



## Webhooks

# Pagamentos

Os webhooks são usados para notificar automaticamente o seu sistema sobre eventos significativos. Eles são enviados como requisições HTTP POST para as URLs configuradas, contendo dados no formato JSON.

## Formato do Objeto de Webhook

Quando um evento ocorre, a notificação é enviada para a URL configurada com o seguinte formato:

```
{
  "event": "nome do evento",
  "token": "token gerado para validação",
  "offerCode": "Código da oferta quando é venda no checkout interno",

  "client": {
    "id": "identificador do cliente",
    "name": "nome do cliente",
    "email": "email do cliente",
    "phone": "telefone do cliente",
    "cpf": "CPF do cliente", // ou null
    "cnpj": "CNPJ do cliente", // ou null

    // É nulo quando for um info produto
    "address": {
      "country": "código do país ex: BR",
      "zipCode": "cep no formato 12345-678",
      "state": "código do estado ex: SP",
      "city": "cidade",
      "neighborhood": "bairro",
      "street": "rua",
```



```

    "number": "número",
    "complement": "complemento"
  },
},

"transaction": {
  "id": "identificador da transação",
  "identifier": "Seu identificador da transação",
  "status": "status da transação",
  "paymentMethod": "método de pagamento",
  "originalCurrency": "moeda do cliente",
  "originalAmount": "valor da transação na moeda do cliente", // número
  "currency": "sua moeda de recebimento",
  "exchangeRate": "quando a moeda do cliente não for BRL", // número
  "installments": "número de parcelas", // número
  "amount": "valor na sua moeda de recebimento", // número
  "createdAt": "data e hora da criação da cobrança", // string ISO
  "payedAt": "data e hora do pagamento" // string ISO ou null,

  // Informado apenas para pagamentos via PIX
  "pixInformation": {
    "qrCode": "00020101065465468949845BR.GOV.BCB.PIX2566api-h.rendin
    "endToEndId": "E123456789012345678901234567890123456789012345" ,
  },

  // Opcional
  "pixMetadata": {
    "payerDocument": "Documento do pagador", // string ou null
    "payerName": "Nome do pagador", // string ou null
    "payerBankName": "Nome do banco do pagador", // string ou null
    "payerBankAccount": "Conta bancária do pagador", // string ou null
    "payerBankBranch": "Agência bancária do pagador", // string ou null

    "receiverDocument": "Documento do recebedor", // string ou null
    "receiverName": "Nome do recebedor", // string ou null
    "receiverPixKey": "Chave PIX do recebedor", // string ou null
    "receiverBankName": "Nome do banco do recebedor", // string ou null
    "receiverBankAccount": "Conta bancária do recebedor", // string
    "receiverBankBranch": "Agência bancária do recebedor" // string
  }
},

// subscription é null quando não for uma cobrança de assinatura

```

```

"subscription": {
  "id": "identificador da assinatura", // string
  "identifier": "Seu identificador da assinatura", // string
  "cycle": "número sequencial da cobrança", // number | Quando for 1º
  "startAt": "data e hora de início", // string ISO
  "intervalType": "tipo de frequência da cobrança", // "DAYS" | "WEEKS"
  "intervalCount": "frequência da cobrança" // number
},

"orderItems": [
  {
    "id": "identificador do item",
    "price": "valor total do item", // numérico
    "product": {
      "id": "identificador do produto",
      "name": "nome do produto",
      "externalId": "seu identificador do produto, para vendas via
    }
  }
],

"trackProps": {
  "key": "value"
  "key1": "value1"
}
}

```

## Possíveis eventos

Os seguintes eventos podem ser monitorados:

TRANSACTION\_CREATED - Transação criada

TRANSACTION\_PAID - Transação paga

TRANSACTION\_CANCELED - Transação cancelada

TRANSACTION\_REFUNDED - Transação estornada

## Métodos de pagamento

Os seguintes `paymentMethod` podem ser monitorados:



BOLETO - Boleto

CREDIT\_CARD - Cartão

## Status da transação

Os seguintes status de transação podem ser monitorados no campo `status`:

`COMPLETED` - Transação concluída (paga)`FAILED` - Transação falhou`PENDING` - Transação pendente`REFUNDED` - Transação estornada`CHARGED_BACK` - Chargeback

## Exemplo de webhook



```
{
  "event": "TRANSACTION_PAID",
  "token": "bhhi3tybfw",
  "offerCode": "ABCK181",
  "client": {
    "id": "11x7ot6j23",
    "name": "John Doe",
    "email": "jondoe@gmail.com",
    "phone": "(11) 9 8888-7777",
    "cpf": "123.456.789-10",
    "cnpj": null,
    "address": {
      "country": "BR",
      "zipCode": "01304-000",
      "state": "SP",
      "city": "São Paulo",
      "neighborhood": "Consolação",
      "street": "Rua Augusta",
      "number": "6312",
      "complement": "6 andar"
    }
  }
}
```

```
    },  
    "Payaction": {  
      "id": "xa69kbub2c",  
      "status": "COMPLETED",  
      "paymentMethod": "CREDIT_CARD",  
      "originalCurrency": "USD",  
      "originalAmount": 20,  
      "currency": "BRL",  
      "amount": 100,  
      "exchangeRate": 5,  
      "installments": 2,  
      "createAt": "2026-02-03T22:24:05.166Z",  
      "payedAt": "2026-02-03T22:25:05.166Z",  
      "pixInformation": {  
        "qrCode": "00020101br.gov.bcb.pix2563pix-h.exemplo.com/pix...",  
        "endToEndId": "E123456789012345678901234567890123456789012345"  
      },  
      "pixMetadata": {  
        "payerDocument": "123.456.789-00",  
        "payerName": "João da Silva",  
        "payerBankName": "Caixa Economica Federal",  
        "payerBankAccount": "12345",  
        "payerBankBranch": "12",  
        "receiverDocument": "987.654.321-00",  
        "receiverName": "Maria da Silva",  
        "receiverPixKey": "98765432100",  
        "receiverBankName": "Banco do Brasil",  
        "receiverBankAccount": "54321",  
        "receiverBankBranch": "21"  
      }  
    },  
    "subscription": null,  
    "orderItems": [  
      {  
        "id": "fe96zw6bdz",  
        "price": 100,  
        "product": {  
          "id": "vhby08vysb",  
          "name": "Curso de marketing",  
          "externalId": "KSA912"  
        }  
      }  
    ],  
  ],  
}
```

```
"trackProps": {  
  "utm_id": "12345",  
  "utm_source": "facebook",  
  "utm_medium": "cpc",  
  "utm_campaign": "lançamento",  
  "utm_content": "newsletter",  
  "utm_term": "summer+venda",  
  "fbclid": "fb.1.1234567890.0987654321",  
  "fbp": "fb.1.0987654321.1234567890",  
  "ip": "179.241.195.127",  
  "country": "BR",  
  "user_agent": "Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/81.0.4012.101 Mobile Safari/537.36",  
  "zip_code": "01304-000",  
  "city": "São Paulo",  
  "state": "SP"  
}
```

## 1. Webhooks Cadastrados Internamente

### O que são?

Os webhooks cadastrados internamente são configurados diretamente no painel da CN Pay. Eles permitem que produtores recebam notificações sobre eventos relacionados a pagamentos feitos pelos seus clientes através do checkout da CN Pay.

### Para Quem é Indicado?

Este tipo de webhook é ideal para produtores que utilizam a plataforma CN Pay para gerenciar suas vendas e desejam receber atualizações automáticas sobre eventos críticos, como transações realizadas, pagamentos concluídos ou estornos. Ele é indicado para aqueles que preferem uma configuração simples e centralizada diretamente no painel da CN Pay, sem a necessidade de integrar diretamente via API.

## Configurando Webhooks Internamente

Para configurar um webhook internamente, siga os seguintes passos:

1. Acesse o painel da CN Pay.



2. Navegue até `Configurações > Webhooks`.

3. Clique no botão **Criar** e preencha os seguintes campos:

**Título do Webhook:** Nome para identificação interna.

**URL alvo do disparo:** URL que receberá as notificações.

**Produtos:** Selecione os produtos para os quais o webhook deve disparar.

**Eventos:** Selecione os eventos que deseja monitorar, como Transação criada, Transação paga, etc.

## Token de Validação

Após a criação do webhook, será gerado um token exclusivo. Este token deve ser usado para validar as notificações recebidas e garantir que foram enviadas pela CN Pay.

## 2. Webhooks via API

Os webhooks via API são configurados diretamente pelo produtor através de chamadas à API da CN Pay. Eles são ideais para integrações mais avançadas, onde a plataforma do produtor necessita receber notificações sobre o andamento das transações.

### Para Quem é Indicado?

Os webhooks via API são indicados para desenvolvedores e empresas que possuem plataformas próprias e que precisam de uma integração mais robusta e personalizada com a CN Pay. Este tipo de webhook é perfeito para quem deseja receber notificações sobre o andamento das transações em tempo real, diretamente na sua aplicação, e exige uma flexibilidade maior para integrar diferentes fluxos de pagamento e automação.

### Configurando Webhooks via API

Para configurar um webhook via API, o produtor deve:

1. Realizar uma requisição POST para a rota específica do método desejado, enviando o `callbackUrl` como um dos parâmetros.
2. No retorno da requisição, além dos dados esperados (como informações de pagamento), será retornado um **token** para validação das futuras notificações.



## Como Funciona?

---

Após a criação do webhook via API, a CN Pay também cria um registro interno para manter o histórico das notificações. Sempre que ocorrer uma atualização na transação (exemplo: pagamento de um Pix), a CN Pay enviará uma notificação para o `callbackUrl` fornecido.

## Limite de Webhooks

Existe um limite de **20 webhooks** por integração. Esse limite é uma trava de segurança para evitar o uso incorreto do `callbackUrl`.

Se você está recebendo o erro `"Você pode criar no máximo 20 webhooks"`, provavelmente está enviando URLs diferentes para cada transação. Consulte nosso artigo [Limite de webhooks via API](#) para entender como resolver esse problema.

## Resposta aos Webhooks

Após processar um webhook, o endpoint do seu sistema deve retornar um status HTTP `2XX` para indicar que o webhook foi recebido e processado com sucesso. Outros códigos de status podem indicar que o webhook não foi aceito e será necessário reenviar a notificação.

---

[← Como usar o split via API](#)

[Transferências →](#)