

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8



EComm - API / Cobrança Online Sicredi

Histórico de Alterações

Versão do Serviço	Versão do Documento	Modificação	Data
1.0	1.1	Inclusão da geração do nossoNumero	22/01/2018
1.0	1.2	Inclusão de mensagens de erro E0032, E0033, E0034 e E0035.	02/02/2018
1.0	1.3	Atualização na validação da Chave Master e Chave de Transação	05/04/2018
1.0	1.4	Inclusão de requisições em PHP e Java	04/09/2018
1.0	1.5	Inclusão da Apresentação e Objetivo	27/12/2018
1.0	1.6	Consulta de Boletos por Período de Datas	28/12/2018
1.0	1.7	Comando de Instrução – Cancelamento Negativação Automática	16/06/2019
1.0	1.8	Inclusão de informação referente aos Postos alfanuméricos	11/09/2019

Elaborado por: Lucas de Castro	Data de Criação: 17/01/2018	Data de Atualização: 11/09/2019
--------------------------------	-----------------------------	---------------------------------

Sumário

APRESENTAÇÃO	3
OBJETIVO	3
ORIENTAÇÕES INICIAIS	3
REGISTRO ONLINE DE BOLETOS	5
OPERAÇÕES DO SERVIÇO	6
Recurso Autenticação	6
Exemplo em PHP:	9
Exemplo em Java:	10
Recurso Emissão	11
Exemplo em PHP:	24
Exemplo em Java:	25
Geração do Nosso Número	26
Recurso Consulta	28
Exemplo em PHP:	34
Exemplo em Java:	35
Recurso Impressão	36
Exemplo em PHP:	39
Exemplo em Java:	40
Recurso Health	41
Exemplo em PHP:	42
Exemplo em Java:	43
Recurso Comando de Instrução	44
Exemplo em PHP:	51
Exemplo em Java:	52
DÚVIDAS FREQUENTES	53

Apresentação

O Cobrança Online Sicredi realiza o registro de boletos em tempo real através de integração com seu sistema Próprio.

Objetivo

O manual do Sicredi Cobrança Online tem como objetivo apresentar as informações necessárias para o registro, geração e consulta de boletos de cobrança de forma online emitidas pelo beneficiário mediante correta integração com seus sistemas via webservices.

Este manual destina-se às pessoas responsáveis pelo desenvolvimento e adaptação dos sistemas das empresas beneficiárias que desejam trabalhar com produto Cobrança Online.

Orientações Iniciais

Para iniciar o processo de integração sistêmica para o registro online de boletos, o associado deve ser previamente cadastrado na Cooperativa de Crédito/Agência como beneficiário de Cobrança e optante pela Cobrança Online.

Ao contratar o produto é gerado o código do beneficiário (código do convênio) e o seu uso será obrigatório na integração troca de informações entre a empresa beneficiária e o Sicredi.

Premissas para promover a integração sistêmica de registro on-line de Boletos:

- Possuir Contrato de Cobrança Sicredi.
- Estar habilitado para o Cobrança Online Sicredi com termo de adesão assinado.
- Integrado o conjunto de registros/campos que devem compor a troca de informações e está baseado nas informações necessárias para a implementação do produto cobrança. do seu sistema de geração/emissão de Boleto de Cobrança.

A Cobrança Online contém as seguintes operações (Serviços EComm-API)

- **Autenticação:** Operação responsável pela autenticação da Chave Master e criação da Chave de Transação dos beneficiários que aderirem o produto Cobrança Online do Sicredi.

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

- **Emissão:** Operação responsável pelo cadastro do pagador, do boleto, do informativo e da mensagem relacionadas ao boleto a ser criado.
 - **Impressão:** Operação responsável pela impressão e reimpressão dos dados de boletos.
- Observação:** quando os boletos são gerados pelo associado é necessário realizar a validação dos boletos pela equipe de homologação antes de liberar em produção.
- **Consulta:** Operação responsável pela consulta da situação dos boletos emitidos para a carteira de cobrança beneficiário.



Fluxo de Registro de Títulos via EComm-API

Todas as movimentações dos boletos criados pelo produto Cobrança Online serão enviadas aos beneficiários pelos processos de Geração dos arquivos Retornos Padrão CNAB240 e CNAB400, dessa forma, para um melhor gerenciamento indicamos adequar o sistema da sua empresa para leitura do Arquivo Retorno e integrá-lo ao sistema de conciliação bancária de sua empresa, através do manual do site do Sicredi .

Registro Online de Boletos

A primeira etapa é o processo de geração da chave Master (código de acesso), que deve ser gerada através do Sicredi Internet Banking pelo Beneficiário da Cobrança - Opção de menu no IBPJ: Cobrança >> Código de Acesso >> Gerar - Esse menu é apresentado exclusivamente para os beneficiários que possuem o campo 'Cobrança Online'. A chave gerada no Sicredi Internet após informado o dispositivo de segurança, deve ser armazenada e esse código gerado é utilizado no recurso de autenticação, e com operações.



Menu com Habilitado para Geração de Chave Master

A chave de transação deverá receber uma chave master válida para ser criada. Esta chave de transação deverá ser gerada com valores aleatórios, aplicado sha256. A chave transação possui validade de 24hrs, a contar da sua data de criação e só será permitido criar nova chave transacional após a chave já gerada expirar. Se for necessária a geração de uma nova chave transacional antes do prazo de 24h deverá ser criada uma nova chave MASTER.

O processo de autenticação do Beneficiário é a primeira etapa do processo.

No passo 2 do fluxo, uma vez obtida a confirmação do registro, o Beneficiário poderá direcionar o Pagador para qualquer Banco com a garantia de que o Boleto de Cobrança foi registrado corretamente e, na liquidação poderá ser validado on-line com a base centralizada de Boletos pelo Banco recebedor.

Operações do Serviço

Recurso Autenticação

Descrição da Operação Autenticação

A operação **POST** “autenticacao” é responsável por criar uma chave criptografada, denominada “chaveTransacao”, baseada na Chave Master. O formato de entrada e saída é JSON, considerando que a saída sempre é composta da entidade a ser retornada e um código HTTP dentro de um *Response*.

Observação: deve ser respeitada a orientação da segurança da informação referente a expiração da Chave de Transação, observando o parâmetro “dataExpiracao” de retorno do serviço (expira a cada 24 horas). Caso o beneficiário possua mais de uma conta, cada conta terá que ter a sua própria Chave Master, podendo assim, gerar a Chave de Transação para cada conta específica. Não há a possibilidade de renovar o token de transação antes da expiração. Somente com a geração de outra Chave Master será possível gerar uma nova Chave de Transação antes da expiração.

ENTRADA:

Ord	Nome	Tipo	Tipo de Parâmetro	Tamanho	Descrição	Obrigatório
1	token	String	HeaderParam	64	Chave Master gerada pelo Internet Bank	Obrigatório
2	Content-Type	application/json	HeaderParam		Indica o tipo de arquivo do recurso	Obrigatório

SAÍDA:

Ord	Nome	Tipo	Tipo de Parâmetro	Formatação	Descrição
1	chaveTransacao	String	JSON	64bits	Chave de Transação para ser utilizada
2	dataExpiracao	String	JSON	yyyy-MM-dd'T'HH:mm:ss'Z'	Data de expiração para controle da chave de transação

RETORNO (Response)

Código	Descrição
HTTP_OK (200)	Operação realizada com sucesso

FALHAS (WebApplicationException):

Status	Código	Mensagem	Parâmetros de entrada sujeitos a crítica	Descrição
HTTP_NOT_FOUND (404)	0005	Ocorreu um erro ao realizar operação de autenticação.	Não possui	Erros desconhecidos ocasionados por falhas de ambiente/negócio. Por favor, entrar em contato com o suporte
HTTP_NOT_FOUND (404)	0001	Não foi possível gerar o Token solicitado!	token	Erros ocasionados por falhas de ambiente/negócio.
HTTP_NOT_FOUND (404)	0002	Não foi possível criptografar o token!	token	Erros ocasionados por falhas de ambiente/negócio.
HTTP_NOT_FOUND (404)	0003	Não foi localizado o Token Master informado!	token	Erros ocasionados por falhas de ambiente/negócio.
HTTP_NOT_FOUND (404)	0004	Existe um Token de Transação válido cadastrado!	token	Erros ocasionados por falhas de ambiente/negócio.
HTTP_BAD_REQUEST (400)	0010	Campo obrigatório em branco.	token	Erros ocasionados por parâmetros obrigatórios não informados ou inválidos.
HTTP_BAD_REQUEST (400)	0011	Tamanho de campo inválido.	token	Erros ocasionados por parâmetros com tamanho inválido.

REQUEST - POST: <https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/autenticacao>

RESPONSE:

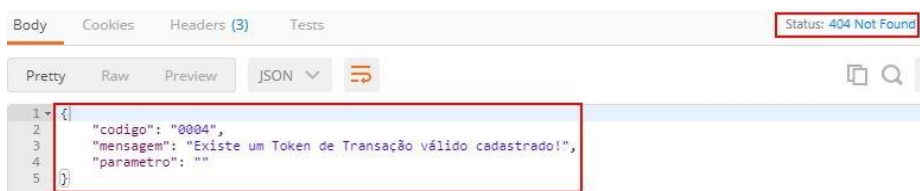
```
{
  "chaveTransacao":"14F671760619965B97FF5CEE9BE10F4792A9F3BAB8DED900F23A633A4C9FEB5",
  "dataExpiracao":"2017-04-26T20:15:28.685-03:00"
}
```

Execução (usando ferramenta Postman):

- Cenário Positivo



- Cenário de Erro <NOT_FOUND>



- Cenário de Erro <BAD_REQUEST>



Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

Exemplo em PHP:

```
function recursoAutenticacao() {
    $curl = curl_init();
    curl_setopt_array($curl, array(
        CURLOPT_URL=>
"https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/autenticacao",
        // eh indicado a ativacao de verificacao SSL p/ ambiente de producao
        CURLOPT_SSL_VERIFYPEER => 0 ,
        CURLOPT_SSL_VERIFYHOST => 0,
        // curl_setopt($ch, CURLOPT_CAINFO, 'local/crt.crt');

        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_CUSTOMREQUEST => "POST",
        CURLOPT_HTTPHEADER => array(
            "Content-Type: application/json",
            "token: SUA\_CHAVE\_MASTER\_AQUI"
        ),
    ));
    $response = curl_exec($curl);
    $info = curl_getinfo($curl);
    $err = curl_error( $curl ) ;
    if ($err) {
        echo "cURL Error: " . \$err;
    } else {
        echo "Body da Resposta: " . json\_encode\(\$response\);
    }
    curl_close( $curl ) ;
    return "Recurso Autenticacao -> HTTP Status Code: " . \$info\["http\_code"\];
}
```

Elaborado por: Lucas de Castro	Data de Criação: 17/01/2018	Data de Atualização: 11/09/2019
--------------------------------	-----------------------------	---------------------------------

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

Exemplo em Java:

```

public static void autenticacao() {

    try {

        URL url = new
URL("https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/autenticacao");

        HttpURLConnection con = (HttpURLConnection) url.openConnection();

        con.setRequestMethod("POST");

        con.setDoOutput(true);

        con.setRequestProperty("token", "SUA_CHAVE_MASTER_AQUI");

        String jsonString = "";

        InputStreamReader in = new InputStreamReader( con.getInputStream());

        BufferedReader br = new BufferedReader( in );

        String text = "";

        while ((text = br.readLine()) != null) {

            jsonString += text;

        }

        System.out.println("Resposta da Autenticacao: " + jsonString);

        in.close();

        br.close();

    } catch (MalformedURLException ex) {

        Logger.getLogger(Teste.class.getName()).log(Level.SEVERE, null, ex);

    } catch (IOException ex) {

        Logger.getLogger(Teste.class.getName()).log(Level.SEVERE, null, ex);

    }

}

```

Recurso Emissão

Descrição da Operação Emissão

A operação POST “emissao” é responsável pela geração do boleto de Cobrança. O formato de entrada e saída é JSON, considerando que a saída sempre é composta da entidade a ser retornada e um código HTTP dentro de um *Response*.

ENTRADA:

	Nome	Tipo	Tipo de Parâmetro	Tamanho	Formatação	Descrição	Obrigatório
1	token	String	Header Param	64	Sem formatação	Chave Transacional gerada pelo recurso /autenticacao	Obrigatório
2	Content-Type	application /json	Header Param		Sem formatação	Estabelece o tipo de conteúdo da requisição	Obrigatório
3	agencia	String	Body	4	Sem formatação	Código da Cooperativa do beneficiário	Obrigatório
4	posto	String	Body	2	Sem formatação	Código do Posto do beneficiário. Quando alfanumérico deverá ser informado “01”.	Obrigatório
5	cedente	String	Body	5	Sem formatação	Código do Convênio de Cobrança do beneficiário	Obrigatório
6	nossoNumero	String	Body	9	Sem formatação	Nosso Número	Opcional
7	codigoPagador	String	Body	5	Sem formatação	Código Pagador	Opcional
8	tipoPessoa	String	Body	1	Domínios: 1 ou 2	Tipo de pessoa do pagador podendo ser: 1 - PESSOA FÍSICA 2 - PESSOA JURÍDICA	Obrigatório
9	cpfCnpj	String	Body	14	Sem formatação	CPF ou CNPJ do pagador	Obrigatório
10	nome	String	Body	40	Sem formatação	Nome do pagador	Obrigatório
11	endereco	String	Body	40	Sem formatação	Endereço do pagador	Opcional. Será obrigatório se o código do pagador não for informado.
12	cidade	String	Body	25	Sem formatação	Cidade do pagador	Opcional. Será obrigatório se o código do pagador não for informado.

13	uf	String	Body	2	Sem formatação	Estado do pagador	Opcional. Será obrigatório se o código do pagador não for informado.
14	cep	String	Body	8	Sem formatação	CEP do pagador	Obrigatório
15	telefone	String	Body	11	Sem formatação	Telefone mais DDD do pagador	Opcional. Será obrigatório se o código do pagador não for informado.
16	email	String	Body	40	xxx@xx.xx	Endereço de email do pagador	Opcional
17	especieDocumento	String	Body	1	Domínios: A B C D E G H I J K O	Espécie de documento do título podendo ser: A - DUPLICATA MERCANTIL (DM) B - DUPLICATA RURAL (DR) C - NOTA PROMISSORIA (NP) D - NOTA PROMISSORIA RURAL (NR) E - NOTA DE SEGURO (NS) G - RECIBO (RC) H - LETRA DE CAMBIO (LC) I - NOTA DE DEBITO (ND) J - DUPLICATA DE SERVICO (DS) K - OUTROS (OS) O - BOLETO OFERTA (OFE)	Obrigatório
18	codigoSacadorAvalista	String	Body	3	Sem formatação	Código do sacador avalista.	Opcional
19	seuNumero	String	Body	10	Sem formatação	Número de controle interno do beneficiário que referencia o pagador.	Obrigatório
20	dataVencimento	String	Body	10	DD/MM/YYYY	Data de vencimento do boleto	Obrigatório
21	valor	BigDecimal	Body	14,2	Sem formatação	Valor do boleto	Obrigatório
22	tipoDesconto	String	Body	1	Domínios: A ou B	Tipo de desconto podendo ser: A - VALOR B - PERCENTUAL	Obrigatório

23	valorDesconto1	BigDecimal	Body	14,2	Sem formatação	Valor de desconto 1	Opcional. Será obrigatório se o dataDesconto1 for informado.
24	dataDesconto1	String	Body	10	DD/MM/YYYY	Data limite para concessão de desconto1	Opcional. Será obrigatório se o valorDesconto1 for informado. Não pode ser igual a outras dataDescontos e nem maior que vencimento.
25	valorDesconto2	BigDecimal	Body	14,2	Sem formatação	Valor de desconto 2	Opcional. Será obrigatório se o dataDesconto2 for informado.
26	dataDesconto2	String	Body	10	DD/MM/YYYY	Data limite para a concessão de desconto 2	Opcional. Será obrigatório se o valorDesconto2 for informado. Não pode ser igual a outras dataDescontos e nem maior que vencimento.
27	valorDesconto3	BigDecimal	Body	14,2	Sem formatação	Valor de desconto 3	Opcional. Será obrigatório se o dataDesconto3 for informado.
28	dataDesconto3	String	Body	10	DD/MM/YYYY	Data limite para a concessão de desconto 3	Opcional. Será obrigatório se o valorDesconto3 for informado. Não pode ser igual a outras dataDescontos e nem maior que vencimento.
29	tipoJuros	String	Body	1	Domínios: A ou B	Tipo de Juros, podendo ser: A - VALOR B - PERCENTUAL	Obrigatório
30	juros	BigDecimal	Body	14,2	Sem formatação	Valor de juros a cobrar por dia	Opcional
31	multas	BigDecimal	Body	5,2	Sem formatação	Valor de multas a cobrar	Opcional

32	desconto Antecipado	BigDecimal	Body	8,2	Sem formatação	Valor de desconto antecipado a cobrar	Opcional. Deve ser menor que o valor.
33	informativo	String	Body	80	Sem formatação	Texto do Informativo	Opcional
34	mensagem	String	Body	300	Sem formatação	Texto da linha da mensagem	Opcional
35	codigoMensagem	String	Body	4	Sem formatação	Código da mensagem	Opcional
36	numDiasNegativacaoAuto	Integer	Body	3	Sem formatação	Número de dias para Negativação Automática. 0 (Zero) para não negativar automaticamente. Quando preenchido com 3 ou 4 dias o sistema comandará negativação em dias úteis após o vencimento. Quando preenchido acima de 4 dias, o sistema comandará negativação em dias corridos após o vencimento.	Opcional

SAÍDA:

Ord	Nome	Tipo	Tamanho	Formatação	Descrição
1	linhaDigitável	String	47	Sem formatação	Linha digitável
2	codigoBanco	String	3	748	Código do Banco
3	nomeBeneficiario	String	40	Sem formatação	Nome do Beneficiário
4	enderecoBeneficiario	String	40	Sem formatação	Endereço do Beneficiário
5	cpfCnpjBeneficiario	String	14	Sem formatação	CPF/CNPJ do Beneficiário
6	cooperativaBeneficiario	String	4	Sem formatação	Código da Cooperativa do Beneficiário
7	postoBeneficiario	String	2	Sem formatação	Código da UA do Beneficiário. Quando alfanumérico deverá ser informado "01".
8	codigoBeneficiario	String	5	Sem formatação	Código do Beneficiário
9	dataDocumento	String	10	Sem formatação	Data do Documento
10	seuNumero	String	10	Sem formatação	Seu Número
11	especieDocumento	String	1	Sem formatação	Espécie de Documento
12	aceite	String	1	Sem formatação	Aceite
13	dataProcessamento	String	10	Sem formatação	YYYY-MM-DD
14	nossoNumero	BigDecimal	9	Sem formatação	Nosso Número
15	especie	String	4	Sem formatação	Espécie: REAL (fixo)
16	valorDocumento	BigDecimal	14,2	Sem formatação	Valor do Documento
17	dataVencimento	String	10	YYYY-MM-DD	Data de Vencimento

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

18	nomePagador	String	40	Sem formatação	Nome do Pagador
19	cpfCnpjPagador	String	14	Sem formatação	CPF/CNPJ do Pagador
20	enderecoPagador	String	40	Sem formatação	Endereço do Pagador
21	dataLimiteDesconto	String	10	YYYY-MM-DD	Data Limite Desconto
22	valorDesconto	BigDecimal	14,2	Sem formatação	Valor Desconto/Abatimento
23	jurosMulta	BigDecimal	9,2	Sem formatação	Juros/Multa
24	instrucao	String	800	Sem formatação	Instruções
25	informativo	String	80	Sem formatação	Informativo
26	codigoBarra	String	44	Sem formatação	Código de Barras

RETORNO (Response):

Status	Descrição
HTTP_CREATED (201)	Operação realizada com sucesso

FALHAS (WebApplicationException):

Status	Código	Mensagem	Parâmetros de entrada sujeitos a crítica	Descrição
HTTP_NOT_FOUND (404)	E0005	Ocorreu um erro ao realizar operação de autenticação	Não possui	Erros ocasionados por falhas de ambiente/negócio. Por favor entrar em contato com o suporte.
HTTP_NOT_FOUND (404)	E0006	Ocorreu um erro ao realizar operação de emissão	Não possui	Erros ocasionados por falhas de ambiente/negócio. Por favor entrar em contato com o suporte.
HTTP_NOT_FOUND (404)	E0007	Ocorreu um erro ao realizar operação de consulta	Não possui	Erros ocasionados por falhas de ambiente/negócio. Por favor entrar em contato com o suporte.
HTTP_NOT_FOUND (404)	E0008	Ocorreu um erro ao realizar operação de impressão	Não possui	Erros ocasionados por falhas de ambiente/negócio. Por favor entrar em contato com o suporte.

HTTP_BAD_REQUEST (400)	E0010	Campo obrigatório em branco	Qualquer parâmetro que consta como obrigatório na listagem de parâmetros de entrada pode ser criticado	Erros ocasionados por parâmetros obrigatórios não informados ou inválidos.
HTTP_BAD_REQUEST (400)	E0011	Tamanho de campo inválido.	Qualquer parâmetro que exceda o tamanho que foi informado na listagem de parâmetros de entrada pode ser criticado.	Erros ocasionados por parâmetros com tamanho inválido.
HTTP_BAD_REQUEST (400)	E0012	Campo com caracteres especiais e/ou alfanuméricos.	Qualquer parâmetro que não contenha formatação na listagem de parâmetros de entrada pode ser criticado.	Erros ocasionados por parâmetros com caracteres não permitidos.
HTTP_BAD_REQUEST (400)	E0013	Campo diferente de '1' e '2'	tipoPessoa	Erro ocasionado ao informar um tipoPessoa diferente de '1' ou '2' no parâmetro tipoPessoa.
HTTP_BAD_REQUEST (400)	E0014	Tipo de Pessoa '1', CPF informado é maior ou menor que 11 posições.	cpfCnpj	Erro ocasionado por CPF informado incorretamente para tipoPessoa '1'.
HTTP_BAD_REQUEST (400)	E0015	Tipo de Pessoa '2', CNPJ informado é maior ou menor do que 14 posições.	cpfCnpj	Erro ocasionado por CNPJ informado incorretamente para tipoPessoa '2'.
HTTP_BAD_REQUEST (400)	E0016	E-mail informado fora do padrão xxx@xxx.xx	email	Erro ocasionado por email fora do padrão.
HTTP_BAD_REQUEST (400)	E0017	Campo diferente de 'A' (VALOR) ou 'B' (PERCENTUAL)	tipoDesconto tipoJuros	Erro ocasionado por parâmetro tipoDesconto/tipoJuros diferente de 'A' ou 'B'

HTTP_BAD_REQUEST (400)	E0018	Campo diferente de 'A', 'B', 'C', 'D', 'E', 'G', 'H', 'I', 'J', 'K', 'O'.	especieDocumento	Erro ocasionado por parâmetro especieDocumento diferente de 'A', 'B', 'C', 'D', 'E', 'G', 'H', 'I', 'J', 'K', 'O'.
HTTP_BAD_REQUEST (400)	E0019	Campo obrigatório para código de pagador em branco.	endereço cidade uf telefone	Erro ocasionado por código pagador em branco , tornando obrigatório o preenchimento de endereço, cidade, uf e telefone.
HTTP_BAD_REQUEST (400)	E0020	Valor de desconto maior que valor do documento.	descontoAntecipado o valorDesconto1 valorDesconto2 valorDesconto3	Erro ocasionado por informar parâmetro descontoAntecipado/ valorDesconto1/ valorDesconto2/ valorDesconto3 com valor maior ou igual ao parâmetro valor.
HTTP_BAD_REQUEST (400)	E0021	Valor de desconto preenchido e data de desconto em branco.	valorDesconto1 dataDesconto1 valorDesconto2 dataDesconto2 valorDesconto3 dataDesconto3	Erro ocasionado por valor de desconto preenchido, porém data de desconto em branco.
HTTP_BAD_REQUEST (400)	E0022	Valor de desconto em branco e data de desconto preenchida.	valorDesconto1 dataDesconto1 valorDesconto2 dataDesconto2 valorDesconto3 dataDesconto3	Erro ocasionado por valor de desconto em branco, porém data de desconto preenchida.
HTTP_BAD_REQUEST (400)	E0023	Data de desconto maior que data de vencimento.	dataDesconto1 dataDesconto2 dataDesconto3	Erro ocasionado pr data de desconto maior que a data de vencimento.
HTTP_BAD_REQUEST (400)	E0024	Não foram encontrados resultados para os filtros de consulta informados.	filtro	Mensagem de retorno sem resultado.

HTTP_BAD_REQUEST (400)	E0025	Ocorreu um erro ao realizar operação de comando de instrução.	Não possui	Erros ocasionados por falhas de ambiente/negócio. Por favor, entrar em contato com o suporte.
HTTP_BAD_REQUEST (400)	E0026	Datas de descontos iguais	dataDesconto1 dataDesconto2 dataDesconto3	Campo de datas com o mesmo valor.
HTTP_BAD_REQUEST (400)	E0027	Formato de data inválido. Formato correto dd/MM/yyyy.	dataDesconto1 dataDesconto2 dataDesconto3	Formato inválido das datas.
HTTP_BAD_REQUEST (400)	E0028	Data de desconto menor que data atual.	dataDesconto1 dataDesconto2 dataDesconto3	Data com valor menor que data de emissão ou se não possui data emissão, valida data do dia corrente.
HTTP_BAD_REQUEST (400)	E0030	Data de vencimento menor que a data desconto.	dataVencimento	Data de vencimento menor que dataDesconto1 ou dataDesconto2 ou dataDesconto3.
HTTP_BAD_REQUEST (400)	E0031	Data de vencimento menor que data atual.	dataVencimento	Data de vencimento menor que data atual.
HTTP_BAD_REQUEST (400)	E0033	Chave Transação inválida para este beneficiário.	token	Chave de transação não pertence ao beneficiário.
HTTP_BAD_REQUEST (400)	E0034	CPF inválido	cpfCnpj	Dígito verificador do CPF está inválido
HTTP_BAD_REQUEST (400)	E0035	CNPJ inválido	cpfCnpj	Dígito verificador do CNPJ está inválido
HTTP_BAD_REQUEST (400)	E2001	ORA-20001: EToken de Transação inválido.	token	Retorno da validação do token de transação.
HTTP_NOT_FOUND (404)	-20001	CCRL0139 : Campo em Branco! (Campo: Nome)	nome	Campo em branco.
HTTP_NOT_FOUND (404)	-20001	NULL0000: Parâmetro recebido excede o tamanho válido: <nome>	nome	Nome maior que 40 caracteres.

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

HTTP_NOT_FOUND (404)	-20001	CCRL0139 : Campo em Branco! (Campo: Agência).	agencia	Campo em branco
HTTP_NOT_FOUND (404)	-20001	NULL0000 : Parâmetro recebido excede o tamanho valido: <agência>	agencia	Campo com tamanho superior a 4 caracteres
HTTP_NOT_FOUND (404)	-20001	CCRL0019 : Agência Inválida. (Agência informada: <agência>).	agencia	Agência não existente
HTTP_NOT_FOUND (404)	-20001	CCRL0139 : Campo em Branco! (Campo: Posto).	posto	Campo em branco.
HTTP_NOT_FOUND (404)	-20001	NULL0000 : Parâmetro recebido excede o tamanho valido: <posto>	posto	Campo com tamanho superior a 2 caracteres.
HTTP_NOT_FOUND (404)	-20001	NULL0000 : Parâmetro recebido excede o tamanho valido: <código do cedente>	cedente	Campo com mais de 5 caracteres
HTTP_NOT_FOUND (404)	-20001	CCRL0008: Código de Beneficiário inválido. (Código informado: <código do beneficiário>).	codigoPagador	Código do pagador inexistente.
HTTP_NOT_FOUND (404)	-20001	NULL0000 : Parâmetro recebido excede o tamanho valido: <UF>	uf	Campo com mais de 2 caracteres.
HTTP_NOT_FOUND (404)	-20001	NULL0000 : Parâmetro recebido excede o tamanho valido: <CEP>	cep	Campo com mais de 8 caracteres.
HTTP_NOT_FOUND (404)	-20001	NULL0000: Parâmetro recebido excede o tamanho valido: <Endereco>	endereco	Campo com mais de 40 caracteres

HTTP_NOT_FOUND (404)	-20001	NULL0000: Parâmetro recebido excede o tamanho valido: <CPF/CNPJ>	cpfCnpj	Campo com mais de 14 caracteres
HTTP_NOT_FOUND (404)	-20001	CCRL0140 : CPF/CGC Inválido! (CPF/CGC Informado Inválido: <CPF/CNPJ>).	cpfCnpj	CPF/CNPJ inválido.
HTTP_NOT_FOUND (404)	-20001	NULL0000: Parâmetro recebido excede o tamanho valido: <Tipo pessoa>	tipoPessoa	Campo com mais de 1 carácter.
HTTP_NOT_FOUND (404)	-20001	CCRL0113 : CEP inválido. (Não cadastrado no Sistema da Carteira de Cobrança!).	cep	CEP não cadastrado.
HTTP_BAD_REQUEST (400)	-20001	ORA-20001: ECCRL0170 : Praça do sacado não cadastrada	cep	CEP inválido. Caso queira desativar essa crítica, entre em contato com sua Unidade de Atendimento SICREDI e peça a desativação de validação de CEP.
HTTP_NOT_FOUND (404)	-20001	ECCRL0164 : Dígito Verificador do NOSSO NUMERO invalido	nossoNumero	Formato inválido do nossoNumero. Verificar o cálculo do dígito verificador.
HTTP_NOT_FOUND (404)	-20001	ORA-20001: EORA-02291: integrity constraint (JDCOBRANCA_OWNER.FKJD NPCDSTLEG_L2J_OPTPGD_UF	uf	UF inválido. Verificar o campo UF, deve conter a sigla de um estado brasileiro válido.
HTTP_NOT_FOUND (404)	-20001	ORA-20001: ECCRL0238 : Título já cadastrado.		No momento da emissão, está sendo passado o nossoNumero de um título já cadastrado.

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

REQUEST - POST:

<https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/emissao>

HEADER:

token: **SEU_TOKEN_DE_TRANSACAO**

Content-Type: application/json

BODY (JSON):

```
{
  "agencia": "0116",
  "posto": "08",
  "cedente": "68062" ,
  "tipoPessoa": "1",
  "cpfCnpj": "10531369943" ,
  "nome": "teste",
  "cep": "91760110",
  "especieDocumento": "B",
  "codigoSacadorAvalista": "000" ,
  "seuNumero": "1234567890",
  "dataVencimento": "31/12/2018" ,
  "valor": 100.99,
  "tipoDesconto": "A",
  "valorDesconto1": 10.99,
  "dataDesconto1": "29/10/2018",
  "valorDesconto2": 5 ,
  "dataDesconto2": "30/10/2018",
  "valorDesconto3": 1.99,
  "dataDesconto3": "31/10/2018",
  "tipoJuros": "B" ,
  "juros": 1,
  "descontoAntecipado": 12.99,
  "mensagem": "Mensagem gerada pelo teste de integracao",
  "codigoMensagem": "",
  "endereco": "Av Franca, 123",
  "cidade": "Porto Alegre" ,
  "uf": "RS" ,
  "telefone": "51-999999999",
  "informativo": "Informativo gerado pelo teste de integracao"
}
```

Elaborado por: Lucas de Castro	Data de Criação: 17/01/2018	Data de Atualização: 11/09/2019
--------------------------------	-----------------------------	---------------------------------

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

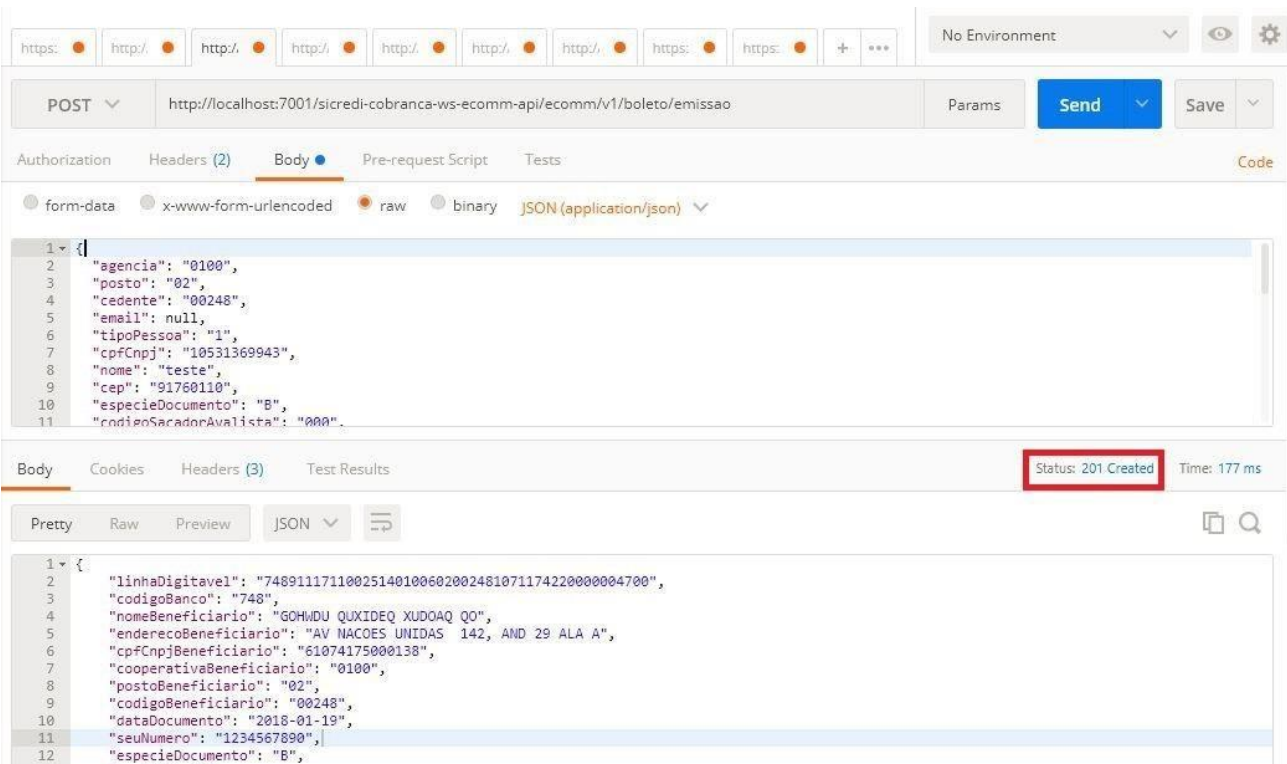
RESPONSE:

```
{
  "linhaDigitavel": "74891118100010510116608680621045677550000010099",
  "codigoBanco": "748",
  "nomeBeneficiario": "NOME DO BENEFICIÁRIO",
  "enderecoBeneficiario": "ENDEREÇO DO BENEFICIÁRIO",
  "cpfCnpjBeneficiario": "91544098000101",
  "cooperativaBeneficiario": "0116",
  "postoBeneficiario": "08",
  "codigoBeneficiario": "68062",
  "dataDocumento": "2018-09-06",
  "seuNumero": "1234567890",
  "especieDocumento": "B",
  "aceite": "N",
  "dataProcessamento": "2018-09-06",
  "nossoNumero": 181001051,
  "especie": "REAL",
  "valorDocumento": 100.99,
  "dataVencimento": "2018-12-31",
  "nomePagador": "TESTE",
  "cpfCnpjPagador": "10531369943",
  "enderecoPagador": "AV FRANCA, 123",
  "dataLimiteDesconto": "2018-10-29",
  "valorDesconto": 0, "jurosMulta": 1,

  "instrucao": "Mensagem gerada pelo teste de integracao\rAPOS VENCIMENTO COBRAR MORA DIARIA DE R$ 1.01.\rCONCEDER DESCONTO DE R$ 10.99 SE PAGO ATE 29/10/2018.\rCONCEDER DESCONTO DE R$ 12.99 POR DIA DE ANTECIPACAO.\r",
  "informativo": "Informativo gerado pelo teste de integracao\r",
  "codigoBarra": "74896775500000100991118100105101160868062104"
}
```

Execução (usando ferramenta Postman):

- Cenário Positivo:



- Cenário de erro <NOT_FOUND>



- Cenário de erro <BAD_REQUEST>:



Exemplo em PHP:

```
function recursoEmissao() {
    $curl = curl_init();
    curl_setopt_array($curl, array(
        CURLOPT_URL =>
            "https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/emissao",
            //eh indicado a ativacao de verificacao SSL p/ ambiente de producao
        CURLOPT_SSL_VERIFYPEER => 0,
        CURLOPT_SSL_VERIFYHOST => 0,
        // curl_setopt($ch, CURLOPT_CAINFO, 'local/crt.crt');
        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_CUSTOMREQUEST => "POST",
        CURLOPT_POSTFIELDS => "{\n"
            . "\"agencia\" : \"0116\", \n"
            . "\"posto\" : \"08\", \n"
            . "\"cedente\" : \"68062\", \n"
            . "\"codigoPagador\" : \"12345\", \n"
            . "\"tipoPessoa\" : \"2\", \n"
            . "\"cpfCnpj\" : \"91544098000101\", \n"
            . "\"nome\" : \"Anderson A D Nunes\", \n"
            . "\"especieDocumento\" : \"O\", \n"
            . "\"cep\" : \"91350070\", \n"
            . "\"codigoSacadorAvalista\" : \"000\", \n"
            . "\"seuNumero\" : \"12345\", \n"
            . "\"dataVencimento\" : \"01/11/2018\", \n"
            . "\"valor\" : \"0.01\", \n"
            . "\"tipoDesconto\" : \"A\", \n"
            . "\"tipoJuros\" : \"B\", \n"
            . "\"informativo\" : \"Boleto gerado pelo teste da integração\", \n"
            . "}"
    ), CURLOPT_HTTPHEADER => array(
        "Content-Type: application/json",
        "token: SUA_CHAVE_TRANSACIONAL_AQUI"
    ));
    $response = curl_exec( $curl );
    $info = curl_getinfo($curl);
    $err = curl_error($curl);

    if ($err ) { echo "cURL Error: " . $err; }

    else { echo "Body da Resposta: " . json_encode($response); }

    curl_close($curl);

    return "Recurso Emissao -> HTTP Status Code: " . $info["http_code"]; }
```


Exemplo em Java:

```

public static void emissao() {
    try {
        URL url = new
URL("https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/emissao"
        HttpURLConnection con = ( HttpURLConnection) url.openConnection();
        con.setRequestMethod("POST");
        con.setDoOutput(true);
        con.setRequestProperty("Content-Type", "application/json");
        con.setRequestProperty("token", "SUA_CHAVE_TRANSACIONAL_AQUI");
        String input = "{\n"
            + "\"agencia\" : \"0116\", \n"
            + "\"posto\" : \"08\", \n"
            + "\"cedente\" : \" 68062\", \n"
            + "\"codigoPagador\" : \"12345\", \n"
            + "\"tipoPessoa\" : \" 2\", \n"
            + "\"cpfCnpj\" : \"91544098000101\", \n"
            + "\"nome\" : \" Anderson A D Nunes\", \n"
            + "\"especieDocumento\" : \" O\", \n"
            + "\"cep\" : \"91350070\", \n"
            + "\"codigoSacadorAvalista\" : \" 000\", \n"
            + "\"seuNumero\" : \"12345\", \n"
            + "\"dataVencimento\" : \" 01/11/2018\", \n"
            + "\"valor\" : \"0.01\", \n"
            + "\"tipoDesconto\" : \"A\", \n"
            + "\"tipoJuros\" : \"B\", \n"
            + "\"informativo\" : \"Boleto gerado pelo teste da integração\", \n"
            + "}";
        OutputStream os = con.getOutputStream();
        os.write(input.getBytes()); os.flush();
        String jsonString = "";
        InputStreamReader in = new InputStreamReader ( con.getInputStream());
        BufferedReader br = new BufferedReader(in);
        String text = "";
        while ((text = br.readLine()) != null) { jsonString += text; }
        System.out.println("Resposta da Emissao: " + jsonString);
        os.close(); in.close(); br.close();
    } catch (MalformedURLException ex) {
        Logger.getLogger(Teste.class.getName()).log(Level.SEVERE, null, ex);
    } catch (IOException ex) {
        Logger.getLogger(Teste.class.getName()).log(Level.SEVERE, null, ex); }
}

```

Geração do Nosso Número

Se **não for enviado** o campo “nossoNumero” será retornado um “nossoNumero” **gerado automaticamente** pelo *webservice* EComm-API. Caso seja enviado o campo “nossoNumero” na emissão do boleto, segue a regra para a criação:

REGRA PARA GERAÇÃO DO NOSSO NUMERO:

AAbnnnnnd => formato do nossoNumero onde **AA** é o ano, **b** é o indicador de geração do nosso número, **nnnnn** é o sequencial e **d** é o dígito verificador.

Fórmula para cálculo do dígito verificador pelo módulo 11

- a) Relacionar os códigos da cooperativa de crédito/agência cedente (**aaaa**), posto cedente (**pp**), do cedente (**cccc**), ano atual (**yy**), indicador de geração do nosso número (**b**) e o número sequencial do cedente (**nnnnn**): **aaaappccccyybnnnnn**;

- b) atribuir os pesos (de 2 a 9) correspondentes para cada dígito, começando da direita para a esquerda, efetuando cada multiplicação:

a	a	a	a	p	p	c	c	c	c	c	y	y	b	n	n	n	n	n
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
4	3	2	9	8	7	6	5	4	3	2	9	8	7	6	5	4	3	2
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=
4a	3a	2a	9a	8p	7p	6c	5c	4c	3c	2c	9y	8y	7b	6n	5n	4n	3n	2n

- c) somar o resultado de cada multiplicação;
- d) dividir o resultado da soma por 11 (onze);
- e) identificar o resto da divisão;
- f) dígito verificador será o resultado da subtração: 11 - resto da divisão. Se o resultado da subtração for 10 (dez) ou 11 (onze), o dígito verificador será 0 (zero).

Exemplo:

cooperativa de crédito/agência cedente _____ : 0100

posto _____ : 02




cedente _____ : 00248

ano _____: 18

byte da geração _____: 2 (nosso número gerado pelo cedente) número

sequencial _____: 00001

Cálculo para encontrar o DV do “Nosso número”

Agência				Posto		Cedente					Ano		Byte	Sequencial					
0	1	0	0	0	2	0	0	2	4	8	1	8	2	0	0	0	0	1	 Dados
x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
4	3	2	9	8	7	6	5	4	3	2	9	8	7	6	5	4	3	2	 Pesos
=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	=	
0	3	0	0	0	14	0	0	8	12	16	9	64	14	0	0	0	0	2	 Multiplicações

Somatório do resultado de cada multiplicação= 142

Ⓟ $142 / 11 = 12,909$

Ⓟ $12 \times 11 = 132$

Ⓟ $142 - 132 = 10$

Ⓟ $11 - 10 = 1$ (Caso o resultado seja 10 ou 11, o DV será 0)

DV = 1

Exemplo do nossoNumero gerado = 182000011

A 10		Byte	Sequencial					DV
1	8	2	0	0	0	0	1	1

Recurso Consulta

Descrição da Operação Consulta

A operação **GET** “consulta” é responsável pela consulta da situação de boletos. O formato de entrada e saída é JSON, considerando que a saída sempre é composta da entidade a ser retornada e um código HTTP dentro de um Response.

ENTRADA:

Ord	Nome	Tipo	Tipo de Parâmetro	Tamanho	Formatação	Descrição	Obrigatório
1	token	String	HeaderParam	64	Sem formatação	Chave Transação da operação autenticação	Obrigatório
2	agencia	String	QueryParam	4	Sem formatação	Código da cooperativa do Beneficiário	Obrigatório
3	cedente	String	QueryParam	5	Sem formatação	Código do Beneficiário	Obrigatório
4	posto	String	QueryParam	2	Sem formatação	Código do posto do Beneficiário. Quando o Posto for alfanumérico deverá ser informado “01”.	Obrigatório
5	nossoNumero	String	QueryParam	9	Sem formatação	Nosso Número do Boleto	Opcional. Se não informado, deverá ser informado os campos tipoData, dataInicio e dataFim.
6	tipoData	String	QueryParam		<ul style="list-style-type: none"> DATA_EMISSAO DATA_VENCIMENTO DATA_LIQUIDACAO DATA_BAIXA 	Valor que indicará a data a ser consultada	Opcional. Se não informado deverá ser informado o campo nossoNumero
7	dataInicio	String	QueryParam	10	Formatação dd/mm/yyyy	Data de início da consulta por intervalo de data.	Opcional. Se não informado deverá ser informado o campo nossoNumero.

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

8	dataFim	String	QueryParam	10	Formatação dd/mm/yyyy	Data final da consulta por intervalo de data.	Opcional. Se não informado deverá ser informado o campo nossoNumero.
---	---------	--------	------------	----	-----------------------	---	--

SAÍDA:

Ord	Nome	Tipo	Tipo de Parâmetro	Formatação	Descrição
1	seuNumero	String	JSON	Sem formatação	Seu Número
2	nossoNumero	String	JSON	Sem formatação	Nosso Número
3	nomePagador	String	JSON	Sem formatação	Nome do Pagador
4	valor	String	JSON	Sem formatação	Valor do Documento
5	valorLiquidado	String	JSON	Sem formatação	Valor Liquidado
6	dataEmissao	Date	JSON	YYYY-MM-DD	Data Emissão
7	dataVencimento	Date	JSON	YYYY-MM-DD	Data Vencimento
8	dataLiquidacao	Date	JSON	YYYY-MM-DD	Data da Liquidação
9	situacao	String	JSON	<ul style="list-style-type: none"> • EM CARTEIRA • LIQUIDADO • BAIXADO POR SOLICITACAO • BAIXADO POR PROTESTO • EM CARTORIO • AGUARDANDO ENTRADA EM CARTORIO • REJEITADO 	Situação

RETORNOS (Response):

Código	Descrição
HTTP_OK (200)	Operação realizada com sucesso.

FALHAS (WebApplicationException)

Status	Código	Mensagem	Parâmetros de entrada sujeitos a crítica	Descrição
HTTP_NOT_FOUND (404)	0007	Ocorreu um erro ao realizar operação de consulta.	Não possui	Erros desconhecidos ocasionados por falhas de ambiente/negócio. Por favor, entrar em contato com o suporte.

HTTP_NOT_FOUND (404)	0024	Não foram encontrados resultados para os filtros de consulta informados.	Não possui	Erros desconhecidos ocasionados por falhas de ambiente/negócio.
HTTP_BAD_REQUEST (400)	0010	Campo obrigatório em branco	Qualquer parâmetro que consta como obrigatório na listagem de parâmetros de entrada pode ser criticado.	Erros ocasionados por parâmetros obrigatórios não informados ou inválidos.
HTTP_BAD_REQUEST (400)	0011	Tamanho de campo inválido.	Qualquer parâmetro que exceda o tamanho que foi informado na listagem de parâmetros de entrada pode ser criticado.	Erros ocasionados por parâmetros com tamanho inválido.
HTTP_BAD_REQUEST (400)	0012	Campo com caracteres especiais e/ou alfanuméricos.	Qualquer parâmetro que não contenha formatação na listagem de parâmetros de entrada pode ser criticado.	Erros ocasionados por parâmetros com caracteres não permitidos.
HTTP_BAD_REQUEST (400)	E0037	Intervalo de datas maior que 90 dias.	dataFim	Erro ocasionado por inserir um intervalo maior que 90 dias.
HTTP_BAD_REQUEST (400)	E0036	Data de inicio maior que data fim.	dataInicio	Erro ocasionado por inserir uma data inicial maior que a data final.
HTTP_BAD_REQUEST (400)	E0038	Preencha o campo nossoNumero ou os campos dataInicio, dataFim e tipoData.	nossoNumero	Erro ocasionado pela ausência dos parâmetros nossoNumero; ou tipoData, dataInicio e dataFim.
HTTP_BAD_REQUEST (400)	E0039	Preencha o campo tipoData com um dos valores: DATA_EMISSAO, DATA_VENCIMENTO, DATA_BAIXA ou DATA_LIQUIDACAO.	tipoData	Erro ocasionado por inserção incorreta no campo tipoData.

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

REQUEST-GET:

<https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/consulta?agencia=0101&posto=19&cedente=68062&nossoNumero=181000853>

RESPONSE (quando informado parâmetro de entrada nossoNumero):

```
[
  {
    "seuNumero": "1234567891",
    "nossoNumero": "181000853",
    "nomePagador": "TESTE",
    "valor": "10",
    "valorLiquidado": "0",
    "dataEmissao": "2017-07-06",
    "dataVencimento": "2017-08-27",
    "situacao": "EM CARTEIRA"
  }
]
```

REQUEST-GET:

https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/consulta?agencia=0100&cedente=00248&posto=02&dataInicio=31/12/2018&dataFim=31/12/2018&tipoData=DATA_EMISSAO

RESPONSE (quando **não** informado parâmetro de entrada nossoNumero, e informado os campos tipoData, dataInicio e dataFim pesquisando pela DATA_EMISSAO):

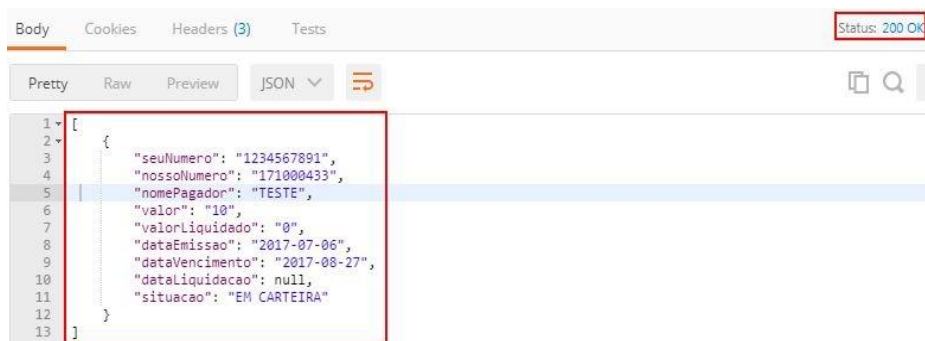
```
[
  {
    "seuNumero": "1234567891",
    "nossoNumero": "191007112",
    "nomePagador": "TESTE",
    "valor": "10",
    "valorLiquidado": "0",
    "dataEmissao": "2018-12-31",
    "dataVencimento": "2019-01-10",
    "situacao": "EM CARTEIRA"
  },
  {
    "seuNumero": "1234567891",
    "nossoNumero": "191006884",
    "nomePagador": "TESTE",
    "valor": "10",
    "valorLiquidado": "0",
    "dataEmissao": "2018-12-31",
    "dataVencimento": "2019-01-05",
    "situacao": "REJEITADO"
  },
  {
    "seuNumero": "1234567891",
    "nossoNumero": "191006892",
    "nomePagador": "TESTE",
    "valor": "10",
  }
]
```



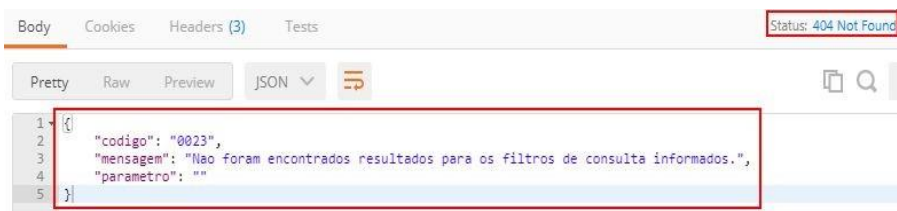
```
"valorLiquidado": "0",
"dataEmissao": "2018-12-31",
"dataVencimento": "2019-01-20",
"situacao" : "BAIXADO POR SOLICITACAO" },
{
  "seuNumero": "1234567891",
  "nossoNumero": "191006906",
  "nomePagador": "TESTE",
  "valor": "10",
  "valorLiquidado": "10",
  "dataEmissao": "2018-12-31",
  "dataVencimento": "2019-01-31",
  "situacao": "LIQUIDADO"
}
]
```

EXECUÇÃO (usando ferramenta Postman):

- Cenário positivo



- Cenário de Erro <NOT_FOUND>:



Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

Exemplo em PHP:

```
function recursoConsulta() {
    $curl = curl_init();
    $params = array('agencia' => '0116','posto' => '08','cedente' => '68062','nossoNumero' => '181000853');
    $url =
"https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/consulta".'?'.
    http_build_query($params);
    curl_setopt_array($curl, array(
        CURLOPT_URL => $url,
        //eh indicado a ativacao de verificacao SSL p/ ambiente de producao
        CURLOPT_SSL_VERIFYPEER => 0,
        CURLOPT_SSL_VERIFYHOST => 0,
        // curl_setopt($ch, CURLOPT_CAINFO, 'local/crt.crt');

        CURLOPT_CUSTOMREQUEST => "GET",
        CURLOPT_HTTPHEADER => array(
            "Content-Type: application/json",
            "token: SUA\_CHAVE\_TRANSACIONAL\_AQUI"
        ),
    ));
    $response = curl_exec($curl);
    $err = curl_error( $curl );
    if ($err) {
        echo "cURL Error: " . $err;
    } else {
        $info = curl_getinfo($curl);
    }
    curl_close($curl);
    return "Recurso Consulta -> HTTP Status Code: " . $info["http\_code"];
}
```

Exemplo em Java:

```
public static void consulta() {  
    try{  
        URL url = new  
URL("https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/consul  
ta?agencia=0116&posto=08&cedente=68062&nossoNumero=181000853");  
        HttpURLConnection con = ( HttpURLConnection) url.openConnection();  
        con.setRequestMethod( "GET");  
        con.setRequestProperty("Content-Type", "application/json" );  
        con.setRequestProperty("token", "SUA_CHAVE_TRANSACIONAL_AQUI" );  
        String jsonString = "";  
        InputStreamReader in = new InputStreamReader( con.getInputStream());  
        BufferedReader br = new BufferedReader(in);  
        String text = "";  
        while ((text = br.readLine()) != null) {  
            jsonString += text;  
        }  
        System.out.println("Resposta da Consulta: " + jsonString);  
        in.close() ;  
        br.close();  
    } catch (MalformedURLException ex) {  
        Logger.getLogger(Teste.class.getName()).log( Level.SEVERE, null, ex) ;  
    } catch (IOException ex) {  
        Logger.getLogger(Teste.class.getName()).log(Level.SEVERE, null, ex);  
    }  
}
```

Recurso Impressão

Descrição da Operação Impressão

A operação GET “impressao” é responsável pela impressão e reimpressão dos boletos de Cobrança. O formato de entrada e saída são JSON, considerando que a saída sempre é composta da entidade a ser retornada e um código HTTP dentro de um *Response*.

ENTRADA:

Ord	Nome	Tipo	Tipo de Parâmetro	Formatação	Descrição	Obrigatório
1	token	String	HeaderParam	64	Chave de transação gerada na operação de autenticação.	Obrigatório
2	agencia	String	QueryParam	4	Código da Cooperativa do Beneficiário.	Obrigatório
3	cedente	String	QueryParam	5	Código do beneficiário.	Obrigatório
4	nossoNumero	String	QueryParam	15	Nosso número do boleto.	Obrigatório
5	posto	String	QueryParam	2	Código do Posto do beneficiário. Quando o Posto for alfanumérico deverá ser informado “01”.	Obrigatório

SAÍDA:

Ord	Nome	Tipo	Tipo de Parâmetro	Formatação	Descrição
1	mensagem	String	JSON	Sem formatação	Mensagem de sucesso ou erro.
2	arquivo	byte[]	JSON	Sem formatação	Array de bytes do arquivo contendo os dados do boleto para a geração no formato PDF.

RETORNO (Response):

Código	Descrição
HTTP_OK (200)	Operação realizada com sucesso.

FALHAS (WebApplicationException):

Status	Código	Mensagem	Parâmetros de entrada sujeitos a crítica	Descrição
HTTP_NOT_FOUND (404)	0008	Ocorreu um erro ao realizar operação de impressão.	Não possui	Erros desconhecidos ocasionados por falhas de ambiente/negócio. Por favor, entrar em contato com o suporte.
HTTP_NOT_FOUND (404)	20001	ORA-20001: EToken de Transação inválido	token	Erros ocasionados por falhas de ambiente/negócio.
HTTP_BAD_REQUEST (400)	0010	Campo obrigatório em branco	Qualquer parâmetro que consta como obrigatório na listagem de parâmetros de entrada pode ser criticado.	Erros ocasionados por parâmetros obrigatórios não informados ou inválidos.
HTTP_BAD_REQUEST (400)	0011	Tamanho de campo inválido	Qualquer parâmetro que exceda o tamanho que foi informado na listagem de parâmetros de entrada pode ser criticado.	Erros ocasionados por parâmetros com tamanho inválido.
HTTP_BAD_REQUEST (400)	0012	Campo com caracteres especiais e/ou alfanuméricos.	Qualquer parâmetro que não contenha formatação na listagem de parâmetros de entrada pode ser criticado.	Erros ocasionados por parâmetros com caracteres não permitidos.

REQUEST - GET:

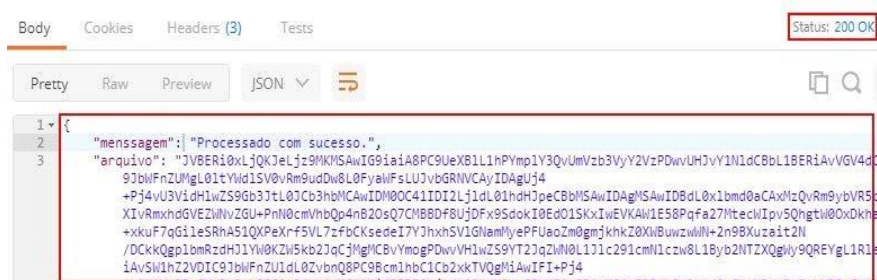
<https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/impressao?agencia=0101&cedente=02166&nossoNumero=171000088&posto=29>

RESPONSE:

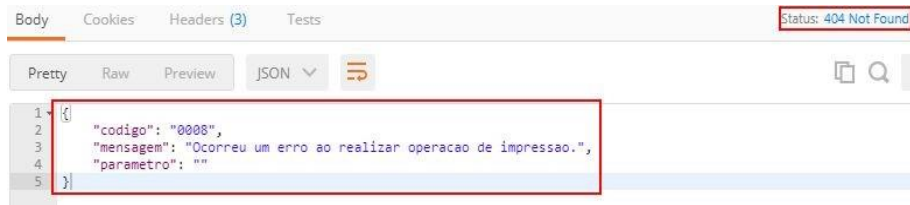
```
{
  "mensagem": "Processado com sucesso.",
  "arquivo": "JVBE7U2 --restante do byte array omitido--"
}
```

EXECUÇÃO (usando ferramenta Postman):

- Cenário positivo:



- Cenário de erro <NOT_FOUND>:



- Cenário de erro <BAD_REQUEST>:



Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

Exemplo em PHP:

```
function recursolImpressao() {
    $curl = curl_init();
    $params = array(
        'agencia' => '0116',
        'posto' => '08',
        'cedente' => '68062',
        'nossoNumero' => '181000853'
    );
    $url =
"https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/impressao"
    . '?' . http_build_query($params);
    curl_setopt_array($curl, array(
        CURLOPT_URL => $url,
        //eh indicado a ativacao de verificacao SSL p/ ambiente de producao
        CURLOPT_SSL_VERIFYPEER => 0,
        CURLOPT_SSL_VERIFYHOST => 0,
        // curl_setopt($ch, CURLOPT_CAINFO, 'local/crt.crt');

        CURLOPT_CUSTOMREQUEST => "GET",
        CURLOPT_HTTPHEADER => array(
            "Content-Type: application/json",
            "token: SUA\_CHAVE\_TRANSACIONAL\_AQUI"
        ) ,
    ));
    $response = curl_exec($curl);
    $err = curl_error($curl);
    if ($err) {
        echo "cURL Error: " . $err;
    } else
    {
        $info = curl_getinfo($curl);
    }
    curl_close( $curl );
    return "Recurso Impressao -> HTTP Status Code: " . $info["http_code"];
}
```

Elaborado por: Lucas de Castro	Data de Criação: 17/01/2018	Data de Atualização: 11/09/2019
--------------------------------	-----------------------------	---------------------------------

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

Exemplo em Java:

```

public static void impressao() {
    try {
        URL url = new
URL("https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/impre
ssao?agencia=0116&posto=08&cedente=68062&nossoNumero=181000853");
        HttpURLConnection con = (HttpURLConnection) url.openConnection();
        con.setRequestMethod("GET");
        con.setRequestProperty("Content-Type", "application/json");
        con.setRequestProperty("token", "SUA_CHAVE_TRANSACIONAL_AQUI");
        System.out.println("Impressao: " + con.getResponseCode());
        String jsonString = "";
        InputStreamReader in = new InputStreamReader(con.getInputStream());
        BufferedReader br = new BufferedReader(in);
        String text = "";
        while ((text = br.readLine()) != null) {
            jsonString += text;
        }
        System.out.println("Resposta da Impressao: " + jsonString);
        in.close();
        br.close();
    } catch (MalformedURLException ex) {
        Logger.getLogger(Teste.class.getName()).log(Level.SEVERE, null, ex);
    } catch (IOException ex) {
        Logger.getLogger(Teste.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```


Recurso Health

Descrição da Operação Health

A operação **GET** “health” é responsável pela verificação da disponibilidade do webservice de Cobrança.

REQUEST - GET:

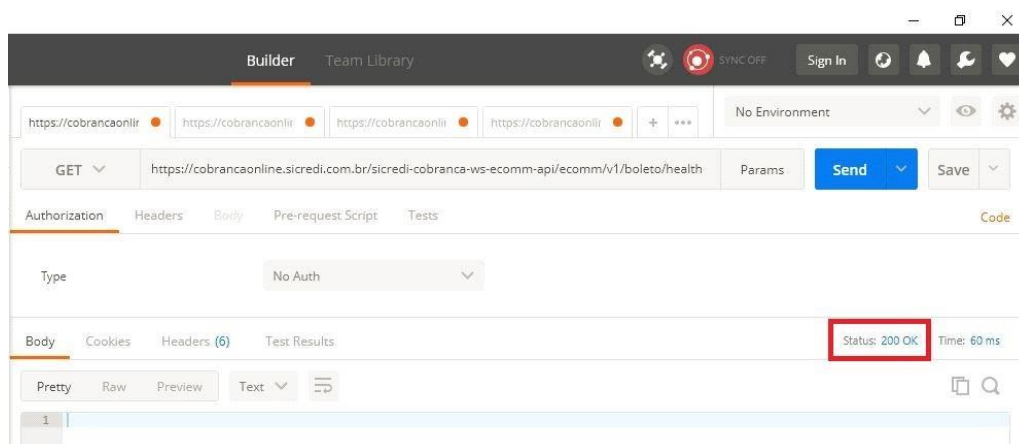
<https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/health>

RETORNO (Response):

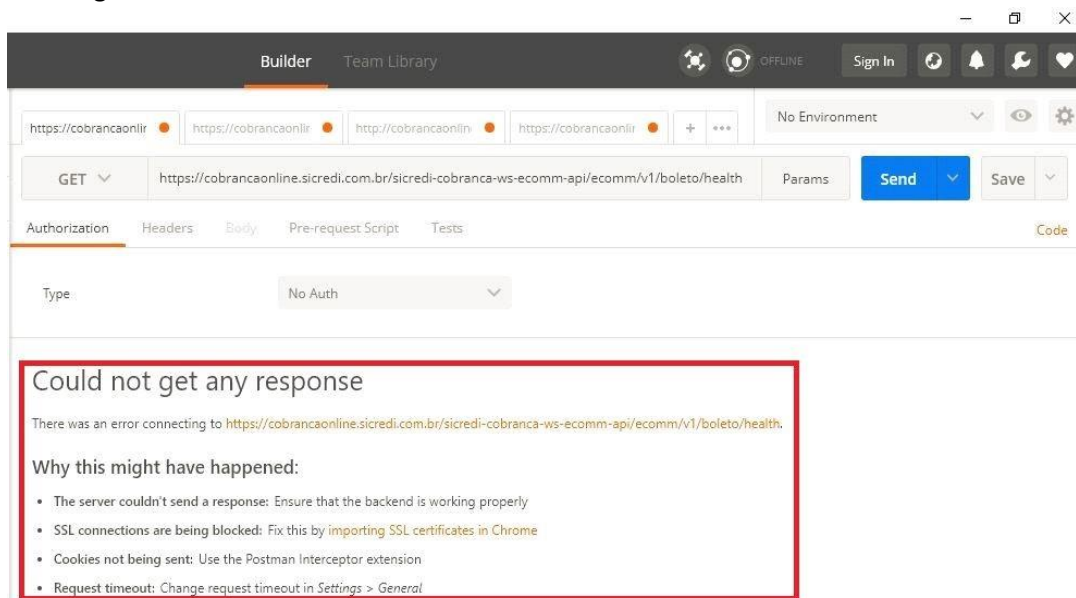
Status	Descrição
HTTP_OK (200)	Status OK

Execução (usando ferramenta Postman):

- Cenário Positivo:



- Cenário Negativo:



Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

Exemplo em PHP:

```
function recursoHealth()
{
    $curl = curl_init();
    curl_setopt_array($curl, array( CURLOPT_URL
        =>
        "https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/health",
        //eh indicado a ativacao de verificacao SSL p/ ambiente de producao
        CURLOPT_SSL_VERIFYPEER => 0 ,
        CURLOPT_SSL_VERIFYHOST => 0 ,
        // curl_setopt($ch, CURLOPT_CAINFO, 'local/crt.crt');

        CURLOPT_RETURNTRANSFER => true,
        CURLOPT_CUSTOMREQUEST => "GET",
        CURLOPT_HTTPHEADER => array(
            "Content-Type: application/json",
        ),
    ));
    $response = curl_exec($curl )    ;
    $err = curl_error($curl);
    if ($err) {
        echo "cURL Error: " . $err;
    } else {
        $info = curl_getinfo($curl);
    }
    curl_close($curl);
    return "Recurso Health -> HTTP Status Code: " . $info["http_code"];
}
```

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

Exemplo em Java:

```

public static void health() {
    try {

        URL url = new
URL("https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/health
");
        HttpURLConnection con = (HttpURLConnection) url.openConnection();
        con.setRequestMethod ( "GET" );
        int status = con.getResponseCode();
        System.out.println("Health: " + status);
        // Read response
        BufferedReader br = new BufferedReader (new InputStreamReader(con.getInputStream()));
        StringBuffer jsonString = new StringBuffer();
        String line;
        while ((line = br.readLine()) != null) { jsonString.append(line);}
        br.close();
        con.disconnect() ;
        System.out.println("Resposta do Health: " + jsonString);
    } catch (MalformedURLException ex) {
        Logger.getLogger(Teste.class.getName()).log (Level.SEVERE, null, ex );
    } catch ( IOException ex) {
        Logger.getLogger( Teste.class.getName ()).log (Level.SEVERE, null, ex );
    }
}

```

Recurso Comando de Instrução

Descrição da Operação Comando de Instrução

A operação **POST** “comandoInstrucao” é responsável por alterar dados dos títulos via serviço. Para executar com sucesso este serviço é necessário

ENTRADA:

Ord	Nome	Tipo	Formatação	Descrição	Obrigatório
1	agencia	String	Sem formatação	Agência do Beneficiário	Obrigatório
2	posto	String	Sem formatação	Posto do Beneficiário. Quando o Posto for alfanumérico deverá ser informado “01”.	Obrigatório
3	cedente	String	Sem formatação	Código do Beneficiário	Obrigatório
4	nossoNumero	String	Sem formatação	Nosso Número do Título	Obrigatório
5	instrucaoComando	String	<ul style="list-style-type: none"> PEDIDO_BAIXA CONCESSAO_ABATIMENTO CANCELAMENTO_ABATIMENTO_CONCEDIDO ALTERACAO_VENCIMENTO ALTERACAO_SEU_NUMERO PEDIDO_PROTESTO SUSTAR_PROTESTO_BAIXAR_TITULO SUSTAR_PROTESTO_MANTER_CARTEIRA ALTERACAO_OUTROS_DADOS 	Comando que será realizado na alteração do boleto. Caso seja informado ALTERACAO_OUTROS_DADOS será obrigatória a inclusão do campo complementoInstrucao	Obrigatório
6	complementoInstrucao	String	<ul style="list-style-type: none"> DESCONTO JUROS_DIA DESCONTO_DIA_ANTECIPACAO DATA_LIMITE_CONCESSAO_DESCONTO CANCELAMENTO_PROTESTO_AUTOMATICO CANCELAMENTO_NEGATIVACAO_AUTOMATICA 	Complemento do comando que será realizado na alteração do boleto	Condicional
7	<u>Campo de Alteração</u>	String	Sem formatação	Nome do campo que deseja mudar	Obrigatório

SAÍDA:

Ord	Nome	Tipo	Tipo de Parâmetro	Formatação	Descrição
1	codigo	String	JSON	Sem formatação	Código da mensagem.
2	mensagem	String	JSON	Sem formatação	Mensagem de sucesso ou erro.
3	parametro	String	JSON	Sem formatação	Parâmetro em que houve erro.

RETORNO (Response):

Status	Descrição
--------	-----------

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

HTTP_OK (200)	Status OK
---------------	-----------

FALHAS:

Status	Código	Mensagem	Descrição
HTTP_BAD_REQUEST (400)	E0032	Data de desconto já cadastrada.	Erro ao executar o comando de instrução para alteração de data de desconto.
HTTP_NOT_FOUND (404)	0116	Instrução inválida. Preenchimento obrigatório. Informe o campo obrigatório.	Campo exigido na operação não foi preenchido.

Quando um Comando de Instrução é realizado com sucesso, a seguinte resposta aparecerá:

```
{
  "codigo": "E0029",
  "mensagem": "Comando de Instrução realizado com sucesso!",
  "parametro": null
}
```

Caso não consiga executar algum comando, o campo "mensagem" dirá o erro em questão.

Segue o corpo dos JSON's dos comandos possíveis:

- PEDIDO_BAIXA - Consultar o Termo de Adesão com sua Unidade de Atendimento Sicredi p/ ver se está com Baixa Automática de boletos vencidos. Caso esteja, o boleto vencido terá sua baixa efetuada automaticamente.

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000352",
  "instrucaoComando": "PEDIDO_BAIXA",
  "complementoInstrucao": null
}
```

- CONCESSAO_ABATIMENTO

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000360",
  "instrucaoComando": "CONCESSAO_ABATIMENTO",
  "complementoInstrucao": null,
  "valor1": 3
}
```

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

- CANCELAMENTO_ABATIMENTO_CONCEDIDO

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000360",
  "instrucaoComando": "CANCELAMENTO_ABATIMENTO_CONCEDIDO",
  "complementoInstrucao": null
}
```

- ALTERACAO_VENCIMENTO

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000352",
  "instrucaoComando": "ALTERACAO_VENCIMENTO",
  "complementoInstrucao": null,
  "tipoVencimento": "DATA_ESPECIFICA",
  "data1": "30/06/2018"
}
```

- ALTERACAO_SEU_NUMERO

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000360",
  "instrucaoComando": "ALTERACAO_SEU_NUMERO",
  "complementoInstrucao": null,
  "seuNumero": "1234"
}
```

- PEDIDO_PROTESTO - Somente em título vencido. Por padrão, todos os boletos emitidos pelo EComm-API, não são protestados.

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000360",
}
```

Elaborado por: Lucas de Castro	Data de Criação: 17/01/2018	Data de Atualização: 11/09/2019
--------------------------------	-----------------------------	---------------------------------

```

    "instrucaoComando": "PEDIDO_PROTESTO",
    "complementoInstrucao": null
}

```

● SUSTAR_PROTESTO_BAIXAR_TITULO

```

{
    "agencia": "0100",
    "posto": "02",
    "cedente": "00248",
    "nossoNumero": "181000360",
    "instrucaoComando": "SUSTAR_PROTESTO_BAIXAR_TITULO"
}

```

● SUSTAR_PROTESTO_MANTER_CARTEIRA

```

{
    "agencia": "0100",
    "posto": "02",
    "cedente": "00248",
    "nossoNumero": "181000360",
    "instrucaoComando": "SUSTAR_PROTESTO_MANTER_CARTEIRA"
}

```

● DESCONTO

```

{
    "agencia": "0100",
    "posto": "02",
    "cedente": "00248",
    "nossoNumero": "181000360",
    "instrucaoComando": "ALTERACAO_OUTROS_DADOS",
    "complementoInstrucao": "DESCONTO", "valor1": 1
}

```

● JUROS_DIA - Muda somente o valor, o tipo de juros não muda (tipo A: valor fixo, tipo B: percentual). O tipo de juros é definido no momento da emissão do boleto.

```

{
    "agencia": "0100",
    "posto": "02",
    "cedente": "00248",
    "nossoNumero": "181000360",
    "instrucaoComando": "ALTERACAO_OUTROS_DADOS",
    "complementoInstrucao": "JUROS_DIA", "valor1": 2.5 }

```


Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

- DESCONTO_DIA_ANTECIPACAO

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000379",
  "instrucaoComando": "ALTERACAO_OUTROS_DADOS",
  "complementoInstrucao": "DESCONTO_DIA_ANTECIPACAO",
  "valor1": 1.5
}
```

- DATA_LIMITE_CONCESSAO_DESCONTO - Altera somente a data, o valor não. Caso o boleto emitido esteja somente com uma data de desconto cadastrada, irá alterar somente a data cadastrada. Se o boleto foi emitido sem nenhuma data de desconto, nada será alterado.

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000360",
  "instrucaoComando": "ALTERACAO_OUTROS_DADOS",
  "complementoInstrucao": "DATA_LIMITE_CONCESSAO_DESCONTO", "data1": "10/05/2018"
}
```

- CANCELAMENTO_PROTESTO_AUTOMATICO - Cancela solicitação de protesto automático do boleto, desde de que o mesmo ainda não esteja em fluxo de protesto como: “Em cartório“, “Protestado” ou “Aguardando sustação de protesto”.

```
{
  "agencia": "0100",
  "posto": "02",
  "cedente": "00248",
  "nossoNumero": "181000360",
  "instrucaoComando": "ALTERACAO_OUTROS_DADOS",
  "complementoInstrucao": "CANCELAMENTO_PROTESTO_AUTOMATICO "
}
```

- CANCELAMENTO_NEGATIVACAO_AUTOMATICA – Cancela solicitação de negativação automática do boleto, desde de que o mesmo ainda não esteja em fluxo de negativação como: “Aguardando entrada de negativação”, “Negativado” ou “Aguardando exclusão de negativação”.

```
{
  "agencia": "0100",
```

Elaborado por: Lucas de Castro	Data de Criação: 17/01/2018	Data de Atualização: 11/09/2019
--------------------------------	-----------------------------	---------------------------------

```
"posto": "02",  
"cedente": "00248",  
"nossoNumero": "181000360",  
"instrucaoComando": "ALTERACAO_OUTROS_DADOS",  
"complementoInstrucao": "CANCELAMENTO_NEGATIVACAO_AUTOMATICA"  
}
```

Exemplo em PHP:

```
function recursoComandoInstrucao() {
    $curl = curl_init();
    curl_setopt_array( $curl, array(
        CURLOPT_URL =>
"https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/comandoInstrucao",
        //eh indicado a ativacao de verificacao SSL p/ ambiente de producao
        CURLOPT_SSL_VERIFYPEER => 0,
        CURLOPT_SSL_VERIFYHOST => 0,
        // curl_setopt($ch, CURLOPT_CAINFO, 'local/crt.crt');

        CURLOPT_CUSTOMREQUEST => "POST",
        CURLOPT_POSTFIELDS => "{\n"
            . "\"agencia\": \"0116\", \n"
            . "\"posto\": \"08\", \n"
            . "\"cedente\": \"68062\", \n"
            . "\"nossoNumero\": \"181000853\", \n"
            . "\"instrucaoComando\": \"ALTERACAO_VENCIMENTO\", \n"
            . "\"complementoInstrucao\": null, \n"
            . "\"tipoVencimento\": \"DATA_ESPECIFICA\", \n"
            . "\"data1\": \"03/11/2018\" \n"
            . "}" ,
        CURLOPT_HTTPHEADER => array(
            "Content-Type: application/json",
            "token: SUA_CHAVE_TRANSACIONAL_AQUI"
        ) ,
    ));
    $response = curl_exec( $curl);
    $err = curl_error($curl);
    if ($err) {
        echo "cURL Error: " . $err;
    } else {
        $info = curl_getinfo( $curl );
    }
    curl_close( $curl );
    return "Recurso Comando de Instrucao -> HTTP Status Code: " . $info["http_code"];
}
```

Exemplo em Java:

```

public static void comandoInstrucao() {
    try {

        URL url = new
URL("https://cobrancaonline.sicredi.com.br/sicredi-cobranca-ws-ecomm-api/ecomm/v1/boleto/coma
ndoInstrucao");
        HttpURLConnection con = (HttpURLConnection) url.openConnection();
        con.setRequestMethod("POST");
        con.setDoOutput( true );
        con.setRequestProperty("Content-Type", "application/json");
        con.setRequestProperty("token", "SUA_CHAVE_TRANSACIONAL_AQUI");
        //comando de instrucao p/ alterar data de vencimento
        String input = "{\n
            + "\"agencia\" : \" 0116\", \n
            + "\"posto\" : \"08\", \n
            + "\"cedente\" : \" 68062\", \n
            + "\"nossoNumero\" : \"181000853\", \n
            + "\"instrucaoComando\" : \"ALTERACAO_VENCIMENTO\", \n
            + "\"complementoInstrucao\" : null, \n
            + "\"tipoVencimento\" : \" DATA_ESPECIFICA\", \n
            + "\"data1\" : \"02/11/2018\" \n
            + \"}\"";
        OutputStream os = con.getOutputStream();
        os.write(input.getBytes());
        os.flush();
        System.out.println("Comando de Instrucao: " + con.getResponseCode());
        // Read response
        String jsonString = "";
        InputStreamReader in = new InputStreamReader( con.getInputStream());
        BufferedReader br = new BufferedReader(in);
        String text = "";
        while ((text = br.readLine()) != null) { jsonString += text; }
        System.out.println ("Resposta do Comando de Instrucao: " + jsonString);
        os.close(); in.close(); br.close();
    } catch (MalformedURLException ex) {
        Logger.getLogger( Teste.class.getName()).log (Level.SEVERE, null, ex);
    } catch (IOException ex) {
        Logger.getLogger( Teste.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

Dúvidas Frequentes

1. Como é gerado o Token Master?

1.1 - No Internet Banking Sicredi, efetuar o *login* com a sua conta desejada



1.2 - Seleciona a opção Cobrança (caso não apareça, entre em contato com a sua Unidade de Atendimento Sicredi e solicite o uso do *webservice* EComm-API).



1.3 - Selecione a opção Gerar em Código de Acesso.



1.4 - Ler o Termo de Adesão, clicar em avançar e confirmar os dados. Será pedido o dispositivo de segurança, é o mesmo utilizado em todas as demais transações.

Página Inicial Cobrança Gerar Código de Acesso

1 Informar Dados **2** Confirmar Dados **3** Finalizar

Código de Acesso

Esse é o código de acesso necessário para autenticação do serviço de registro de boletos via Cobrança Online.

Conforme Termo de Adesão da Cobrança, é de sua exclusiva e intransferível responsabilidade a utilização e guarda da chave de acesso aqui gerada, não cabendo a imputação ao Sicredi de quaisquer ônus decorrentes de sua inadequada ou indevida utilização.

☒ Li e concordo com o termo de adesão.

cancelar avançar

1.5 - Depois de realizado os passos acima, será gerado o Token Master no campo Autenticação Eletrônica .

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

2. Como é gerado o Token Transacional?

O Token Transacional é gerado quando uma requisição ao recurso /autenticacao é executado, obtendo no retorno o token com sua respectiva data de expiração. É muito importante que seja guardado o Token Transacional pois não há uma forma de consulta posterior. Em caso de perda do Token Transacional, existem duas soluções:

1 - Esperar a expiração do Token Transacional. Demora 24 horas após gerado.

2 - Gerar um novo Token Master e usando-o para gerar um novo Transacional. Pode ser feito a qualquer momento. O Token Master é gerado no Internet Banking Sicredi logado com a conta do beneficiário.

3. Há alguma forma de recuperar o Token Transacional gerado?

Não. É muito importante que salve/guarde o Token Transacional.

4. Em caso do Token Master ficar pendente de autorização?

Isso ocorre porque a conta tem dois ou mais sócios, logo, a maioria dos sócios deve autorizar a geração do Token Master. Basta logar no Internet Banking Sicredi e autorizar o procedimento.

5. Caso ocorra um problema na comunicação com o webservice EComm-API e o Token Transacional não seja gerado, ficará sem emitir boleto até que o Token expire ou que seja gerada uma nova Chave Master?

Caso exista problema na comunicação com o webservice EComm-API, nenhuma chave será gerada, basta requisitar novamente o recurso /autenticacao para obter o Token Transacional. Caso exista problema no tratamento da chave gerada (ex.: sua aplicação recebeu o Token Transacional e por algum problema não conseguiu salvá-lo), logo, terá que aguardar a expiração do Token Transacional gerado ou gerar uma nova chave master.

6. No campo especieDocumento, qual opção devo utilizar para emissão de títulos?

Essa informação é referente à natureza do negócio. Entre em contato com sua Unidade de Atendimento Sicredi para saber qual espécie de documento se adequa melhor a sua empresa. A identificação incorreta da espécie do documento não vai impedir que o boleto seja pago e nem que o crédito seja efetuado na conta do beneficiário, mas pode ocasionar na impossibilidade de se protestar o boleto caso venha a ser necessário.

7. O campo endereço fica muito pequeno perto do que necessita ser colocado nele como: logradouro, número e complemento. Isso passa 40 caracteres na maioria das vezes. Pode gerar algum problema?

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

Não há problema em ter apenas 40 caracteres. O título será válido para pagamento. Caso seja necessário uma consulta para saber o endereço completo do pagador, basta usar o campo CEP que estará preenchido.

8. Nos campos mensagem e instrucao, é possível inserir quebra de linha?

Não há como inserir quebra de linha. A quebra de linha ocorrerá automaticamente a cada 80 caracteres. É provável que a separação de sílabas, na quebra de linha, ocorra de maneira inadequada, mas isso não invalida o título.

9. O campo cidade tem limite de 25 caracteres. Caso o nome da cidade exceda esse limite, como proceder?

Pode ser emitido títulos cujo o nome da cidade exceda 25 caracteres, o que acontecerá é que o nome será cortado depois do vigésimo quinto caracter, mas isso não invalida o boleto pois ainda consta o CEP em caso de ser necessário uma consulta para saber o nome completo da cidade e bairro.

10. Emissão de títulos, via *webservice* EComm-API, será registrada na CIP (Câmara Interbancária de Pagamentos) independente do horário do dia?

Via *webservice*, o registro na CIP demora apenas alguns segundos, logo, o título pode ser pago na sequência. Títulos enviados via arquivo CNAB (240 ou 400) após das 18hrs têm seu registro na CIP somente no próximo dia útil.

11. Existe ambiente de homologação/teste?

Não. Somente ambiente de produção.

12. O que acontece com boletos vencidos?

Boletos vencidos tem a sua baixa executada automaticamente de acordo com a quantidade de dias acordado no Termo de Adesão do beneficiário. Por exemplo: caso o beneficiário tenha 60 dias para baixa automática de títulos, o boleto ainda estará válido para pagamento, e somente 60 dias depois de vencido que será dado baixa automática. Quando um boleto é baixado, fica impossibilitado de pagamento ou alterações, e também não é mais cobrada taxas de manutenção do boleto.

13. Quem monta o layout do boleto (.PDF) e distribui os dados no boleto é o recurso /impressao do webservice da Sicredi?

Manual EComm - API		
	Projeto: Plataforma ECommerce – Cobrança Online	Versão 1.8

O *webservice* EComm-API é responsável pela montagem do .PDF do título em formato de *array* de *bytes*. Através de ferramentas de conversão, é possível gerar o .PDF do título usando o *array* de *bytes*.

14. Qual é a melhor utilização do campo mensagem na emissão de um boleto?

É um campo dedicado para instruir o operador de caixa bancário. Esse campo tem quebra automática de linha a cada 80 caracteres e pode exibir no máximo 300 caracteres (gerando até 4 linhas). Seguindo do campo mensagem será incluído as demais instruções do boleto como: vencimento, possíveis multas, juros e descontos, ou seja, no .PDF do boleto (na parte 'Instruções - texto de responsabilidade do beneficiário') aparecerá o campo 'mensagem' + demais instruções.

15. Como não há ambiente de homologação/teste, pode ser efetuado testes normalmente no ambiente de produção que não gerará custos?

Não. Confirme com a sua Unidade de Atendimento Sicredi se na emissão, impressão e comando de instrução de boletos consta alguma taxa operacional, isso depende de como foi feito o contrato de adesão com o Sicredi.

16. Em casos de boletos não pagos, como proceder?

Por padrão os boletos emitidos pelo EComm-API não são protestados automaticamente. Uma vez emitido o boleto, ele continuará existindo gerando apenas taxas de manutenção (consultar com a sua Unidade de Atendimento Sicredi qual foi a taxa acordada e a quantidade de dias para a Baixa Automática ser executada). Para baixar um título, basta usar o recurso /comandoInstrucao solicitando PEDIDO_BAIXA.

17. Caso queira alterar apenas o Valor do Boleto, tem algum comando de instrução?

Não há comando de instrução para tal. Terá que dar baixa no boleto existente e emitir um novo com o valor desejado.

18. Posso cadastrar títulos com CEP inválido?

Sim, basta desativar a opção de Valida CEP. Entre em contato com sua Unidade de Atendimento Sicredi para solicitar a desativação. A consequência de um CEP inválido no título é a impossibilidade de protesto.