

# Documentação da API

Esta documentação fornece informações detalhadas sobre como interagir com nossos endpoints de API para criar tokens de cartão, processar transações de compra (via Cartão de Crédito, Boleto e PIX), recuperar detalhes de pagamentos e configurar webhooks.

---

## Índice

- [Autenticação e Configuração](#)
    - [Obtenção das Chaves de API](#)
    - [Configuração de Webhooks](#)
  - [Endpoints](#)
    - [1. Criar Token de Cartão](#)
    - [2. Criar Transação de Compra](#)
      - [Transação com Cartão de Crédito](#)
      - [Transação com Boleto](#)
      - [Transação com PIX](#)
    - [3. Obter Detalhes do Pagamento](#)
  - [Webhooks](#)
    - [Estrutura do Webhook](#)
    - [Exemplo de Payload do Webhook](#)
    - [Eventos e Condições de Envio](#)
  - [Notas](#)
- 

## Autenticação e Configuração

Para utilizar a API, é necessário autenticar todas as requisições e configurar webhooks para receber notificações sobre o status das transações.

### Obtenção das Chaves de API

1. **Criação de Conta:** Primeiro, crie uma conta em nossa plataforma acessando `/platform/api-keys`.
2. **Geração das Chaves:** Após o login, acesse a seção **Chaves de API** para gerar suas chaves:
  - **Public Key:** Chave pública utilizada em algumas requisições.
  - **Secret Key:** Chave secreta que deve ser mantida em segurança e é usada para autenticação nas requisições.

**Importante:** Nunca compartilhe sua `Secret Key`. Ela é necessária para autenticação e deve permanecer confidencial.

### Configuração de Webhooks

1. **Acesso à Configuração:** Após criar sua conta, acesse `/platform/webhooks` para configurar seus webhooks.
2. **Registro do Webhook:** Forneça a URL do seu endpoint que receberá as notificações.

3. **Seleção de Eventos:** Escolha os eventos que deseja receber notificações:
- `onPixCreated` : Notificação quando um PIX é criado.
  - `onBuyApproved` : Notificação quando uma compra é aprovada.
  - `onChargeback` : Notificação em caso de chargeback.
  - `onRefund` : Notificação quando um reembolso é processado.
4. **Salvamento:** Após configurar, salve as alterações para ativar os webhooks.

## Endpoints

### 1. Criar Token de Cartão

Crie um token para um cartão de crédito que será usado em transações subsequentes.

- **Endpoint:** `POST /transaction.createCardToken`
- **URL:** `https://example.com.br/api/v1/transaction.createCardToken`

**Requisição**

**Cabeçalhos:**

- `Content-Type:` `application/json`
- `Authorization:` `<secretKey>`

**Parâmetros do Corpo:**

Parâmetro	Tipo	Obrigatório	Descrição
<code>cardNumber</code>	<code>string</code>	Sim	Número do cartão de crédito.
<code>cardCvv</code>	<code>string</code>	Sim	Código CVV do cartão de crédito.
<code>cardExpirationMonth</code>	<code>string</code>	Sim	Mês de expiração (2 dígitos).
<code>cardExpirationYear</code>	<code>string</code>	Sim	Ano de expiração (2 dígitos).
<code>holderName</code>	<code>string</code>	Sim	Nome do titular do cartão.
<code>holderDocument</code>	<code>string</code>	Sim	Documento (CPF) do titular do cartão.

**Exemplo de Requisição**

```
curl -X POST "https://example.com.br/api/v1/transaction.createCardToken" \
-H "Content-Type: application/json" \
-H "Authorization: <secretKey>" \
-d '{
  "cardNumber": "4000000000000010",
  "cardCvv": "123",
  "cardExpirationMonth": "10",
  "cardExpirationYear": "30",
  "holderName": "João Silva",
  "holderDocument": "01234567890"
}'
```

**Resposta**

Sucesso (200 OK)

```
{
  "token": "abcd1234efgh5678ijkl9012"
}
```

Parâmetros da Resposta:

Parâmetro	Tipo	Descrição
token	string	Token representando o cartão de crédito

2. Criar Transação de Compra

Crie uma transação de compra usando um dos métodos de pagamento: Cartão de Crédito, Boleto ou PIX.

- **Endpoint:** POST /transaction.purchase
- **URL:** https://example.com.br/api/v1/transaction.purchase

Requisição

Cabeçalhos:

- Content-Type: application/json
- Authorization: <secretKey>

Parâmetros do Corpo:

Parâmetro	Tipo	Obrigatório	Descrição
name	string	Sim	Nome completo do cliente.
email	string	Sim	Endereço de email do cliente.
cpf	string	Sim	CPF do cliente (11 dígitos).
phone	string	Sim	Número de telefone do cliente (8-12 dígitos).
paymentMethod	string	Sim	Método de pagamento: "PIX", "CREDIT_CARD" OU "BILLET".
creditCard	objeto	Condicional	Detalhes do cartão de crédito (obrigatório se paymentMethod for "CREDIT_CARD").
cep	string	Não	CEP do endereço.
complement	string	Não	Complemento do endereço.
number	string	Não	Número do endereço.
street	string	Não	Rua do endereço.
district	string	Não	Bairro do endereço.
city	string	Não	Cidade do endereço.
state	string	Não	Estado (UF) do endereço.

utmQuery	string	Não	Parâmetros UTM para rastreamento.
checkoutUrl	string	Não	URL de checkout.
referrerUrl	string	Não	URL de referência.
externalId	string	Não	Identificador externo para a transação.
postbackUrl	string	Não	URL para receber notificações de postback.
amount	number	Sim	Valor total em centavos (mínimo 500).
traceable	boolean	Sim	Indica se a transação é rastreável.
items	array	Sim	Lista de itens incluídos na transação.

Parâmetros do Objeto `creditCard` (obrigatório se `paymentMethod` for `"CREDIT_CARD"`):

Parâmetro	Tipo	Obrigatório	Descrição
token	string	Sim	Token obtido em <code>transaction.createCardToken</code> .
installments	number	Sim	Número de parcelas (entre 1 e 12).

Parâmetros do Objeto `items`:

Parâmetro	Tipo	Obrigatório	Descrição
unitPrice	number	Sim	Preço unitário em centavos.
title	string	Sim	Título ou descrição do item.
quantity	number	Sim	Quantidade do item.
tangible	boolean	Sim	Indica se o item é tangível.

Exemplos por Método de Pagamento

Transação com Cartão de Crédito

```
curl -X POST "https://example.com.br/api/v1/transaction.purchase" \
-H "Content-Type: application/json" \
-H "Authorization: <secretKey>" \
-d '{
  "name": "João Silva",
  "email": "joao.silva@example.com",
  "cpf": "01234567890",
  "phone": "16999999999",
  "paymentMethod": "CREDIT_CARD",
  "creditCard": {
    "token": "abcd1234efgh5678ijkl9012",
    "installments": 3
  },
  "amount": 5000,
  "traceable": true,
  "items": [
    {
      "unitPrice": 5000,
```

```
        "title": "Assinatura Premium",
        "quantity": 1,
        "tangible": false
    }
]
}'
```

#### Transação com Boleto

```
curl -X POST "https://example.com.br/api/v1/transaction.purchase" \
-H "Content-Type: application/json" \
-H "Authorization: <secretKey>" \
-d '{
  "name": "Maria Souza",
  "email": "maria.souza@example.com",
  "cpf": "09876543210",
  "phone": "16988888888",
  "paymentMethod": "BILLET",
  "amount": 7500,
  "traceable": true,
  "items": [
    {
      "unitPrice": 7500,
      "title": "Compra de E-book",
      "quantity": 1,
      "tangible": false
    }
  ]
}'
```

#### Transação com PIX

```
curl -X POST "https://example.com.br/api/v1/transaction.purchase" \
-H "Content-Type: application/json" \
-H "Authorization: <secretKey>" \
-d '{
  "name": "Ana Pereira",
  "email": "ana.pereira@example.com",
  "cpf": "12345678901",
  "phone": "16977777777",
  "paymentMethod": "PIX",
  "amount": 10000,
  "traceable": true,
  "items": [
    {
      "unitPrice": 10000,
      "title": "Acesso a Curso Online",
      "quantity": 1,
      "tangible": false
    }
  ]
}'
```

## Resposta

Sucesso (200 OK)

```
{
  "id": "txn_123456789",
  "customId": "cust_987654321",
  "installments": 3,
  "transactionId": "tran_123456789",
  "chargeId": null,
  "expiresAt": "2024-12-31T23:59:59Z",
  "dueAt": null,
  "approvedAt": null,
  "refundedAt": null,
  "rejectedAt": null,
  "chargebackAt": null,
  "paymentProvider": "GatewayDePagamento",
  "availableAt": null,
  "pixQrCode": null,
  "pixCode": null,
  "billetUrl": null,
  "billetCode": null,
  "customerId": "cust_123456789",
  "status": "PENDING",
  "address": null,
  "district": null,
  "number": null,
  "complement": null,
  "city": null,
  "state": null,
  "zipCode": null,
  "amount": 5000,
  "transactionFee": 150,
  "taxSeller": 50,
  "taxPlatform": 100,
  "amountSeller": 4850,
  "amountPlatform": 100,
  "amountMaster": 0,
  "amountGarantee": 0,
  "taxGarantee": 0,
  "garanteeReleaseAt": null,
  "approvedEmailSentAt": null,
  "traceable": true,
  "method": "CREDIT_CARD",
  "deliveryStatus": null,
  "createdAt": "2023-12-04T12:34:56Z",
  "updatedAt": "2023-12-04T12:34:56Z",
  "utmQuery": null,
  "checkoutUrl": null,
  "referrerUrl": null,
  "externalId": null,
  "postbackUrl": null
}
```

Parâmetros da Resposta:

Parâmetro	Tipo	Descrição
id	string	Identificador único da transação.
customId	string	Identificador personalizado da transação.
installments	number	Número de parcelas (se aplicável).
transactionId	string	Identificador do provedor de pagamento.
chargeId	string	Identificador da cobrança (se aplicável).
expiresAt	datetime	Data de expiração da transação.
dueAt	datetime	Data de vencimento (para boletos).
status	string	Status da transação ("PENDING", "APPROVED", etc.).
amount	number	Valor da transação em centavos.
method	string	Método de pagamento utilizado.
traceable	boolean	Indica se a transação é rastreável.
createdAt	datetime	Data e hora de criação da transação.
updatedAt	datetime	Data e hora da última atualização da transação.
Outros campos	vários	Detalhes adicionais dependendo do status da transação.

3. Obter Detalhes do Pagamento

Recupere detalhes de uma transação específica.

- **Endpoint:** GET /transaction.getPayment
- **URL:** https://example.com.br/api/v1/transaction.getPayment

Requisição

Cabeçalhos:

- Authorization: <secretKey>

Parâmetros de Consulta:

Parâmetro	Tipo	Obrigatório	Descrição
id	string	Sim	Identificador único da transação.

Exemplo de Requisição

```
curl -X GET "https://example.com.br/api/v1/transaction.getPayment?id=txn_123456789" \
-H "Authorization: <secretKey>"
```

Resposta

Sucesso (200 OK)

```
{
  "id": "txn_123456789",
  "amount": 5000,
  "status": "APPROVED",
  "method": "CREDIT_CARD",
  "billetCode": null,
  "billetUrl": null,
  "pixCode": null,
  "pixQrCode": null,
  "customId": "cust_987654321",
  "dueAt": null,
  "expiresAt": "2024-12-31T23:59:59Z",
  "installments": 3,
  "items": [
    {
      "id": "item_123456789",
      "unitPrice": 5000,
      "quantity": 1,
      "title": "Assinatura Premium"
    }
  ],
  "customer": {
    "name": "João Silva",
    "email": "joao.silva@example.com",
    "cpf": "01234567890",
    "phone": "16999999999"
  },
  "deliveryStatus": null,
  "trackingCode": null,
  "createdAt": "2023-12-04T12:34:56Z",
  "updatedAt": "2023-12-04T12:34:56Z"
}
```

**Parâmetros da Resposta:**

Parâmetro	Tipo	Descrição
id	string	Identificador único da transação.
amount	number	Valor da transação em centavos.
status	string	Status da transação ("PENDING", "APPROVED", etc.).
method	string	Método de pagamento utilizado.
billetCode	string	Código do boleto (se aplicável).
billetUrl	string	URL do boleto (se aplicável).
pixCode	string	Código PIX (se aplicável).
pixQrCode	string	QR Code PIX (se aplicável).
customId	string	Identificador personalizado da transação.
dueAt	datetime	Data de vencimento (se aplicável).



expiresAt	datetime	Data de expiração da transação.
installments	number	Número de parcelas (se aplicável).
items	array	Lista de itens na transação.
customer	objeto	Detalhes do cliente.
createdAt	datetime	Data e hora de criação da transação.
updatedAt	datetime	Data e hora da última atualização da transação.
Outros campos	vários	Detalhes adicionais dependendo do tipo de transação.

## Webhooks

Os webhooks permitem que sua aplicação receba notificações em tempo real sobre eventos relacionados às transações, como aprovações, recusas e criação de PIX.

### Estrutura do Webhook

Quando um evento configurado ocorre, enviaremos uma requisição POST para a URL que você forneceu ao configurar o webhook. O payload (corpo) dessa requisição terá a seguinte estrutura:

#### Parâmetros do Payload:

Parâmetro	Tipo	Descrição
paymentId	string	Identificador único do pagamento.
externalId	string	Identificador externo da transação (se fornecido).
checkoutUrl	string	URL do checkout (se aplicável).
referrerUrl	string	URL de referência (se aplicável).
customId	string	Identificador personalizado da transação.
status	string	Status atual da transação ("PENDING", "APPROVED", etc.).
paymentMethod	string	Método de pagamento utilizado.
deliveryStatus	string	Status da entrega (se aplicável).
totalValue	number	Valor total da transação em centavos.
netValue	number	Valor líquido para o vendedor em centavos.
pixQrCode	string	QR Code PIX (se aplicável).
pixCode	string	Código PIX (se aplicável).
billetUrl	string	URL do boleto (se aplicável).
billetCode	string	Código do boleto (se aplicável).
expiresAt	datetime	Data de expiração da transação.

dueAt	datetime	Data de vencimento (para boletos).
installments	number	Número de parcelas (se aplicável).
utm	string	Parâmetros UTM para rastreamento (se aplicável).
items	array	Lista de itens incluídos na transação.
customer	objeto	Detalhes do cliente.
createdAt	datetime	Data e hora de criação da transação.
updatedAt	datetime	Data e hora da última atualização da transação.
approvedAt	datetime	Data e hora de aprovação da transação (se aplicável).
refundedAt	datetime	Data e hora de reembolso da transação (se aplicável).
chargebackAt	datetime	Data e hora de chargeback da transação (se aplicável).
rejectedAt	datetime	Data e hora de rejeição da transação (se aplicável).

**Parâmetros do Objeto `customer` :**

Parâmetro	Tipo	Descrição
id	string	Identificador único do cliente.
name	string	Nome do cliente.
email	string	Email do cliente.
cpf	string	CPF do cliente.
cep	string	CEP do cliente.
phone	string	Telefone do cliente (com DDI).
complement	string	Complemento do endereço.
number	string	Número do endereço.
street	string	Rua do endereço.
city	string	Cidade do endereço.
state	string	Estado (UF) do endereço.
district	string	Bairro do endereço.
createdAt	datetime	Data de criação do cadastro.
updatedAt	datetime	Data da última atualização.

**Exemplo de Payload do Webhook**

```
{
  "paymentId": "txn_123456789",
  "externalId": null,
  "checkoutUrl": null,
```

```

"referrerUrl": null,
"customId": "cust_987654321",
"status": "APPROVED",
"paymentMethod": "CREDIT_CARD",
"deliveryStatus": null,
"totalValue": 5000,
"netValue": 4850,
"pixQrCode": null,
"pixCode": null,
"billetUrl": null,
"billetCode": null,
"expiresAt": "2024-12-31T23:59:59Z",
"dueAt": null,
"installments": 3,
"utm": null,
"items": [
  {
    "id": "item_123456789",
    "unitPrice": 5000,
    "quantity": 1,
    "title": "Assinatura Premium",
    "tangible": false
  }
],
"customer": {
  "id": "cust_123456789",
  "name": "João Silva",
  "email": "joao.silva@example.com",
  "cpf": "01234567890",
  "cep": "01001000",
  "phone": "+5516999999999",
  "complement": null,
  "number": "123",
  "street": "Rua Exemplo",
  "city": "São Paulo",
  "state": "SP",
  "district": "Centro",
  "createdAt": "2023-12-04T12:34:56Z",
  "updatedAt": "2023-12-04T12:34:56Z"
},
"createdAt": "2023-12-04T12:34:56Z",
"updatedAt": "2023-12-04T12:34:56Z",
"approvedAt": "2023-12-04T12:35:00Z",
"refundedAt": null,
"chargebackAt": null,
"rejectedAt": null
}

```

## Eventos e Condições de Envio

Com base no código fornecido, o webhook será acionado nas seguintes condições:

- Transações **PENDENTES**:

- Se o método de pagamento for **CARTÃO DE CRÉDITO** e o evento `onPixCreated` estiver habilitado, o status enviado será `"IN_REVIEW"`.
  - Se o método de pagamento for **PIX** e o evento `onPixCreated` estiver habilitado.
  - **Transações APROVADAS:**
    - Se o evento `onBuyApproved` estiver habilitado.
  - **Transações com CHARGEBACK:**
    - Se o evento `onChargeback` estiver habilitado.
  - **Transações REEMBOLSADAS:**
    - Se o evento `onRefund` estiver habilitado.
- 

## Notas

- **Formatos de Data e Hora:** Todos os campos de data e hora estão no formato ISO 8601 ( `YYYY-MM-DDTHH:mm:ssZ` ).
  - **Valores Monetários:** Todos os valores monetários estão em **centavos**. Por exemplo, `5000` representa `R$50,00`.
  - **Formatos de CPF e Telefone:**
    - **CPF:** Deve conter 11 dígitos (exemplo: `"01234567890"`).
    - **Telefone:** Deve conter entre 8 e 12 dígitos e incluir o código do país (exemplo: `"5516999999999"`).
  - **Valor Mínimo da Transação:** O valor mínimo para uma transação é de `500` centavos ( `R$5,00` ).
  - **Rastreabilidade:** Defina `traceable` como `true` para habilitar o rastreamento da transação.
  - **Cabeçalho de Autorização:** Deve ser incluído em **todas** as requisições para autenticação.
  - **Tratamento de Erros:** Em caso de erros, a API retornará códigos de status HTTP apropriados e mensagens detalhadas.
  - **Configuração Prévia:** Certifique-se de configurar os webhooks e obter as chaves de API antes de iniciar a integração.
- 

Obrigado por escolher nossos serviços!