción enviando next_amount en 5.3 Actualizar suscripción .

Día de cobro (billing_day)

El comportamiento depende de billing_frequency:

- **monthly** — billing_day es el día del mes (1-31). Si se omite, el día de cobro queda fijado al día en que cada usuario se suscribe (ej.: alguien que se suscribe el 23 paga el 23 de cada mes en adelante).
- **weekly** — billing_day representa un día de la semana. Si se omite, el día de cobro queda fijado al día de la semana en que cada usuario se suscribe.

billing_day	Día
1	Lunes
2	Martes
3	Miércoles
4	Jueves
5	Viernes
6	Sábado
7	Domingo

- **bimonthly, quarterly, yearly** — billing_day y billing_month son obligatorios.

Prorrateso (prorate)

Solo aplica a planes con billing_day definido.

- `prorate: true` — la primera cuota se cobra proporcional al momento de la suscripción; desde el siguiente ciclo se cobra la cuota completa. Ej.: un plan que cobra el día 1 de cada mes, si alguien se suscribe el 15, paga la mitad de la cuota ese día y la cuota completa el 1 del mes siguiente.
- `prorate: false` — se cobra la cuota completa sin importar el día de suscripción.

Duración (`billing_cycles`)

- Con `billing_cycles` — el plan tiene una cantidad fija de cobros.
- Sin `billing_cycles` — el plan es indefinido, hasta que la suscripción se cancele.

4.2 Listar planes

```
GET https://api.geopagos.io/api/v1.0/plans
```

Query params

Parámetro	Tipo	Requerido	Descripción
<code>limit</code>	number	Sí	Cantidad de resultados por página
<code>offset</code>	number	Sí	Desplazamiento para paginación
<code>status</code>	string	No	Filtra por estado del plan
<code>pricingType</code>	string	No	<code>fixed</code> o <code>dynamic</code>
<code>billingFrequency</code>	string	No	Frecuencia de facturación (ver 4.3 Crear plan)
<code>currency</code>	string	No	Código de moneda
<code>search</code>	string	No	Búsqueda por texto
<code>amount</code>	number	No	Filtra por monto
<code>externalReference</code>	string	No	Referencia externa del plan

Ejemplo de request

```
curl -X GET "https://api.geopagos.io/api/v1.0/plans?limit=20&offset=0&status=active" \
-H "Authorization: Bearer <token>"
```

Response 200

```
{
  "data": [
    {
      "plan_id": "plan_123",
      "subscription_url": "https://...",
      "name": "Plan Mensual",
      "amount": 1000,
      "currency": "ARS",
      "billing_frequency": "monthly",
      "status": "active",
      "created_at": "2026-01-01T00:00:00Z",
      "updated_at": "2026-01-01T00:00:00Z"
    }
  ],
  "total": 1,
  "limit": 20,
  "offset": 0
}
```

4.3 Crear plan

POST https://api.geopagos.io/api/v1.0/plans

Body — application/json

Campo	Tipo	Requerido	Notas
name	string	Sí	—
description	string	No	—
pricing_type	string	Sí	Valores: fixed, dynamic
amount	number	Condicional	Requerido si pricing_type es fixed
currency	string	Sí	—
billing_frequency	string	Sí	Valores: weekly, monthly, bimonthly, quarterly, yearly
billing_cycles	number	No	Cantidad total de ciclos de facturación
billing_day	number	No	Día del mes/semana en que se factura
prorate	boolean	No	—
billing_month	number	No	—
trial_period_days	number	No	—
status	string	No	Valores: active, inactive
redirect_url	string	No	—
external_reference	string	No	—
merchant_name	string	Sí	—
merchant_fiscal_id	string	No	—
merchant_id	number	Sí	—
image_content	string	No	Imagen en base64
image_file_name	string	No	—
background_color	string	No	—

Ejemplo de request

```
curl -X POST "https://api.geopagos.io/api/v1.0/plans" \
-H "Authorization: Bearer <token>" \
-H "Content-Type: application/json" \
-d '{
  "name": "Plan Mensual",
  "pricing_type": "fixed",
  "amount": 1000,
  "currency": "ARS",
  "billing_frequency": "monthly",
  "merchant_name": "Mi Comercio",
  "merchant_id": 456
}'
```

Response 2xx — mismo esquema que un ítem de GET /plans.

4.4 Actualizar plan

```
PUT https://api.geopagos.io/api/v1.0/plans/{planId}
```

Body — todos los campos opcionales

name, description, amount, currency, billing_frequency, billing_cycles, trial_period_days, status ("active"|"inactive"), redirect_url, external_reference, merchant_name, merchant_fiscal_id.

Response 200 — mismo esquema que GET /plans/{planId}.

4.5 Desactivar plan

No existe un DELETE para planes. Para desactivar un plan se usa el mismo endpoint de actualización enviando el estado inactive:

```
PUT https://api.geopagos.io/api/v1.0/plans/{planId}
```

```
{ "status": "inactive" }
```

Response 200 — mismo esquema que GET /plans/{planId}.

Importante: cuando un plan es cancelado (desactivado), todas las suscripciones vigentes de ese plan **siguen activas**. Desactivar un plan no cancela las suscripciones existentes; si se quieren cancelar, debe hacerse individualmente según [5.4 Cancelar suscripción](#).

4.6 Obtener plan

```
GET https://api.geopagos.io/api/v1.0/plans/{planId}
```

Response 200

```
{
  "plan_id": "plan_123",
  "subscription_url": "https://...",
  "name": "Plan Mensual",
  "amount": 1000,
  "currency": "ARS",
  "billing_frequency": "monthly",
  "status": "active",
  "created_at": "2026-01-01T00:00:00Z",
  "updated_at": "2026-01-01T00:00:00Z"
}
```

5. Subscriptions

5.1 Listar suscripciones

```
GET https://api.geopagos.io/api/v1.0/subscriptions
```

Query params

Parámetro	Requerido	Descripción
limit	Sí	—
offset	Sí	—
planId	No	Filtra por plan
customerId	No	Filtra por cliente
status	No	—
createdAfter	No	Fecha ISO 8601
createdBefore	No	Fecha ISO 8601
search	No	Búsqueda por nombre, email, documento o referencia externa del cliente

Response 200

```
{
  "data": [
    {
      "subscription_id": "sub_123",
      "customer_id": "cus_456",
      "plan_id": "plan_123",
      "payment_method_id": "pm_789",
      "status": "active",
      "start_date": "2026-01-01T00:00:00Z",
      "current_period_start": "2026-07-01T00:00:00Z",
      "current_period_end": "2026-08-01T00:00:00Z",
      "next_billing_date": "2026-08-01T00:00:00Z",
      "billing_cycles_completed": 6,
      "created_at": "2026-01-01T00:00:00Z",
      "updated_at": "2026-07-01T00:00:00Z"
    }
  ],
  "pagination": { "total": 1, "limit": 20, "offset": 0 }
}
```

Campos completos de cada suscripción: subscription_id, customer_id, plan_id, payment_method_id?, status, start_date, trial_end_date?, current_period_start, current_period_end, next_billing_date?, end_date?, cancelled_at?, billing_cycles_completed, billing_cycles_remaining?, last_amount?, next_amount?, next_amount_description?, external_reference?, terms_acceptance?, metadata?, created_at, updated_at.

5.2 Obtener suscripción

```
GET https://api.geopagos.io/api/v1.0/subscriptions/{subscriptionId}
```

Response 200 — mismo esquema que un ítem de GET /subscriptions.

5.3 Actualizar suscripción

```
PUT https://api.geopagos.io/api/v1.0/subscriptions/{subscriptionId}
```

Body — todos los campos son opcionales

Campo	Tipo	Notas
status	string	Valores: active, paused, cancelled
payment_method_id	string<uuid>	—
next_amount	number	—
next_amount_description	string	—
pause_until	string<date-time>	Fecha hasta la cual queda pausada
pause_reason	string	—
cancel_at_period_end	boolean	—
cancellation_reason	string	—
external_reference	string	Máximo 128 caracteres
metadata	object	—

Response 200 — mismo esquema que GET /subscriptions/{subscriptionId}.

5.4 Cancelar suscripción

```
DELETE https://api.geopagos.io/api/v1.0/subscriptions/{subscriptionId}
```

Body — todos los campos son opcionales

Campo	Tipo
cancel_at_period_end	boolean
cancellation_reason	string
metadata	object

Ejemplo de request

```
curl -X DELETE "https://api.geopagos.io/api/v1.0/subscriptions/sub_123" \
-H "Authorization: Bearer <token>" \
-H "Content-Type: application/json" \
-d '{ "cancel_at_period_end": true, "cancellation_reason": "Solicitado por el cliente" }'
```

Response 200 — mismo esquema que GET /subscriptions/{subscriptionId}.

6. Charges

6.1 Listar cobros

```
GET https://api.geopagos.io/api/v1.0/charges
```

Query params (snake_case)

Parámetro	Requerido	Descripción
limit	Sí	—
offset	Sí	—
subscription_id	No	—
status	No	Valores: pending, processing, succeeded, failed, cancelled
created_after	No	—
created_before	No	—
search	No	Búsqueda por nombre, email, documento o referencia externa del cliente

Response 200

```
{
  "data": [
    {
      "charge_id": "chg_123",
      "subscription_id": "sub_123",
      "amount": 1000,
      "currency": "ARS",
      "status": "paid",
      "billing_period_start": "2026-07-01T00:00:00Z",
      "billing_period_end": "2026-08-01T00:00:00Z",
      "attempt_count": 1,
      "max_attempts": 3,
      "created_at": "2026-07-01T00:00:00Z",
      "updated_at": "2026-07-01T00:00:00Z",
      "customer_name": "Juan Pérez",
      "customer_email": "juan@example.com",
      "plan_name": "Plan Mensual",
      "payment_method_last4": "4242",
      "payment_method_brand": "visa"
    }
  ],
  "pagination": { "total": 1, "limit": 20, "offset": 0 }
}
```

Campos completos: charge_id, subscription_id, amount, currency, status, reference_number?, billing_period_start, billing_period_end, attempt_count, max_attempts, last_attempt_at?, next_attempt_at?, failure_reason?, failure_code?, created_at, updated_at, customer_name, customer_email, plan_name, payment_method_last4, payment_method_brand, external_reference?, attempts?.

6.2 Obtener cobro

```
GET https://api.geopagos.io/api/v1.0/charges/{chargeId}
```

Response 200 — mismo esquema que un ítem de GET /charges, incluyendo el detalle completo de attempts[] (historial de intentos de cobro).

7. Customers

7.1 Listar clientes

```
GET https://api.geopagos.io/api/v1.0/customers
```

Query params

Parámetro	Requerido
limit	Sí
offset	Sí
email	No
name	No
createdAfter	No

Response 200

```
{
  "data": [
    {
      "customer_id": "cus_456",
      "email": "juan@example.com",
      "name": "Juan Pérez",
      "created_at": "2026-01-01T00:00:00Z",
      "updated_at": "2026-01-01T00:00:00Z"
    }
  ],
  "pagination": { "total": 1, "limit": 20, "offset": 0 }
}
```

Campos completos: customer_id, merchant_id?, tenant_identifier?, email, name, phone?, document?, metadata?, active_subscriptions_count?, created_at, updated_at.

7.2 Obtener cliente

```
GET https://api.geopagos.io/api/v1.0/customers/{customerId}
```

Response 200 — mismo esquema que un ítem de GET /customers.

7.3 Métodos de pago de un cliente

```
GET https://api.geopagos.io/api/v1.0/customers/{customerId}/payment-methods
```

Query params (opcionales)

Parámetro	Valores válidos
type	credit, debit, prepaid
status	active, inactive, expired
brand	visa, mastercard, amex, diners, cabal

Response 200

```
{
  "data": [
    {
      "payment_method_id": "pm_789",
      "customerId": "cus_456",
      "type": "credit",
      "brand": "visa",
      "last4": "4242",
      "expMonth": 12,
      "expYear": 2028,
      "holderName": "Juan Perez",
      "holderDocument": {
        "type": "DNI",
        "number": "12345678"
      },
      "status": "active",
      "metadata": {},
      "createdAt": "2026-01-01T00:00:00Z",
      "updatedAt": "2026-01-01T00:00:00Z"
    }
  ],
  "count": 1
}
```

Campos de cada elemento de data[]: payment_method_id (<uuid>, único campo en snake_case), customerId (<uuid>), type, brand, last4, expMonth (integer), expYear (integer), holderName, holderDocument ({ type, number }), status, metadata (object), createdAt (<date-time>), updatedAt (<date-time>).

Nota de casing: a diferencia del resto de la respuesta (camelCase), payment_method_id se mantiene en snake_case.

7.4 Suscripciones de un cliente

```
GET https://api.geopagos.io/api/v1.0/customers/{customerId}/subscriptions
```

Query params: limit, offset (siempre), status (opcional).

Response 200 — mismo esquema que GET /subscriptions.

8. Payment Methods

8.1 Obtener método de pago

```
GET https://api.geopagos.io/api/v1.0/payment-methods/{paymentMethodId}
```

Response 200

```
{
  "payment_method_id": "pm_789",
  "customer_id": "cus_456",
  "type": "credit",
  "brand": "visa",
  "last4": "4242",
  "exp_month": 12,
  "exp_year": 2028,
  "holder_name": "Juan Perez",
  "status": "active",
}
```

```
"created_at": "2026-01-01T00:00:00Z",
"updated_at": "2026-01-01T00:00:00Z"
}
```

9. Manejo de errores

- Los errores HTTP se devuelven con el código de estado correspondiente (4xx/5xx).
 - Si el header Authorization no está presente o es inválido, la petición falla antes de llegar a procesarse.
 - Se recomienda a los integradores loguear siempre status y el body de la respuesta para diagnóstico, dado que el detalle específico del error puede variar según el recurso.
 - Ante timeouts o errores de red, reintentar con backoff exponencial es la estrategia recomendada, especialmente para operaciones de lectura (GET).
-

10. Resumen de endpoints

Recurso	Método	Endpoint
Plans	GET	/api/v1.0/plans
Plans	POST	/api/v1.0/plans
Plans	PUT	/api/v1.0/plans/{planId}
Plans	GET	/api/v1.0/plans/{planId}
Subscriptions	GET	/api/v1.0/subscriptions
Subscriptions	GET	/api/v1.0/subscriptions/{subscriptionId}
Subscriptions	PUT	/api/v1.0/subscriptions/{subscriptionId}
Subscriptions	DELETE	/api/v1.0/subscriptions/{subscriptionId}
Charges	GET	/api/v1.0/charges
Charges	GET	/api/v1.0/charges/{chargeId}
Customers	GET	/api/v1.0/customers
Customers	GET	/api/v1.0/customers/{customerId}
Customers	GET	/api/v1.0/customers/{customerId}/payment-methods
Customers	GET	/api/v1.0/customers/{customerId}/subscriptions
Payment Methods	GET	/api/v1.0/payment-methods/{paymentMethodId}