
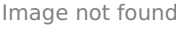


Emissão Sicoob - Via API

Analista Responsável: Gustavo Henrique Braga Fernandes Esse documento visa descrever os processos básicos para a integração com a API do Banco Sicoob. Inicialmente será implementada a emissão de boleto

- [Primeiros Passos - Liberação - criação aplicativo](#)
- [Sandbox \(homologação\)](#)
- [Endpoints Produção \(Exemplos JSON\)](#)

Primeiros Passos - Liberação - criação aplicativo

| | | |
|--|---|------------------------------------|
|   | Projeto/Sistema: Emissão Sicoob - Via API | Versão do Template: 1.1 |
| | Processo: Primeiros Passos - Liberação - criação aplicativo e sandbox | Versão do Documento: 1.0 |
| | Responsável(eis): Gustavo Fernandes | Data: 02/01/2025 |

1. Introdução

Este processo visa exemplificar o cadastro para libeação e criação do aplicativo, para emissão de boletos no banco Sicoob - via API.


Processos Relacionados

- Sandbox (homologação)
- Endpoints Produção (Exemplos JSON)

Especificação Funcional

PROC001 CADASTRO NO PORTAL:

Antes de tudo, será necessário possuir uma conta no "Portal Developers - Sicoob".

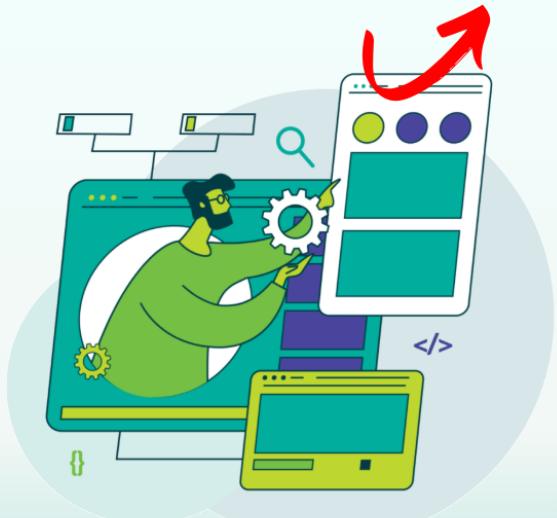



Catálogo de APIsDocumentaçãoEmpresas parceirasCadastrtoEntrar

Portal do Sicoob para DESENVOLVEDORES

Aqui o **desenvolvedor** pode utilizar nossos serviços financeiros de forma ágil e segura para aprimorar a sua experiência na utilização das **nossas APIs**.

Comece agora





Fale com a Alice

Será necessário preencher todos os dados para cadastro, inclusive o **CPF** do representante da empresa.

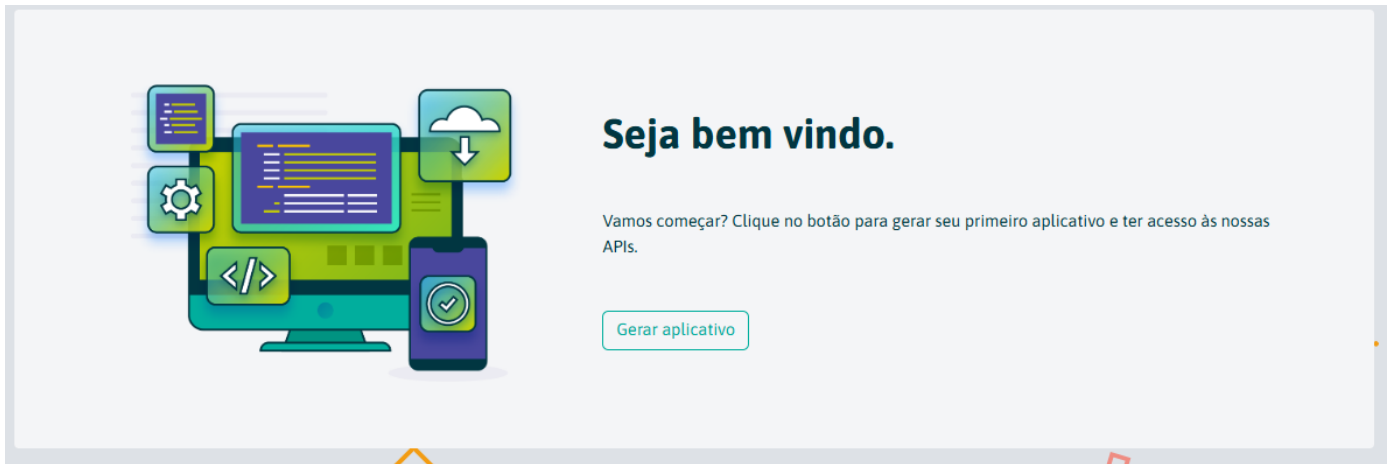
Cadastre-se

Faça o seu cadastro para ter acesso às apis:

| Pessoa Jurídica | Pessoa Física |
|--|---|
| Nome fantasia <input type="text" value="Digite nome fantasia"/> | CNPJ <input type="text" value="00.000.000/0000-00"/> |
| Dados do representante da empresa | |
| Nome completo <input type="text" value="Digite seu nome sem abreviações"/> | CPF <input type="text" value="000.000.000-00"/> |
| E-mail <input type="text" value="Digite o seu e-mail"/> | Confirmar e-mail <input type="text" value="Confirme o seu e-mail"/> |
| Descrição <input type="text" value="Conte-nos um pouco sobre seu projeto..."/> | Contato <input type="text" value="(00) 00000-0000"/> |
| <input type="checkbox"/> Eu aceito os termos de uso . | |
| <div>Criar cadastro</div> | |

Gerar Aplicativo (Ambiente produção):

Após realizar o acesso, cadastrando-se no portal, será necessário criar um novo aplicativo junto ao cliente.



1. Será necessário ser um cooperado e utilizar o AppSicoob para conseguir.
(Fazer o procedimento junto com um cliente que já é cliente do banco Sicoob)



2. Seguir o passo a passo conforme documentação:

[Passo a passo criação aplicativo](#)

Os cooperados que possuem aplicativos da API de Cobrança Bancária V1 no fluxo de *Authorization Code* não serão impactados. As novas solicitações deverão ocorrer no fluxo *Client Credential* e oportunamente deve ser realizada a migração dos cooperados cujo fluxo ainda esteja no método de autenticação antigo.

Para criar seus aplicativos, siga os passos abaixo:

- 1) Faça login no Portal Developers e acesse o menu Meus Aplicativos.
- 2) Acesse a opção Nova Aplicação.
- 3) Caso o seu cadastro esteja desatualizado há algum tempo, será exigido a atualização das informações de sua conta.
- 4) Faça login com dados de sua cooperativa, conta-corrente ou chave de acesso e senha do Aplicativo Sicoob Mobile Banking para confirmar a criação do aplicativo em sua conta.
- 5) Será enviado um código de verificação no seu **APP Sicoob**. Caso não tenha recebido o código, vá ao ícone de pesquisa do aplicativo Sicoob, busque por "Notificação". Ao acessar, irá aparecer as opções de "Notificação" e "Novidades", caso esteja habilitado, por gentileza, desabilite e habilite novamente.
- 6) prossiga com o cadastro do aplicativo seguindo os passos da tela de geração da aplicação que será apresentada.
- 7) Após a geração, as aplicações de acesso e as URLs de produção estarão disponíveis no menu Meus Aplicativos para cada aplicativo criado.

Confira abaixo vídeos de passo a passo para a criação de aplicativos no Portal Developers do Sicoob.

Autenticação

O fluxo de autenticação *Client Credentials* pode ser acessado pela URL abaixo:

Geração do token:



<https://auth.sicoob.com.br/auth/realms/cooperado/protocol/openid-connect/token>

Você poderá visualizar o passo a passo de geração de token via Postman no vídeo disponibilizado no link a seguir.

Vídeo exemplo aqui.

O fluxo de autenticação utilizado pelas APIs do Sicoob requer a utilização de **certificado digital** emitido por uma entidade certificadora **ICP Brasil** e deve ser emitido para o **CNPJ do cooperado**, quando PJ (Pessoa Jurídica) e para CPF do cooperado quando PF (Pessoa Física).

Sandbox (homologação)

| | | |
|--|---|------------------------------------|
|   | Projeto/Sistema: Emissão Sicoob - Via API | Versão do Template: 1.1 |
| | Processo: Sandbox (homologação) | Versão do Documento: 1.0 |
| | Responsável(eis): Gustavo Fernandes | Data: 08/01/2025 |

1. Introdução

Este processo visa exemplificar a utilização do ambiente sandbox para registro de boletos no banco Sicoob - via API.

Processos Relacionados

[Primeiros Passos - Liberação - criação aplicativo](#)

[Endpoints Produção \(Exemplos JSON\)](#)

Especificação Funcional

PROC002 SANDBOX (HOMOLOGAÇÃO):

O ambiente de Sandbox foi criado especificamente para desenvolvedores que desejam testar as APIs do Sicoob.

1) Acesse suas credenciais de teste:

Antes de começar a utilizar o ambiente de sandbox, é necessário obter suas credenciais de teste. Estas:

Client ID

9b5e603e428cc477a2841e2683c92d21

Access token (Bearer)

OBS: Vale ressaltar que estas credenciais não são válidas para produção!

2) Autenticação:

É necessário fornecer no Header *Authorization* das requisições o Access token fornecido

Endereços de Sandbox (Endpoints):

API Cobrança Bancária:

<https://sandbox.sicoob.com.br/sicoob/sandbox/cobranca-bancaria/v3>

API Cobrança Bancária Pagamentos:

<https://sandbox.sicoob.com.br/sicoob/sandbox/cobranca-bancaria-pagamentos/v3>

API Conta Corrente:

<https://sandbox.sicoob.com.br/sicoob/sandbox/conta-corrente/v4>

API Convênios Pagamentos:

<https://sandbox.sicoob.com.br/sicoob/sandbox/convenios-pagamentos/v2>

API Investimentos - RDC:

<https://sandbox.sicoob.com.br/sicoob/sandbox/investimentos/v2>

API Open Finance - Iniciação de Pagamento:

<https://sandbox.sicoob.com.br/sicoob/sandbox/payments/v2/itp>

API Pix Pagamentos:

<https://sandbox.sicoob.com.br/sicoob/sandbox/pix-pagamentos/v2>

API Pix Recebimentos:

<https://sandbox.sicoob.com.br/sicoob/sandbox/pix/api/v2>

API Poupança:

<https://sandbox.sicoob.com.br/sicoob/sandbox/poupanca/v3>

API SPB Transferências:

<https://sandbox.sicoob.com.br/sicoob/sandbox/spb/v2>

Exemplos de requisição

Consultar Cobrança Imediata PIX

```
curl --location --request GET
'https://sandbox.sicoob.com.br/sicoob/sandbox/pix/api/v2/cob/: TXID' \
--header 'Authorization: Bearer {{Access Token}}' \
--header 'client_id: {{client_id}}' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json'
```

Obs: O TXID é um path param que deve ser preenchido com o identificador único do QR Code. Ele deve conter de 27 a 36 caracteres.

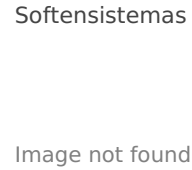
Consultar Boleto

```
curl --location -g --request GET 'https://sandbox.sicoob.com.br/sicoob/sandbox/cobranca-bancaria/v3/boletos?numeroContrato={{numContrato}}&modalidade=1&nossoNumero=integer' \
--header 'Authorization: Bearer {{Access Token}}' \
--header 'client_id: {{client_id}}' \
--header 'Accept: application/json' \
--header 'Content-Type: application/json'
```

Obs: Os Headers seguirão um padrão para todas as APIs.

Endpoints Produção

(Exemplos JSON)

| | | |
|--|--|------------------------------------|
|  | Projeto/Sistema: Emissão Sicoob - Via API | Versão do Template: 1.1 |
| | Processo: Endpoints Produção (Exemplos JSON) | Versão do Documento: 1.0 |
| | Responsável(eis): Gustavo Fernandes | Data: 08/01/2025 |

1. Introdução

Este processo visa exemplificar a utilização de endpoints, envio e retornos para registro de boletos no banco Sicoob - via API.

Processos Relacionados

[Primeiros Passos - Liberação - criação aplicativo](#)

[Sandbox \(homologação\)](#)

Especificação Funcional

PROC003 API DE COBRANÇA BANCÁRIA V3:

Esta API disponibiliza serviços para recebimento de valores referentes às vendas de produtos e serviços da sua empresa, por meio de boletos de cobrança, pagos em toda a rede bancária.

Funcionalidades

- Gerenciamento de Boletos
- Alteração de informações de pagadores de boletos
- Negativação de pagadores
- Protesto de boletos
- Movimentação

Especificações de uso da API

<https://developers.sicoob.com.br/portal/documentacao?slugItem=apis&slugSubItem=cobranca-bancaria-v3>

Questões Técnicas:

1. Cobrança Bancária Pagamentos

BASE URL: <https://api.sicoob.com.br/pagamentos/v3>

(Esta API disponibiliza funcionalidades para realização de pagamentos de boletos de Cobrança Bancária.)

1.1 Pagamento (Pagamento de Boletos de Cobrança)

1.1.1 Serviço para consultar boleto:

| Método HTTP | URL |
|-------------|-------------------------|
| GET | /boletos/{codigoBarras} |

Parâmetros:

| Name | Description |
|--|--|
| client_id * required string (header) | <input type="text" value="client_id"/> |
| codigoBarras * required string (path) | Número de código de barras com 44 posições ou linha digitável com 47 posições do <input type="text" value="codigoBarras"/> |
| numeroConta * required integer(\$int64) (query) | Número da Conta Habilitada para Pagamentos via API. <input type="text" value="numeroConta"/> |
| dataPagamento string(\$date) (query) | Data do pagamento do boleto. Se não for informada o pagamento será para data atual Formato: yyyy-MM-dd Exemplo: 2022-07-30 <input type="text" value="dataPagamento"/> |

1.1.2 Resposta esperada:

| Code | Description |
|------|-------------|
| 200 | OK |

1.1.3 Exemplo retorno:

```
{
  "resultado": {
    "numeroInstituicaoEmissora": 756,
    "nomeInstituicaoEmissora": "Banco Cooperativo do Brasil",
    "tipoPessoaBeneficiario": "F",
    "numeroCpfCnpjBeneficiario": "12345678900",
    "nomeRazaoSocialBeneficiario": "José da Silva",
    "nomeFantasiaBeneficiario": "Estilo Tech",
    "tipoPessoaBeneficiarioFinal": "2",
    "numeroCpfCnpjBeneficiarioFinal": "12345678900",
    "nomeRazaoSocialBeneficiarioFinal": "João das Flores",
    "nomeFantasiaBeneficiarioFinal": "Quitanda do João",
    "tipoPessoaPagador": "F",
    "numeroCpfCnpjPagador": "12345678900",
    "nomeRazaoSocialPagador": "Maria José",
    "nomeFantasiaPagador": "Rosa Maria da Silva",
    "codigoBarras": "string",
    "numeroLinhaDigitavel": "string",
    "dataVencimentoBoleto": "2021-04-20",
    "dataLimitePagamentoBoleto": "2021-04-25",
    "valorBoleto": 152.3,
    "valorAbatimentoDesconto": 0,
    "valorMultaMora": 0,
    "valorPagamento": 152.3,
    "dataPagamento": "2021-04-24",
    "permiteAlterarValor": true,
    "consultaEmContingencia": false,
    "codigoEspecieDocumento": 25,
    "codigoSituacaoBoletoPagamento": "12",
    "nossoNumero": "123789",
    "numeroDocumento": "123456",
    "identificadorConsulta": "hash",
    "descricaoInstrucaoDesconto1": "Conceder desconto de 10% até 05/04/2021",
    "descricaoInstrucaoDesconto2": "Conceder desconto de 5% até 10/04/2021",
    "descricaoInstrucaoDesconto3": "Conceder desconto de 2% até 05/04/2021",
```

```
"descricaoInstrucaoValorMinMax": "O Valor Mínimo é R$ 0,01. O Valor Máximo é R$ 99.999.999,99",
  "bloquearPagamento": true,
  "mensagemBloqueioPagamento": "Pagamento bloqueado"
}
```

1.1.4 Respostas possíveis:

| Code | Description |
|------|--|
| 204 | Requisição processada com sucesso e não há conteúdo a ser retornado. |
| 400 | Possíveis erros de negócio. |
| 406 | Possíveis erros de inconsistência nos dados passados. |
| 500 | Erro interno. |

1.1.5 Exemplo retorno: (Mesmo padrão de retorno para o 400, 406 e 500)

```
{
  "mensagens": [
    {
      "mensagem": "string",
      "codigo": "string"
    }
  ]
}
```

1.1.6 Serviço para efetuar o pagamento ou agendamento de boleto.

| Método HTTP | URL |
|-------------|------------------------------------|
| POST | /boletos/pagamentos/{codigoBarras} |

Parâmetros:

| Name | Description |
|---|---|
| client_id * required string (header) | <div>client_id</div> |
| x-idempotency-key * required string (header) | <p>Este parâmetro é usado para garantir a idempotência das requisições, ajudando Para cada requisição que deseja ser tratada de forma idempotente, envie um va Esse valor deve ser uma string única que segue um padrão específico: come seguido do número da conta corrente (com até 14 dígitos), e termina com um U Ex:4342-8901234-550e8400-e29b-41d4-a716-446655440000.</p> <p>Especificação do UUID</p> <div>x-idempotency-key</div> |
| codigoBarras * required string (path) | <p>Número de código de barras do boleto com 44 posições</p> <div>codigoBarras</div> |
| boletoPagamento * required object (body) | <p>Dados do boleto a ser pago ou agendado</p> <div>Example Value Model</div> |

Exemplo valor | Modelo: **(boletoPagamento)**

```
{
  "identificadorConsulta": "hash",
  "valorBoleto": 152.3,
  "valorDescontoAbatimento": 0,
  "valorMultaMora": 0,
  "descricaoObservacao": "Boleto da Escola",
  "aceitaValorDivergente": true,
  "numeroCpfCnpjPortador": "12345678900",
  "nomePortador": "Rosa Maria da Silva",
  "amount": 152.3,
  "date": "2021-04-24",
  "debtorAccount": {
    "issuer": 1234,
    "number": 1234569,
    "accountType": 0,
    "personType": 0
  }
}
```

1.1.7 Resposta esperada:

| Code | Description |
|------|-------------|
| 200 | OK |

1.1.8 Exemplo retorno:

```
{
  "resultado": {
    "numeroAgencia": "0001-9",
    "nomeAgencia": "Agência Sede",
    "numeroConta": 1234569,
    "nomeProprietarioContaCorrente": "José da Silva",
    "numeroLinhaDigitavel": "string",
    "numeroInstituicaoEmissora": 756,
    "nomeInstituicaoEmissora": "Banco Cooperativo do Brasil",
    "numeroCpfCnpjBeneficiario": "12345678900",
    "nomeRazaoSocialBeneficiario": "José da Silva",
    "nomeFantasiaBeneficiario": "Estilo Tech",
    "numeroCpfCnpjBeneficiarioFinal": "12345678900",
    "nomeRazaoSocialBeneficiarioFinal": "João das Flores",
    "nomeFantasiaBeneficiarioFinal": "Quinatda dojão",
    "numeroCpfCnpjPagador": "12345678900",
    "nomeRazaoSocialPagador": "Maria José",
    "nomeFantasiaPagador": "Rosa Maria da Silva",
    "dataVencimento": "2018-09-20",
    "valorBoleto": 100.36,
    "valorAbatimentoDesconto": 0,
    "valorMultaMora": 60.36,
    "valorPagamento": 255.63,
    "dataPagamento": "2019-10-20",
    "situacaoPagamento": "Efetivado",
    "descricaoDetalheSituacao": "Saldo no momento da rejeição em 10/03/2021 às 22:01:10: R$ 343,05.",
    "dataHoraCadastro": "2019-10-20T12:30:22.000Z",
    "aceitaValorDivergente": true,
    "nossoNumero": "756",
    "numeroDocumento": "123456",
    "descricaoObservacao": "Boleto da Escola",
    "descricaoOuvidoria": "OUVIDORIA AGIBANK: 08007250996",
```

```
"descricaoTituloComprovante": "PAGAMENTO DE BOLETO",  
"idPagamento": 1983450,  
"numeroAutenticacaoPagamento": "89C3E9FD-1A37-40BE-A85B-69AF118D336A"  
}  
}
```

1.1.1.9 Respostas possíveis:

| Code | Description |
|------|--|
| 204 | Requisição processada com sucesso e não há conteúdo a ser retornado. |
| 400 | Possíveis erros de negócio. |
| 406 | Possíveis erros de inconsistência nos dados passados. |
| 500 | Erro interno. |

1.1.1.10 Exemplo retorno: (Mesmo padrão de retorno para o 400, 406 e 500)

```
{  
  "mensagens": [  
    {  
      "mensagem": "string",  
      "codigo": "string"  
    }  
  ]  
}
```

1.1.1.11 Serviço para consultar um comprovante de um pagamento efetuado.

| Método HTTP | URL |
|-------------|--|
| GET | /boletos/pagamentos/{idPagamento}/comprovantes |

Parâmetros:

| Name | Description |
|--|--|
| client_id * required string (header) | <input type="text" value="client_id"/> |
| idPagamento * required integer(\$int64) (path) | <input type="text" value="idPagamento"/> |
| numeroConta * required integer(\$int64) (query) | <input type="text" value="numeroConta"/> |

1.1.12 Resposta esperada:

| Code | Description |
|------|-------------|
| 200 | OK |

1.1.13 Exemplo retorno:

```
{
  "resultado": {
    "numeroAgencia": "0001-9",
    "nomeAgencia": "Agência Sede",
    "numeroConta": 1234569,
    "nomeProprietarioContaCorrente": "José da Silva",
    "numeroLinhaDigitavel": "string",
    "numeroInstituicaoEmissora": 756,
    "nomeInstituicaoEmissora": "Banco Cooperativo do Brasil",
    "numeroCpfCnpjBeneficiario": "12345678900",
    "nomeRazaoSocialBeneficiario": "José da Silva",
    "nomeFantasiaBeneficiario": "Estilo Tech",
    "numeroCpfCnpjBeneficiarioFinal": "12345678900",
    "nomeRazaoSocialBeneficiarioFinal": "João das Flores",
    "nomeFantasiaBeneficiarioFinal": "Quinatda dojão",
    "numeroCpfCnpjPagador": "12345678900",
    "nomeRazaoSocialPagador": "Maria José",
    "nomeFantasiaPagador": "Rosa Maria da Silva",
    "dataVencimento": "2018-09-20",
    "valorBoleto": 100.36,
    "valorAbatimentoDesconto": 0,
```



```

    "valorMultaMora": 60.36,
    "valorPagamento": 255.63,
    "dataPagamento": "2019-10-20",
    "situacaoPagamento": "Efetivado",
    "descricaoDetalheSituacao": "Saldo no momento da rejeição em 10/03/2021 às 22:01:10: R$
343,05.",
    "dataHoraCadastro": "2019-10-20T12:30:22.000Z",
    "aceitaValorDivergente": true,
    "nossoNumero": "756",
    "numeroDocumento": "123456",
    "descricaoObservacao": "Boleto da Escola",
    "descricaoOuvidoria": "OUVIDORIA AGIBANK: 08007250996",
    "descricaoTituloComprovante": "PAGAMENTO DE BOLETO",
    "idPagamento": 1983450,
    "numeroAutenticacaoPagamento": "89C3E9FD-1A37-40BE-A85B-69AF118D336A"
  }
}

```

1.1.14 Respostas possíveis:

| Code | Description |
|------|--|
| 204 | Requisição processada com sucesso e não há conteúdo a ser retornado. |
| 400 | Possíveis erros de negócio. |
| 406 | Possíveis erros de inconsistência nos dados passados. |
| 500 | Erro interno. |

1.1.15 Exemplo retorno: (Mesmo padrão de retorno para o 400, 406 e 500)

```

{
  "mensagens": [
    {
      "mensagem": "string",
      "codigo": "string"
    }
  ]
}

```

1.1.16 Serviço para cancelar um agendamento de pagamento.

| Método HTTP | URL |
|-------------|--|
| DELETE | /boletos/pagamentos/agendamentos/{idPagamento} |

Parâmetros:

| Name | Description |
|---|--|
| client_id * required string (header) | <input type="text" value="client_id"/> |
| idPagamento * required integer(\$int64) (path) | Número de identificação do agendamento no SISBR. <input type="text" value="idPagamento"/> |
| cancelamento * required object (body) | Dados da Conta Habilitada para Pagamentos via API. Example Value Model |

Exemplo valor | Modelo: (**cancelamento**)

```
{
  "numeroConta": 1234569
}
```

1.1.17 Resposta esperada:

| Code | Description |
|------|--|
| 204 | Requisição processada com sucesso e não há conteúdo a ser retornado. |

1.1.18 Respostas possíveis:

| Code | Description |
|------|---|
| 400 | Possíveis erros de negócio. |
| 406 | Possíveis erros de inconsistência nos dados passados. |
| 500 | Erro interno |

1.1.19 Exemplo retorno: (Mesmo padrão de retorno para o 400, 406 e 500)

```
{
  "mensagens": [
```

```
{
  "mensagem": "string",
  "codigo": "string"
}
```

1.1.20 Serviço para consultar um comprovante de um pagamento efetuado através do idempotency

| Método HTTP | URL |
|-------------|--|
| GET | /boletos/pagamentos/{idempotency}/idempotency/compro vantes |

Parâmetros:

client_id * required
string
(header)

idempotency * required
string
(path)

Este parâmetro é usado para garantir a idempotência das requisições, ajudando a ev

1.1.21 Resposta esperada:

| Code | Description |
|------|-------------|
| 200 | OK |

1.1.22 Exemplo retorno:

```
{
  "resultado": {
    "numeroAgencia": "0001-9",
    "nomeAgencia": "Agência Sede",
    "numeroConta": 1234569,
    "nomeProprietarioContaCorrente": "José da Silva",
    "numeroLinhaDigitavel": "string",
    "numeroInstituicaoEmissora": 756,
    "nomeInstituicaoEmissora": "Banco Cooperativo do Brasil",
```

```

"numeroCpfCnpjBeneficiario": "12345678900",
"nomeRazaoSocialBeneficiario": "José da Silva",
"nomeFantasiaBeneficiario": "Estilo Tech",
"numeroCpfCnpjBeneficiarioFinal": "12345678900",
"nomeRazaoSocialBeneficiarioFinal": "João das Flores",
"nomeFantasiaBeneficiarioFinal": "Quinatda dojão",
"numeroCpfCnpjPagador": "12345678900",
"nomeRazaoSocialPagador": "Maria José",
"nomeFantasiaPagador": "Rosa Maria da Silva",
"dataVencimento": "2018-09-20",
"valorBoleto": 100.36,
"valorAbatimentoDesconto": 0,
"valorMultaMora": 60.36,
"valorPagamento": 255.63,
"dataPagamento": "2019-10-20",
"situacaoPagamento": "Efetivado",
"descricaoDetalheSituacao": "Saldo no momento da rejeição em 10/03/2021 às 22:01:10: R$
343,05.",
"dataHoraCadastro": "2019-10-20T12:30:22.000Z",
"aceitaValorDivergente": true,
"nossoNumero": "756",
"numeroDocumento": "123456",
"descricaoObservacao": "Boleto da Escola",
"descricaoOuvidoria": "OUVIDORIA AGIBANK: 08007250996",
"descricaoTituloComprovante": "PAGAMENTO DE BOLETO",
"idPagamento": 1983450,
"numeroAutenticacaoPagamento": "89C3E9FD-1A37-40BE-A85B-69AF118D336A"
}
}

```

1.1.23 Respostas possíveis:

| Code | Description |
|------|--|
| 204 | Requisição processada com sucesso e não há conteúdo a ser retornado. |
| 400 | Possíveis erros de negócio. |
| 406 | Possíveis erros de inconsistência nos dados passados. |
| 500 | Erro interno. |

1.1.19 Exemplo retorno: (Mesmo padrão de retorno para o 400, 406 e 500)

```
{
  "mensagens": [
    {
      "mensagem": "string",
      "codigo": "string"
    }
  ]
}
```

AVISO IMPORTANTE

Desativação da API Cobrança Bancária V2

Informamos que a API Cobrança Bancária **V2** será descontinuada em **30 de abril de 2025**. Essa medida está sendo tomada em função do lançamento da **API Cobrança Bancária V3**, que já está disponível e oferece melhorias significativas.

Orientamos que os cooperados que ainda utilizam a versão 2 realizem a migração para a versão 3 para garantir a continuidade dos serviços e o acesso às novas funcionalidades. Abaixo está o cronograma da desativação:

- **03/01/2025 Desativação da criação de credenciais para novas integrações:**
Para incentivar a adoção da nova versão, não será mais possível criar credenciais para a API Cobrança Bancária V2. A criação de credenciais estará disponível apenas para a V3.
- **30/04/2025 Interrupção do funcionamento da API Cobrança Bancária V2:**
A partir dessa data, a V2 da API será desativada. Todos os integradores e empresas parceiras que ainda utilizam a versão V2 devem migrar para a V3 antes desta data para evitar impactos nos serviços.

Acesso para conferir o catálogo completo das API's:

[Catálogo de API's](#)

