

Integração PDV Avanço — Consulta de Descontos e Envio de Venda (Parceiros CRM)

“ Objetivo

Documentar o fluxo PDV → Parceiro de Fidelidade/Descontos (ex.: "cresce vendas" como placeholder), com exemplos de requisição e retorno, e a explicação dos campos utilizados. O documento é **genérico** e pode ser aplicado a qualquer parceiro que exponha endpoints equivalentes.

Visão geral do fluxo

1. **Consulta de descontos** (pré-venda): o PDV envia os itens do carrinho + identificação do cliente → **API do parceiro** retorna descontos aplicáveis e dados do cliente.
2. **Envio da venda** (pós-fechamento): o PDV registra a venda efetuada (itens, pagamento, total) → **API do parceiro** confirma pontos/saldo/benefícios concedidos e retorna os saldos finais.

“ **Observação:** os nomes de host, caminhos e cabeçalhos abaixo são **exemplos**.
Adapte-os ao parceiro real (ex.: `https://{host-parceiro}/admin/api/v2/...`).

Padrões comuns

- **Formato:** JSON (UTF-8).
- **Métodos:** `POST` para as operações descritas.
- **Timeout recomendado no PDV:** 30s.
- **Autenticação:** via cabeçalhos proprietários do parceiro (ex.: `X-AdminUser-Email`, `X-AdminUser-Token`).
- **Idioma:** quando suportado, enviar `Accept-Language: pt-BR`.

Cabeçalhos (exemplo)

```
Accept: */*
Content-Type: application/json
Accept-Language: pt-BR,pt;q=0.8,en-US;q=0.6,en;q=0.4
charset: utf-8
X-AdminUser-Email: <email-de-integracao@parceiro>
X-AdminUser-Token: <token-de-integracao>
```

1) Consulta de Descontos (pré-venda)

Endpoint (exemplo):

```
POST https://{host-parceiro}/admin/api/v2/shops
```

Request — Exemplo de `curl`

```
curl -k -m 30 -s -S \
  -H "Accept: */*" \
  -H "Content-Type: application/json" \
  -H "Accept-Language: pt-BR,pt;q=0.8,en-US;q=0.6,en;q=0.4" \
  -H "charset: utf-8" \
  -H "X-AdminUser-Email: <email-de-integracao@parceiro>" \
  -H "X-AdminUser-Token: <token-de-integracao>" \
  -X POST \
  --data-binary @inicia_parceiro.json \
  "https://{host-parceiro}/admin/api/v2/shops" \
  -o retorno_consulta.json --stderr erro_consulta.log -D header_consulta.txt
```

Request — Corpo (exemplo)

```
{
  "items": [
    {
      "category": "046",
      "code": "00000000000017",
    }
  ]
}
```

```

    "product_name": "CREME AMEND ALMOND B",
    "quantity": 1.0,
    "subcategory": "241",
    "unit_type": "um",
    "unit_value": 6.99
  }
],
"registration": "05940778690"
}

```

Significado dos campos (request)

Campo	Tipo	Obrigatório	Descrição
<code>items</code>	Array de objeto	Sim	Lista de itens no carrinho do PDV para avaliação de desconto.
<code>items[].category</code>	string	Recomendado	Código de categoria mercadológica do item no PDV.
<code>items[].subcategory</code>	string	Opcional	Código de subcategoria mercadológica.
<code>items[].code</code>	string	Sim	Identificador do produto (EAN/PLU/código interno).
<code>items[].product_name</code>	string	Recomendado	Descrição do produto.
<code>items[].quantity</code>	number	Sim	Quantidade do item.
<code>items[].unit_type</code>	string	Recomendado	Unidade de medida (ex.: <code>um</code> , <code>kg</code> , <code>lt</code>).
<code>items[].unit_value</code>	number	Sim	Preço unitário vigente no PDV (antes de desconto).
<code>registration</code>	string	Sim	Identificação do cliente para o parceiro. Ex.: CPF (somente números).

“ Boas práticas:

- Enviar `unit_value` já no **valor de venda** do PDV; o parceiro retornará descontos/propostas.
- Se houver mix de unidades (ex.: peso/quilagem), alinhar previamente com o parceiro como enviar e arredondar.

Response — Exemplo

```
{
  "response": {
    "code": 200,
    "discounts": {
      "products": [],
      "subtotal": 0.0
    },
    "shop_id": "0909092413",
    "user": {
      "address": {
        "adjunct": "",
        "city": "Belo Horizonte",
        "district": "Santa Efigênia",
        "number": 131,
        "phone": "(31) 33333-3333",
        "state": "Minas Gerais",
        "street": "Avenida Brasil",
        "zipcode": "30140-000"
      },
      "name": "Heberth Minelli",
      "st_token": "11177821-d40a-40f8-9b58-9f50978682a9",
      "total_balance": 100.0,
      "total_points": 1969
    }
  }
}
```

Significado dos campos (response)

Campo	Tipo	Descrição
<code>response.code</code>	number	Código de status interno do parceiro (200 = sucesso). Não substitui o HTTP status.
<code>response.discounts.products</code>	array	Detalhes de descontos por produto (quando aplicável).
<code>response.discounts.subtotal</code>	number	Desconto total a ser aplicado no subtotal do carrinho (se houver).

Campo	Tipo	Descrição
<code>response.shop_id</code>	string	Identificador da loja/estabelecimento no parceiro.
<code>response.user</code>	objeto	Dados do cliente no ecossistema do parceiro (endereços, saldos, pontos).
<code>response.user.st_token</code>	string	Token de sessão/identificação do cliente no parceiro (usar conforme política do parceiro).
<code>response.user.total_balance</code>	number	Saldo monetário de benefícios (se aplicável).
<code>response.user.total_points</code>	number	Pontos acumulados (se aplicável).

“ **Aplicação no PDV:** o PDV pode aplicar descontos retornados (por item e/ou subtotal) e exibir dados do cliente (nome, pontos, etc.), conforme regras da loja.

2) Envio da Venda (pós-fechamento)

Endpoint (exemplo):

POST `https://{host-parceiro}/admin/api/v2/sales`

“ O caminho acima é **exemplo**. Alguns parceiros usam `/shops/sales`, `/orders`, etc. Alinhe o endpoint definitivo na homologação.

Request — Corpo (exemplo)

```
{
  "discounts": [],
  "items": [
    {
      "category": "046",
      "code": "000000000000017",
      "product_name": "CREME AMEND ALMOND B",
      "quantity": 1.0,
```

```

    "subcategory": "241",
    "total_value": 6.99,
    "unit_type": "um",
    "unit_value": 6.99
  },
  {
    "category": "048",
    "code": "000000000000031",
    "product_name": "LEITE UHT PIRACANJUB",
    "quantity": 1.0,
    "subcategory": "269",
    "total_value": 1.0,
    "unit_type": "um",
    "unit_value": 1.0
  }
],
"payment": {
  "coupon": "02290621.99",
  "date": "2025-09-09T11:12:20",
  "paymethod": [
    { "paymethod": 1, "value": 7.99 }
  ],
  "total": 7.99
},
"registration": "05940778690"
}

```

Significado dos campos (request)

Campo	Tipo	Obrigatório	Descrição
discounts	array	Opcional	Lista de descontos efetivamente aplicados na venda (quando o parceiro exige detalhar).
items	array	Sim	Itens efetivamente vendidos. Campos seguem a semântica da consulta, com adição de total_value.

Campo	Tipo	Obrigatório	Descrição
items[].total_value	number	Sim	Valor total do item (já considerando quantidade e descontos).
payment	objeto	Sim	Dados do pagamento da venda.
payment.coupon	string	Recomendado	Identificador do cupom/ECF/NSU no PDV (para auditoria).
payment.date	string (ISO-8601)	Sim	Data/hora de fechamento da venda no PDV (ex.: YYYY-MM-DDThh:mm:ss).
payment.paymethod[]	array	Sim	Parcelas/meios de pagamento utilizados.
payment.paymethod[].paymeth od	number/string	Sim	Código do meio de pagamento (tabela acordada com o parceiro).
payment.paymethod[].value	number	Sim	Valor pago com o respectivo meio.
payment.total	number	Sim	Total da venda (soma dos pagamentos).
registration	string	Sim	Identificação do cliente no parceiro (ex.: CPF).

“ **Conciliação:** `payment.total` deve bater com a soma de `paymethod[].value`. Para múltiplos meios, adicione entradas no array.

Response — Exemplo

```
{
  "response": {
    "balance": 0.0,
    "code": 200,
    "future_balance": 0.0,
    "points": 8,
    "user": {
      "address": {
        "adjunct": "",
        "city": "Belo Horizonte",
```

```
{
  "district": "Santa Efigênia",
  "number": 131,
  "phone": "(31) 33333-3333",
  "state": "Minas Gerais",
  "street": "Avenida Brasil",
  "zipcode": "30140-000"
},
{
  "name": "Heberth Minelli",
  "st_token": "11177821-d40a-40f8-9b58-9f50978682a9",
  "total_balance": 100.0,
  "total_points": 1977
}
}
```

Significado dos campos (response)

Campo	Tipo	Descrição
<code>response.code</code>	number	Código de processamento no parceiro (200 = sucesso).
<code>response.points</code>	number	Pontos concedidos na venda.
<code>response.balance</code>	number	Saldo monetário utilizado/remanescente após a venda (quando aplicável).
<code>response.future_balance</code>	number	Saldo futuro (benefícios liberados em D+N, quando houver).
<code>response.user.total_points</code>	number	Pontos totais do cliente após a venda.
<code>response.user.total_balance</code>	number	Saldo total do cliente após a venda (quando aplicável).

Considerações de implementação (PDV)

- **Idempotência:**

- Recomenda-se enviar um **identificador único** por venda (ex.: `payment.coupon` + data) e, se o parceiro suportar, um cabeçalho `Idempotency-Key` para evitar

lançamentos duplicados em reenvios.

- **Tratamento de erros:**

- Mapear HTTP status (ex.: 4xx/5xx) e também `response.code` do corpo quando houver. Implementar retentativas **somente** para erros transitórios.

- **Encoding:**

- Garantir UTF-8 para nomes de produtos e logradouros; caso o parceiro retorne acentos com encoding incorreto, normalizar para exibição no PDV.

- **Tabelas de referência:**

- `paymethod` (meios de pagamento) e eventuais códigos de categoria/subcategoria devem ser alinhados e versionados entre PDV e parceiro.

- **Segurança:**

- Armazenar `X-AdminUser-Token` em cofre seguro. Nunca registrar token em logs de nível INFO/DEBUG.

- **Observabilidade:**

- Logar request-id, latência, tamanho de payload e resumo do resultado (sem PII sensível) para facilitar diagnósticos.

Anexos — Arquivos de exemplo

`inicia_parceiro.json`

```
{
  "items": [
    {
      "category": "046",
      "code": "000000000000017",
      "product_name": "CREME AMEND ALMOND B",
      "quantity": 1.0,
      "subcategory": "241",
      "unit_type": "um",
      "unit_value": 6.99
    }
  ],
  "registration": "05940778690"
}
```

`retorno_consulta.json` (amostra)

```
{
  "response": {
    "code": 200,
    "discounts": { "products": [], "subtotal": 0.0 },
    "shop_id": "0909092413",
    "user": {
      "name": "Heberth Minelli",
      "st_token": "11177821-d40a-40f8-9b58-9f50978682a9",
      "total_balance": 100.0,
      "total_points": 1969
    }
  }
}
```

venda_enviada.json

```
{
  "discounts": [],
  "items": [
    {
      "category": "046",
      "code": "000000000000017",
      "product_name": "CREME AMEND ALMOND B",
      "quantity": 1.0,
      "subcategory": "241",
      "total_value": 6.99,
      "unit_type": "um",
      "unit_value": 6.99
    },
    {
      "category": "048",
      "code": "000000000000031",
      "product_name": "LEITE UHT PIRACANJUB",
      "quantity": 1.0,
      "subcategory": "269",
      "total_value": 1.0,
      "unit_type": "um",
      "unit_value": 1.0
    }
  ]
}
```

```
}
],
"payment": {
  "coupon": "02290621.99",
  "date": "2025-09-09T11:12:20",
  "paymethod": [ { "paymethod": 1, "value": 7.99 } ],
  "total": 7.99
},
"registration": "05940778690"
}
```

retorno_venda.json (amostra)

```
{
  "response": {
    "balance": 0.0,
    "code": 200,
    "future_balance": 0.0,
    "points": 8,
    "user": {
      "name": "Heberth Minelli",
      "st_token": "11177821-d40a-40f8-9b58-9f50978682a9",
      "total_balance": 100.0,
      "total_points": 1977
    }
  }
}
```

Checklist de Homologação

- ☐
- ☐ Confirmar **endpoints definitivos** (consulta e venda) e autenticação.
- ☐
- ☐ Validar **mapeamento de** `paymethod` (códigos e formas aceitas).
- ☐
- ☐ Verificar regras de **arredondamento** e casas decimais em itens/descontos.
- ☐

- ☐ Ensaiar cenários com **múltiplos meios de pagamento** e cancelamento/estorno (se aplicável).
 - ☐
 - ☐ Testar comportamento **offline/timeout** e diretrizes de reenvio (com idempotência).
-

“ **Observação:** os exemplos de dados (CPF, nomes, tokens, códigos) são **fictícios** e servem apenas para demonstração. Substitua por valores reais na sua integração.

Revision #2

Created 23 September 2025 00:53:57 by Luiz Paulo

Updated 23 September 2025 01:19:49 by Luiz Paulo