



## Wallet Hub Linx

### Manual de Integração – Automações (PDVs)

Área: P&D  
Data: Julho/2024  
Versão: 2.9

<b>1.</b>	<b>Controle de Revisões .....</b>	<b>4</b>
<b>2.</b>	<b>Descrição da Documentação.....</b>	<b>5</b>
<b>3.</b>	<b>Glossário.....</b>	<b>6</b>
<b>4.</b>	<b>Ambientes .....</b>	<b>7</b>
<b>5.</b>	<b>Autenticação no DevCenter .....</b>	<b>9</b>
5.1.	<i>Headers da Requisição .....</i>	9
<b>6.</b>	<b>Relação de chamadas à API .....</b>	<b>10</b>
6.1.	<i>Ativar ou desativar PDV.....</i>	10
Método pos/status .....	10	
Objeto formataqr .....	11	
Exemplo de Requisição .....	11	
Retorno da Requisição .....	11	
6.2.	<i>Criar Cobrança.....</i>	12
Método payment .....	12	
Objeto payments .....	13	
Objeto payer .....	14	
Objeto client_phone .....	14	
Objeto client_identification .....	14	
Objeto split .....	14	
Objeto order_items .....	15	
Objeto shipping_address .....	15	
Objeto specific_wallets .....	16	
Exemplo de Requisição .....	16	
Retorno da Requisição .....	18	
6.3.	<i>Recebimento de Callback.....</i>	19
6.4.	<i>Confirmar Pagamento .....</i>	22
Método payment order confirm.....	22	
Objeto payments .....	22	
Exemplo de Requisição .....	22	
Retorno da Requisição .....	23	
6.5.	<i>Cancelar Pagamento .....</i>	23
Método payment cancel.....	24	
Exemplo de Requisição .....	24	
Retorno da Requisição .....	24	
6.6.	<i>Consultar Carteira/ Wallet .....</i>	25
Retorno da Requisição .....	25	
<b>7.</b>	<b>Consultar Cobrança – Contingência .....</b>	<b>27</b>
Retorno da Requisição .....	28	
Rate Limit .....	32	
<b>8.</b>	<b>Integrações Parceiros .....</b>	<b>33</b>
8.1.	<i>Integração Padrão .....</i>	33
8.2.	<i>Integração PIX Saque.....</i>	33
8.3.	<i>Integração PIX Troco .....</i>	34

8.4.	<i>Integração NuPay</i> .....	35
8.5.	<i>Integração PIX Paga Leve</i> .....	35
8.5.1.	<i>Pré Aprovação – PIX Parcelado</i> .....	35
	Método installment-availability.....	35
	Exemplo de Requisição .....	36
	Retorno da Requisição .....	36
8.6.	<i>Integração PicPay E-Commerce</i> .....	37
8.7.	<i>Integração Pede Pronto</i> .....	37
8.8.	<i>Integração Split de Pagamento</i> .....	37
9.	<b>Status Page QRLinx</b> .....	<b>39</b>
10.	<b>Tabelas de Status e Erros</b> .....	<b>41</b>
10.1.	<i>Status do pagamento</i> .....	41
10.2.	<i>Status do cancelamento</i> .....	41
10.3.	<i>Status PIX</i> .....	41
10.4.	<i>Tabela de erros</i> .....	42

## 1. Controle de Revisões

Data	Histórico das revisões	Versão	Revisado por
Jan-23	Atualização de diversos itens necessários para melhor entendimento do conteúdo.	1.8	P&D – Rodrigo Mendes
Mar-23	Inclusão de informação sobre a cobrança específica (specific_wallets) e API para consulta de carteiras.	1.9	P&D – Rodrigo Mendes
Mar-23	Atualização de informações sobre as APIs de consultas e cancelamento.	2.0	P&D – Rodrigo Mendes
Mai-23	Inclusão das informações sobre o Rate Limit.	2.1	P&D – Rodrigo Mendes
Mai-23	Melhoria da informação apresentada sobre os Status de Cancelamento no item 9.2.	2.2	P&D – Rodrigo Mendes
Jun-23	Inclusão de campos específicos para transações com a Pede Pronto.	2.3	P&D – Rodrigo Mendes
Ago-23	Inclusão das informações sobre o “integration_type_pix” e “report” nas consultas de transações.	2.4	P&D – Rodrigo Mendes
Out-23	Alteração da collection do postman; Alteração no processo de solicitação das credenciais de HMG.	2.5	P&D – Rodrigo Mendes
Nov-23	Atualização do item 6.2 com novos campos na criação a cobrança. Incluído tópico específico sobre as integrações e suas particularidades (item 8).	2.6	P&D – Rodrigo Mendes
Fev-24	Inclusão de novos campos na criação a cobrança com NuPay e Paga Leve. Inclusão dos itens (8) Integração NuPay e Integração PIX Paga Leve.	2.7	P&D – Rodrigo Mendes
Mai-24	Inclusão da API para consulta PIX Parcelado na Paga Leve.	2.8	P&D – Rodrigo Mendes
Jul-24	Inclusão de informações sobre a Status Page.	2.9	P&D – Rodrigo Mendes

## 2. Descrição da Documentação

Este documento contém definições, conceitos, parâmetros técnicos e exemplos das funcionalidades que compõem a solução “QRLinx - Linx Payhub”, tendo o propósito de orientar profissionais (desenvolvedores e analistas) envolvidos no seu processo de integração.

O QRLinx é uma solução inovadora desenvolvida pela Linx. Com o QRLinx, o seu estabelecimento comercial:

- Não precisa manter as placas com os QR Codes de cada carteira parceira em cima do balcão. Você apresenta um único QRCode na tela do seu ponto de venda (PDV) que pode ser lido e processado por todas as carteiras integradas<sup>1</sup> com a nossa solução e aceitas pelo seu estabelecimento;
- O processo de conciliação de vendas é integrado em um só lugar, facilitando a conferência dos valores recebidos;
- Reduz a possibilidade de fraudes no processo de pagamento.

---

<sup>1</sup> A lista atualizada de carteiras digitais integradas com o QRLinx deve ser obtida com o departamento comercial.

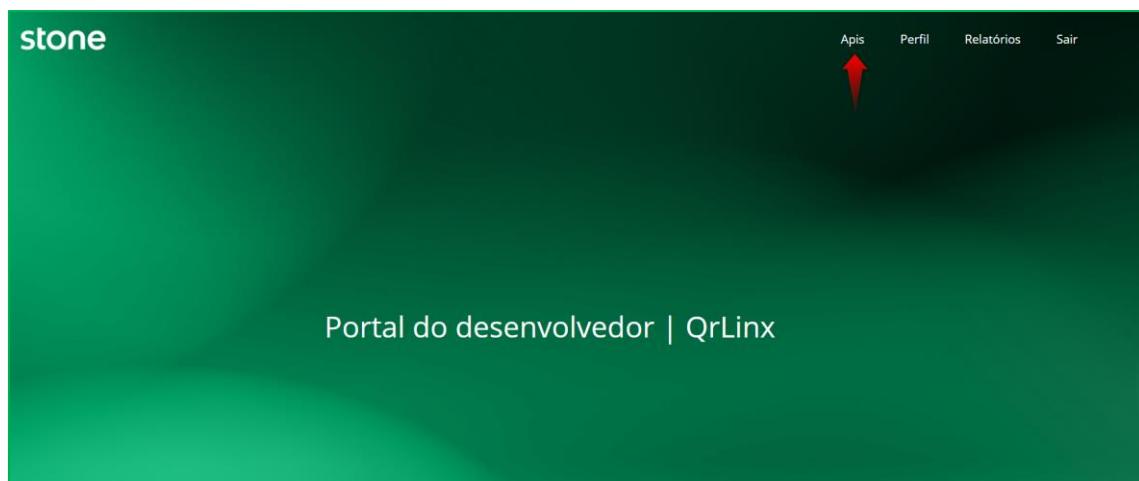
### 3. Glossário

Termo	Conceito
<b>Callback</b>	É um mecanismo de controle de fluxo que visa beneficiar processos assíncronos.
<b>Collection</b>	Uma coleção de APIs, para realização de testes no sistema.
<b>DevCenter</b>	Portal com as APIs e produtos da Linx.
<b>Estabelecimento Comercial (EC)</b>	Estabelecimento físico ou virtual que comercializa mercadorias ou serviços ( <i>e-commerce, marketplaces</i> , loja parceira, prestadores de serviços diversos, dentre outros).
<b>IdEndToEnd</b>	É o identificador único de uma transação Pix.
<b>PDV</b>	Ponto de Venda.
<b>PaymentId</b>	É o identificador único de uma cobrança gerada no QrLinx, seu formato padrão é um Guid.
<b>PSP</b>	Provedor em soluções de pagamento.
<b>Subscription Key</b>	Chave de produção e homologação para utilização das APIs no DevCenter.
<b>QueryString</b>	É um modelo clássico de manutenção do estado da página. Elas são nada mais do que conjuntos de pares/valores anexados a URL.
<b>QR Code</b>	Código de barras bidimensional legível apenas para máquinas que permite a inclusão de informações diversas sobre um determinado item ou fenômeno. Na Wallet Hub, os QR Codes contêm as informações necessárias à identificação e processamento de uma ordem de venda.
<b>URL</b>	Forma padronizada de representação de diferentes documentos, mídia e serviços de rede na internet, capaz de fornecer a cada documento um endereço único.
<b>WebHook</b>	É uma forma de recebimento de informações, que são passadas quando um evento acontece.

## 4. Ambientes

Para a liberação do ambiente de testes é necessário efetuar os passos abaixo:

1. Efetuar cadastro no portal DevCenter: <https://new-devcenter.linxpayhub.com.br/>
2. Enviar e-mail para [rodrigo.mendes@linx.com.br](mailto:rodrigo.mendes@linx.com.br) e [suporte-wallet@linx.com.br](mailto:suporte-wallet@linx.com.br) com as seguintes informações:
  - a. Título do e-mail: Liberação de acesso ao ambiente de homologação / DevCenter;
  - b. E-mail que foi cadastrado no DevCenter;
  - c. CNPJ para cadastro no ambiente de homologação do QRLinx.
3. Será solicitado a liberação das suas credenciais no DevCenter. Você receberá um e-mail comunicando a liberação do acesso ao ambiente de homologação;
4. Para obter a chave do DevCenter (Subscription Key) é necessário acessar novamente o portal do DevCenter, na aba “APIs” do menu superior;



5. Buscar o produto “Wallet Hub - QrLinx - PDVs”, e clicar sobre API;

A screenshot of the DevCenter API details page. The top navigation bar is visible with the "APIs" tab selected. The main content area shows a table with one row for the "Wallet Hub - QrLinx - PDVs" API. The columns are "Name", "Description", and "Type". The data in the table is: Name: "Wallet Hub - QrLinx - PDVs", Description: "PDVs", and Type: "REST". There are also filter and search options at the top of the table.

6. Será aberto uma tela com todas as APIs transacionais disponíveis para as automações, escolha uma delas e clique em “Try it”;

The screenshot shows the API definition for the 'Ativar/desativar ponto de venda' operation. The 'Try it' button is highlighted with a red arrow.

7. No campo authorization perceba que apenas a credencial de HMG está disponível;

The screenshot shows the 'Authorization' section for the 'Ativar/desativar ponto de venda' operation, highlighting the 'Primary: rodrigomendes' credential.

8. O valor da Subscription-Key está disponível mais abaixo, lembre-se que este é um dado sensível e somente automação poderá terá acesso a chave disponibilizada.

The screenshot shows the 'Headers' section for the 'Ativar/desativar ponto de venda' operation, highlighting the 'Ocp-Apim-Subscr.' header value.

Obs.: A Subscription Key do DevCenter é gerada para a integração em Homologação. A chave para integração em produção será gerada após validação da integração em Homologação pela equipe de Produtos do Wallet Hub, e permanecerá a mesma, independente de quantos estabelecimentos (CNPJs) utilizarem a aplicação integrada à Wallet Hub.

## 5. Autenticação no DevCenter

### 5.1. Headers da Requisição

Para realizar qualquer requisição à API da Wallet Hub é sempre necessário informar os 3 campos abaixo no header de requisição. Todos são de preenchimento **obrigatório**:

Campo	Descrição
<b>Ocp-Apim-Subscription-Key</b>	Chave de integração fornecida na plataforma DevCenter. Haverá uma chave para Homologação e outra para Produção, e elas serão únicas para a plataforma integrada, independente de quantos CNPJs usem a mesma.
<b>Apim-Company-Target</b>	Homologação: “linx-box” Produção: “linx”
<b>Apim-Company-Channel</b>	Identificação do canal, sendo de livre escolha, orientamos que o preenchimento seja com o nome da automação + CNPJ da Loja que realizará a solicitação (Ex.: AutoSystem-12345678901234)

## 6. Relação de chamadas à API

Todas as chamadas para o QRLinx **homologação** a inicial da URL sempre será:

<https://new-api.linxpayhub.com.br/wallet-hub-qrlinx>.

Todas as chamadas para o QRLinx em **produção** a inicial da URL sempre será:

<https://api.linxpayhub.com.br/wallet-hub-qrlinx>.

Para que possa realizar transações no ambiente de homologação foi desenvolvido um aplicativo de teste que simula transações de PIX, este aplicativo está disponível apenas para smartphones com sistema operacional android, segue o link para download:

<https://static.linxpayhub.com.br/qrlinx/apps/app-teste-qrlinx-v3.apk>.

### 6.1. Ativar ou desativar PDV

Para que um PDV possa solicitar a criação de uma cobrança, é necessário realizar o cadastro do PDV, ou seja, ativação junto ao QRLinx.

A obrigatoriedade e os valores de algumas propriedades variam conforme o tipo de ativação. Quando aplicável, as diferenças serão explicitadas no detalhamento das propriedades de cada método.

A operação para **Ativar/Desativar PDV** é feita com um *PATCH* para `{url}/v1/pos/status` com as propriedades abaixo:

#### Método pos/status

Parâmetro	Descrição	Tipo	Obrigatório
status	Estado transacional do PDV. Valores aceitos: "active" e "desactive".	String (8)	Sim
store_cnpj	CNPJ do EC que está realizando a venda.	String (14)	Sim
pos_description	Identificador alfanumérico do PDV, gerado pelo QRLinx.	String (30)	Não
external_code	Código associado ao PDV, de escolha livre do EC. Sugerimos a inclusão de uma identificação única para cada PDV.	String (128)	Sim

formataqr	Define o tipo de QRCode, que o PDV utilizará, atualmente apenas um valor é objeto utilizado.		Sim
channel	Canal de vendas utilizado pelo EC. Valores aceitos: "Físico" e "Online".	String (7)	Não

Caso o Channel fornecido na ativação do PDV seja “online” os campos com os dados do cliente que está realizando a compra se tornam obrigatórios.

### Objeto formataqr

Parâmetro	Descrição	Tipo	Obrigatório
Type	Valor esperado: "0000000000000002"	String (14)	Sim
description	Valor esperado: "Ativacao Rapida"	String (14)	Sim

### Exemplo de Requisição

```
{
  "status": "active",
  "store_cnpj": "12345678901234",
  "pos_description": "QRLINX124FGT",
  "external_code": "PDV00256",
  "formatqr": {
    "type": "0000000000000002",
    "description": "Ativacao Rapida"
  },
  "channel": "Fisico"
}
```

### Retorno da Requisição

O QrLinx retornará à informação que o PDV foi ativado com sucesso, informando o pos\_id, dado este necessário para a criação da cobrança.

```
{
  "success": true,
  "message": "Ponto de venda ativado com sucesso",
  "data": {
    "pos_id": "94c4566c-de09-4783-8f5b-97d859a342f2",
    "store_cnpj": "12345678901234",
    "pos_description": "QRLINXPBHQGE",
    "date_active": "2022-05-04T09:18:43.4119869",
    "channel": 0
  }
}
```

Caso não ocorra sucesso na requisição será retornado um erro com um código e descrição “LINX105” do que ocorreu, os códigos de erros estão disponíveis no final deste manual (Item 9).

```
{
  "error": {
    "code": "LINX105",
    "type": 400,
    "message": "Bad Request",
    "languages": [
      {
        "tag": "en-US",
        "description": "Cnpj inválido!"
      }
    ]
  }
}
```

## 6.2. Criar Cobrança

Para realizar a geração de um QRCode de pagamento o PDV deverá acionar API “Criar Cobrança”, nela é necessário fornecer dados do produto como também do cliente que está realizando a compra caso necessário. API retornará uma string para geração da imagem do QRCode como também algumas informações importantes da transação.

**Atenção:** O json de criação de cobrança poderá ter campos específicos para cada tipo de integração, ou seja, para criar uma cobrança com Split de pagamento é necessário enviar campos que não são obrigatórios com o Mercado Pago por exemplo, devido a isso no momento de acionar o QRLinx os campos obrigatórios para cada integração deverão ser respeitados.

A criação de um pagamento se dá por intermédio de um *POST* para [\[url\]/v1/payment](#). Ao criar a cobrança, ela terá o status “6 – Aguardando Pagamento”.

### Método payment

Parâmetro	Descrição	Tipo	Obrigatório
order_numberext	Número da ordem de venda atribuído pelo EC.	String (36)	Sim

pos_id	Identificação do PDV que está realizando a venda.	Guid (36)	Sim
amount	Valor total da venda.	Decimal (7,2)	Sim
subtotal	Subtotal da venda.	Decimal (7,2)	Sim
discount_value	Valor do desconto concedido.	Decimal (7,2)	Não
add_value	Valor adicionado à venda. Ex.: frete, embalagem.	Decimal (7,2)	Não
servicesAmount	Valor dos serviços, campo obrigatório para criar cobrança com a Pede Pronto.	Decimal (7,2)	Não
subsidizedAmount	Valor de subsídios, campo obrigatório para criar cobrança com a Pede Pronto.	Decimal (7,2)	Não
specific_wallets	Informação necessária para criação de cobrança específica.	objeto	Não
is_pix_upfront	Campo para criação de PIX parcelado junto a Paga leve.	String (true/false)	Não
payments	Informações necessárias à realização do pagamento.	objeto	Sim
payer	Informações do pagador.	objeto	Não
client_phone	Telefone do pagador	objeto	Não
client_identification	Natureza jurídica do pagador (Pessoa Física/Jurídica). Valores aceitos: "PF" e "PJ".	objeto	Não
order_items	Itens da ordem de venda.	objeto	Não
shipping_address	Endereço para entrega da compra.	objeto	Não

### Objeto payments

Parâmetro	Descrição	Tipo	Obrigatório
discount_value	Valor do desconto concedido.	Decimal (7,2)	Não
value	Valor total da venda.	Decimal (7,2)	Sim
url	Endereço do webhook fornecido pelo EC para o recebimento dos callbacks da transação.	String (max)	Não
payer	Informações do pagador.	Objeto	Não
expiration_time	Tempo em segundos da expiração da cobrança criada.	Integer (10)	Não
soft_descriptor	Identificação da loja, este campo é necessário para integração junto a PicPay e-commerce.	String (7)	Não
auto_capture	Caso o E-commerce necessite de confirmação do valor posterior à autorização da pessoa consumidora, é preciso enviar o campo com valor false,	String (true/false)	Não

	indicando que o valor autorizado será apenas reservado temporariamente (captura tardia), caso contrário enviar true. Este campo é necessário para integração junto a PicPay e-commerce.
withdraw_pix	Representa o valor que o cliente deseja sacar na loja, ou seja, utilizar o PIX SAQUE. Decimal (7,2) Não
change_pix	Representa o valor que o cliente deseja de troco na transação, ou seja, utilizar o PIX TROCO. Decimal (7,2) Não

**Objeto payer**

Parâmetro	Descrição	Tipo	Obrigatório
first_name	Primeiro nome do pagador.	String (30)	Não
last_name	Último nome do pagador.	String (30)	Não
email	E-mail do pagador.	String (50)	Não
external_code	Código do cliente no PSP de dados do EC.	String (36)	Não
client_phone	Dados do telefone do pagador.	objeto	Não

**Objeto client\_phone**

Parâmetro	Descrição	Tipo	Obrigatório
area_code	Código de área do telefone do pagador.	String (4)	Não
number	Número do telefone do pagador.	String (12)	Não
extension	Ramal do telefone do pagador.	String (4)	Não

**Objeto client\_identification**

Parâmetro	Descrição	Tipo	Obrigatório
type	Natureza jurídica do pagador. Valores Aceitos: "CPF" para pessoa física; "CNPJ" para pessoa jurídica)	String (5)	Não
number	Número do CPF/CNPJ do pagador. Campo obrigatório para integrações com NuPay e Paga Leve.	String (15)	Não

**Objeto split**

Parâmetro	Descrição	Tipo	Obrigatório
-----------	-----------	------	-------------

amount	Divisão do valor que será aplicada pelo split de acordo com a informação fornecida no type.	Decimal (7,2)	Não
recipients_document	Identificação do cliente recebedor.	String (50)	Não
type	Tipo de divisão do split é esperado o campo “percentage” (percentual) ou “amount” (valor).	String (50)	Não
options	Objeto referente as regras de taxa e devolução.	Objeto	Não
charge_processing_fee	Indica se o recebedor vinculado à regra será cobrado pelas taxas da transação.	String (true/false)	Não
charge_remainder_fee	No caso de restos de divisão de taxas entre os dois recebedores exista, aquele com esse parâmetro 'true' será cobrado desses centavos.	String (true/false)	Não
liable	Indica se o recebedor atrelado assumirá os riscos de estorno da transação.	String (true/false)	Não

### Objeto order\_items

Parâmetro	Descrição	Tipo	Obrigatório
name	Descrição do item.	String (150)	Não
quantity	Quantidade do item.	Decimal (7,2)	Não
amount	Valor total do item.	Decimal (7,2)	Não
subtotal	Subtotal do item.	Decimal (7,2)	Não
discount_value	Valor do desconto concedido.	Decimal (7,2)	Não
add_value	Valor adicionado à venda. Ex.: frete, embalagem.	Decimal (7,2)	Não
number_external	Código do item.	String (14)	Não
ncm	Natureza jurídica do pagador (Pessoa Física/Jurídica).	String (8)	Não
percentageDiscount	Percentual de desconto, campo obrigatório para integração com a Pede Pronto.	Decimal (7,2)	Não
subsidized	Subsídio, campo obrigatório para integração com a Pede Pronto.	String (true/false)	Não

### Objeto shipping\_address

Parâmetro	Descrição	Tipo	Obrigatório
street	Descrição do logradouro.	String (60)	Não
postal_code	CEP do logradouro.	String (10)	Não

number	Número do logradouro.	String (15)	Não
complement	Complemento do endereço.	String (40)	Não
state	Estado/UF do endereço.	String (2)	Não
district	Bairro do logradouro.	String (60)	Não
City	Cidade do logradouro.	String (60)	Não

### Objeto specific\_wallets

Este objeto é necessário para criação de uma cobrança específica, ou seja, quando a loja não deseja criar uma cobrança com todas as carteiras disponíveis.

O campo é uma lista, pois o PDV pode escolher criar a cobrança apenas com uma carteira ou mais de uma. Quando o PDV for enviar a chamada para criação de cobrança ele deverá enviar uma das opções da lista abaixo:

Nome
AME
CASH_BERTI
E_PICPAY
MERCADO_PAGO
NUPAY
PEDE_PRONTO
PICPAY
PIX

O specific wallets aceita apenas o nome da carteira, caso a loja não possua a informação de quais carteiras estão ativas para transacionar, o PDV poderá utilizar API “[Consulta Carteiras](#)”, essa API retorna as carteiras ativas da loja.

O campo não é obrigatório.

### Exemplo de Requisição

```
{
    "order_numberext": "1234",
    "pos_id": "096152EB-49D2-4634-B613-438E20EC3D35",
    "amount": 0.01,
    "subtotal": 0.01,
    "discount_value": 0.0,
    "add_value": 0.0,
    "servicesAmount": 0,
    "subsidizedAmount": 0,
    "specific_wallets": [
        "AME"
    ]
}
```

```

    "PIX", "MERCADO_PAGO"
],
"is_pix_upfront": true,
"payments": [
{
    "discount_value": 0.0,
    "value": 0.02,
    "url": "https://webhook.site/a5ef-c056bfd5a889",
    "payer": {
        "first_name": "Rodrigo",
        "last_name": "Mendes",
        "email": "rodrigo.mendes@linx.com.br",
        "external_code": "0",
        "client_phone": {
            "area_code": "11",
            "number": "982232155",
            "extension": "0"
        },
        "client_identification": {
            "type": "CPF",
            "number": "12345678901"
        }
    },
    "expiration_time": 100000,
    "split": [
{
        "amount": 0,
        "recipients_document": "string",
        "type": "string",
        "options": {
            "charge_processing_fee": true,
            "charge_remainder_fee": true,
            "liable": true
        }
    }
],
    "soft_descriptor": "string",
    "auto_capture": true,
    "withdraw_pix": 0,
    "change_pix": 0
}
],
"order_items": [
{
    "name": "produto1",
    "quantity": 1,
    "amount": 1.50,
    "subtotal": 1.50,
    "discount_value": 0.0,
    "add_value": 0.0,
    "number_external": "123",
}
]
}
]
```

```

        "ncm": "12345678",
        "percentageDiscount": 0,
        "subsidized": true
    }
],
"shipping_address": {
    "street": "Rua Pan",
    "postal_code": "09572550",
    "number": "397",
    "complement": "",
    "state": "SP",
    "district": "Nova Gerty",
    "city": "São Caetano do Sul"
}
}
}

```

### Retorno da Requisição

O QRLinx retornará as informações sobre a cobrança gerada, contendo o id do pagamento, a string do QR Code e as respectivas carteiras aceitas pela loja, o QR retornado é o dinâmico e sempre será gerado um novo a cada criação de cobrança.

É retornando também o “qrlink”, uma opção para as automações utilizarem o link de pagamento do QRLinx, além de conter também a opção “Copia e Cola” do PIX.

É de suma importância que pelo menos o PaymentId seja registrado e salvo pela automação, pois este dado é necessário para consultar o pagamento caso ocorra alguma instabilidade no fluxo transacional.

Caso o PDV utilize o QR Code em tela automação deverá gerar a imagem do QR Code e exibir também as carteiras digitais cadastradas para a loja, o QRLinx envia também o logo de cada carteira digital credenciada.

```

"success": true,
"message": "pagamento criado com sucesso, utilize a consulta do pagamento para saber o status do pagamento.",
"data": {
    "payment_id": "2af10f23-8064-4c5e-a4c5-348bd965b425",
    "order_id": "933c50c6-a70b-41d3-a1da-21f8f02a873d",
    "payment_status": {
        "id": 6,
        "name": "Aguardando Pagamento"
    },
    "date_time": "2022-05-04T09:29:36.3764226",
    "expiration_date": "2022-05-04T09:35:36.3764226",
    "qrcode": "00020101021226840014br.gov.bcb.pix2562pix-h.stone.com.br/pix/v2/3ca0268b-0ac1-4702-bde5-9e7ea4ac5daf27970021BR.COM.LINXPAYHUB.API0168LINX0nNB9w2IGMYduJVpqcLiY5Zlz5n0weq6Llh-fHCT-1kQT-"
}

```

```

ondc-
GsQf6sWZjPQw52040000530398654041.505802BR5914EAD PLATAFORMA6014RIO DE JANEIRO62290525b8be302c
ea30467cad7e4144763048FF3",
  "qrlink": "https://paymentqr.linx.com.br/payment/2af10f23-8064-4c5e-a4c5-348bd965b425",
  "wallets": [
    {
      "payment_link": null,
      "cnpj": "0000000000000000",
      "name": "Pix",
      "trading_name": "Pix",
      "logo_link": "https://static.linipayhub.com.br/images/logos/LogoPix.jpg",
      "logo_link_download": "https://static.linipayhub.com.br/images/logos/LogoPix.jpg"
    },
    {
      "payment_link": null,
      "cnpj": "22896431000110",
      "name": "Picpay Servicos S.A",
      "trading_name": "PicPay",
      "logo_link": "https://static.linipayhub.com.br/images/logos/LogoPicPay.png",
      "logo_link_download": ""
    }
  ]
}

```

Consulte o [Manual de Boas Práticas](#) do QRlinx para mais informações sobre o processo de criação da cobrança.

### 6.3. Recebimento de Callback

Ao realizar a criação de uma cobrança automação deverá fornecer uma URL para recebimento de callback do QRlinx, automação será notificada sempre que uma alteração na cobrança ocorrer.

Além da opção de callback o QRlinx disponibiliza uma API para consulta de status da transação, mas orientamos que essa API seja utilizada apenas no modelo de contingência, caso não ocorra o recebimento do webhook. A informação para utilização da API de consulta está disponível no [item 7](#).

A url deste serviço deve ser enviada na requisição da criação da cobrança o QRlinx realiza as seguintes notificações:

- Cobrança criada e com status aguardando pagamento;
- Após a realização do pagamento e com o status pagamento realizado;
- Caso ocorra um desfazimento em execução (não ocorre um estorno) e com o status desfeito;

- Caso ocorra um desfazimento transacional (ocorre um estorno) e com o status desfeito;
- Caso ocorra um cancelamento (ocorre um estorno) da cobrança com o status cancelado.

Segue abaixo exemplo dos callbacks enviados pelo QRlinx:

- Cobrança criada;

```
{
  "payment_id": "6093d4fd-78f0-447b-be8b-7891497ace1e",
  "order_id": "e18563f8-1c49-491b-a9b2-763e70b33d71",
  "discount_value": 0,
  "value": 0.02,
  "date_time": "2023-01-19T11:22:53.1774379",
  "payment_method": "",
  "remaining_amount": 0.02,
  "order": {
    "order_number_ext": "896819"
  },
  "refunds": [],
  "api_version": 0,
  "id_end_to_end": "E60701190202301191424DY5RKDFHGOT",
  "payment_status": {
    "id": 6,
    "name": "Pendente"
  },
  "status_pix": "Transferido",
  "client_payer_payments": []
}
```

- Pagamento realizado;

```
{
  "payment_id": "6093d4fd-78f0-447b-be8b-7891497ace1e",
  "order_id": "00000000-0000-0000-0000-000000000000",
  "transaction_id": "1122127312",
  "discount_value": 0,
  "value": 0.02,
  "date_time": "2023-01-19T11:22:53.1774379",
  "expire_date": "2023-01-20T15:09:33.1774063",
  "payment_method": "PIX",
  "remaining_amount": 0.02,
  "order": null,
  "refunds": [],
  "api_version": "v1",
  "id_end_to_end": "E60701190202301191424DY5RKDFHGOT",
  "payment_status": {
    "id": 7,
    "name": "Pagamento Realizado"
  },
  "status_pix": "Transferido",
  "payment_date": "2023-01-19T11:24:38",
  "refunds": []
}
```

```

"wallet": {
    "cnpj": "0000000000000000",
    "trading_name": "Pix",
    "mdr": 0,
    "deadline": 0
},
"client_payer_payments": []
}

```

- **Cancelamento**

```

{
    "payment_id": "6093d4fd-78f0-447b-be8b-7891497ace1e",
    "order_id": "e18563f8-1c49-491b-a9b2-763e70b33d71",
    "transaction_id": "1122127312",
    "discount_value": 0,
    "value": 0.02,
    "date_time": "2023-01-19T11:22:53.1774379",
    "payment_method": "PIX",
    "remaining_amount": 0,
    "order": {
        "order_number_ext": "896819"
    },
    "refunds": [
        {
            "refund_id": "71df1fff-11eb-49a9-d489-08daf43f2f7c",
            "value": 0.02,
            "date_time": "2023-01-19T11:27:17.4639823",
            "refund_status_code": 2,
            "cancel_numberext": "253780",
            "financial_transaction": true,
            "api_version": "V2",
            "refund_status": "Ok"
        }
    ],
    "api_version": 0,
    "id_end_to_end": "E60701190202301191424DY5RKDFHGOT",
    "payment_status": {
        "id": 4,
        "name": "Estornado"
    },
    "status_pix": "Devolvido",
    "payment_date": "2023-01-19T11:24:38",
    "wallet": {
        "mdr": 0,
        "deadline": 0
    },
    "client_payer_payments": []
}

```

## 6.4. Confirmar Pagamento

Após o QrLinx receber o callback de “pagamento realizado” pela carteira digital ou PIX, o pagamento passará para o “status 7 – Pagamento Realizado”. Automação deverá realizar a confirmação da cobrança por intermédio de um POST para [{url}/v1/payment/order/confirm](#), porém não é necessário aguardar o retorno caso automação não utilize os dados apresentados no report. O objeto “REPORT” é o campo que possui os dados do pagamento, a automação poderá utilizar esses dados para complementar o cupom fiscal.

Assim que o pagamento for confirmado passará para “status 5 – Pagamento Finalizado”.

### Método payment order confirm

Parâmetro	Descrição	Tipo	Obrigatório
pos_id	Id do PDV que realizou a venda.	String (36)	Sim
payments	Agrega as informações necessárias à confirmação do pagamento.	objeto	Sim

### Objeto payments

Parâmetro	Descrição	Tipo	Obrigatório
payment_id	Identificação única do pagamento informada no momento da criação.	String (36)	Sim
discount_value	Valor do desconto concedido (caso exista).	Decimal (7,2)	Não
value	Valor total da venda.	Decimal (7,2)	Sim

### Exemplo de Requisição

```
{
  "pos_id": "A2C2223F-A9AE-4F07-A866-0CEDF34AB46B",
  "payments": [
    {
      "payment_id": "f9130830-ee84-4857-80c6-7cbaedc243b9",
      "discount_value": 0.0,
      "value": 1.50
    }
  ]
}
```

### Retorno da Requisição

```
{
  "success": true,
  "message": "pagamentos finalizados com sucesso",
  "data": {
    "pos_id": "A2C2223F-A9AE-4F07-A866-0CEDF34AB46B",
    "date_time": "2022-05-04T12:05:49.9229518",
    "date_confirm": "2022-05-04T12:14:20.8810858",
    "payments": [
      {
        "payment_id": "ac0bf2ac-7278-493f-98ba-f1c867cb1f53",
        "payment_methods": "CREDITO",
        "report": "##FORMA PAGAMENTO.:CREDITO##WALLET_AUTH.:Carteira Teste Linx##AUT.:731aa60##CODIGO TERM.:QRQLINX6KVKG##CODIGO ESTAB.:05761098000113##DOC.:ac0bf2ac-7278-493f-98ba-f1c867cb1f53##DATA.:04/05/2022 12:14:20##CPF.:##MOD.:QRQLINX##VALOR.:1.50##IDENDTOEND.:",
        "order": {
          "order_number_ext": "fe665c80-8f71-499b-915c-e757f8da12af",
          "integration_type_pix": {
            "id": 3,
            "name": "Bradesco"
          }
        }
      ],
      "id_end_to_ends": []
    }
  }
}
```

Consulte o [Manual de Boas Práticas](#) do QRlinx para mais informações sobre o processo de confirmação do pagamento.

## 6.5. Cancelar Pagamento

A pedido do cliente (desistência da compra; utilização de outra forma de pagamento etc.) ou por motivos de força maior (p.ex. falta de energia) o pagamento pode ser desfeito ou cancelado.

O QRlinx possui duas situações quando ocorre uma solicitação de cancelamento:

- Desfazimento: Transação foi cancelada antes da transação financeira;
- Cancelada: Transação foi cancelada após a confirmação do PDV e gerou um estorno para o cliente.

Para desfazer ou cancelar o pagamento, devemos realizar um *PATCH* para [{url}/v2/payment/cancel](#) com os parâmetros abaixo.

O serviço de cancelamento é assíncrono, com isso, API sempre retorna o status Pendente na requisição, pois será necessário cancelar a transação na carteira responsável pelo pagamento.

Após a solicitação do cancelamento é necessário realizar uma consulta do status do pagamento ([Item 7](#)), para garantir que o desfazimento ou cancelamento tenha ocorrido com sucesso.

Os status de estorno e desfazimento possíveis podem ser consultados no [item 9.2](#) deste manual.

### Método payment cancel

Parâmetro	Descrição	Tipo	Obrigatório
payment_id	Identificação única do pagamento no QRLinx.	String (36)	Sim
pos_id	Id do PDV que realizou a venda.	String (36)	Sim
value	Valor da venda.	Decimal (7,2)	Sim
cancel_numberext	Número da ordem gerada pela automação para controle dos cancelamentos.	String (60)	Sim
reason_message	Informar o motivo que está ocorrendo o desfazimento.	String (255)	Não

### Exemplo de Requisição

```
{
  "payment_id": "cce95190-a305-4ff9-8b5b-4d57d9ab16f6",
  "pos_id": "A2C2223F-A9AE-4F07-A866-0CEDF34AB46B",
  "value": 1.50,
  "cancel_numberext": "123456",
  "reason_message": "Venda teste, realizado o desfazimento"
}
```

### Retorno da Requisição

```
{
  "success": true,
  "message": "Cancelamento solicitado com sucesso.",
  "data": {
    "refund_id": "92ea9e05-3c9e-45f6-3765-08da2decb940",
    "value": 1.50,
    "date_time": "2022-05-04T13:41:00.8667916",
    "refund_status_code": 1,
    "cancel_numberext": "1234",
    "financial_transaction": false,
    "api_version": "V2",
    "refund_status": "Pendente"
  }
}
```

Consulte o [Manual de Boas Práticas](#) do QRLinx para mais informações sobre o processo de cancelamento do pagamento.

## 6.6. Consultar Carteira/ Wallet

Automação poderá consultar as carteiras que estão disponíveis para uma determinada loja, essa consulta se faz necessária na geração de uma cobrança específica, pois assim o PDV poderá solicitar a geração do QRCode apenas para as carteiras desejadas.

A consultada ocorre por intermédio de um *GET* para `{url}/v1/wallet/storewallet?cnpj={cnpj}`.

### Retorno da Requisição

Este é o retorno de uma consulta de carteira:

```
{
  "success": true,
  "message": "Carteiras retornadas com sucesso",
  "data": [
    {
      "cnpj": "22896431000110",
      "name": "Picpay Servicos S.A",
      "trading_name": "PicPay",
      "logo_link": "https://static.linxpayhub.com.br/images/logos/LogoPicPay.png",
      "logo_link_download": "https://static.linxpayhub.com.br/images/logos/LogoPicPay.png",
      "specific_wallet": "PICPAY"
    },
    {
      "cnpj": "0000000000000000",
      "name": "Pix",
      "trading_name": "Pix",
      "logo_link": "https://static.linxpayhub.com.br/images/logos/LogoPix.jpg",
      "logo_link_download": "https://static.linxpayhub.com.br/images/logos/LogoPix.jpg",
      "specific_wallet": "PIX",
      "integrationTypePixModel": {
        "id": 7,
        "name": "Pagar.me",
        "priority": "Primary"
      }
    },
    {
      "cnpj": "0000000000000000",
      "name": "Pix",
      "trading_name": "Pix",
      "logo_link": "https://static.linxpayhub.com.br/images/logos/LogoPix.jpg",
      "logo_link_download": "https://static.linxpayhub.com.br/images/logos/LogoPix.jpg",
      "specific_wallet": "PIX",
      "integrationTypePixModel": {
        "id": 5,
        "name": "Santander",
        "priority": "Secondary"
      }
    }
  ]
}
```

Na consulta o QRLinx retorna o nome da carteira como também o logo de cada uma, assim automação poderá exibir o logo e o cliente selecionar qual cobrança específica deseja criar.

Importante observar que tivemos dois retornos de PIX neste exemplo acima, isso é devido a loja possuir duas credenciais de PIX ativa, o PDV deverá sempre considerar o “Primary” como referência, já que é a chave primária e que todas as transações são direcionadas. O PIX com a informação “Secondary” somente é ativado quando o provedor primário está sendo afetado por alguma instabilidade, porém essa informação não é enviada para o PDV pois o MultiPSP mantém o cliente transacionando mesmo durante um processo de instabilidade.

Caso a loja não possua nenhuma carteira ativa será retornado uma lista vazia com a seguinte informação, “A loja consultada não possui carteiras ativas para transacionar”.

## 7. Consultar Cobrança – Contingência

Automação poderá consultar as informações de suas cobranças criadas, porém este serviço de consulta deverá atuar como uma contingência, caso automação não tenha recebido a resposta do webhook. Também poderá ser utilizado para consultas recorrentes nos casos que automação não possua o serviço de webhook disponível nos PDVs.

As consultadas ocorrem por intermédio de um *GET* para `{url}/v1/payment/check/{payment_id}`.

Segue abaixo a descrição de cada campo retornado pela API:

Parâmetro	Descrição
message	Informação sobre o status atual da cobrança.
data	Campo que contém as informações da cobrança consultada.
payment_id	Id da cobrança gerado pelo QRLinx.
order_id	Id da ordem gerado pelo QRLinx.
transaction_id	Id da transação gerado no autorizador da transação.
discount_value	Valor de desconto, caso seja aplicado pela loja.
value	Valor da cobrança criada pelo PDV.
date_time	Data e hora do momento em que a cobrança foi gerada no QRLinx.
expire_date	Data e hora que a cobrança será expirada caso não ocorra o pagamento.
payment_method	Método de pagamento utilizado pelo cliente para pagamento da cobrança.
remaining_amount	Valor que foi pago pelo cliente.
order	Objeto que contém a ordem informada pelo PDV.
order_number_ext	Descrição da ordem fornecida pelo PDV.
integration_type_pix	Objeto que identifica a integração PIX utilizada pela loja.
id	Enum das integrações PIX.
name	Nome da integração PIX utilizada pela loja.
refunds	Objeto que contém as tentativas de cancelamento solicitadas pelo PDV.
refund_id	Id da solicitação de cancelamento gerado pelo QRLinx.
value	Valor do cancelamento solicitado pelo PDV.
date_time	Data e hora da solicitação do cancelamento.
refund_status_code	Enum que representa o status do cancelamento, tabela de status disponível no <a href="#">item 9.2</a> .
cancel_numberext	Id de identificação do cancelamento gerado pelo PDV.
financial_transaction	Informação se o cancelamento teve movimentação financeira ou não
api_version	Versão da API de cancelamento utilizada pelo PDV.

refund_status	Status do cancelamento disponível no <a href="#">item 10.2</a> .
api_version	Versão da API transacional utilizada pelo PDV.
Id_end_to_end	Id de identificação da transação PIX gerado pelo BACEN.
payment_status	Objeto que contém o status da transação.
id	Enum que representa o status da cobrança, tabela de status disponível no <a href="#">item 10.1</a> .
name	Descrição do status da cobrança disponíveis no <a href="#">item 10.1</a> .
status_pix	Descrição do status da transação PIX, tabela de status disponível no <a href="#">item 10.3</a> .
payment_date	Data e hora que ocorreu o pagamento da cobrança.
wallet	Objeto que contém a informação da carteira que realizou o pagamento, no QRInx o PIX é registrado como uma carteira digital.
cnpj	CNPJ da carteira digital, para as transações de PIX é retornado o valor “00000000000000”.
trading_name	Nome da carteira que realizou o pagamento da cobrança.
mdr	Taxa que a loja possui acordada com a carteira.
deadline	Prazo para recebimento acordado com a carteira.
client_payer_payments	Objeto que contém a informação do cliente e banco pagador.
cnpj	CNPJ da instituição que realizou o pagamento da cobrança.
client_payer	Objeto que contém dados do cliente.
name	Nome do cliente que realizou o pagamento da cobrança.
document_type	Tipo de documento CPF ou CNPJ.
document_number	Número do documento truncado.
account	Objeto que contém informação do banco.
bank	Objeto que contém dados do banco.
name	Nome do banco que realizou o pagamento da transação.
ispb	Identificador dos bancos no sistema de transferência de reserva do Banco Central.
report	Informações da confirmação do pagamento, para que automação possa reutilizar os dados caso preciso.

## Retorno da Requisição

Este é o retorno de uma consulta de cobrança antes da realização do pagamento:

```
{
  "success": true,
  "message": "Processo de pagamento em andamento",
  "data": {
    "payment_id": "2af10f23-8064-4c5e-a4c5-348bd965b425",
    "order_id": "933c50c6-a70b-41d3-a1da-21f8f02a873d",
    "discount_value": 0.00,
    "value": 1.50,
    "date_time": "2022-05-04T09:29:36.376Z",
  }
}
```

```

"expiration_date": "2022-05-04T09:35:36.3764226",
"payment_method": "",
"remaining_amount": 1.50,
"order": {
    "order_number_ext": "80231",
    "integration_type_pix": {
        "id": 3,
        "name": "Bradesco"
    }
},
"refunds": [],
"api_version": "V1",
"payment_status": {
    "id": 6,
    "name": "Aguardando Pagamento"
},
"status_pix": "Pendente",
"client_payer_payments": []
}
}

```

E este é o retorno de uma consulta após a realização do pagamento com PIX:

```

{
    "success": true,
    "message": "Finalização do pagamento realizada com sucesso",
    "data": {
        "payment_id": "2f28cebc-bd93-4986-b9dc-bca56c1437f3",
        "order_id": "ead37877-6fe8-45ea-93be-89fc53588298",
        "transaction_id": "1291986617",
        "discount_value": 0.00,
        "value": 31.68,
        "date_time": "2023-03-28T17:50:09.037Z",
        "expire_date": "2023-03-28T18:40:09.037Z",
        "payment_method": "PIX",
        "remaining_amount": 31.68,
        "order": {
            "order_number_ext": "20230328002953100",
            "integration_type_pix": {
                "id": 3,
                "name": "Bradesco"
            }
        },
        "refunds": [],
        "api_version": "V1",
        "id_end_to_end": "E60701190202303282050DY5V1D47X8M",
        "payment_status": {
            "id": 5,
            "name": "Finalizado"
        },
        "status_pix": "Transferido",
    }
}

```

```

"payment_date": "2023-03-28T17:50:40.000Z",
"wallet": {
    "cnpj": "0000000000000000",
    "trading_name": "Pix",
    "mdr": 0.0,
    "deadline": 0
},
"client_payer_payments": [
{
    "cnpj": "60701190000104",
    "client_payer": {
        "name": "FERNANDO FERNANDO FERNANDES",
        "document_type": "CPF",
        "document_number": "***11111**"
    },
    "account": {
        "bank": {
            "name": "ITAÚ UNIBANCO S.A.",
            "ispb": "60701190"
        }
    }
},
],
"report": "##FORMA PAGAMENTO.:PIX##WALLET_AUTH.:Pix##AUT.:LINX0008D
B9E8B852EBCC1X1E69730421D8##CODIGO TERM.:QRLINXEZJ3WL##CODIGO ESTAB.
:33438250050008##DOC.:7dfb33df-a8b9-401b-9be7-297fa96da26c##DATA.:01
/01/0001 00:00:00##CPF.:##MOD.:QRLINX##VALOR.:36.69##IDENDTOEND.:
E2812760320230816190400075890839"
}
}
}

```

E este é o retorno de uma consulta após a solicitação do cancelamento:

```
{
"success": true,
"message": "Estorno do pagamento realizada com sucesso",
"data": {
    "payment_id": "65fe7f87-d251-48ef-aecd-9559189629a1",
    "order_id": "ce906652-a855-422d-bf8e-39cc6d38e267",
    "transaction_id": "c2d80d9",
    "discount_value": 0.00,
    "value": 5.00,
    "date_time": "2023-03-27T15:38:07.609Z",
    "expire_date": "2023-03-27T17:38:07.609Z",
    "payment_method": "CREDITO",
    "remaining_amount": 0.00,
    "order": {
        "order_number_ext": "9e2511c8-c269-40de-8e27-4a720f7f5019",
        "integration_type_pix": {
            "id": 3,
            "name": "PIX"
        }
    }
}
}
```

```
        "name": "Bradesco"
    }
},
"refunds": [
{
    "refund_id": "1c600587-80c7-4c02-e4c6-08db2ea4e6e8",
    "value": 5.00,
    "date_time": "2023-03-27T15:41:53.125Z",
    "refund_status_code": 2,
    "cancel_numberext": "944528",
    "financial_transaction": true,
    "api_version": "V2",
    "refund_status": "Ok"
},
],
"api_version": "V1",
"payment_status": {
    "id": 4,
    "name": "Cancelado"
},
"status_pix": "Pendente",
"payment_date": "2023-03-27T15:38:27.323Z",
"wallet": {
    "cnpj": "20893225000121",
    "trading_name": "Carteira Teste Linx",
    "mdr": 0.0,
    "deadline": 0
},
"client_payer_payments": [
{
    "cnpj": "20893225000121",
    "account": {
        "bank": {
            "name": "Carteira Teste Linx"
        }
    }
}
]
}
```

## Rate Limit

A rota de consulta de pagamento (check/{payment\_id}) tem um rate limit vinculado.

Para não ser impactado por essa tratativa de segurança, a consulta de pagamento deve ser realizada após 10 segundos do QRCode aparecer em tela e a cobrança em questão pode ser consultado 5 vezes a cada minuto, desta forma automação não será impactada pelo rate limit.

O retorno do rate limit é o seguinte: `{"statusCode": 429, "message": "Rate limit is exceeded. Try again in 23 seconds."}`.

Toda vez que este retorno for recebido, ou seja, o status code 429 for recebido após uma consulta da automação, isso significa que foi atingido o rate limit deste pagamento, dentro da mensagem do body de resposta tem em quantos segundos pode ser realizada uma nova consulta de pagamento.

Para facilitar, dentro do header de resposta da requisição, é retornado 3 header auxiliares.

- **Remaining-Calls:** Quantidade de chamadas restantes que estão disponíveis para realizarem a consulta da cobrança dentro do período de 1 minuto

- **Total-Calls:** Quantidade de chamadas totais que estão disponíveis dentro do período de 1 minuto

Headers		200 OK	711 ms	1.25 KB	Save as Example	...
Key	Value					
Transfer-Encoding	chunked					
Content-Type	application/json; charset=utf-8					
Content-Encoding	gzip					
Vary	Accept-Encoding					
Request-Context	appld=clid-v1:f6748e45-2df0-471d-b7da-150be46fc3...					
Strict-Transport-Security	max-age=2592000					
X-Powered-By	ASP.NET					
Host	api.linxpayhub.com.br					
Set-Cookie	api.linxpayhub.com.br					
Remaining-Calls	6					
Total-Calls	60					
Date	Tue, 02 May 2023 20:39:23 GMT					

- **Retry-After:** Caso o rate limit seja atingido, ou seja, receber um http status code 429, este header indica em quanto tempo você pode fazer uma nova consulta para este pagamento.

Headers		429 Too Many Requests	122 ms	241 B	Save as Example	...
Key	Value					
Content-Length	84					
Content-Type	application/json					
Retry-After	19					
Total-Calls	60					
Date	Tue, 02 May 2023 20:38:59 GMT					

Reforçando que o rate limit é por cobrança criada, e não pela automação ou pelo pdv.

## 8. Integrações Parceiros

No momento de realizar a criação de uma cobrança é necessário que automação/PDV informe alguns dados para geração do QRCode. Porém algumas integrações solicitam campos específicos que se tornam obrigatórios para criação com sucesso da cobrança.

Abaixo iremos discriminar os campos básicos para criação de uma cobrança (Integração Padrão) e as integrações que demandam campos específicos.

### 8.1. Integração Padrão

Na integração padrão temos os campos fundamentais que toda automação deve enviar no momento de gerar uma cobrança, abaixo segue as carteiras e provedores de PIX neste modelo:

- **Carteiras:** Ame, Cash Berti, Mercado Pago, PicPay;
- **PIX:** Stone, Pagarme, Bradesco, Cielo, Hiper, Pefisa, Realize, Santander, Saqpay.

Segue abaixo os campos que são obrigatórios na criação da cobrança e aqueles que são apenas recomendados caso automação possua:

Parâmetro	Obrigatório	Recomendado
order_numberext	X	
pos_id	X	
amount	X	
subtotal	X	
payments	X	
payments.value	X	
payments.expiration_time		X
order_items		X

### 8.2. Integração PIX Saque

O Pix Saque permitirá que todos os clientes de qualquer participante do Pix realizem um saque em um dos pontos que ofertar o serviço. Estabelecimentos comerciais, redes de caixas eletrônicos (ATMs) compartilhados e os próprios participantes do Pix, por meio de seus ATMs próprios, poderão ofertar o serviço. Para ter acesso aos recursos em espécie, basta que o cliente faça um Pix para o agente de saque, em dinâmica similar à de um Pix normal, a partir da leitura de um QRCode mostrado ao cliente ou a partir do aplicativo do prestador do serviço.

Importante que para gerar um PIX Saque automação deverá enviar o campo “payments.value” zerado, já que não está ocorrendo uma compra e sim apenas um saque junto ao estabelecimento.

Segue abaixo os campos que são obrigatórios na criação da cobrança PIX Saque:

Parâmetro	Obrigatório	Recomendado
order_numberext	X	
pos_id	X	
amount	X	
subtotal	X	
payments	X	
withdraw_pix	X	

Não é permitido gerar uma cobrança de PIX Saque e PIX Troco em conjunto, caso os campos específicos sejam enviados o QRLinx retornará erro na criação da cobrança.

### 8.3. Integração PIX Troco

No caso do Pix Troco, a dinâmica é idêntica, com a diferença que o saque de recursos em espécie acontece junto com a realização de uma compra no agente de saque. Nesse caso, o Pix é feito pelo valor total (compra + saque). O extrato do cliente evidenciará o valor correspondente ao saque e o valor correspondente à compra.

Importante que para gerar um PIX Troco automação deverá enviar o campo “payments.value” conforme a compra realizada pelo cliente, e no campo “change\_pix” informar o valor que o cliente deseja sacar/receber de troco. Por fim o campo “amount” representa a soma dos dois campos acima na criação da cobrança.

Segue abaixo os campos que são obrigatórios na criação da cobrança PIX Saque:

Parâmetro	Obrigatório	Recomendado
order_numberext	X	
pos_id	X	
Amount	X	
Subtotal	X	
Payments	X	
payments.value	X	
change_pix	X	

Não é permitido gerar uma cobrança de PIX Saque e PIX Troco em conjunto, caso os campos específicos sejam enviados o QRLinx retornará erro na criação da cobrança.

## 8.4. Integração NuPay

Na integração com a Nupay, o pagamento não depende da leitura de um QRCode, assim que o PDV solicitar a criação da cobrança o cliente do lojista será notificado via push no celular, para concluir o pagamento no próprio aplicativo da Nubank. Caso o PDV solicite uma cobrança específica com a NuPay o QRLinx não irá retornar a string do QRCode.

A pedido da NuPay é recomendado que na utilização dessa integração seja utilizado o campo “specific\_wallets”, gerando assim uma cobrança específica com a carteira. [Clique aqui](#) para visualizar o modelo da integração.

Segue abaixo os campos que são obrigatórios na criação da cobrança e aqueles que são apenas recomendados caso automação possua:

Parâmetro	Obrigatório	Recomendado
payer	X	
client_identification	X	
specific_wallets		X

## 8.5. Integração PIX Paga Leve

Na integração com a Paga Leve temos uma integração de PIX com a opção do parcelamento, devido a isso um novo campo na criação da cobrança passa a ser obrigatório para geração da cobrança com sucesso.

Disponibilizamos uma API específica com a Paga Leve para que a loja possa consultar a disponibilidade do PIX Parcelado, assim o lojista poderá oferecer um parcelamento de pix antes da finalização da transação:

### 8.5.1. Pré Aprovação – PIX Parcelado

Automação poderá consultar se o cliente possui o PIX Parcelado de forma pré-aprovada, ou seja, de acordo com o valor da transação e o CPF do cliente a Paga Leve analisa uma oferta de parcelamento da transação PIX.

A consultada da pré-aprovação ocorre por intermédio de um *POST* para {url}/v1/payment/installment-availability.

#### Método installment-availability

Parâmetro	Descrição	Tipo	Obrigatório
pos_id	Id do PDV que realizará a venda.	String (36)	Sim
value	Valor da venda.	Decimal (7,2)	Sim
document	Número do CPF do cliente que está realizando a compra.	String (11)	Sim

**Exemplo de Requisição**

```
{
  "pos_id": "A2C2223F-A9AE-4F07-A866-0CEDF34AB46B",
  "value": 1.50,
  "document": "12345678901"
}
```

**Retorno da Requisição**

```
{
  "sucesso": true,
  "message": "Consulta realiza com sucesso",
  "data": {
    "status": "APPROVED"
  }
}
```

Essa API possui apenas duas mensagens de retorno “Approved” ou “Declined”, assim a loja terá a informação se o PIX Parcelado está aprovado ou não para o seu cliente.

Caso o cliente do lojista seja aprovado no pagamento do PIX parcelado automação deverá enviar o campo “is\_pix\_upfront” com o valor “false”. O QRlinx retornará no campo “payment\_link” o link de pagamento da Paga Leve que poderá ser exibido ou enviado para o cliente concluir o processo de pagamento.

Na transação de PIX normal em que não envolve o parcelamento o campo “is\_pix\_upfront” não precisa ser enviado, caso seja enviado deverá conter o valor “true”, o QRlinx retornar o QRCode para pagamento no PDV, mas também retornará o “payment\_link” da Paga Leve que poderá ser exibido ou enviado para o cliente concluir o processo de pagamento.

[Clique aqui](#) para visualizar um modelo de integração com a Paga Leve.

Segue abaixo os campos que são obrigatórios na criação da cobrança com Paga leve, lembrando que os campos da integração padrão também deve ser considerados:

Parâmetro	Obrigatório	Recomendado
payer	X	
client_identification	X	
is_pix_upfront	X	
client_phone		X

## 8.6. Integração PicPay E-Commerce

Na integração com e-PicPay, que são transações específicas para o e-commerce, a loja possui as opções de realizar uma captura tardia, parcial ou até mesmo se preciso realizar um estorno parcial após a captura da transação.

O processo de captura tardia depende do campo `auto_capture` seja enviada de forma correta, onde automação/PDV irá informar se loja pretende realizar uma captura tardia ou não. Também é necessário a identificação do cliente comprador.

Segue abaixo os campos que são obrigatórios na criação da cobrança com e-PicPay, lembrando que os campos da integração padrão também deve ser considerados:

Parâmetro	Obrigatório	Recomendado
<code>payer</code>	X	
<code>client_identification</code>	X	
<code>soft_descriptor</code>		X
<code>auto_capture</code>	X	

## 8.7. Integração Pede Pronto

Na integração com a Pede Pronto, é de suma importância automação/PDV informar se a transação possui taxa de serviço como também se terá algum subsídio relacionado a cobrança. Também é solicitado a identificação do cliente comprador e os itens da compra acompanhado de suas taxas e subsídio caso possua.

Segue abaixo os campos que são obrigatórios na criação da cobrança com Pede Pronto, lembrando que os campos da integração padrão também deve ser considerados:

Parâmetro	Obrigatório	Recomendado
<code>servicesAmount</code>	X	
<code>subsidizedAmount</code>	X	
<code>payer</code>	X	
<code>order_items</code>	X	

## 8.8. Integração Split de Pagamento

O Split de Pagamento é um meio da loja dividir o valor da transação com outros recebedores, uma função utilizada por grandes marketplaces.

Para utilização dessa integração é necessário que o lojista possua a função Split ativa tanto no QRLinx quanto no portal da PagarMe, logo após ativação o usuário poderá criar os recebedores vinculados a chave PIX no portal do QRLinx.

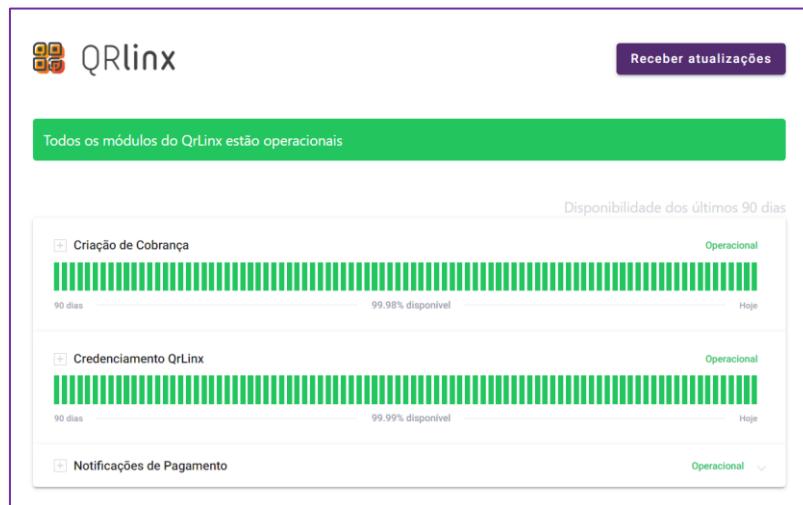
Segue abaixo o objeto obrigatório na criação da cobrança com Split, lembrando que os campos da integração padrão também deve ser considerados:

Parâmetro	Obrigatório	Recomendado
Payments.split	X	

## 9. Status Page QRLinx

Para auxiliar os clientes e também as automações sobre os incidentes que estão ocorrendo nos provedores de PIX parceiros ou até mesmo no QRLinx, disponibilizamos uma status page com as informações do incidente e data de início e fim do ocorrido. Clique aqui para acessar à página.

A status page exibe as informações do QRLinx em tempo real e se os nossos módulos estão operacionais ou não:

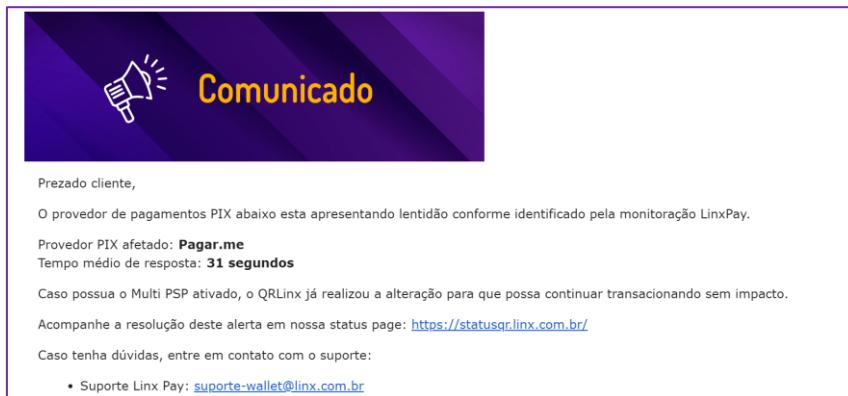


As informações sobre os incidentes são armazenadas por até 7 dias.

É possível realizar o cadastro para receber as notificações por e-mail, passando as informações de Nome, E-mail e Telefone. Sempre que um incidente ocorrer será disparado um e-mail para informando a indisponibilidade no parceiro.

The form is titled "Receber atualizações" and has two options: "E-mail/Telefone" (selected) and "Webhook". A descriptive text below says: "Receba notificações por e-mail sempre que o Payhub criar, atualizar ou resolver um incidente." There are three input fields: "Nome", "E-mail", and "(xx) XXXXX-XXXX". At the bottom is a "Se inscrever" button.

Este é uma modelo da notificação por e-mail:



É possível realizar também o cadastro de um webhook, desta forma o QRlinx notificará o sistema/automação para que possa informar os lojistas diretamente:

The form is titled 'Receber atualizações' and has two tabs: 'E-mail/Telefone' and 'Webhook', with 'Webhook' being selected. Below the tabs, there is a description: 'Receba notificações de webhook sempre que o Payhub criar, atualizar ou resolver um incidente.' There are two input fields: 'E-mail' and 'Endpoint', both with placeholder text. At the bottom is a blue button labeled 'Se inscrever'.

Após a realização do cadastro será enviado um e-mail para confirmação do webhook, é necessário realizar a confirmar para finalizar o cadastro. Realizada a confirmação o primeiro webhook enviado será de status do cadastro com a informação “CHECK”.

Segue abaixo o exemplo de notificação informando o incidente via webhook:

```
{
  "Id": "c61ec785-f048-4edd-98b3-66829414691a",
  "IntegrationType": "PIX",
  "Problem": "SLOWNESS",
  "Status": "OPEN",
  "Provider": "SICOOB",
  "StarDate": "2024-07-29T21:38:00.0287821Z"
}
```

Segue abaixo o exemplo de notificação informando o encerramento do incidente via webhook:

```
{
  "Id": "c61ec785-f048-4edd-98b3-66829414691a",
  "IntegrationType": "PIX",
  "Problem": "SLOWNESS",
  "Status": "CLOSED",
  "Provider": "SICOOB",
  "StarDate": "2024-07-29T21:38:00.0287821Z",
  "EndDate": "2024-07-29T22:18:00.0287821Z"
}
```

Importante destacar que os horários acima estão com no padrão UTC.

## 10. Tabelas de Status e Erros

### 10.1. Status do pagamento

Status	Descrição
6 – Aguardando Pagamento	Ocorre quando o pagamento foi criado pelo PDV.
7 – Pagamento Realizado	Ocorre quando o pagamento foi autorizado.
5 – Finalizado	Ocorre quando o pagamento foi confirmado pelo PDV.
9 – Desfeito	Ocorre quando o PDV solicita o cancelamento antes de ser autorizado.
4 – Cancelado	Ocorre quando o PDV solicita o cancelamento após a confirmação.

### 10.2. Status do cancelamento

Status	Descrição
0 - Erro	Ocorre quando a solicitação de cancelamento apresenta falha na solicitação junto ao PSP, automação não necessita mais consultar o status da transação, recomendamos acionar o RC da Linx para entendimento da falha ou o cliente realizar uma nova tentativa via Portal do QRLinx.
1 - Pendente	Ocorre quando a solicitação de cancelamento está em processamento.
2 - Ok	Ocorre quando a solicitação de cancelamento foi autorizada.
3 - Negado	Ocorre quando a solicitação de cancelamento não foi autorizada, e a transação permanece com seu status anterior, recomendamos acionar o RC da Linx para entendimento da falha ou o cliente realizar uma nova tentativa via Portal do QRLinx.

### 10.3. Status PIX

Status	Descrição
1 - Devolvido	Valor transacionado foi estornado com sucesso.
2 – Devolução Pendente	Transação está no processo de devolução.
3 – Não Devolvido	Valor não pode ser estornado, necessário acionar a sustentação para análise do motivo.
4 – Pendente	Aguardando notificação de pagamento.
5 – Transferido	Transação ocorreu com sucesso.
6 – Desfeito	QRCode expirou sem notificação do pagamento.

## 10.4. Tabela de erros

### Código de Erro    Descrição do erro

LINX001	Data inválida
LINX002	Existem campos obrigatórios a serem preenchidos
LINX003	Carteira parceira não está apta a realizar pagamento na loja escolhida
LINX004	Carteira parceira não está apta a realizar pagamento no Wallet Hub
LINX005	Ponto de venda não encontrado
LINX006	Ponto de venda inativo
LINX007	Ponto de venda já informado e cancelado
LINX008	Ponto de venda já informado e finalizado
LINX009	CNPJ inválido
LINX010	Código da transação inválido
LINX014	Loja inativa
LINX016	Não existe pagamento em execução a ser desistido
LINX017	Não foi possível cancelar o pagamento por problemas externos
LINX018	Não foi possível confirmar o pagamento por problemas externos
LINX019	Não foi possível excluir o pagamento por problemas externos
LINX020	O código do pagamento é inválido
LINX022	O pagamento informado consta cancelado
LINX023	O pagamento informado consta excluído
LINX024	O pagamento informado consta finalizado
LINX025	Pagamento não consta em estado finalizado
LINX026	Pagamento não consta em estado realizado
LINX027	Pagamento não encontra em estado de aguardando pagamento
LINX028	Pedido inválido
LINX029	Pedido não está aguardando a confirmação
LINX030	QRCode Inválido
LINX031	Status do pagamento inválido
LINX032	Valor do desconto maior que o valor do pagamento
LINX033	Valor do pagamento não pode ser zerado
LINX040	Ao criar um pedido, só pode ser enviado um pagamento por vez
LINX041	Não existe um desconto para a data atual

LINX042	Já existe uma carteira parceira em processo de pagamento
LINX043	Loja não encontrada
LINX044	Os pagamentos enviados são diferentes do contido no sistema
LINX045	A quantidade de pagamentos com status 'Pagamento Realizado' não confere com a enviada para confirmação
LINX046	O valor informado do pagamento é diferente do contido no sistema
LINX047	O valor do desconto informado no pagamento é diferente do contido no sistema
LINX048	O comprimento da string está incorreto
LINX049	Não foi possível obter as informações do QRCode
LINX050	Houve um erro na conversão
LINX051	Não foi possível gerar um QRCode corretamente
LINX052	QRCodeGenerator não conhecido
LINX054	O pedido está com o status 'Pedido Finalizado'
LINX055	Ponto de venda inválido
LINX056	Pedido não encontrado
LINX057	Foi enviado 2 paymentId diferentes
LINX058	Integrações que enviam o payment, não devem enviar o topic ou id via querystring
LINX059	Objeto de integração inválido
LINX060	Tópico inválido
LINX061	Objeto de integração não encontrado
LINX062	Formato inválido
LINX063	Já existe um ponto de venda com esta descrição para a loja
LINX064	Identificador de ponto de venda não encontrado
LINX066	Desconto para o dia atual não encontrado
LINX067	TransactionId é diferente do pagamento informado
LINX068	Ocorreu um erro com uma integração. Verifique a mensagem
LINX069	Ponto de venda não aceito para confirmar o pedido
LINX071	Versão da API incorreta
LINX072	O ponto de venda não é da loja informada
LINX073	A carteira não foi encontrada

LINX074	Não foi possível realizar todos os estornos do pagamento com a integradora
LINX075	Cliente não encontrado
LINX076	Já existe uma loja cadastrada com este CNPJ
LINX077	Pagamento não encontrado
LINX081	Ocorreu um erro no cancelamento, solicitar cancelamento via email!
LINX082	External reference da integração é diferente
LINX083	External reference está em um formato inválido
LINX084	Valor do pagamento da integração é diferente
LINX089	Software do PDV não encontrado
LINX091	Pagamento não encontra em estado de aguardando pagamento
LINX092	Não foi encontrado a configuração de integração para esta conciliação
LINX093	Valor de pagamento inválido
LINX094	Existem campos obrigatórios a serem preenchidos
LINX095	Erro ao enviar a mensagem para a fila
LINX097	Erro ao obter localização
LINX098	A lista de lojas não foi enviada
LINX099	A loja só pode ser ativada ou inativada. Utilize apenas Ativo ou Inativo no Status
LINX0ER	Erro não identificado
LINX100	Já existe este relacionamento entre marca e carteira. Atualize o já existente
LINX101	Não foi encontrado o relacionamento entre marca e carteira para ser atualizado
LINX103	Loja já usou todas as licenças disponíveis para credenciamento de ponto de venda
LINX104	A Data Inicial não deve ser superior a Data Final
LINX106	Não é possível autorizar um pagamento sem o código da autorização transaction_id.
LINX107	O Software PDV Id é inválido.
LINX108	Configuração da carteira não encontrada!
LINX109	Já existe está configuração para carteira!
LINX110	O endereço fornecido é inválido.

LINX111	Já existe uma configuração para este tipo de integração com esta carteira
LINX112	Não foi encontrada uma integrationHttpRequest para ser atualizada
LINX113	IntegrationAuthenticationConfigId diferente do existente
LINX114	A integração não possui um IntegrationAuthenticationConfig para ser atualizado
LINX115	Você enviou um integrationHttpRequest com IntegrationHttpParameters, porém, ela não existe para ser atualizada
LINX116	O IntegrationHttpParameterId {0} - não é existe para ser atualizado
LINX117	A carteira não pode ser alterada.
LINX120	Apim-Company-Target e Apim-Company-Channel precisam ser enviadas no header
LINX121	Pagamento não encontra em estado de aguardando confirmação de cancelamento
LINX122	Pagamento não encontra em estado de aguardando confirmação de cancelamento
LINX123	Callback não aceito
LINX124	O estorno não foi encontrado para adicionar o código externo
LINX125	Callback não aceito
LINX127	Não autorizado
LINX128	Identificador(es) inválidos(s)
LINX129	Callback aceito
LINX132	Status não aceito no filtro
LINX133	Configuração não encontrada
LINX134	A soma dos valores informados para as pendências e dos PDVs ativos para esta loja, ultrapassam a quantidade máxima de PDVs permitidos para essa loja. Por favor, consulte a quantidade máxima de PDVs em Consulta Lojas.
LINX135	Email não encontrado
LINX136	Lista de lojas é vazia ou nula
LINX138	Método de pagamento já criado!

LINX139	Método de pagamento não encontrado
LINX140	Ordem de serviço não encontrada
LINX141	Display Model não encontrado
LINX142	Esse modelo de display deverá ser igual a quantidade total de PDVs preenchidos no credenciamento.
LINX143	QRCode não apto para transacionar
LINX146	Url é obrigatório para pagamento online.
LINX147	A descrição e external code enviados não se referem ao mesmo ponto de venda.
LINX148	Não é possível alterar o channel de um ponto de venda que já foi ativado.
LINX149	Carteira não apta para forma de pagamento online.
LINX150	A ordem do pagamento informado é diferente do contido no sistema.
LINX151	Ponto de venda não apto para forma de pagamento online.
LINX152	Dados do endereço deve ser preenchido.
LINX153	Dados de identificação do cliente deve ser preenchido.
LINX154	Dados do telefone deve ser preenchido.
LINX155	Dados do cliente deve ser preenchido.
LINX156	Essa loja ainda não está vinculada com a carteira informada para exclusividade.
LINX157	QRCode não apto para esta carteira.
LINX158	Modelo de QRCode inválido, utilize o Gerar QRCode para Transferência.
LINX159	O valor a ser estornado não pode ser maior que o valor total.
LINX160	A carteira em questão, ainda não está apta para realizar cancelamento parcial.
LINX161	Pagamento já possui um cancelamento parcial em andamento, para realizar outro cancelamento parcial, informe o valor a ser cancelado.
LINX162	Pagamento Pix não encontrado na base de dados.
LINX163	Ocorreu um erro na tentativa de estornar pagamento no pix.
LINX164	Pagamento não estornado no pix, timeout.
LINX165	Vínculo entre a loja e carteira não encontrado
LINX166	O tipo de integração não existe
LINX167	Dict não encontrado

LINX171	Falha na leitura do arquivo .xls/.xlsx
LINX172	Não é permitido para este endpoint o credenciamento para a integração
LINX173	Campos ExternalCode e ExternalCodeType devem ser preenchidos para integração MundiPagg
LINX174	O tipo no campo ExternalCodeType só pode ser ClientSecret para integração MundiPagg
LINX175	Erro ao deserializar json.
LINX176	Estorno não efetuado.
LINX177	Cancelamento em duplicidade.
LINX178	O valor do cancelamento deve ser igual ao do pagamento.
LINX179	O CancelNumberExt é obrigatório.
LINX196	Não foi possível gerar a cobrança pix.
LINX200	Existe uma solicitação de cancelamento pendente, aguarde a confirmação do cancelamento.



**0800 701 5607**  
**linx.com.br**

Visite nossas páginas  
nas mídias sociais:

