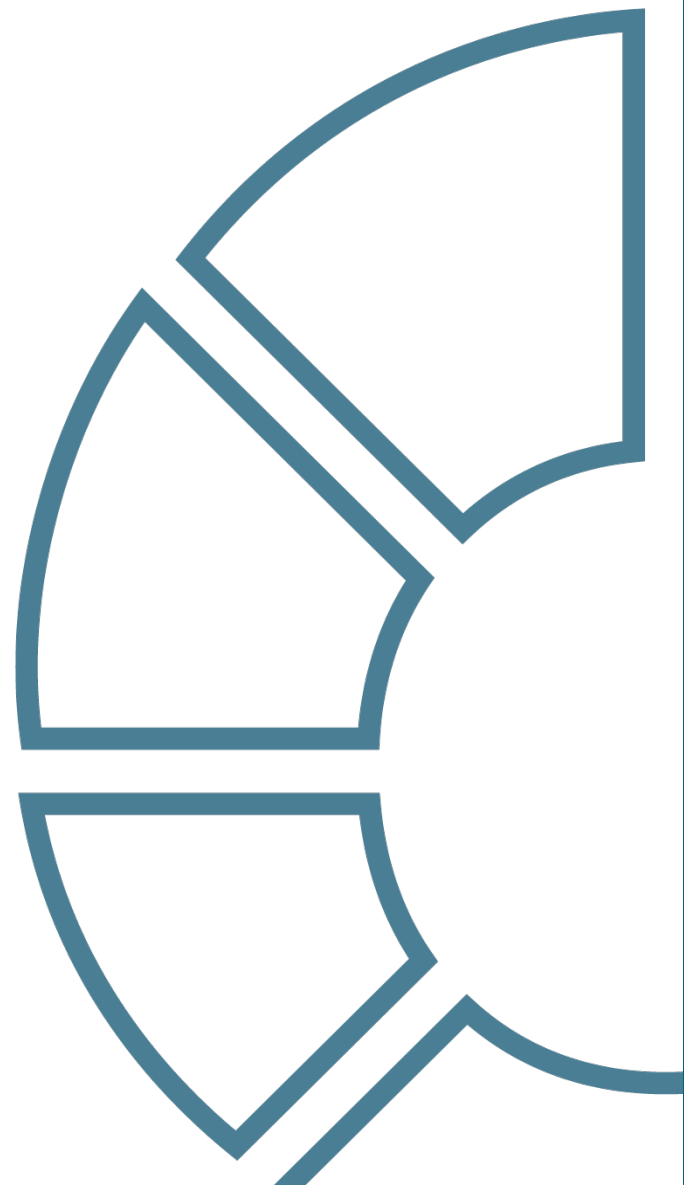




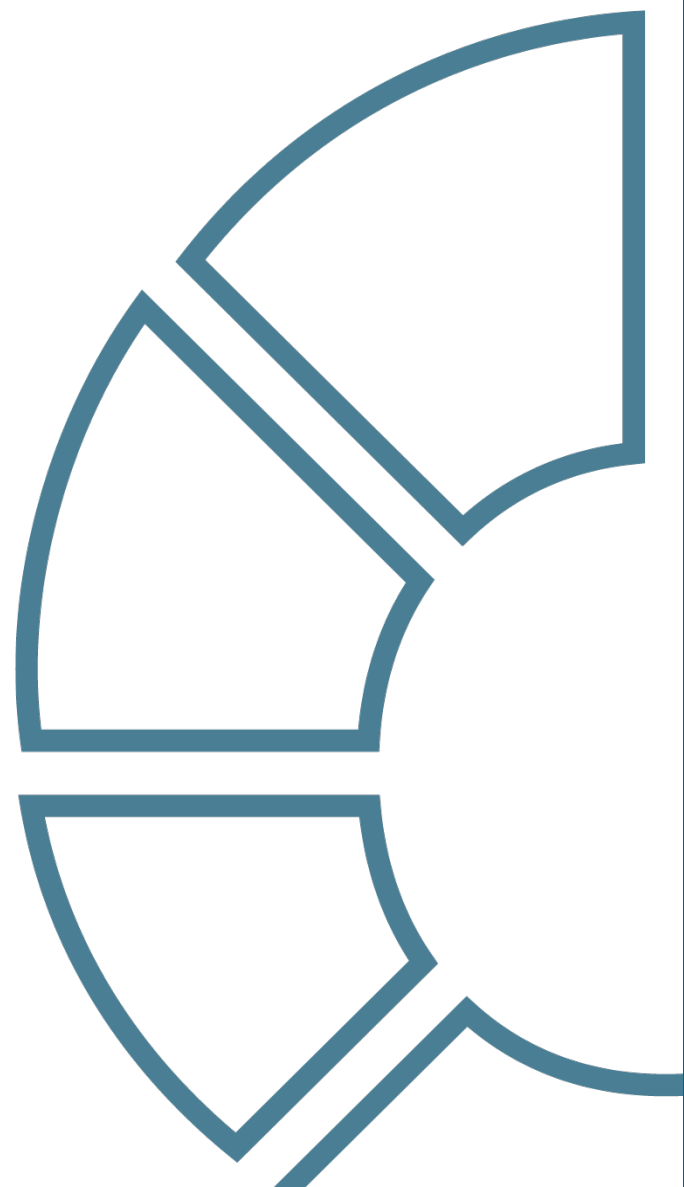
GUIA DE IMPLEMENTAÇÃO DA
VERSÃO 2.0 DAS APIS DE DADOS
CADASTRAIS E TRANSACIONAIS

CONTEÚDO DO GUIA



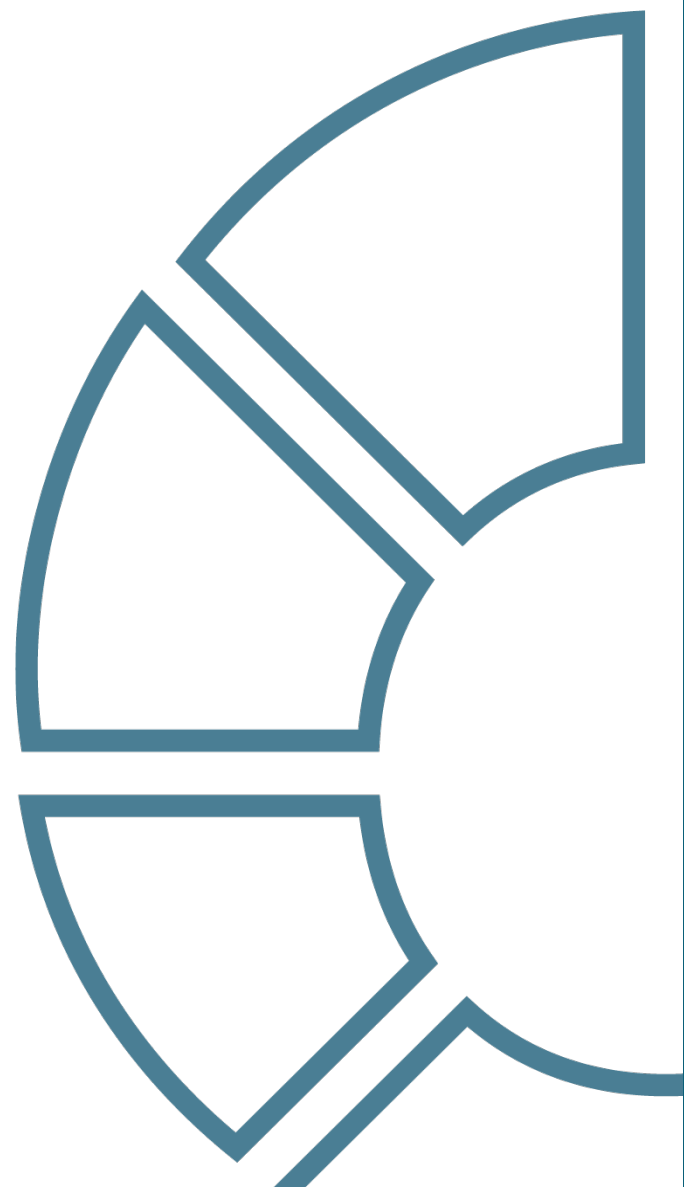
01	Introdução	11
	1.1. Detalhamento das alterações em relação a versão 1.0.X das APIs	13
	Descrição macro das alterações aplicadas às APIs de Dados Cadastrais e Transacionais em relação aos <i>payloads</i>	15
02	Política Versionamento	17
	2.1. Tabela de versões de lançamento das APIs	18
	2.2. Regras de versionamento semântico	19
	2.3. Prazos para implementação de versionamento	20
	2.4. Status para as fases da política de versionamento	21
	2.5. Para novos entrantes	22
03	Convivência de versões	23
	3.1. Atualização do diretório	25

CONTEÚDO DO GUIA



04 Controle de Eficiência do Ecossistema	27
4.1. Tempo de resposta	28
4.2. Controle de Timeout	28
4.3. Regras para revisão dos limites de tráfego e operacionais	29
4.4. Limites Operacionais	29
Contabilização de chamadas	31
Paginação	34
4.5. Limites de Tráfego	38
Quanto ao TPM (Transação por minuto):	38
Quanto ao TPS:	39
4.6. Criação de 2 novos endpoints	41
4.7. Quanto ao erro 422	42
4.8. Casos de Erro	43

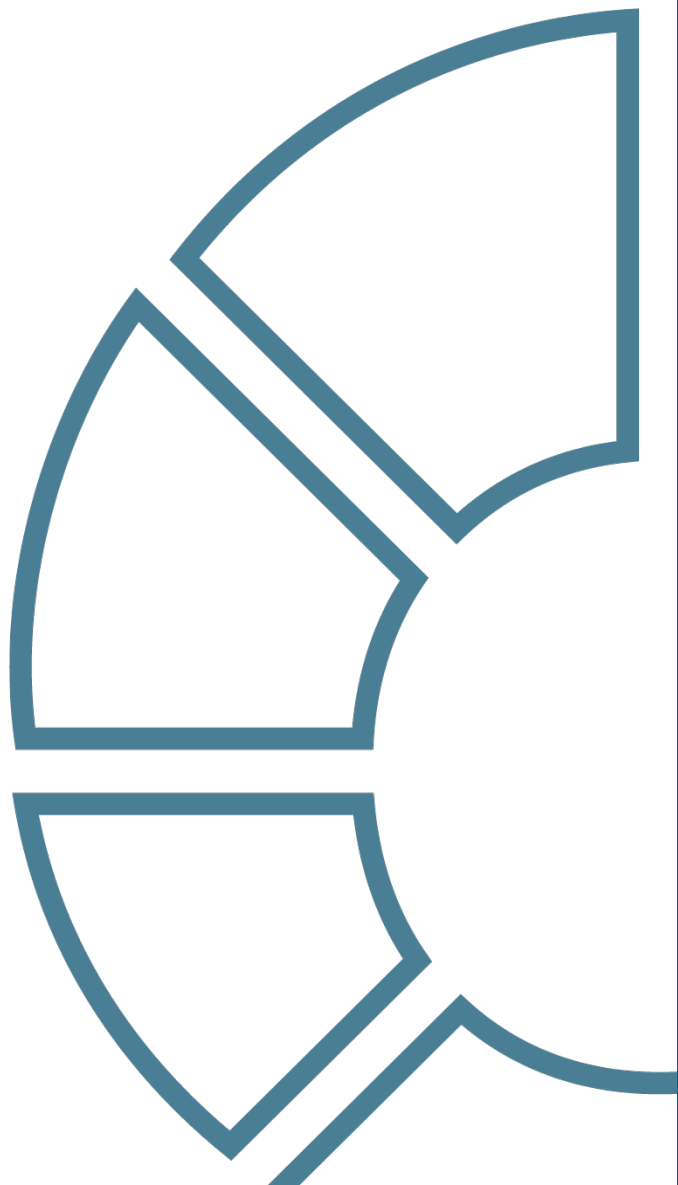
CONTEÚDO DO GUIA



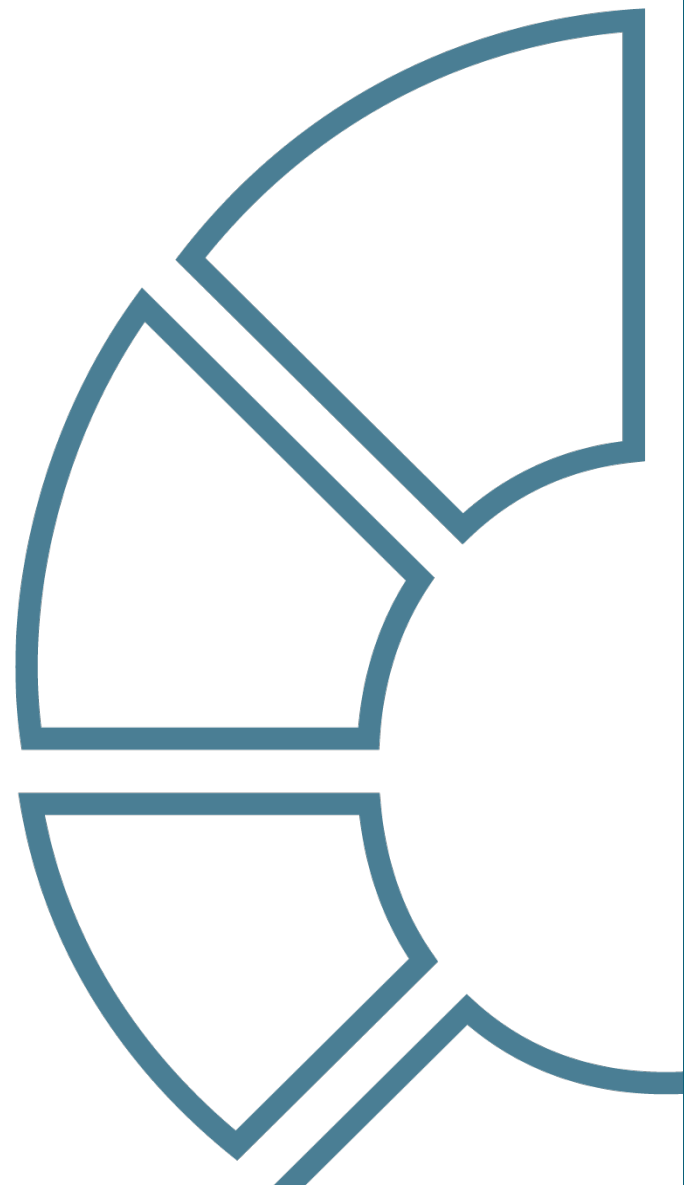
05 API de Consentimento (Consents)	45
5.1. Considerações Gerais	46
5.2. Sobre a Jornada de Consentimento	51
5.3. Regras de Negócios para implementação pela Instituição Transmissora	53
5.4. Regras para o preenchimento do campo status do Recurso (AWAITING_AUTHORISATION, AUTHORISED e REJECTED) e a transição entre status.	54
5.5. Detalhamento do motivo de rejeição do objeto Rejection	55
5.6. Convivência dos consentimentos nas versões V1 e V2 da API Consents	59
5.7. Transição de status	60
5.8. Tratamento para usuário final que não é cliente da Instituição Transmissora	61
5.9. O que foi alterado	61

CONTEÚDO DO GUIA

06	API de Recursos (Resources)	62
6.1.	Considerações Gerais	63
6.2.	Casos de uso e o seu respectivo status a ser informado pela API de Recursos (Resources)	64
6.3.	Transição de status da API de Recursos (Resources)	65
6.4.	Recomendação uso de polling	66
6.5.	Comportamento da API de Contas com relação a Resources	67
6.6.	Regras de negócio para implementação pela Instituição Transmissora	68
6.7.	Regras para o preenchimento do campo status do Recurso (PENDING_AUTHORISATION, AVAILABLE, UNAVAILABLE, TEMPORARILY_UNAVAILABLE) e a transição entre status	69
6.8.	Códigos de uso para o erro 403 na API de accounts (Contas)	70
6.9.	Regras de negócios para implementação de consumo pela Instituição Receptora	71

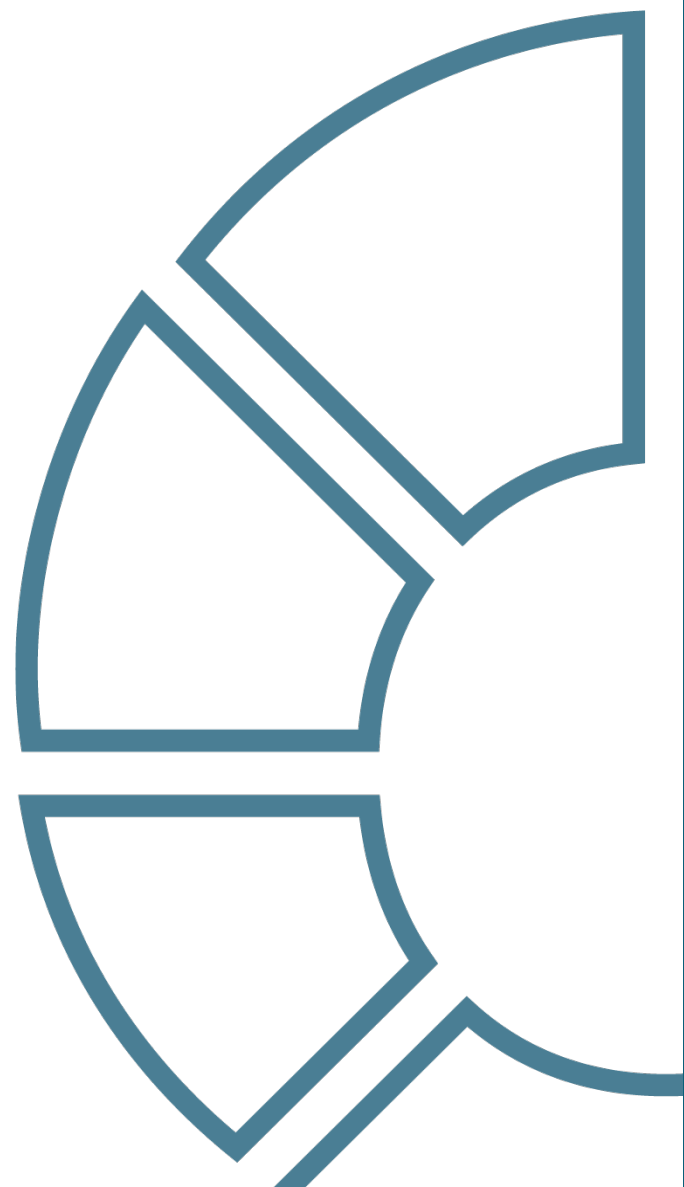


CONTEÚDO DO GUIA



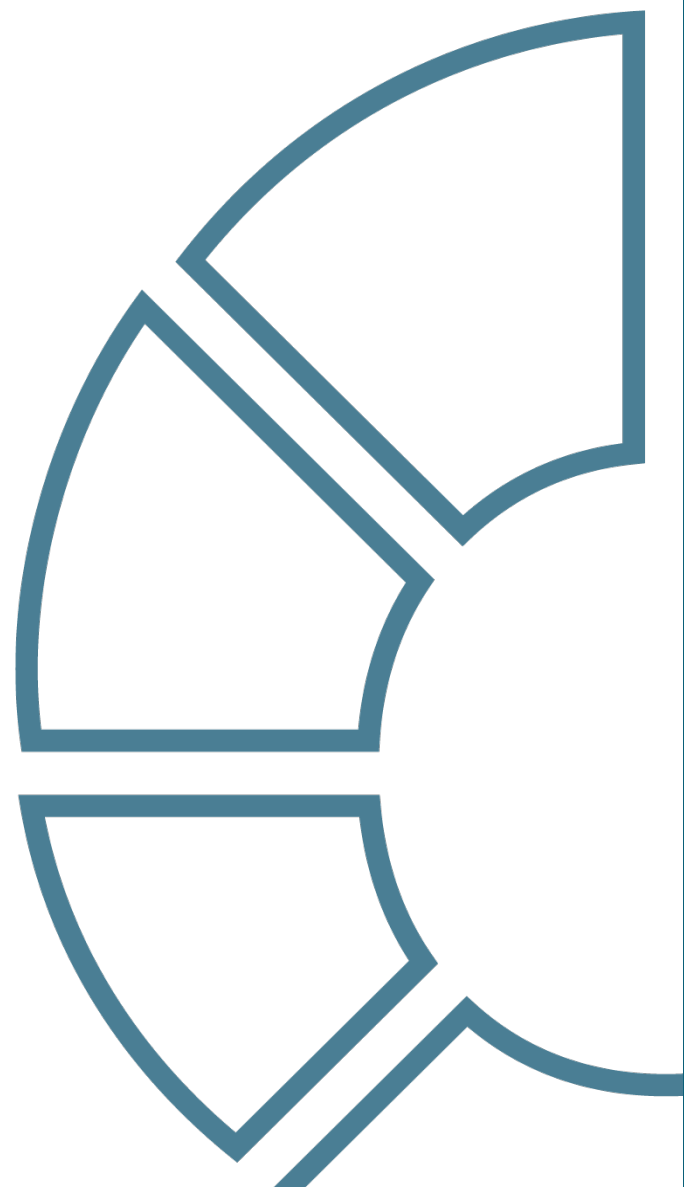
6.10. Regras quanto a busca de atualizações	72
6.11. Tabela descritiva das possibilidades de interação entre API de Recursos (Resources) e API de Contas (Accounts)	73
6.12. Comportamento da API de Dados Cadastrais com relação a Resources	74
6.13. Códigos de uso para o erro 403 na API de dados Cadastrais (Customers)	77
6.14. Comportamento da API de Cartões com relação a Resources	78
6.15. Regras de negócio de implementação pela Instituição Transmissora	79
6.16. Códigos de uso para o erro 403 na API Credit Cards Accounts (Cartão de Crédito)	80
6.17. Regras de negócio para implementação de consumo pela Instituição Receptora	81
6.18. Regras quanto a busca de atualizações	82
6.19. Tabela descritiva das possibilidades de interação entre API resources e as APIs de Operações de Crédito	83

CONTEÚDO DO GUIA



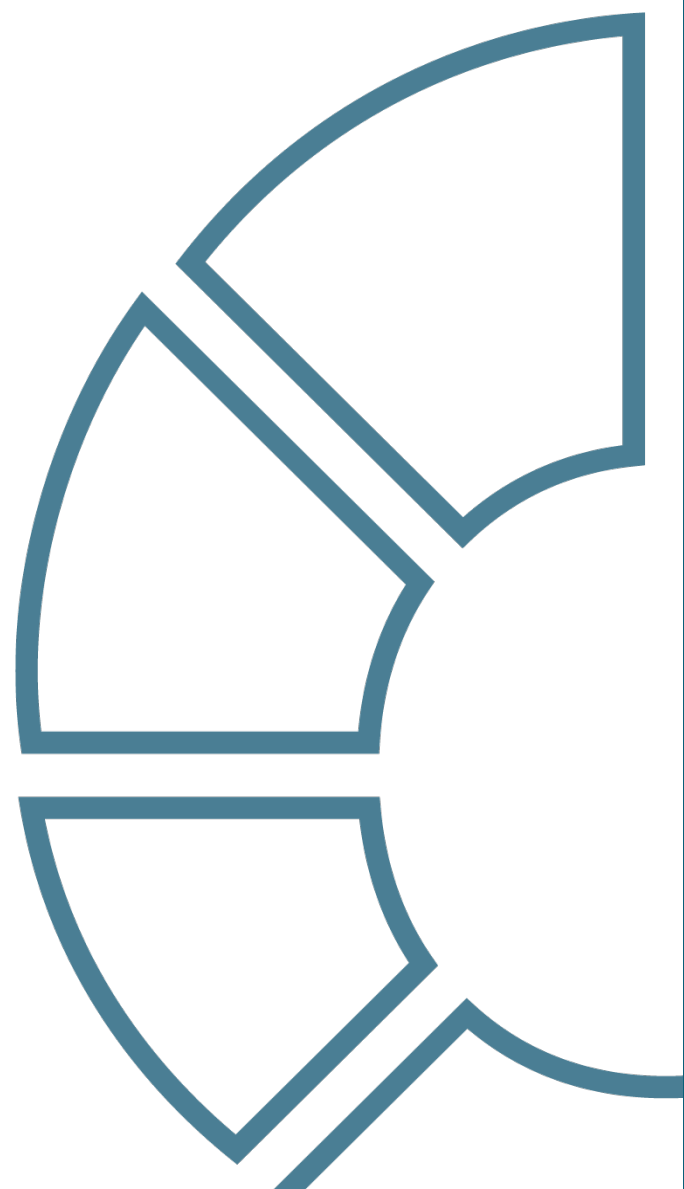
6.20. Comportamento das APIs de Crédito	84
6.21. Visão Geral da API Resources para as APIs de Operações de crédito	84
6.22. Regras de implementação pela Instituição Transmissora	85
6.23. Regras de negócio	88
6.24. Exemplo de retorno JSON	90
6.25. Diagrama representacional dos status da API de Recursos (Resources)	92
6.26. Regras para o preenchimento do campo status do Recurso (PENDING_AUTHORISATION, AVAILABLE, UNAVAILABLE, TEMPORARILY_UNAVAILABLE) e a transição entre status	93
6.27. Casos de uso de Operações de Crédito e o respectivo status a ser informado pela API de Recursos (Resources)	95
6.28. Regras de consumo pela Instituição Receptora	97
6.29. Atualizações na documentação de Resources quanto as regras de busca das informações nas APIs de Operações de Crédito	98
6.30. Tabela descritiva das possibilidades de interação entre API resources e as APIs de Operações de Crédito	99
6.31. Regras para o uso do status do Recurso	106

CONTEÚDO DO GUIA



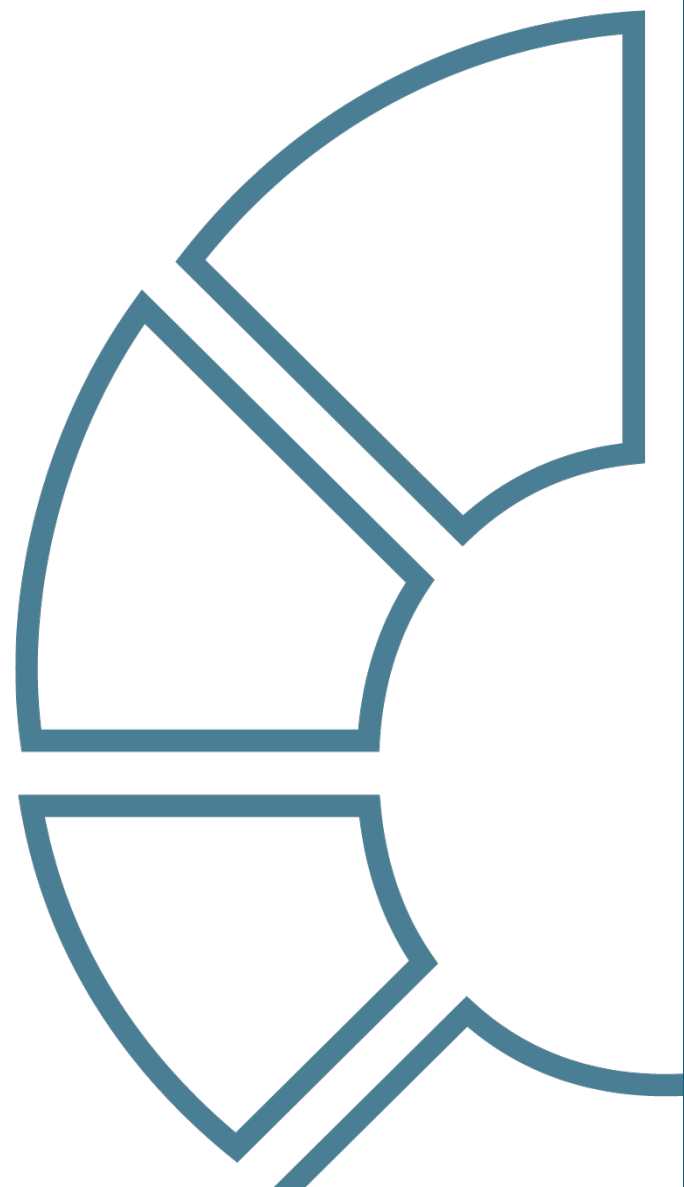
07	API de Dados Cadastrais (Customers)	107
7.1.	API Customers - Open Finance Brasil	108
7.2.	Orientações	109
7.3.	O que foi alterado	110
08	API de Cartões de Crédito (Credit-Cards-Accounts)	111
8.1.	API Credit-cards-accounts - Open Finance Brasil	112
8.2.	Orientações	113
8.3.	O que foi alterado	114
09	API de Contas (Accounts)	115
10.1.	API Accounts - Open Finance Brasil	116
10.2.	Orientações	117
10.3.	Diagrama Visual	118
10.4.	O que foi alterado	119

CONTEÚDO DO GUIA



10	API Operações de Crédito: Empréstimos (Loans)	120
	11.1. API Loans - Open Finance Brasil	121
	11.2. Orientações	122
	11.3. O que foi alterado	123
11	API Operações de Crédito: Financiamento (Financings)	124
	12.1. API Financings - Open Finance Brasil	125
	12.2. Orientações	126
	12.3. O que foi alterado	127

CONTEÚDO DO GUIA



12	API Operações de Crédito: Adiantamento a Depositantes (Unarranged-Accounts-Overdraft)	128
13.1.	API Unarranged-accounts-overdraft – Open Finance Brasil	129
13.2.	Orientações	130
13.3.	O que foi alterado	131
13	API Operações de Crédito: Direitos Creditórios Descontados (Invoice-Financings)	132
14.1.	API Invoice-financings – Open Finance Brasil	133
14.2.	Orientações	134
14.3.	O que foi alterado	135



01

Introdução

Este guia de implementação tem por objetivo **facilitar o desenvolvimento das Versões 2.0 das APIs de compartilhamento de dados cadastrais e transacionais**, além de trazer as alterações destas APIs quando comparadas às suas versões 1.0.X.

Além das alterações das APIs, o Guia também traz **regras gerais e casos de uso** para as APIs de Consentimento e Recursos (Consents e Resources), Política de Versionamento e instruções para controle de eficiência no ecossistema.

O Guia é resultado da análise de 403 *tickets* de *Service Desk* registrados pelos participantes do ecossistema, da lista de problemas conhecidos da Fase 2 e mais de 30 propostas debatidas no GT Especificações.

Foram criados 2 novos *endpoints*, para que sejam compartilhadas informações com alta frequência de atualização (**GET /accounts/v2/accounts/{accountId}/transactions-current** e **GET /credit-cards-accounts/v2/accounts/{creditCardAccountId}/transactions-current**).





1.1

Detalhamento das alterações em relação à versão 1.0.X das APIs



Foram feitas **455 alterações nos contratos das APIs de compartilhamento de dados (Fase 2)**, que podem ser encontradas no *changelog* de forma mais detalhada no link: <https://openbankingbrasil.atlassian.net/wiki/spaces/OB/pages/1738152/Change+Log>.



APIs	Quantidade de Ajustes
Consentimento (Consents)	2
Recursos (Resources)	2
Dados Cadastrais (Customers)	106
Contas (Accounts)	24
Cartão de Crédito (Credit-Cards-Accounts)	65
Financiamentos (Financings)	77
Empréstimos (Loans)	52
Direitos Creditórios Descontados (Invoice-Financings)	68
Adiantamento a Depositantes (Unarranged-Accounts-Overdraft)	59
Total	455

DESCRIÇÃO MACRO DAS ALTERAÇÕES APLICADAS ÀS APIS DE DADOS CADASTRAIS E TRANSACIONAIS EM RELAÇÃO AOS PAYLOADS

- Nenhum campo poderá mais receber o valor NA:
 - **Obrigatórios**, foram revistas as mandatoriedades a fim de atender as regras de negócios de cada um dos produtos;
 - **Opcionais**, para este caso basta não enviar o campo;
 - **Condicionais**, se comportam como opcionais, exceto quando o cenário se encaixar na restrição incluída na descrição;
 - Mesmo conceito aplicado para enums com o valor `NAO_APLICA`;
 - Removidos todos os atributos `Nullable` dos campos, que foram analisados obedecendo critérios de mandatoriedade e cenários de negócio.
- Com o objetivo de normalizar o formato dos dados entre as pontas, foram acrescentados *patterns regex* em todos os campos elegíveis.
- Foram normalizados os atributos dos campos de acordo com o seu tipo de dado.
- Todos os campos `additionalInfo` foram alterados para opcional ou condicional de acordo com suas características de negócio. Para os campos condicionais foram acrescentadas restrições de cenário de uso.



DESCRIÇÃO MACRO DAS ALTERAÇÕES APLICADAS ÀS APIS DE DADOS CADASTRAIS E TRANSACIONAIS EM RELAÇÃO AOS PAYLOADS

- Foram alterados todos os campos numéricos do tipo `double` para `string`, com isso garantimos que exista um formato de dados padrão, tornando possível que todas as casas, considerando suas peculiaridades sistêmicas, consigam realizar o tráfego das informações.
- Revista a mandatoriedade dos `arrays`, considerando os seguintes cenários:
 - **Obrigatórios**, passam a receber `minItems. 0`, considerando que caso o bloco de informações não exista, ele não seja enviado. Nesse cenário atendemos ao conjunto de informações relativas a um produto, onde um cenário específico não pode ser atendido, mas ele existe como característica do produto, é como quem está enviando dizer “não tenho essa informação para este produto”.
 - **Opcionais e Condicionais**, passam a receber `minItems. 1`, revista a opcionalidade para não existência das informações em um determinado cenário de negócio, basta não enviar o `array`, caso exista, deve ser preenchido pelo menos um elemento.
- Inserção e remoção de campos atendendo às especificidades de cada produto (API).





02

Política de Versionamento



A política de versionamento estabelece as **regras para o lançamento** de novas versões das APIs existentes e também os prazos de retornos do ecossistema quanto a ajustes nas versões *Release Candidates* e ainda os prazos de convivência das versões diferentes destas APIs.

TABELA DE VERSÕES DE LANÇAMENTO DAS APIS

Etapa	Descrição	Prazo ¹	Entregável	Portal a ser publicado
Versão Alpha:	Estágio de desenvolvimento para avaliação de testes internos - versão inicial, onde ainda existe a possibilidade de alterações na estratégia de desenvolvimento da API	Não se aplica	<ul style="list-style-type: none"> Escopo inicial da API Planilha inicial de campos e descrições Publicação no portal sobre o lançamento da API, seu escopo e planilha de trabalho Fornecedor produz a primeira versão do <i>Swagger</i> - 1.0.0 <i>Alpha</i> 	Consumo interno da Convenção
Versão Beta:	Estágio suficientemente evoluído para avaliações e testes externos - pode ter ainda mudanças significativas	10 dias úteis	<ul style="list-style-type: none"> Avaliação do <i>Swagger</i> inicial Publicação da versão 1.0.0 Beta 	Área do Desenvolvedor - <i>Draft</i>
Versão Release Candidate:	Quase pronta para lançamento final, sem novos aprimoramentos de recursos, limitando alterações a correções de <i>bugs</i> e questões de segurança - versão mais estruturada e próxima ao que será a versão estável	+10 dias úteis	<ul style="list-style-type: none"> Avaliação no <i>Swagger</i> 1.0.0 Beta Publicação da versão 1.0.0 RC Desenvolvimento do Motor de Conformidade Massa de dados 	Área do Desenvolvedor
Versão Estável:	Versão de lançamento - versão com as especificações prontas para uso	Não se aplica	<ul style="list-style-type: none"> Versão estável da API - 1.0.0 (<i>Yaml-Swagger</i> e Dicionário de dados) 	Área do Desenvolvedor

1 - Prazo para receber retorno do ecossistema, para eventuais correções e ajustes



REGRAS DE VERSIONAMENTO SEMÂNTICO

Dado um número de versão baseado no modelo MAJOR.MINOR.PATCH, incremente a:

- 1. Versão Maior (MAJOR):** quando fizer mudanças incompatíveis na API,
- 2. Versão Menor (MINOR):** quando adicionar funcionalidades mantendo compatibilidade, e
- 3. Versão de Correção (PATCH):** quando corrigir falhas mantendo a compatibilidade.

Rótulos adicionais para pré-lançamento (*pre-release*) e metadados de construção (*build*) estão disponíveis como extensão ao formato MAJOR.MINOR.PATCH.

Cod. Versão: 1.0.0

O versionamento de cada API é iniciado pela versão 1.0.0

*Serão seguidas as regras de versionamento semântico 2.0 disponíveis em: <https://semver.org/lang/pt-BR/>

Exemplo Precedência de versões:

1.0.0-alpha < 1.0.0-alpha.1 < 1.0.0-alpha.beta < 1.0.0-beta < 1.0.0-beta.2 < 1.0.0-beta.11 < 1.0.0-rc.1 < 1.0.0



PRAZOS PARA IMPLEMENTAÇÃO DE VERSIONAMENTO

Versionamento	Release	Notificação de mudança	Implementação	Depreciação	Exige recertificação?
Major	6 meses	60 dias antes do dia D	Até 120 dias depois do dia D	210 dias depois do dia D	Sim
Minor	3 meses	30 dias antes do dia D	Até 60 dias depois do dia D	-	Avaliar no pacote
Patch	1 mês	Até 15 dias antes do dia D	Até 30 dias depois do dia D	-	Não

É necessário existir um tempo de convivência de versões diferentes no ecossistema

Dia D: dia da publicação da especificação estável da versão pelo ecossistema

Release: intervalo mínimo entre duas versões estáveis daquela categoria

Notificação de mudança: esclarecimento das mudanças que ocorrerão na API, acompanhadas de potenciais versões: beta ou *release candidate*

Implementação: é o prazo que as Transmissoras/Detentoras terão para se adequar à nova especificação, incluindo certificação e implantação

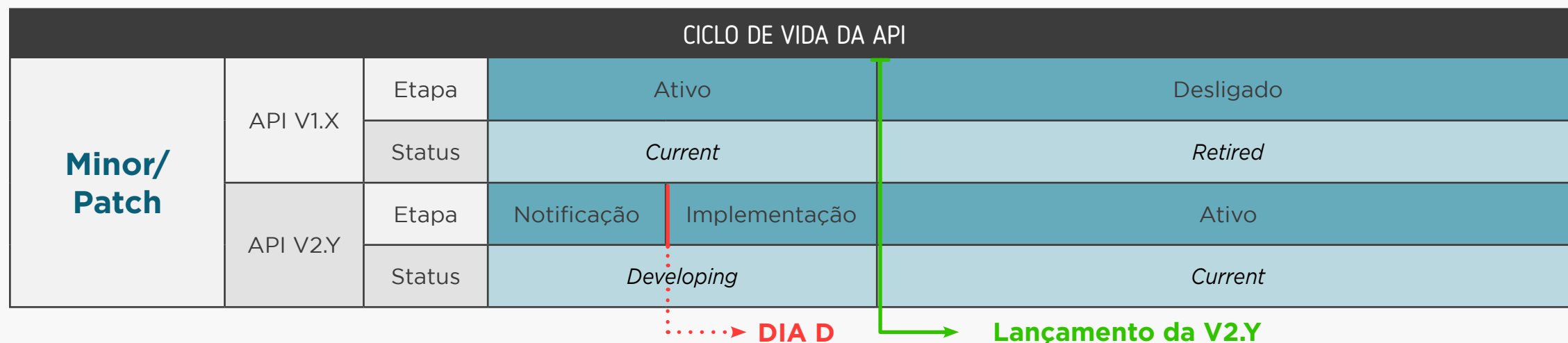
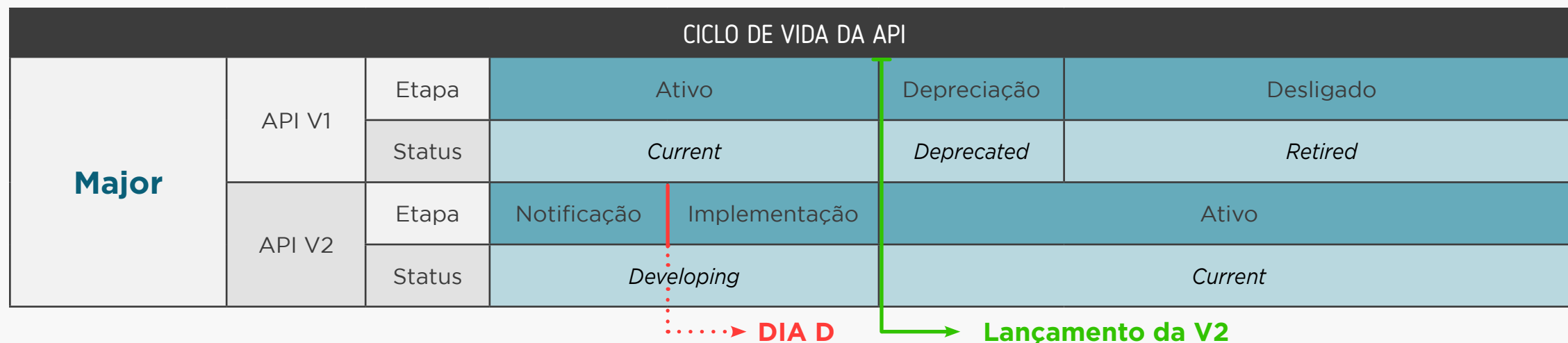
Depreciação: a Transmissora/Detentora precisa manter as duas versões no ar, pois é o período de adaptação do ecossistema (até 90 dias contados desde a data de implementação ou 210 dias desde o dia D)

Exige recertificação?

- **Sim:** há necessidade de nova certificação
- **Avaliar no pacote:** há necessidade de verificar de acordo com a solicitação de ajuste
- **Não:** não há necessidade de nova certificação



STATUS PARA AS FASES DA POLÍTICA DE VERSIONAMENTO



Se faz necessária a criação de novos status baseados nas etapas da política de versionamento, os quais serão publicados no Portal do Desenvolvedor junto com as versões das APIs.

1. **Developing:** abrange o período da notificação da nova versão até o final da implementação nas casas.
2. **Current:** abrange o período após a implementação até o lançamento da próxima versão.
3. **Deprecated:** abrange o período de depreciação até o momento que a versão é desligada do ambiente produtivo.
4. **Retired:** abrange o período após a depreciação da versão, ou seja, quando a API estará desligada.



PARA NOVOS ENTRANTES

Espera-se que os novos participantes do ecossistema implementem a última versão da API publicada com o status “*Current*” e sigam o ciclo de vida da mesma a partir da etapa em que se encontra.





03

Convivência de versões

Até a depreciação definitiva das APIs, existirá o período de convivência, no qual as Instituições Transmissoras terão que disponibilizar as APIs da v1.0.X e v2.0.0. Desta forma as Receptoras podem fazer a migração de versão, de forma a manter a disponibilidade.

Por exemplo:

GET /loans/v1/contracts

GET /loans/v2/contracts





ATUALIZAÇÃO DO DIRETÓRIO

Segue abaixo instruções para atualização no Diretório, referentes a inclusão da versão 2.0.0 das APIs.

Atualmente existe o campo “Version” que suporta o versionamento semântico, sendo possível adicionar múltiplas APIs do mesmo grupo (ex: *Accounts*) com versões diferentes, bastando a Instituição participante adicionar múltiplas entradas na tabela de “Recursos da API”

Em sequência, os campos necessários para a adição de uma API no diretório, incluindo os dois campos supracitados

 [Clique aqui para encontrar demais informações no Guia de Operação do Diretório Central.](#)



ATUALIZAÇÃO DO DIRETÓRIO

Edit accounts ✕

Api Resource Information

Api Family Type accounts	Version* 1.0.0
Certification URI* https://webhook.site/998e92b7-8102-4549-94'	
Start date* 13/05/2022	End date* 13/05/2023
Current status Self-Certified	Api Resource Id b3826276-7f0d-4984-afca-d0854b5aec9b

CANCEL SAVE



04

Controle de Eficiência do Ecossistema



TEMPO DE RESPOSTA (SLA)

Os *endpoints* foram classificados de acordo com a frequência de atualização dos dados, conforme segue:

- **Alta frequência de alteração.**
- **Média frequência de alteração.**
- **Baixa frequência de alteração.**

Foi decidido a aplicação de diferentes tempos de resposta (SLAs) com base nas classificações de frequência dos *endpoints*:

- **Alta frequência: 1.500ms (1,5s)**
- **Média frequência: 2.000ms (2,0s)**
- **Baixa frequência: 4.000ms (4,0s)**

CONTROLE DE TIMEOUT

Padronização do *timeout* de tempo de resposta do servidor (*server*) e do tempo de espera resposta do consumidor (*client*) em **15 segundos**

Definição do erro **HTTP Status Code 504 - Gateway Timeout** como resposta do **atingimento do TIMEOUT do servidor**



REGRAS PARA REVISÃO DOS LIMITES DE TRÁFEGO E OPERACIONAIS

As revisões para limites de tráfego e operacionais deverão ocorrer após três meses da data de lançamento em produção da Versão 2.0 das APIs.

LIMITES OPERACIONAIS

Foram estabelecidos limites operacionais com a definição de um controle de chamadas no âmbito do cliente e recurso/consentimento, conforme segue:

- **1 chamada = 1 solicitação a 1 endpoint**
- Categorização de limites por:
 - Período: mensal, com período de aferição iniciando no primeiro dia do mês e encerrando no último dia do mês;
 - Cliente: CPF/CNPJ (não o consentimento);
 - IF (*Organization Id*);
 - *Endpoint*;
 - **Critério adicional:**
 1. Consentimento: quando *endpoint* retorna lista de recursos; ou
 2. Recurso / Produto: Quando *endpoint* retorna detalhes de um recurso.





LIMITES OPERACIONAIS

As APIs de Dados Abertos, Iniciação de Pagamentos, APIs de Segurança, Consentimento (Consents) e Recursos (Resources), não terão aplicáveis sobre si os limites operacionais.

A referência de fuso horário para reiniciar a contagem de limites operacionais será 0:00 do horário de Brasília.

A implementação dos controles pode ser realizada de **forma assíncrona**, isto é, a Instituição Transmissora pode, ao processar um **GET**, verificar se o limite já foi atingido (ainda que com alguma defasagem) e em um processo assíncrono processar a lógica e, se necessário, atualizar a indicação de que o limite foi atingido.

Alternativamente a implementação dos controles pode ser realizada também de forma síncrona, isto é, realizando a validação dos limites operacionais de forma online.

A implementação dos limites operacionais é opcional pelas Transmissoras. Uma vez que sejam implementados, devem garantir o consumo mínimo estabelecido, sendo facultado à Transmissora a possibilidade de ampliar estes limites, mas vedada a implementação de limites inferiores aos estabelecidos.

Exemplo: para os *endpoints* de alta frequência está estabelecido o limite de 240 chamadas ao mês. Assim, uma Transmissora poderia limitar em 270, mas não poderia limitar em 210.

Caso a Instituição Transmissora consiga atender a todas as requisições recebidas dentro do seu controle de limite de tráfego (TPS/TPM), atendendo às quantidades mínimas de limites operacionais estabelecidas, a implementação do controle de limites operacionais se torna opcional.



CONTABILIZAÇÃO DE CHAMADAS

- Chamadas que devem ser consideradas na contabilização:

- **Status Code 2XX:** respostas com sucesso.

- Chamadas que não deverão ser consideradas na contabilização:

- **Status Code 5XX:** respostas com erro no servidor de APIs.
- **Status Code 4XX:** incluindo o 403 (Forbidden), 404 (Not Found).

- Re-chamadas para fins de paginação não deverão ser contabilizadas.
- Criação de novo status code para atingimento de limite - Código 423.

A implementação do parâmetro *pagination key* é obrigatório para as instituições implementarem o limite operacional.





4 Controle de Eficiência do Ecosistema

Endpoint	Classificação	SLA	Limite Operacional
Saldos da conta	Alta frequência de alteração	1.500ms (1,5s)	Consulta 420x ao mês*
Limites da conta	Alta frequência de alteração	1.500ms (1,5s)	Consulta 420x ao mês*
Limites de cartão de crédito	Alta frequência de alteração	1.500ms (1,5s)	Consulta 240x ao mês
Transações de cartão de crédito / recente	Alta frequência de alteração	1.500ms (1,5s)	Consulta 240x ao mês
Transações da conta / recente	Alta frequência de alteração	1.500ms (1,5s)	Consulta 240x ao mês
Consents (Limites aplicados à API de Consents serão tratados separadamente, dada a sua criticidade na jornada do usuário do OPB)	Alta frequência de alteração	1.500ms (1,5s)	Não se aplica
Resources (Não se aplicam limites de chamada operacionais para API de Resources)	Alta frequência de alteração	1.500ms (1,5s)	Não se aplica
Fatura do cartão de crédito	Média frequência	2.000ms (2,0s)	Consulta 30x ao mês
Transações de cartão de crédito por fatura	Média frequência	2.000ms (2,0s)	Consulta 30x ao mês
Empréstimos	Média frequência	2.000ms (2,0s)	Consulta 30x ao mês
Financiamentos	Média frequência	2.000ms (2,0s)	Consulta 30x ao mês
Adiantamento a depositantes	Média frequência	2.000ms (2,0s)	Consulta 30x ao mês
Direitos Creditórios Descontados	Média frequência	2.000ms (2,0s)	Consulta 30x ao mês
Pagamentos do contrato	Média frequência	2.000ms (2,0s)	Consulta 30x ao mês
Parcelas do contrato	Média frequência	2.000ms (2,0s)	Consulta 30x ao mês

*Exceção do limite operacional máximo mensal



4 Controle de Eficiência do Ecosistema

Endpoint	Classificação	SLA	Limite Operacional
Identificação Pessoa Natural	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Identificação Pessoa Jurídica	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Qualificação Pessoa Natural	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Qualificação Pessoa Jurídica	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Relacionamento Pessoa Natural	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Relacionamento Pessoa Jurídica	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Lista de cartões de crédito	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Lista de contas	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Identificação da conta	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Contrato	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Garantias do contrato	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Transações de cartão de crédito / histórico	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Transações de conta / histórico	Baixa frequência	4.000ms (4,0s)	Consulta 4x ao mês
Endpoints de Canal de Atendimento	Baixa frequência	4.000ms (4,0s)	Não se aplica
Endpoints produtos e serviços	Baixa frequência	4.000ms (4,0s)	Não se aplica

PAGINAÇÃO NO CONTEXTO DOS LIMITES OPERACIONAIS

Para a não contabilização de rechamadas, considerando que eventuais chamadas que possuam paginação devam ser interpretadas como uma única chamada, foi estabelecido a criação de um *query parameter* adicional para funcionar como identificador de rechamadas.

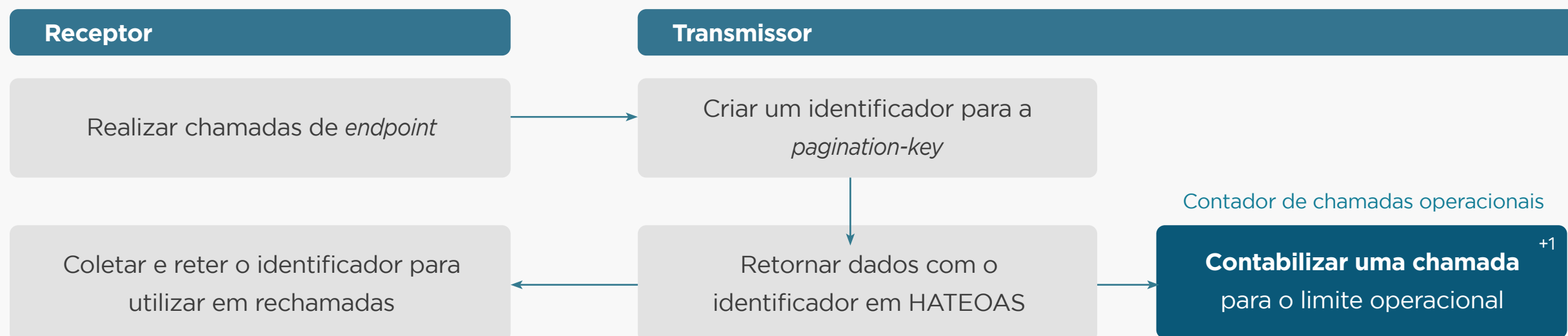
- Esse novo parâmetro deve ter o nome **pagination-key**.
- Cabe à Transmissora criar o identificador e enviá-lo via HATEOAS no retorno da chamada.
- O tempo máximo de utilização do identificador pelo Receptor é de 60 minutos.
- A Transmissora deve implementar validações para garantir a coerência da utilização do identificador, de acordo com as regras de limites operacionais.
- Caso a Receptora utilize um **pagination-key** inválido ou expirado, a Transmissora deverá gerar um novo **pagination-key** retornando o resultado com sucesso. Esta chamada será contabilizada para os limites operacionais.
- Essa implementação deve ser realizada em todos *endpoints* atuais com paginação.
- A Transmissora poderá gerar um único ID para controle das chamadas das próximas páginas.



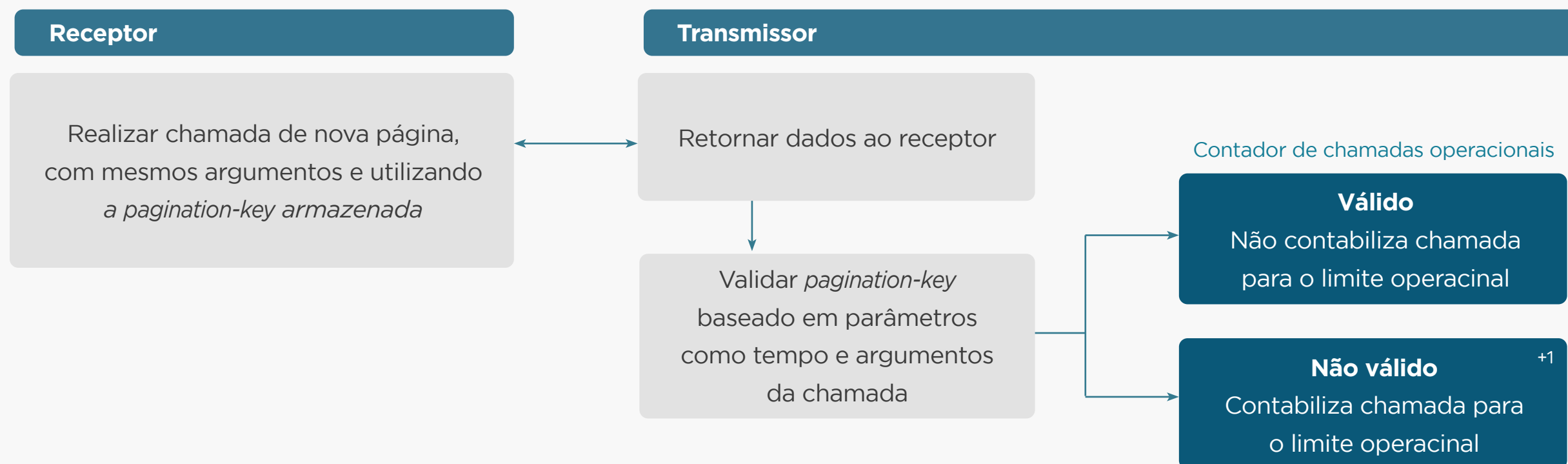


Exemplo de primeira chamada e chamada de paginação:

1ª Chamada



Chamadas de Paginação





Exemplo do uso do pagination-key:

Primeira chamada

GET /accounts/v2/accounts/12345678/transactions?page=1&page-size=1000

```
{
  "data" : [
    "..."
  ],
  "links" : {
    "self" : "https://api.banco.com.br/open-banking/accounts/v2/accounts/12345678/transactions?page=1&page-size=1000&pagination-key=123456",
    "first" : "https://api.banco.com.br/open-banking/accounts/v2/accounts/12345678/transactions?page=1&page-size=1000&pagination-key=123456",
    "next" : "https://api.banco.com.br/open-banking/accounts/v2/accounts/12345678/transactions?page=2&page-size=1000&pagination-key=123456"
  },
  "meta" : {
    "requestDateTime" : "2021-05-21T08.30.00Z"
  }
}
```



Exemplo do uso do pagination-key:

Próximas chamadas

GET /accounts/v2/accounts/12345678/transactions?page=2&page-size=1000&pagination-key=123456

```
{
  "data" . [
    "..."
  ],
  "links" . {
    "self" . "https://api.banco.com.br/open-banking/accounts/v2/accounts/12345678/transactions?page=1&page-size=1000&pagination-key=123456",
    "prev" . "https://api.banco.com.br/open-banking/accounts/v2/accounts/12345678/transactions?page=9&page-size=1000&pagination-key=123456",
    "first" . "https://api.banco.com.br/open-banking/accounts/v2/accounts/12345678/transactions?page=1&page-size=1000&pagination-key=123456",
    "next" . "https://api.banco.com.br/open-banking/accounts/v2/accounts/12345678/transactions?page=2&page-size=1000&pagination-key=123456"
  },
  "meta" . {
    "requestDateTime" . "2021-05-21T08.30.00Z"
  }
}
```

LIMITES DE TRÁFEGO

Quanto ao TPM (Transações por Minuto):

- **APIs Dados Abertos (Open Data)** - consumo público: Manter os limites por IP, estabelecendo um limite mínimo de 500 TPM.
- **APIs de Dados Cadastrais e Transacionais** – autenticada por cliente: Retirar contexto de chamadas por IP, considerando chamadas por *IF/organisationId* e por *endpoint*, de acordo com slide 15. De acordo com a segmentação que segue:

- Alta frequência: 2.000 TPM.
- Média frequência: 1.000 TPM.
- Baixa frequência: 500 TPM.

Nos casos abaixo **não são aplicáveis** limites de TPM:

- **APIs de segurança** (*Token* – Consumo Aauth 2.0 (FAPI), *Token* – DCR/DCM).
- **Endpoints da API de Consentimento** (*Consents*) (*GET*, *POST* e *DELETE*)

O erro “**HTTP Status Code 429 – Too many requests**” passa a ser utilizado de forma exclusiva para pedidos de consumo que excedam os limites de TPM.





LIMITES DE TRÁFEGO

Quanto ao limite de TPS:

Manter limites de 300 TPS (requisições/segundo globalmente), levando em consideração as características do **gatilho para ampliação do TPS**.

- Dado que os Receptores também devem se estruturar para garantir um consumo eficiente de dados (guardando dados históricos ao invés de buscá-los a cada consulta), caso o atingimento dos limites de um Transmissor seja concentrado em 1 único Receptor acima de 30% durante a quinzena, o gatilho para aumento de infraestrutura não precisa ser considerado e a Instituição Receptora deve ser avisada via *ticket* do *Service Desk*.
- **Fica definida a utilização do código HTTP Status Code 529** (*Site is overloaded*) para retorno quando atingido o **limite de TPS** estabelecido.
- Estas definições passarão por **revisão** para reavaliação dos limites estabelecidos **a cada 3 meses a partir da data entrada em produção**.



LIMITES DE TRÁFEGO

Gatilho para ampliação do TPS

- Caso uma Transmissora atinja o limite de 300 TPS, a mesma deve ampliar sua capacidade de infraestrutura para possibilitar um acréscimo de 150 TPS ao limite anterior. Tal aumento deve ocorrer novamente a cada vez que o limite vigente naquela Transmissora for atingido, até o teto de 900 TPS.
- Cada Instituição Transmissora precisará realizar este cálculo preventivamente.
- Caso a Instituição tenha aumentado a sua infraestrutura para disponibilizar um TPS maior do que os 300, mas tenha elasticidade para diminuir de acordo com a demanda, poderá fazer isso até o limite mínimo de 300 TPS.

- **Formato de cálculo para Monitoramento:**

- Limite máximo de 10% de atingimento de 90% do TPS estabelecido por quinzena, em 3 quinzenas consecutivas.
- Cálculo de TPS a cada 1 Segundo.
- Contador de segundos com TPS maior que 90% do regulamento atual (inicial 270 TPS).
- Caso ao final da quinzena o contador atinja um número maior que 10%, a quinzena deve ser contabilizada.
- Caso a Instituição contabilize 3 quinzenas consecutivas ultrapassando o limite percentual, o TPS deve ser acrescido em 150TPS ao limite anterior (máximo 900 TPS).
- Tempo para adequação deve ser de no máximo 2 meses após a identificação do gatilho.

CRIAÇÃO DE 2 NOVOS ENDPOINTS

Foram criados *endpoints* para compartilhamento de transações recentes das API's de Contas e Cartões de Crédito (Accounts e Credit-Cards-Accounts). Seguem as características referentes aos novos *endpoints*:

- O dicionário de dados dos novos *endpoints* seguem rigorosamente os mesmos campos dos *endpoints* '**GET /accounts/v1/accounts/{accountId}/transactions**' e '**GET /credit-cards-accounts/v1/accounts/{creditCardAccountId}/transactions**'.
- O nome dos novos *endpoints* será o mesmo nome dos atuais com a adição do sufixo *current*: **GET /accounts/v2/accounts/{accountId}/transactions-current** e '**GET /credit-cards-accounts/v2/accounts/{creditCardAccountId}/transactions-current**'.
- O período a ser considerado para apresentação de transações nestes novos *endpoints* será de até 7 dias (7 dias anteriores da consulta, incluindo o dia da consulta (D-6), de acordo com o campo de critério utilizado).
- Para os novos *endpoints* serão utilizados os mesmos critérios de consulta dos *endpoints* atuais.





QUANTO AO ERRO 422

Foi adicionado aos *Swaggers* a possibilidade de retornar o código **HTTP Status Code 422 - accepted**. Neste momento este código de retorno deve ser utilizado quando algum parâmetro da consulta for enviado semanticamente incorreto, ou seja, desrespeitando uma regra de negócio. Para os seguintes *endpoints*:

Cartões:

```
/accounts/{creditCardAccountId}/bills  
/accounts/{creditCardAccountId}/{billId}/transactions  
/accounts/{creditCardAccountId}/transactions  
/accounts/{creditCardAccountId}/transactions-current
```

Contas:

```
/accounts/{accountId}/transactions  
/accounts/{accountId}/transactions-current
```

Uso do código **HTTP Status Code 422 - accepted** para os demais *endpoints*, será objeto de estudo e publicação em momento oportuno.

CASOS DE ERRO

Foram mapeados alguns erros esperados durante o fluxo do usuário no ecossistema. Existem orientações específicas já tratadas no [Guia de Experiência do Usuário](#). Aqui, trataremos a visão mais técnica do que pode ser feito em caso de erro.

- Durante o redirecionamento do Consentimento, não se encontrou o *browser* no dispositivo do usuário: Além da orientação ao usuário, deve-se perceber que o fluxo de Consentimento continua pendente. Como a intenção de Consentimento segue válida por 60 minutos, deve-se tentar utilizar o mesmo Consentimento para tentar novamente a autorização do usuário.
- Durante o redirecionamento do Consentimento, houve um problema inesperado no servidor do Transmissor (HTTP Code 4xx ou 5xx): Além da orientação ao usuário, deve-se perceber que o fluxo de Consentimento continua pendente. Como a intenção de Consentimento segue válida por 60 minutos, deve-se tentar utilizar o mesmo Consentimento para tentar novamente a autorização do usuário.
- Durante o redirecionamento do Consentimento, houve um problema inesperado no servidor do Transmissor (Timeout, queda de Internet, etc.): Além da orientação do usuário, deve-se perceber que o fluxo de Consentimento continua pendente. Como a intenção de Consentimento segue válida por 60 minutos, deve-se tentar utilizar o mesmo consentimento para tentar novamente a autorização do usuário.



CASOS DE ERRO

- Durante o redirecionamento do Consentimento, houve um problema com o Login do usuário (erro de credenciais): Além da orientação ao usuário, deve-se perceber que o fluxo de Consentimento continua pendente. Como a intenção de Consentimento segue válida por 60 minutos, deve-se tentar utilizar o mesmo Consentimento para tentar novamente a autorização do usuário.
- Após o Consentimento do lado do Receptor, o usuário não finaliza o fluxo de compartilhamento deixando o fluxo pendente. Como a intenção de Consentimento segue válida por 60 minutos, deve-se tentar utilizar o mesmo Consentimento para tentar novamente a autorização do usuário.
- Após o Consentimento do lado do Receptor, o usuário rejeita o compartilhamento. O consentimento é alterado para o status **REJECTED** e não pode mais ser utilizado para compartilhamento de dados.
- Após o Consentimento completo, o app do Receptor não consegue acessar os dados por indisponibilidade: Deve-se tentar novamente. *Access Token* ainda é válido, portanto deve-se tentar novamente sem maiores prejuízos técnicos.
- Após o Consentimento completo, o app do Receptor não consegue acessar os dados por pendência de autorização de um terceiro (ex.: Dupla alçada): Deve-se consultar a API de Recursos (Resources) para verificar as pendências e saber como proceder.





05

API de Consentimento

(Consents)



CONSIDERAÇÕES GERAIS

API de Consentimento (Consents) trata **exclusivamente** dos consentimentos para **compartilhamento de dados entre as instituições** do Open Finance.

As informações da Instituição Receptora não trafegam na API de Consentimento (Consents) – a autenticação da Receptora se dá através do DCR, [conforme página DCR do Portal do Desenvolvedor](#).

API de Consentimento (Consents) está acessível através da credencial de acesso da Instituição Receptora (Client Credentials)

Após o POST de criação do consentimento, o status dele na resposta da Transmissora deverá ser **AWAITING_AUTHORISATION**.

O status do consentimento será alterado para **AUTHORISED** somente após autenticação e confirmação por parte do usuário requerente na Instituição Transmissora dos dados.





CONSIDERAÇÕES GERAIS

A descrição do fluxo de consentimento encontra-se disponível no Portal do desenvolvedor e na imagem conforme segue:

1. CONSENTIMENTO

Usuário inicia a jornada na Receptora fornecendo as informações do consentimento (i.e. CPF/CNPJ, agrupamento de dados, transmissora).

2. ENVIO DE INFORMAÇÕES

Receptora envia as informações do consentimento a Transmissora (i.e. CPF/CNPJ e agrupamento de dados).

POST/consents/v1/consents

3. CONSENTID

Transmissora responde com o consentId e status do consentimento como “aguardando autorização”.

4. REDIRECT FLOW

Receptora inicia o fluxo de redirecionamento pelo redirect flow ou decoupled flow (CIBA), redirecionando o usuário à Transmissora.

8. STATUS DO CONSENTIMENTO

Transmissora respondendo com o status do consentimento como “autorizado”.

7. CONSULTA DO CONSENTIMENTO

Receptora consulta o status do consentimento através do **GET/consents/v1/consents/{consentid}**

6. GERAÇÃO DO ACCESS TOKEN

Transmissora utiliza do authorization-code para gerar o Access Token para possibilitar o acesso aos dados pela Receptora.

5. AUTENTICAÇÃO E CONFIRMAÇÃO

Usuário se autentica, seleciona as origens e confirma as informações do consentimento na Transmissora.

O arquivo com o mapeamento completo entre *Roles*, *scopes* e *permissions* está disponibilizado no Portal do Desenvolvedor, no mesmo item acima - descrição do fluxo de consentimento.

 [Clique aqui para acessar o Portal do desenvolvedor.](#)



CONSIDERAÇÕES GERAIS

A Receptora deve enviar obrigatoriamente, no pedido de criação de consentimento, todas as *permissions* dos agrupamentos de dados as quais ela deseja consentimento, conforme tabela abaixo:

Categoria de Dados	Agrupamento	Permissions
Cadastro	Dados Cadastrais PF	CUSTOMERS_PERSONAL_IDENTIFICATIONS_READ RESOURCES_READ
Cadastro	Informações complementares PF	CUSTOMERS_PERSONAL_ADITTIONALINFO_READ RESOURCES_READ
Cadastro	Dados Cadastrais PJ	CUSTOMERS_BUSINESS_IDENTIFICATIONS_READ RESOURCES_READ
Cadastro	Informações complementares PJ	CUSTOMERS_BUSINESS_ADITTIONALINFO_READ RESOURCES_READ
Contas	Saldos	ACCOUNTS_READ ACCOUNTS_BALANCES_READ RESOURCES_READ
Contas	Limites	ACCOUNTS_READ ACCOUNTS_OVERDRAFT_LIMITS_READ RESOURCES_READ
Contas	Extratos	ACCOUNTS_READ ACCOUNTS_TRANSACTIONS_READ RESOURCES_READ



CONSIDERAÇÕES GERAIS

Categoria de Dados	Agrupamento	Permissions
Cartão de Crédito	Limites	CREDIT_CARDS_ACCOUNTS_READ CREDIT_CARDS_ACCOUNTS_LIMITS_READ RESOURCES_READ
Cartão de Crédito	Transações	CREDIT_CARDS_ACCOUNTS_READ CREDIT_CARDS_ACCOUNTS_TRANSACTIONS_READ
Cartão de Crédito	Faturas	CREDIT_CARDS_ACCOUNTS_READ CREDIT_CARDS_ACCOUNTS_BILLS_READ CREDIT_CARDS_ACCOUNTS_BILLS_TRANSACTIONS_READ RESOURCES_READ
Operações de Crédito	Dados do Contrato	LOANS_READ LOANS_WARRANTIES_READ LOANS_SCHEDULED_INSTALMENTS_READ LOANS_PAYMENTS_READ FINANCINGS_READ FINANCINGS_WARRANTIES_READ FINANCINGS_SCHEDULED_INSTALMENTS_READ FINANCINGS_PAYMENTS_READ UNARRANGED_ACCOUNTS_OVERDRAFT_READ UNARRANGED_ACCOUNTS_OVERDRAFT_WARRANTIES_READ UNARRANGED_ACCOUNTS_OVERDRAFT_SCHEDULED_INSTALMENTS_READ UNARRANGED_ACCOUNTS_OVERDRAFT_PAYMENTS_READ INVOICE_FINANCINGS_READ INVOICE_FINANCINGS_WARRANTIES_READ INVOICE_FINANCINGS_SCHEDULED_INSTALMENTS_READ INVOICE_FINANCINGS_PAYMENTS_READ RESOURCES_READ



CONSIDERAÇÕES GERAIS

Para Instituições Transmissoras: caso recebam uma solicitação de consentimento com permissões de dados cadastrais PF e PJ em conjunto, ela deve retornar um erro **HTTP Status Code 400 - Bad Request**. As solicitações de PF e PJ são mutuamente excludentes.

A Instituição Transmissora deve validar o preenchimento correto [desses agrupamentos](#) no momento da geração do consentimento.

Caso a Instituição Receptora envie permissões divergentes ao agrupamento especificado na tabela, a Transmissora deve rejeitar o pedido da Receptora dando retorno **HTTP Status Code 400 - Bad Request**.

A Transmissora deve retornar, da lista de *permissions* requisitadas, apenas o subconjunto de *permissions* por ela suportada, removendo da lista as *permissions* de produtos não suportados e retornando **HTTP Status Code 201 - Created**. Caso não existam permissões funcionais, a Instituição Transmissora deve retornar o erro **HTTP Status Code 422 - Unprocessable Entity**.





SOBRE A JORNADA DE CONSENTIMENTO

A Jornada de Consentimento é mais abrangente do que a implementação da API de Consentimento (Consents). Mais informações relativas a esta jornada estão presentes no Guia de Experiência do usuário do Open Finance Brasil e incluem passos obrigatórios a ser implementados pelas instituições participantes.

Abaixo você pode conferir parte das considerações quanto ao que precisa ser implementado:

- Para as API's de Contas e Cartão de Crédito (Accounts e Credit-Card-Accounts), ao menos um recurso de cada API deve ser compartilhado, caso exista.
- Para as API's de Empréstimo, Financiamentos, Adiantamento a Depositantes e Direitos Creditórios Descontados (Loans, Financings, Unarranged-Accounts-Overdraft e Invoice-Financings), ao menos uma das submodalidades (que pertença a qualquer uma das modalidades mencionadas) deve ser compartilhada.
- No caso das API's de Empréstimo, Financiamentos e Adiantamento a Depositantes (Loans, Financings e Unarranged-accounts-overdraft), se o cliente der o consentimento de um produto que ele ainda não tem contratado na Instituição Transmissora, no momento em que o cliente eventualmente realizar esta contratação, esta nova operação contratada deve ser compartilhada.

SOBRE A JORNADA DE CONSENTIMENTO

- Uma requisição de dados dos clientes não pode ser negada em função do fato de que este cliente não tenha contratado um dos produtos comercializados por esta transmissora, ou seja, a Transmissora deve responder com sucesso às requisições inerentes aos demais scopes concedidos pelo cliente com exceção do produto não contratado.
- O Transmissor deve somente listar as submodalidades que comercializa.
- Somente devem ser selecionados recursos ou submodalidades para as *permissions* solicitadas no consentimento.





REGRAS DE NEGÓCIOS PARA IMPLEMENTAÇÃO PELA INSTITUIÇÃO TRANSMISSORA

- **RN001:** na resposta do POST para criação de consentimento, a Transmissora deve obrigatoriamente retornar, da lista de *permissions* requisitadas, o subconjunto de *permissions* por ela suportada.
- **RN002:** caso a Transmissora não suporte nenhuma das *permissions* solicitadas deve retornar o erro **HTTP Status Code 422 - Unprocessable Entity**.
- **RN003:** caso a Transmissora ainda esteja preparando a listagem dos recursos autorizados da API de Recursos (Resources), deve ser retornado o código **HTTP Status Code 202 - accepted** com o *body* vazio e a Receptora deverá seguir as recomendações de *polling*.
- **RN004:** o prazo de validade de um consentimento é de no máximo 12 meses a partir da data de sua criação.
- **RN005:** caso seja enviado o campo `businessEntity` na criação do consentimento, as permissões solicitadas para as APIs de cadastro devem então ser de PJ. Se as permissões enviadas forem de PF a Transmissora deve rejeitar o pedido com erro **HTTP Status Code 400 - Bad Request**.

REGRAS PARA O PREENCHIMENTO DO CAMPO STATUS DO RECURSO E A TRANSIÇÃO ENTRE STATUS

- **AWAITING_AUTHORISATION:** é o status inicial que deve ser devolvido após a criação do consentimento. Este status deve ser retornado por até 60 minutos, após este tempo, caso o cliente não realize a confirmação do consentimento na Transmissora, o status deve ser automaticamente alterado para **REJECTED**.
- **AUTHORISED:** é o status que permite o consumo do consentimento por parte da Receptora. Pode ser revogado pelo cliente na Transmissora ou na Receptora e expira automaticamente ao término do prazo de validade do consentimento; em qualquer um desses dois casos o status é alterado pela **REJECTED**.
- **REJECTED:** é o status retornado quando o consentimento não está disponível, seja por ação do usuário final ou por expiração do consentimento por prazo. É estado final, não pode ser alterado para outro status.





DETALHAMENTO DO MOTIVO DE REJEIÇÃO DO CONSENTIMENTO - OBJETO REJECTION

Foi incluído na Versão 2.0 da API de Consentimento (Consents) um objeto **Rejection** no retorno do *endpoint* **GET/consents/{consentId}** a ser retornado em caso de consentimento com status **REJECTED** com as informações abaixo listadas:

- **RejectedBy:** define qual das partes envolvidas foi responsável pela rejeição (usuário, Instituição Receptora ou Transmissora)
 - Enum (**USER** - usuário, **ASPSP** - Instituição Transmissora, **TPP** - Instituição Receptora).
- **Reason:** define a razão pela rejeição sendo dividido em:
 - code: Enum que define o código da razão pela qual o consentimento foi rejeitado.
 - **CONSENT_EXPIRED** - consentimento que ultrapassou o tempo limite para autorização.
 - **CUSTOMER_MANUALLY_REJECTED** - cliente efetuou a rejeição do consentimento manualmente através de interação nas instituições participantes.



DETALHAMENTO DO MOTIVO DE REJEIÇÃO DO CONSENTIMENTO - OBJETO REJECTION

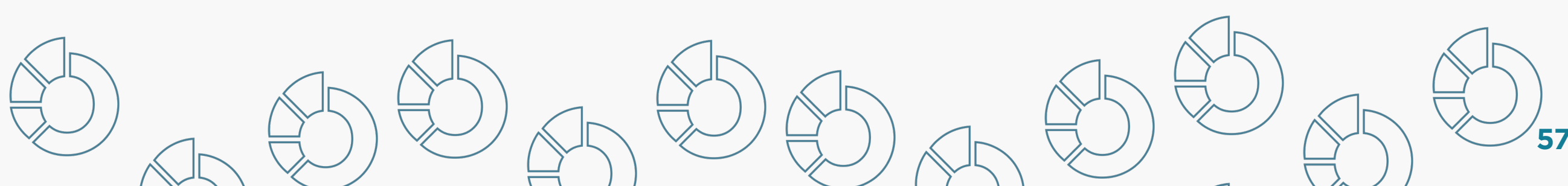
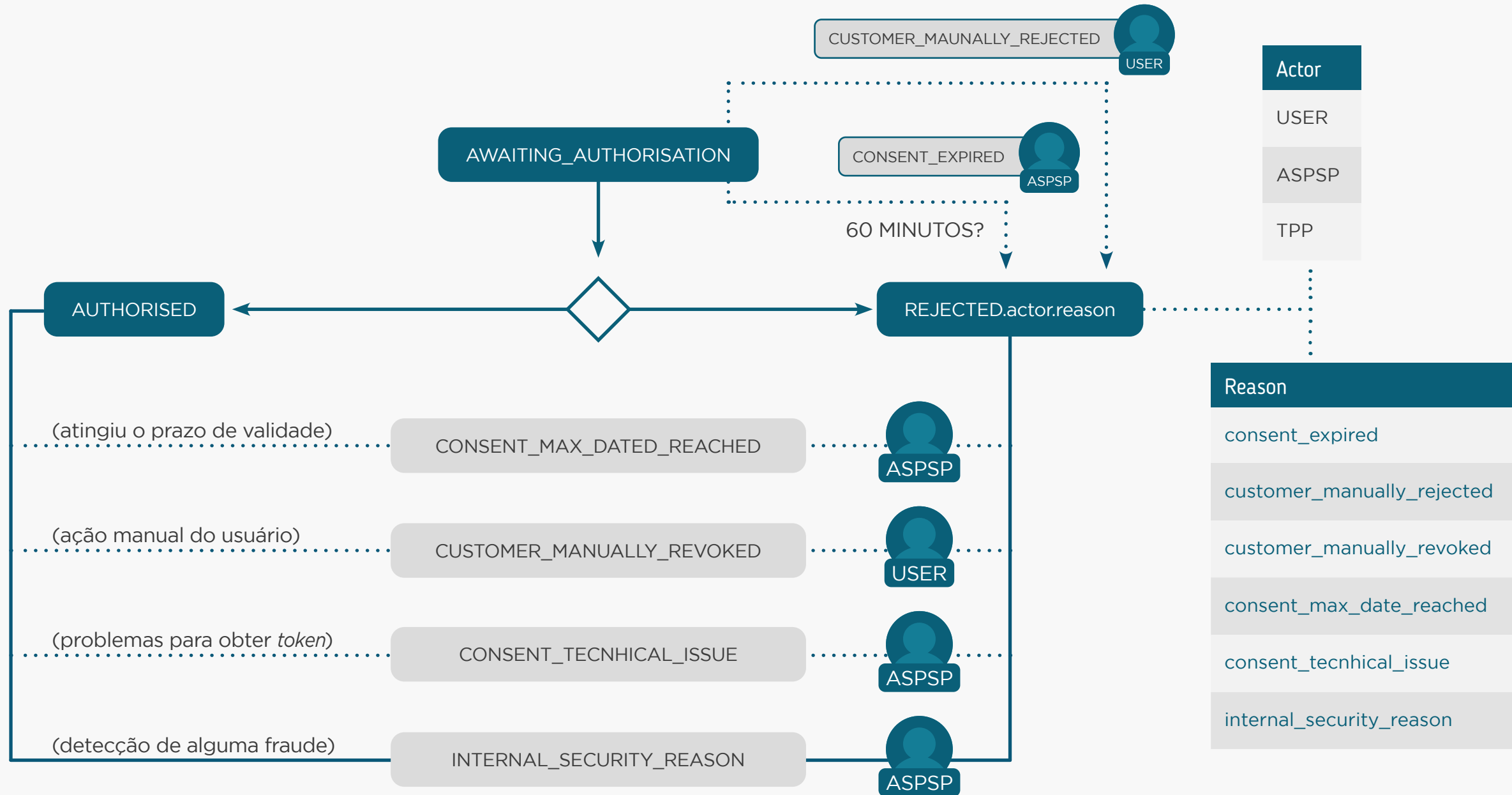
- **CUSTOMER_MANUALLY_REVOKED** - cliente efetuou a revogação após a autorização do consentimento.
 - **CONSENT_MAX_DATE_REACHED** - consentimento que ultrapassou o tempo limite de compartilhamento.
 - **CONSENT_TECHNICAL_ISSUE** - consentimento que foi rejeitado devido a um problema técnico que impossibilita seu uso pela Instituição Receptora, por exemplo: falha associada a troca do *AuthCode* pelo *AccessToken*, durante o processo de *Hybrid Flow*.
 - **INTERNAL_SECURITY_REASON** - Consentimento que foi rejeitado devido as políticas de segurança aplicada pela Instituição Transmissora.
-
- **AdditionalInformation:** Contém informações adicionais a critério da Transmissora.





DETALHAMENTO DO MOTIVO DE REJEIÇÃO DO CONSENTIMENTO - OBJETO REJECTION

Abaixo segue diagrama com o comportamento esperado para o objeto **Rejection**:





DETALHAMENTO DO MOTIVO DE REJEIÇÃO DO CONSENTIMENTO - OBJETO REJECTION

Cenários de Uso	Condição	Status	Reason	Responsável
1. Cliente efetua consentimento, mas jornada é interrompida	> 60 minutos	REJECTED	CONSENT_EXPIRED	ASP
2. Cliente efetua consentimento e no momento de confirmar no transmissor cancela a jornada	action button canceled	REJECTED	CUSTOMER_MANUALLY_REJECTED	USER
3. Cliente efetua consentimento e o prazo de validade expira	dateTime atingindo	REJECTED	CONSENT_MAX_DATE_REACHED	ASP
4. Cliente possui consentimento ativo e decide efetuar a revogação no transmissor	delete	REJECTED	CUSTOMER_MANUALLY_REVOKED	USER
5. Cliente possui consentimento ativo e decide efetuar a revogação no receptor	delete	REJECTED	CUSTOMER_MANUALLY_REVOKED	USER
6. Cliente efetua consentimento e no momento de retorno o receptor (confirmação) ocorre falha para obter o access <i>token</i> . Demais cenários técnicos de falha também poderão ser categorizados neste enum	token failed	REJECTED	CONSENT_TECHNICAL_ISSUE	ASP
7. Cliente efetua consentimento, porém ao chegar no transmissor é identificado algum dado que indique ação fraudulenta	fraud	REJECTED	INTERNAL_SECURITY_REASON	ASP



CONVIVÊNCIA DOS CONSENTIMENTOS NAS VERSÕES V1 E V2 DA API CONSENTS

Consentimentos REVOGADOS com a API CONSENTS (v2.0.0) devem ter o objeto Rejection.

- Quando consultados com a versão anterior da API CONSENTS(v1.0.3), não terão o objeto Rejection.
- Quando consultados com a API CONSENTS (v2.0.0), terão o objeto Rejection.

Consentimentos REVOGADOS com a versão anterior da API CONSENTS (v1.0.3) não devem ter o objeto Rejection.

- Quando consultados com a versão anterior da API CONSENTS (v1.0.3), não terão o objeto Rejection.
- Quando consultados com a API CONSENTS(v2.0.0), não terão o objeto Rejection.

Após o lançamento produtivo da versão 2, consentimentos que forem REVOGADOS por canais ou processos internos da Transmissora deverão obrigatoriamente ter os dados de rejeição preenchidos incluído a mudança de status do consentimento.



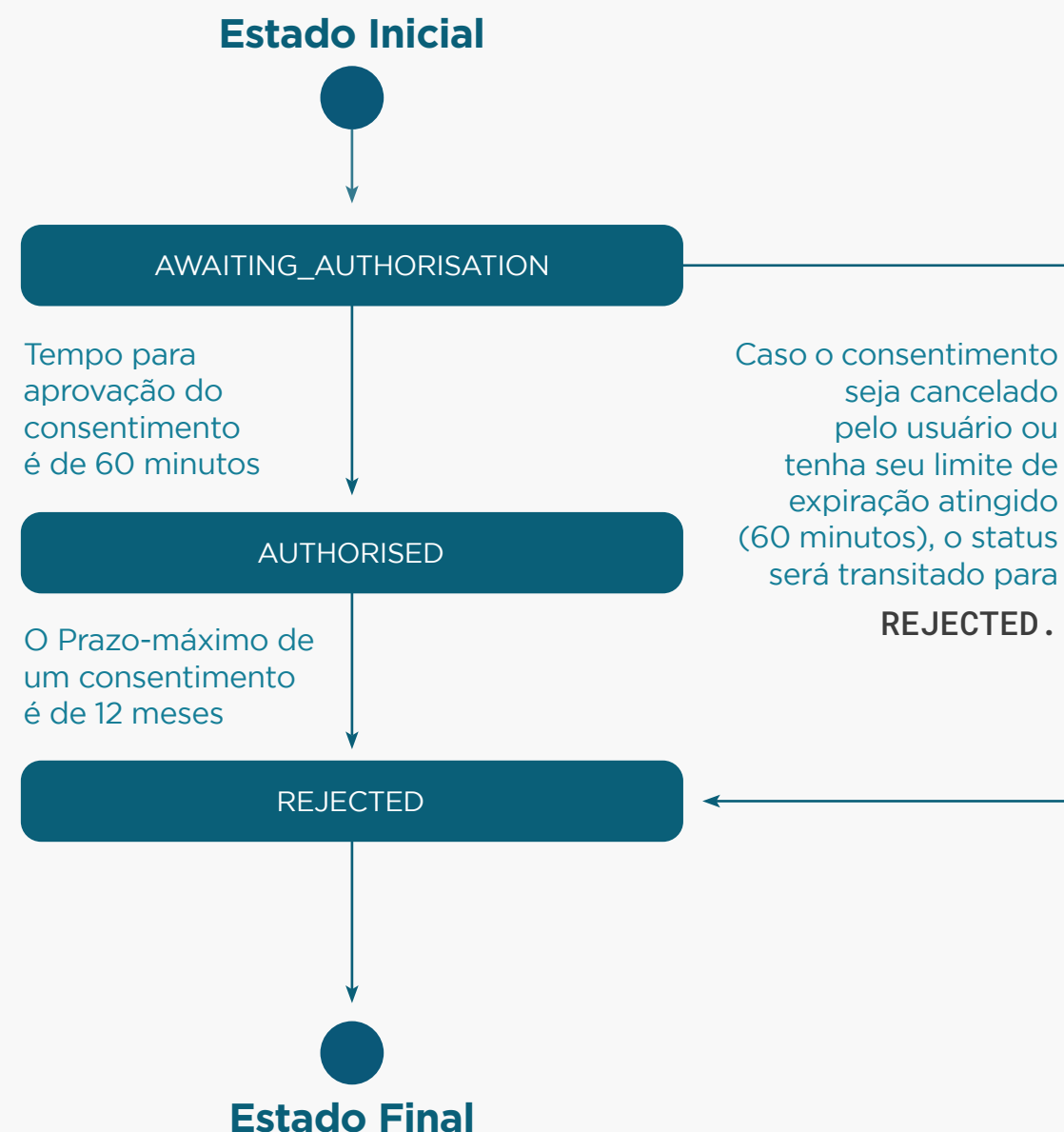
TRANSIÇÃO DE STATUS

Descrição das transições possíveis a partir de cada status

AWAITING_AUTHORISATION: é um status apenas inicial e não deve ser alcançado a partir dos demais status. Pode ser alterado para **AUTHORISED** ou **REJECTED**.

AUTHORISED: pode ser alterado para **REJECTED**.

REJECTED: é o estado final, não pode ser alterado para outro status.



A rejeição pode ser motivada pelos seguintes atores/participantes do ecossistema:

1. Receptora de dados (por ela própria ou pelo usuário);
2. Transmissora de dados;
3. Sistema (expiração).



TRATAMENTO PARA USUÁRIO FINAL QUE NÃO É CLIENTE DA INSTITUIÇÃO TRANSMISSORA

No POST de criação do **consentimento** para compartilhamento de dados, a Transmissora não deve retornar de forma explícita se este usuário final (PF ou PJ) é ou não cliente da instituição. Caso a **Transmissora** receba uma solicitação de intenção de consentimento de um usuário final que **não seja seu cliente**, a resposta **deve ser de sucesso**, como se fosse cliente. Nesse caso, o consentimento iria expirar automaticamente após o tempo máximo de 60 minutos, visto que este usuário final não tem cadastro e não conseguiria se autenticar. Este comportamento de não retornar à informação de que o usuário final não é cliente da instituição visa inibir eventuais usos inadequados do serviço de compartilhamento de dados.

O QUE FOI ALTERADO:

A versão mais recente das mudanças pode ser encontrada em:
[Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)](#)



06

API de Recursos

(Resources)



CONDIÇÕES

GERAIS



- **A API de Recursos (Resources)** está somente disponível para consentimentos que se encontram no status **AUTHORISED**. A Transmissora não pode reduzir o escopo do consentimento, exceto nos casos em que não ofereça produtos daquele agrupamento (*permission*).
- **A API de Recursos (Resources)** retorna uma lista com os recursos vinculados a um consentimento em específico, identificado pelo **consentId** que se encontra vinculado ao *token* enviado no *header* **Authorization**.
- A *permission* específica para consulta à API Resources (RESOURCES_READ) deve ser solicitada pela Instituição Receptora na ocasião do pedido de criação do consentimento.



CASOS DE USO E O SEU RESPECTIVO STATUS A SER INFORMADO PELA API DE RECURSOS (RESOURCES):

Os casos abaixo se aplicam a recursos vinculados a um consentimento autorizado:

- Recursos que se encontram em **plena utilização** e disponíveis nos canais de atendimento eletrônico para os usuários finais devem ter seus status como `AVAILABLE`;
- Recursos que se encontram em **bloqueios temporários**, quando o cliente não possui acesso ao recurso por meio dos canais de atendimento eletrônico, devem retornar o status `TEMPORARILY_UNAVAILABLE`;
- Recursos que se encontram em **bloqueios definitivos**, quando o cliente não possui acesso ao recurso por meio dos canais de atendimento eletrônico, devem retornar o status `UNAVAILABLE`;
- Recursos que se encontram **encerrados**, quando o cliente não possui acesso ao recurso por meio dos canais de atendimento eletrônico, devem retornar o status `UNAVAILABLE`;
- Recursos que foram **migrados** e se encontram **indisponíveis** nos canais de atendimento eletrônico devem retornar o status `UNAVAILABLE`.
- Casos de uso que um cliente final efetue um consentimento para mais de um recurso e um deles exija a aprovação de **múltiplas alçadas** (`PENDING_AUTHORISATION`) e o outro esteja **disponível para consulta** (`AVAILABLE`), o comportamento esperado é que cada recurso tenha seu status representado de forma independente, disponibilizando imediatamente as contas já aprovadas.

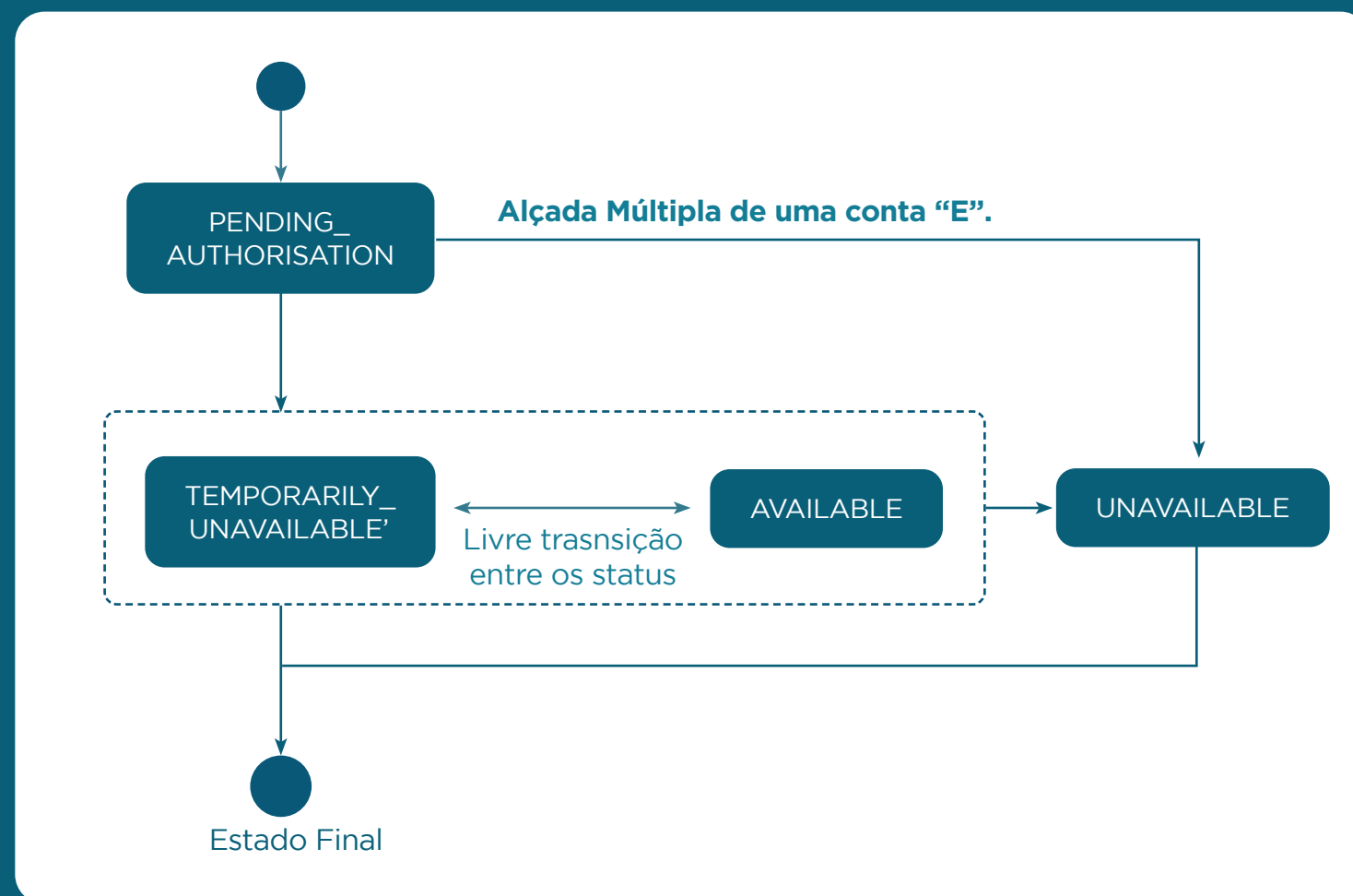


TRANSIÇÃO DE STATUS DA API DE RECURSOS (RESOURCES)

Descrição das transições possíveis a partir de cada status:

- **PENDING_AUTHORISATION** é um status apenas inicial e não deve ser alcançado a partir dos demais status.
- **PENDING_AUTHORISATION** pode ser transicionada para **AVAILABLE**, **TEMPORARILY_UNAVAILABLE** e **UNAVAILABLE**
- **UNAVAILABLE** é um status final e não permite mais transições aos demais status.
- **AVAILABLE** pode ser transicionada para **TEMPORARILY_UNAVAILABLE** e **UNAVAILABLE**
- **TEMPORARILY_UNAVAILABLE** pode ser transicionada para **AVAILABLE** e **UNAVAILABLE**.

Máquina de Estados para transição de status da API de Recursos:



RECOMENDAÇÃO USO DE POLLING

É recomendado que a Instituição Receptora implemente um *retry exponencial* (o tempo de espera entre a última chamada e a próxima chamada da API deve crescer exponencialmente), de forma a não sobrecarregar a API de Recursos (Resources).

Em casos em que seja retornado o código de erro **HTTP Status Code 500 - internal server error** de forma persistente, recomenda-se que a Instituição Receptora consulte as API's comuns para verificar a disponibilidade da API requisitada antes de realizar uma nova tentativa de consulta.

 [Consultar API's comuns](#)





COMPORTAMENTO DA API DE CONTAS COM RELAÇÃO A RESOURCES

Visão da API de Recursos (Resources) para o Tipo Accounts (Contas)

- O identificador único `resourceId` de cada recurso do tipo - ACCOUNT - presente na lista devolvida pela API, corresponde ao identificador único `accountId` utilizado pela **API de Contas (Accounts)**. Por exemplo:

Retorno da API de Recursos (Resource):

```
{
  "data": [
    {
      "resourceId": "12345678990",
      "type": "ACCOUNT",
      "status": "AVAILABLE"
    },
  ],
  ...}

```

Exemplo de utilização do `resourceId` em uma chamada a API de Contas (Accounts): `accounts/v1/accounts/12345678990`

REGRAS DE NEGÓCIO PARA IMPLEMENTAÇÃO PELA INSTITUIÇÃO TRANSMISSORA

- **RN001:** É obrigatório que todos os recursos do tipo - ACCOUNT -compartilhados retornem o identificador único independente do status a ser informado.
- **RN002:** É obrigatório o retorno de todas as contas consentidas pelo cliente independente de seus status (contas não mais acessíveis pelo cliente, mas presente no consentimento dos clientes deve ser informadas com o status **UNAVAILABLE**)
- **RN003:** Caso a Transmissora ainda esteja preparando a listagem dos recursos autorizados da API de Recursos (Resources), deve ser retornado o código **HTTP Status Code 202 - accepted** com o *body* vazio e a Receptora deverá seguir as recomendações de *polling*
- **RN004:** Recursos que estejam em processo de múltiplas alçadas para aprovação DEVEM retornar o status **PENDING_AUTHORISATION**.
- **RN005:** O método de listagem de contas da API de Contas (Accounts) deve retornar apenas a lista de contas que estejam com o status da Resources como **AVAILABLE**
- **RN006:** Cada recurso compartilhado por meio do mesmo consentimento deve ter seus status tratados de forma independente.





Regras para o preenchimento do campo status do Recurso (PENDING_AUTHORISATION, AVAILABLE, UNAVAILABLE, TEMPORARILY_UNAVAILABLE) e a transição entre status

- **PENDING_AUTHORISATION:** indica a existência de pendências para o compartilhamento do recurso, em caso de múltiplas alçadas, quando é necessário o consentimento de mais de um titular. Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** na API de Contas (Accounts) quando utilizado o `accountId` que está neste status.
- **AVAILABLE:** indica que o recurso encontra-se disponível para consulta na API de Contas (Accounts), através do identificador – `accountId` – compartilhado.
- **UNAVAILABLE:** indica que o recurso encontra-se indisponível para consulta na API de Contas (Accounts). Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** nas API's de Contas (Accounts) quando utilizado o `accountId` que está neste status.
- **TEMPORARILY_UNAVAILABLE:** indica que o recurso encontra-se temporariamente indisponível para consulta na API de Contas (Accounts). Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** nas API's de Contas (Accounts) quando utilizado o `accountId` que está neste status(`TEMPORARILY_UNAVAILABLE`).



CÓDIGOS DE USO PARA O ERRO 403 NA API DE CONTAS (ACCOUNTS)

A Transmissora deve preencher do retorno do código **HTTP Status Code 403 - Forbidden**, conforme segue:

- **PENDING_AUTHORISATION:**

```
code='status_RESOURCE_PENDING_AUTHORISATION',  
title='Aguardando autorização de múltiplas alçadas',  
detail=(uso a critério do Transmissor)
```

- **UNAVAILABLE:**

```
code='status_RESOURCE_UNAVAILABLE', title='Recurso  
indisponível',detail=(uso a critério do transmissor)  
TEMPORARILY_UNAVAILABLE. code= '
```

- **TEMPORARILY_UNAVAILABLE:**

```
code=' status_RESOURCE_TEMPORARILY_  
UNAVAILABLE', title='Recurso temporariamente  
indisponível',detail=(uso a critério do Transmissor)
```





REGRAS DE NEGÓCIOS PARA IMPLEMENTAÇÃO DE CONSUMO PELA INSTITUIÇÃO RECEPTORA

Interpretação do status do Recurso quando chamar a API de Contas (Accounts):

- **PENDING_AUTHORISATION:** Neste status a recomendação é que a Receptora aguarde a alteração desse status para um dos demais status executando *polling* na API de Recursos (Resources) (Vide recomendações de *polling*).
- **AVAILABLE:** Indica que o recurso Conta encontra-se disponível para consulta na API de Contas (Accounts) através do identificador `accountId` compartilhado.
- **UNAVAILABLE:** Neste status a recomendação é que a Receptora não consulte a API de Contas (Accounts) em função do dado não estar disponível para esse `accountId`.
- **TEMPORARILY_UNAVAILABLE:** Neste status a recomendação é que a Receptora aguarde a alteração desse status para um dos demais status executando *polling* na API de Recursos (Resources) (Vide recomendações de *polling*).

REGRAS QUANTO A BUSCA DE ATUALIZAÇÕES

- Recomenda-se que a iniciadora consulte a API de Recursos (Resources) logo após identificar que o consentimento se encontra autorizado de modo a obter todas as contas consentidas pelo cliente e seus respectivo identificadores.
- Recomenda-se que a iniciadora caso obtenha um **HTTP Status Code 403 - Forbidden** na API de Contas (Accounts) avalie o JSON retornado no *body* da requisição para identificar as informações adicionais referentes ao erro, como o status específico do *Resources* que impede o consumo da informação.
- Recomenda-se que a Iniciadora consulte a API de Recursos (Resources) caso obtenha um erro na API de Contas (Accounts) de modo a identificar uma possível alteração de status do recurso.





TABELA DESCRITIVA DAS POSSIBILIDADES DE INTERAÇÃO ENTRE API DE RECURSOS (RESOURCES) E API DE CONTAS (ACCOUNTS)

Cenário	Status Conta	API Resources	API Listagem Contas	APIs Dados Contas	Code
Sem consentimento	-	401 unauthorized	401 unauthorized	401 unauthorized	
Com consentimento não autorizado	-	401 unauthorized	401 unauthorized	401 unauthorized	
Com consentimento autorizado (pendente múltipla alçada)	-	200 - Retorna conta com status PENDING_AUTHORISATION	200 Não retorna conta	403 forbidden	STATUS_RESOURCE_PENDING_AUTHORISATION
Com consentimento autorizado (aprovado múltipla alçada)	OK	200 - Retorna conta com status AVAILABLE	200 retorna a conta	200 retorna dados	
Com consentimento autorizado (aprovado múltipla alçada)	Bloqueio	200 - Retorna conta com status TEMPORARY_UNAVAILABLE	200 Não retorna conta	403 forbidden	STATUS_RESOURCE_TEMPORARY_UNAVAILABLE
Com consentimento autorizado (aprovado múltipla alçada)	Encerrada/ Migrada	200 - Retorna conta com status UNAVAILABLE	200 Não retorna conta	403 forbidden	STATUS_RESOURCE_UNAVAILABLE
Com consentimento autorizado (recusado múltipla alçada)	-	200 - Retorna conta com status UNAVAILABLE	200 Não retorna conta	403 forbidden	STATUS_RESOURCE_UNAVAILABLE
Com consentimento autorizado (conta inexistente)	-	200 - Não retorna na lista	200 Não retorna conta	403 forbidden	
Com consentimento revogado ou expirado*		401 unauthorized	401 unauthorized	401 unauthorized	

* Para os cenários de consentimento revogado ou expirado, conforme determinado na especificação de segurança o access token associado ao mesmo é invalidado impossibilitando a consulta as APIs produtos por parte da Receptora.



COMPORTAMENTO DA API DE DADOS CADASTRAIS COM RELAÇÃO A RESOURCES

Regras de negócio para implementação pela transmissora

- **RN001** - Dados Cadastrais não geram recursos listáveis na API de Recursos (Resources). Eventuais fluxos de múltiplas alçadas terão os seus status reportados diretamente ao consultar os dados da API de Dados Cadastrais (Customers).
- **RN002** - Ao consultar a API de Recursos (Resources) em que, no consentimento, se tenham somente permissões de compartilhamento de Dados Cadastrais, deve ser retornado o código 200 com lista vazia da API de Recursos (Resources) (esta RN traz uma mudança com relação ao que estava estabelecido no problema conhecido BCLOG-F02-177).
- **RN003** - Caso a Transmissora ainda esteja preparando a listagem dos recursos autorizados da API de Recursos (Resources), deve ser retornado o código **HTTP Status Code 202 - accepted** com o *body* vazio e a Receptora deverá seguir as recomendações de *polling*.
- **RN004** - Caso a Transmissora não possua cadastro unificado das empresas de sua respectiva marca, ela poderá entregar mais de um registro no *endpoint* de Identificação PN e PJ, devendo esta indicar, no atributo “**companyCnpj**”, a referência do CNPJ das empresas que fazem uso do respectivo cadastro.

6 API de Recursos (Resources)

Considerando o cenário abaixo: Marca 1, com as empresas CNPJ 1, CNPJ 2 e CNPJ 3.
Cadastro Não Unificado

```
{
  "data": [
    {
      "personalId": "teste1",
      "civilName": "Pessoa da Silva",

      "companyCnpj": [
        "CNPJ 1"
      ],

      ... omitido demais campos para exemplificar ...
    },
    {
      "personalId": "teste1",
      "civilName": "Pessoa da Silva",

      "companyCnpj": [
        "CNPJ 2",
        "CNPJ 3"
      ],

      ... omitido demais campos para exemplificar ...
    }
  ],
  "links": {
    "self": "https://api.banco.com.br/open-banking/api/v1/resource",
    "first": "https://api.banco.com.br/open-banking/api/v1/resource",
    "prev": "https://api.banco.com.br/open-banking/api/v1/resource",
    "next": "https://api.banco.com.br/open-banking/api/v1/resource",
    "last": "https://api.banco.com.br/open-banking/api/v1/resource"
  },
  "meta": {
    "totalRecords": 1,
    "totalPages": 1,
    "requestDateTime": "2021-05-21T08.30.00Z"
  }
}
```





6 API de Recursos (Resources)

- **RN005** - Caso a Transmissora possua cadastro unificado das empresas de sua respectiva marca, ela poderá entregar apenas um registro na lista do *endpoint* de Identificação PN e PJ, devendo esta indicar, no atributo “**companyCnpj**”, a referência do CNPJ das empresas que fazem uso do respectivo cadastro.

Considerando o cenário abaixo: Marca 1, com as empresas CNPJ 1, CNPJ 2 e CNPJ 3. Cenário: Cadastro Unificado

```
{
  "data": [
    {
      "personalId": "teste1",
      "civilName": "Pessoa da Silva",

      "companyCnpj": [
        "CNPJ 1",
        "CNPJ 2",
        "CNPJ 3"
      ],
      ... omitido demais campos para exemplificar ...
    }
  ],
  "links": {
    "self": "https://api.banco.com.br/open-banking/api/v1/resource",
    "first": "https://api.banco.com.br/open-banking/api/v1/resource",
    "prev": "https://api.banco.com.br/open-banking/api/v1/resource",
    "next": "https://api.banco.com.br/open-banking/api/v1/resource",
    "last": "https://api.banco.com.br/open-banking/api/v1/resource"
  },
  "meta": {
    "totalRecords": 1,
    "totalPages": 1,
    "requestDateTime": "2021-05-21T08.30.00Z"
  }
}
```

- **RN006** - Se, a critério do transmissor, for aplicado fluxo de múltipla alçada para acesso aos Dados Cadastrais, deverá ser aplicado os status apresentado a seguir.



CÓDIGOS DE USO PARA O ERRO 403 NA API DE DADOS CADASTRAIS (CUSTOMERS)

A Transmissora deve preencher do retorno do código **HTTP Status Code 403 - Forbidden**, conforme segue:

- **PENDING_AUTHORISATION:** `code='status_RESOURCE_PENDING_AUTHORISATION', title='Aguardando autorização de múltiplas alçadas', detail=(uso a critério do Transmissor)`
- **UNAVAILABLE:** `code='status_RESOURCE_UNAVAILABLE', title='Recurso indisponível', detail=(uso a critério do transmissor)`
- **TEMPORARILY_UNAVAILABLE:** `code=' status_RESOURCE_TEMPORARILY_UNAVAILABLE', title='Recurso temporariamente indisponível', detail=(uso a critério do transmissor)`





COMPORTAMENTO DA API DE CARTÕES COM RELAÇÃO A RESOURCES

Visão da API de Recursos (Resources) para o Tipo de Cartão de Crédito (Credit-Cards-Accounts)

- O identificador único `resourceId` de cada recurso do tipo `CREDIT_CARD_ACCOUNT` presente na lista devolvida pela API, corresponde ao identificador único `creditCardAccountId` utilizado pela API de Cartão de Crédito (Credit-Cards-Accounts). Por exemplo:

Retorno da API de Recursos (Resources):

```
{
  "data": [
    {
      "resourceId": "12345678990",
      "type": " CREDIT_CARD_ACCOUNT ",
      "status": "AVAILABLE"
    },
  ],
  ...}
```

Exemplo de utilização do `resourceId` em uma chamada a API de Cartão de Crédito (Credit-Cards-Accounts):
`credit-cards-accounts/v1/accounts/12345678990`

- Cartões (ex. virtual / adicional / físico) bloqueados temporariamente não impactam o status da conta cartão associada ao consentimento.



REGRAS DE NEGÓCIO DE IMPLEMENTAÇÃO PELA INSTITUIÇÃO TRANSMISSORA

Regras para o preenchimento do campo status do Recurso (PENDING_AUTHORISATION, AVAILABLE, UNAVAILABLE, TEMPORARILY_UNAVAILABLE) e a transição entre status.

- **PENDING_AUTHORISATION:** indica a existência de pendências para o compartilhamento do recurso, em caso de múltiplas alçadas, quando é necessário o consentimento de mais de um titular. Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** na API de Cartão de Crédito (Credit-Cards-Accounts) quando utilizado o `creditCardAccountId` que está neste status.
- **AVAILABLE:** indica que o recurso `CREDIT_CARD_ACCOUNT` encontra-se disponível para consulta na API de Cartão de Crédito (Credit-Cards-Accounts) através do identificador `creditCardAccountId` compartilhado.
- **UNAVAILABLE:** indica que o recurso `CREDIT_CARD_ACCOUNT` encontra-se indisponível para consulta na API de Cartão de Crédito (Credit-Cards-Accounts). Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** na API de Cartão de Crédito (Credit-Cards-Accounts) quando utilizado o `creditCardAccountId` que está neste status.
- **TEMPORARILY_UNAVAILABLE:** indica que o recurso `CREDIT_CARD_ACCOUNT` encontra-se temporariamente indisponível para consulta na API de Cartão de Crédito (Credit-Cards-Accounts). Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** na API de Cartão de Crédito (Credit-Cards-Accounts) quando utilizado o `creditCardAccountId` que está neste status.



CÓDIGOS DE USO PARA O ERRO 403 NA API DE CARTÃO DE CRÉDITO (CREDIT-CARDS-ACCOUNTS):

A Transmissora deve preencher do retorno do código **HTTP Status Code 403 - Forbidden**, conforme segue:

- **PENDING_AUTHORISATION:**

```
code='status_RESOURCE_PENDING_AUTHORISATION', title='Aguardando  
autorização de múltiplas alçadas', detail=(uso a critério do transmissor)
```

- **UNAVAILABLE:**

```
code='status_RESOURCE_UNAVAILABLE', title='Recurso  
indisponível', detail=(uso a critério do transmissor)
```

- **TEMPORARILY_UNAVAILABLE:**

```
code='status_RESOURCE_TEMPORARILY_UNAVAILABLE', title='Recurso  
temporariamente indisponível', detail=(uso a critério do transmissor)
```





CÓDIGOS DE USO PARA O ERRO 403 NA API DE CARTÃO DE CRÉDITO (CREDIT-CARDS-ACCOUNTS):

Interpretação do status do Recurso quando chamar a API de Cartão de Crédito (Credit-Cards-Accounts):

- **PENDING_AUTHORISATION:** neste status a recomendação é que a Receptora aguarde a alteração desse status para um dos demais status executando *polling* na API de Recursos (Resources) (Vide recomendações de *polling*).
- **AVAILABLE:** indica que o recurso `CREDIT_CARD_ACCOUNT` encontra-se disponível para consulta na API de Cartão de Crédito (Credit-Cards-Accounts) através do identificador `creditCardAccountId` compartilhado.
- **UNAVAILABLE:** neste status a recomendação é que a Receptora não consulte a API de Cartão de Crédito (Credit-Cards-Accounts) em função do dado não estar disponível para esse `creditCardAccountId`.
- **TEMPORARILY_UNAVAILABLE:** neste status a recomendação é que a Receptora aguarde a alteração desse status para um dos demais status executando *polling* na API de Recursos (Resources) (Vide recomendações de *polling*).

REGRAS QUANTO A BUSCA DE ATUALIZAÇÕES:

- Recomenda-se que a Receptora consulte a API de Recursos (Resources) logo após identificar que o consentimento se encontra autorizado de modo a obter todas as **CREDIT_CARD_ACCOUNT** consentidas pelo cliente e seus respectivo identificadores.
- Recomenda-se que a Receptora caso obtenha um erro **HTTP Status Code 403 - Forbidden** na API de Cartão de Crédito (Credit-Cards-Accounts) avalie o JSON retornado no *body* da requisição para identificar as informações adicionais referentes ao erro, como o status específico do Resources que impede o consumo da informação.
- Recomenda-se que a Receptora consulte a API de Recursos (Resources) caso obtenha um erro na de API de Cartão de Crédito (Credit-Cards-Accounts) de modo a identificar uma possível alteração de status do recurso.





TABELA DESCRITIVA DAS POSSIBILIDADES DE INTERAÇÃO ENTRE API RESOURCES E API CREDIT CARDS ACCOUNTS (ANÁLOGA AO CASO DE ACCOUNTS):

Cenário	Status Conta	API Resources	API Listagem Cartão de Crédito	APIs Dados Contas	Code
Sem consentimento	-	401 unauthorized	401 unauthorized	401 unauthorized	
Com consentimento não autorizado	-	401 unauthorized	401 unauthorized	401 unauthorized	
Com consentimento autorizado (pendente múltipla alçada)	-	200 - Retorna conta com status PENDING_AUTHORISATION	200 Não retorna conta	403 forbidden	STATUS_RESOURCE_PENDING_AUTHORISATION
Com consentimento autorizado (aprovado múltipla alçada)	OK	200 - Retorna conta com status AVAILABLE	200 retorna a conta	200 retorna dados	
Com consentimento autorizado (aprovado múltipla alçada)	Bloqueio	200 - Retorna conta com status TEMPORARY_UNAVAILABLE	200 Não retorna conta	403 forbidden	STATUS_RESOURCE_TEMPORARY_UNAVAILABLE
Com consentimento autorizado (aprovado múltipla alçada)	Encerrada/Migrada	200 - Retorna conta com status UNAVAILABLE	200 Não retorna conta	403 forbidden	STATUS_RESOURCE_UNAVAILABLE
Com consentimento autorizado (recusado múltipla alçada)	-	200 - Retorna conta com status UNAVAILABLE	200 Não retorna conta	403 forbidden	STATUS_RESOURCE_UNAVAILABLE
Com consentimento autorizado (conta inexistente)	-	200 - Não retorna na lista	200 Não retorna conta	404 not found	
Com consentimento revogado ou expirado*		401 unauthorized	401 unauthorized	401 unauthorized	

* Para os cenários de consentimento revogado ou expirado, conforme determinado na especificação de segurança o access token associado ao mesmo é invalidado impossibilitando a consulta as APIs produtos por parte da Receptora.



COMPORTAMENTO DAS APIS DE CRÉDITO

(Loans, Financings, Unarranged-accounts-overdraft e Invoice-financings com relação a Resources)



Visão Geral da API de Recursos (Resources) para as APIs de Operações de crédito

O identificador único **resourceId** de cada recurso do tipo operações de crédito presente na lista devolvida pela API, corresponde ao identificador único **contractId** da API a modalidade correspondente.

REGRAS DE IMPLEMENTAÇÃO PELA INSTITUIÇÃO TRANSMISSORA

Importante:

No caso de operações de crédito o cliente não efetua a confirmação por cada contrato que possui, e sim por modalidade de operação, conforme última coluna da tabela acima, com as opções válidas para os casos de operações de crédito.

A categorização das operações de crédito a serem compartilhadas são chamadas de modalidade no Guia de Experiência do Usuário em função do uso da mesma terminologia “modalidade” na circular 4015, porém essa mesma categorização no doc 3040 é chamada de submodalidade. A fim de manter padronização no Open Finance, consideraremos o termo “modalidade”.

As modalidades definidas na circular 4015 foram baseadas nas submodalidades do 3040, porém não há correspondência exata.

Utilizaremos no texto abaixo o termo “passíveis de compartilhamento” para contratos ativos ou que estiveram ativos nos últimos 12m (considerando o momento da consulta pela Receptora).



6 API de Recursos (Resources)



Tabela com as modalidades e submodalidades das API's de crédito, mais informações podem ser consultadas no [Guia de Experiência do Usuário](#).

Domínio	Descrição	Sub	Descrição	Circular 4015
01	Adiantamentos a depositantes	01	Adiantamentos a depositantes	Adiantamentos a depositantes
		02	Crédito pessoal - com consignação em folha de pagam.	Crédito pessoal - consignado
		03	Crédito pessoal - sem consignação em folha de pagam.	Crédito pessoal - sem consignação
		11	<i>Home equity</i>	<i>Home equity</i>
02	Empréstimos	12	Microcrédito produtivo orientado	Microcrédito
		13	Cheque especial	Cheque especial
		14	Conta garantida	Conta garantida
		15	Capital de giro com prazo de vencimento de até 365 dias	
		16	Capital de giro com prazo de vencimento superior a 365 dias	Capital de giro
		17	Capital de giro com teto rotativo	
		03	Direitos creditório descontados	01
02	Desconto de cheques			Desconto de cheques
03	Antecipação de faturas de cartão de crédito			Antecipação de recebíveis de cartão de crédito
98	Outros direitos creditórios descontados			Desconto de nota promissória
99	Outros títulos descontados			Desconto de nota promissória



Domínio	Descrição	Sub	Descrição	Circular 4015
04		01	Aquisição de bens - veículos automotores	Aquisições de bens móveis
		02	Aquisição de bens - outros bens	
		03	Microcréditos	Microcrédito produtivo orientado
08	Financiamentos	01	Custeio	Rurais
		02	Investimento	
		03	Comercialização	
		04	Industrialização	
09		01	Financiamento habitacional - SFH	Sistema Financeiro da Habilitação (SFH)
		02	Financiamento habitacional - exceto SFH	Sistema Financeiro Imobiliário (SFI)



Regras de negócio:

- **RN001:** É obrigatório que todos os recursos do tipo operações de crédito passíveis de compartilhamento, da modalidade de crédito selecionada pelo cliente retornem o identificador único independente do status ser informado.
- **RN002:** É obrigatório o retorno de todas as operações de crédito passíveis de compartilhamento da modalidade selecionada pelo cliente, independente de seus status. Operação de crédito que já foram objeto de compartilhamento no Open Finance (que em algum momento teve seu status **AVAILABLE**) deve ter seu status alterado para **UNAVAILABLE** na data em que completa 12 meses do encerramento (ou liquidação) do seu contrato.
- **RN003:** Caso a Transmissora ainda esteja preparando a listagem dos recursos autorizados da API de Recursos (Resources), deve ser retornado o código **HTTP Status Code 202 - accepted** com o *body* vazio e a Receptora deverá seguir as recomendações de *polling*
- **RN004:** É obrigatório que todas as operações passíveis de compartilhamento da modalidade escolhida sejam compartilhadas, ou seja, tenham seu **resourceId** corretamente expostos. Isso vale tanto para contratos existentes no momento do consentimento, encerrados nos últimos 12m ou para contratos que venham a ser firmados após esse momento, desde que o consentimento esteja vigente.

Regras de negócio:

- **RN005:** contratos de operações de crédito que estejam em processo de múltiplas alçadas para aprovação DEVEM retornar o status **PENDING_AUTHORISATION**.
- **RN006:** o método de listagem de contratos das API's de Operações de crédito deve retornar apenas a lista de contratos que estejam com o status da Resources como **AVAILABLE**.
- **RN007:** cada recurso (contrato) de uma mesma modalidade de crédito autorizada no consentimento deve ter seus status tratado de forma independente.
- **RN008:** no caso em que um cliente tenha compartilhado alguma modalidade, mas não possua nenhum contrato desta modalidade compartilhada, a resources deve retornar lista vazia, com **HTTP Status Code 200 - OK**.





Exemplo de retorno JSON

Utilizando API de Adiantamento a Depositantes (Unarranged-Accounts-Overdraft) como exemplo, o retorno esperado em JSON para alguns casos, está descrito abaixo:

Levando em consideração que um consentimento possua dois Adiantamentos a Depositantes, neste cenário o retorno de uma chamada a API de Adiantamento a Depositantes (Unarranged-Accounts-Overdraft) seria:

```
{
  "data": [
    {
      "contractId": "xcjklompowsa279212650822221989319aadrtjk",
      "brandName": "Organização A",
      "companyCnpj": "60500998000144",
      "productType": "ADIANTAMENTO_A_DEPOSITANTES",
      "productSubType": "ADIANTAMENTO_A_DEPOSITANTES",
      "ipocCode": "92792126019929279212650822221989319252576"
    },
    {
      "contractId": "989319aad921265powsa27rtjk0822221xcjklom",
      "brandName": "Organização A",
      "companyCnpj": "60500998000144",
      "productType": "ADIANTAMENTO_A_DEPOSITANTES",
      "productSubType": "ADIANTAMENTO_A_DEPOSITANTES",
      "ipocCode": "2094592478592874587523452348523459898798"
    }
  ],
  ...
}
```




Exemplo de retorno JSON

Neste caso a API de Recursos (Resources) deve devolver um `resourceId` para cada `contractId` apresentado pela API de Adiantamento a Depositantes (Unarranged-Accounts-Overdraft):

```
{
  "data": [
    {
      "resourceId": "xcjklompowsa279212650822221989319aadrtjk",
      "type": "UNARRANGED_ACCOUNT_OVERDRAFT",
      "status": "AVAILABLE"
    },
    {
      "resourceId": "989319aad921265powsa27rtjk0822221xcjklom",
      "type": "UNARRANGED_ACCOUNT_OVERDRAFT",
      "status": "AVAILABLE"
    }
  ],
  ...
}
```

Diagrama representacional dos status da API de Recursos (Resources)

Produtos encerrados X Período de vigência do consentimento

Cenários	Consentimento (ex. 4 meses)					
	DIA D	D + 1M	D + 2M	D + 3M	D + 4M	D + 5M
Produto encerrado em D -13M	Não é escopo de compartilhamento					
Produto encerrado em D -11M	AVAILABLE	AVAILABLE	UNAVAILABLE	UNAVAILABLE	UNAVAILABLE	UNAUTHORISED
Produto encerrado em D +2M	AVAILABLE	AVAILABLE	AVAILABLE	UNAVAILABLE	UNAVAILABLE	UNAUTHORISED

*encerrado/liquidado/sofreu portabilidade / cancelado /quitada...

Glossário:

- **Dia D:** dia do consentimento
- **Eixo X:** represente o tempo em meses
- **Eixo em Y:** representa os cenários que são produtos e suas vigências. Ex.: empréstimo que se encerrou em 13 meses.
- O encontro dos eixos representa os status da resouces, naquele tempo para aquele produto.



Regras para o preenchimento do campo status do Recurso (PENDING_AUTHORISATION, AVAILABLE, UNAVAILABLE, TEMPORARILY_UNAVAILABLE) e a transição entre status.

PENDING_AUTHORISATION: indica a existência de pendências para o compartilhamento do recurso, em caso de múltiplas alçadas, quando é necessário o consentimento de mais de um titular. Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** com o conteúdo abaixo:

```
{
  "code": "status_RESOURCE_PENDING_AUTHORISATION",
  "title": "Aguardando autorização de múltiplas alçadas",
  "detail": "(uso a critério do transmissor)"
}
```

AVAILABLE: indica que o recurso da modalidade de operação de crédito escolhida encontra-se disponível para consulta nas APIs Operações de crédito através do identificador `contractId` compartilhado. A Transmissora retorna os dados com **HTTP Status Code 200 - OK**.

UNAVAILABLE: indica que o recurso (contrato) se encontra indisponível para consulta na API de Operação de crédito. Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** com o conteúdo abaixo:

```
{
  "code": "status_RESOURCE_UNAVAILABLE",
  "title": "Recurso indisponível",
  "detail": "(uso a critério do transmissor)"
}
```



Regras para o preenchimento do campo status do Recurso (PENDING_AUTHORISATION, AVAILABLE, UNAVAILABLE, TEMPORARILY_UNAVAILABLE) e a transição entre status.

TEMPORARILY_UNAVAILABLE: indica que o recurso (contrato) se encontra temporariamente indisponível para consulta nas APIs de operações de crédito da modalidade em questão. Neste caso a Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden** com o conteúdo abaixo:

```
{
  "code": "status_RESOURCE_TEMPORARILY_UNAVAILABLE",
  "title": "Recurso temporariamente indisponível",
  "detail": "(uso a critério do transmissor)"
}
```



Casos de uso de Operações de Crédito e o respectivo status a ser informado pela API de Recursos (Resources)

- Operações de crédito que se encontram em **plena utilização** e disponíveis nos canais de atendimento eletrônico para os usuários finais ou empréstimos liquidados nos últimos 12m devem ter seus status como **AVAILABLE**;
- Operações de crédito **portadas para outra instituição** devem parar de ter suas informações compartilhadas, e o recurso correspondente deve ter seu status atualizado para **UNAVAILABLE**
- Operações de crédito que se encontram em **bloqueios temporários**, onde o cliente não possui acesso aos detalhes por meio dos canais de atendimento eletrônico, devem retornar o status **TEMPORARILY_UNAVAILABLE**;
- Operações de crédito que se encontram em **bloqueios definitivos** onde o cliente não possui acesso por meio dos canais de atendimento eletrônico, devem retornar o status **UNAVAILABLE**;
- Operações de crédito que se encontram **liquidadas**, onde o cliente não possui acesso por meio dos canais de atendimento eletrônico, devem retornar o status **UNAVAILABLE**;



Casos de uso de Operações de Crédito e o respectivo status a ser informado pela API de Recursos (Resources)

- Operações de crédito **liquidadas nos últimos 12 meses** em relação à data da consulta pela Receptora devem retornar o status **AVAILABLE**;
- Operações de crédito **liquidadas há mais que 12 meses** em relação a data da consulta pela Receptora devem retornar o status **UNAVAILABLE**;
- Casos de uso que um cliente final efetue um consentimento para modalidade para a qual um dos contratos exija a aprovação de **múltiplas alçadas** (**PENDING_AUTHORISATION**) e o outro contrato esteja **disponível** para consulta (**AVAILABLE**), o comportamento esperado é que cada contrato tenha seu status representado de forma independente, disponibilizando imediatamente os contratos já aprovados.
- Operações de crédito **canceladas** não são escopo de exposição no Open Finance. Contratos nessa situação devem parar de ter suas informações compartilhadas, e o recurso correspondente deve ter seu status atualizado para **UNAVAILABLE**;
- Operações de crédito que tenham **ido para perda** não são escopo de exposição no Open Finance. Contratos nessa situação devem parar de ter suas informações compartilhadas, e o recurso correspondente deve ter seu status atualizado para **UNAVAILABLE**.



REGRAS DE CONSUMO PELA INSTITUIÇÃO RECEPTORA

Regras quanto a interpretação do status do Recurso detalhando quando chamar a API de Operações de crédito:

- **PENDING_AUTHORISATION:** neste status a recomendação é que a Receptora aguarde a alteração desse status para um dos demais status executando polling na API de Recursos (Resources) (Vide recomendações de polling).
- **AVAILABLE:** indica que o recurso encontra-se disponível para consulta na API de operação de crédito da modalidade correspondente através do identificador `contractId` compartilhado
- **UNAVAILABLE:** neste status a recomendação é que a Receptora não consulte a API de operação de crédito em função do dado não estar disponível para esse `contractId`
- **TEMPORARILY_UNAVAILABLE:** neste status a recomendação é que a Receptora aguarde a alteração desse status para um dos demais status executando polling na API de Recursos (Resources) (Vide recomendações de polling).



Atualizações na documentação de Resources quanto as regras de busca das informações nas APIs de Operações de Crédito

- Recomenda-se que a Receptora consulte a API de Recursos (Resources) logo após identificar que o consentimento se encontra autorizado de modo a obter todos os contratos da modalidade consentida pelo cliente e seus respectivos identificadores – lembrando que para operações de crédito pode não haver nenhum contrato no momento da primeira consulta, e posteriormente novos contratos podem aparecer.
- Recomenda-se que a Receptora implemente uma rotina de consulta de novos contratos consumindo a API de Recursos (Resources) de forma periódica. Cabe a Instituição Receptora definir a periodicidade que melhor atende às suas necessidades (lembrando de notificar o seu usuário final sobre esta periodicidade), contudo respeitando o limite de uma requisição ao dia com este propósito, exceto em casos de atualizações ativas pelo cliente.
- Recomenda-se que a Receptora caso obtenha um erro **HTTP Status Code 403 - Forbidden** na API de Empréstimos (Loans) avalie o JSON retornado no *body* da requisição para identificar as informações adicionais referentes ao erro, como o status específico da API de Recursos (Resources) que impede o consumo da informação
- Recomenda-se que a Receptora consulte a API de Recursos (Resources) caso obtenha um erro na API de operação de crédito, de modo a identificar uma possível alteração de status do recurso.



Tabela descritiva das possibilidades de interação entre API de Recursos (Resources) e as APIs de Operações de Crédito

(A tabela abaixo explicita os status esperados par as APIs de Operações de crédito com a utilização da API de Adiantamento a Depositantes (Unarranged-Accounts-Overdraft) como exemplo)

Cenário	Status do recurso na Detentora	API Resources (.../resources/v1/resources)	API Listagem Produto (.../unarranged-accounts-overdraft/v1/contracts)	APIs Dados Produto (.../unarranged-accounts-overdraft/v1/contracts/{contractId})
Sem consentimento	-	401 unauthorized	401 unauthorized	401 unauthorized
Com consentimento não autorizado	-	401 unauthorized	401 unauthorized	401 unauthorized
Com consentimento autorizado (pendente múltipla alçada)	-	200 - Retorna recurso com status PENDING_AUTHORISATION .../resources/v1/resources <pre>[{ "resourceId": "1234", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "PENDING_AUTHORISATION" }]</pre>	200 Não retorna .../unarranged-accounts-overdraft/v1/contracts <pre>{ "data": [] ... }</pre>	403 forbidden .../unarranged-accounts-overdraft/v1/contracts/1234 <pre>{ "code": "status_RESOURCE_PENDING_AUTHORISATION", "title": "Aguardando autorização de múltiplas alçadas", "detail": "xxxxxx" }</pre>



Cenário	Status do recurso na Detentora	API Resources (.../resources/v1/resources)	API Listagem Produto (.../unarranged-accounts-overdraft/v1/contracts)	APIs Dados Produto (.../unarranged-accounts-overdraft/v1/contracts/{contractId})
Com consentimento autorizado (aprovado múltipla alçada)	OK	200 - Retorna o recurso com status AVAILABLE .../resources/v1/resources <pre>[{ "resourceId": "1234", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "AVAILABLE" }, { "resourceId": "4321", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "AVAILABLE" }]</pre>	200 retorna o recurso na lista .../unarranged-accounts-overdraft/v1/contracts <pre>{ "data": [{ "contractId": "1234", "brandName": "Organização A", "companyCnpj": "60500998000144", "productType": "ADIANTAMENTO_A_DEPOSITANTES", "productSubType": "ADIANTAMENTO_A_DEPOSITANTES", "ipocCode": "123456" }, { "contractId": "4321", "brandName": "Organização A", "companyCnpj": "60500998000144", "productType": "ADIANTAMENTO_A_DEPOSITANTES", "productSubType": "ADIANTAMENTO_A_DEPOSITANTES", "ipocCode": "654321" }] ... }</pre>	200 retorna dados para o recurso .../unarranged-accounts-overdraft/v1/contracts/1234 <pre>{ "data": [{ "contractId": "1234", "brandName": "Organização A", "companyCnpj": "60500998000144", "productType": "ADIANTAMENTO_A_DEPOSITANTES", "productSubType": "ADIANTAMENTO_A_DEPOSITANTES", "ipocCode": "123456" }] ... }</pre>



6 API de Recursos (Resources)

Cenário	Status do recurso na Detentora	API Resources (.../resources/v1/resources)	API Listagem Produto (.../unarranged-accounts-overdraft/v1/contracts)	APIs Dados Produto (.../unarranged-accounts-overdraft/v1/contracts/{contractId})
Com consentimento autorizado (aprovado múltipla alçada)	Bloqueio Temporário	200 - Retorna o recurso com status TEMPORARY_UNAVAILABLE .../resources/v1/resources <pre>[{ "resourceId": "1234", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "TEMPORARY_UNAVAILABLE" }, { "resourceId": "4321", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "AVAILABLE" }]</pre>	200 Não retorna na lista o recurso "bloqueado" .../unarranged-accounts-overdraft/v1/contracts <pre>{ "data": [{ "contractId": "4321", "brandName": "Organização A", "companyCnpj": "60500998000144", "productType": "ADIANTAMENTO_A_DEPOSITANTES", "productSubType": "ADIANTAMENTO_A_DEPOSITANTES", "ipocCode": "654321" }] ... }</pre>	403 forbidden .../unarranged-accounts-overdraft/v1/contracts/1234 <pre>{ "code": "status_RESOURCE_PENDING_AUTHORISATION", "title": "Aguardando autorização de múltiplas alçadas", "detail": "(uso a critério do transmissor)" }</pre>



Cenário	Status do recurso na Detentora	API Resources (.../resources/v1/resources)	API Listagem Produto (.../unarranged-accounts-overdraft/v1/contracts)	APIs Dados Produto (.../unarranged-accounts-overdraft/v1/contracts/{contractId})
Com consentimento autorizado (aprovado múltipla alçada)	Liquidado a menos de 12 meses	200 - Retorna o recurso como status AVAILABLE .../resources/v1/resources	200 retorna o recurso na lista .../unarranged-accounts-overdraft/v1/contracts	200 retorna os dados .../unarranged-accounts-overdraft/v1/contracts/1234
		<pre>[{ "resourceId": "1234", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "AVAILABLE" }, { "resourceId": "4321", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "AVAILABLE" }]</pre>	<pre>{ "data": [{ "contractId": "1234", "brandName": "Organização A", "companyCnpj": "60500998000144", "productType": "ADIANTAMENTO_A_DEPOSITANTES", "productSubType": "ADIANTAMENTO_A_DEPOSITANTES", "ipocCode": "123456" }, { "contractId": "4321", "brandName": "Organização A", "companyCnpj": "60500998000144", "productType": "ADIANTAMENTO_A_DEPOSITANTES", "productSubType": "ADIANTAMENTO_A_DEPOSITANTES", "ipocCode": "654321" }] ... }</pre>	<pre>{ "data": [{ "contractId": "1234", "brandName": "Organização A", "companyCnpj": "60500998000144", "productType": "ADIANTAMENTO_A_DEPOSITANTES", "productSubType": "ADIANTAMENTO_A_DEPOSITANTES", "ipocCode": "123456" }, ...] }</pre>



6 API de Recursos (Resources)

Cenário	Status do recurso na Detentora	API Resources (.../resources/v1/resources)	API Listagem Produto (.../unarranged-accounts-overdraft/v1/contracts)	APIs Dados Produto (.../unarranged-accounts-overdraft/v1/contracts/{contractId})
Com consentimento autorizado (aprovado múltipla alçada)	Liquidado a mais de 12 meses (recurso nunca compartilhado na vigência do consentimento)	200 - Não retorna o recurso obs.: como exemplo o recurso de resourceId 1234 foi liquidado a mais de 12 meses .../resources/v1/resources <pre>[{ "resourceId": "1234", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "AVAILABLE" }, { "resourceId": "4321", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "AVAILABLE" }]</pre>	200 - Não retorna o recurso .../unarranged-accounts-overdraft/v1/contracts <pre>{ "data": [{ "contractId": "4321", "brandName": "Organização A", "companyCnpj": "60500998000144", "productType": "ADIANTAMENTO_A_DEPOSITANTES", "productSubType": "ADIANTAMENTO_A_DEPOSITANTES", "ipocCode": "654321" }] ... }</pre>	404 not found .../unarranged-accounts-overdraft/v1/contracts/1234



Cenário	Status do recurso na Detentora	API Recursos (.../resources/v1/resources)	API Listagem Produto (.../unarranged-accounts-overdraft/v1/contracts)	APIs Dados Produto .../unarranged-accounts-overdraft/v1/contracts/{contractId}
Com consentimento autorizado (aprovado múltipla alçada)	Liquidado a mais de 12 meses (recurso já compartilhado na vigência do consentimento)	200 - Retorna o recurso como status UNAVAILABLE .../resources/v1/resources <pre>[{ "resourceId": "1234", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "UNAVAILABLE" }]</pre>	200 - Não retorna o recurso .../unarranged-accounts-overdraft/v1/contracts <pre>{ "data": [] ... }</pre>	403 forbidden .../unarranged-accounts-overdraft/v1/contracts/1234 <pre>{ "code": "status_RESOURCE_UNAVAILABLE", "title": "Aguardando autorização de múltiplas alçadas", "detail": "(uso a critério do transmissor)" }</pre>
Com consentimento autorizado (recusado múltipla alçada)	-	200 - Retorna recurso com status UNAVAILABLE .../resources/v1/resources <pre>[{ "resourceId": "1234", "type": "UNARRANGED_ACCOUNT_OVERDRAFT", "status": "UNAVAILABLE" }]</pre>	200 Não retorna recurso recusado na lista .../unarranged-accounts-overdraft/v1/contracts <pre>{ "data": [] ... }</pre>	403 forbidden .../unarranged-accounts-overdraft/v1/contracts/1234 <pre>{ "code": "status_RESOURCE_UNAVAILABLE", "title": "Aguardando autorização de múltiplas alçadas", "detail": "(uso a critério do transmissor)" }</pre>



6 API de Recursos (Resources)

Cenário	Status do recurso na Detentora	API Recursos (.../resources/v1/resources)	API Listagem Produto (.../unarranged-accounts-overdraft/v1/contracts)	APIs Dados Produto .../unarranged-accounts-overdraft/v1/contracts/{contractId}
Com consentimento autorizado (recurso inexistente)		HTTP 200 - Retorna array vazio .../resources/v1/resources []	200 Não retorna .../unarranged-accounts-overdraft/v1/contracts { "data": [] ... }	404 not found .../unarranged-accounts-overdraft/v1/contracts/1234
Com consentimento revogado ou expirado*		401 unauthorized	401 unauthorized	401 unauthorized

* Para os cenários de consentimento revogado ou expirado, conforme determinado na especificação de segurança o access token associado ao mesmo é invalidado impossibilitando a consulta as APIs produtos por parte da Receptora.



REGRAS PARA O USO DO STATUS DO RECURSO

- **AVAILABLE:** A Transmissora retorna os dados com status **HTTP Status Code 200 - OK**.
- **UNAVAILABLE:** A Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden**, com o conteúdo ao lado:
- **TEMPORARILY_UNAVAILABLE:** A Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden**, com o conteúdo ao lado:
- **PENDING_AUTHORISATION:** A Transmissora deve retornar um erro **HTTP Status Code 403 - Forbidden**, com o conteúdo ao lado:

Durante o consumo da API de Adiantamento a Depositantes (Unarranged-Accounts-Overdraft) pelo Receptor, sendo o status do recurso os itens apresentados abaixo.

```
{
  "code": "status_RESOURCE_UNAVAILABLE",
  "title": "Recurso indisponível",
  "detail": "(uso a critério do transmissor)"
}
```

```
{
  "code": "status_RESOURCE_TEMPORARILY_UNAVAILABLE",
  "title": "Recurso temporariamente indisponível",
  "detail": "(uso a critério do transmissor)"
}
```

```
{
  "code": "status_RESOURCE_PENDING_AUTHORISATION",
  "title": "Aguardando autorização de múltiplas alçadas",
  "detail": "(uso a critério do transmissor)"
}
```

O QUE FOI ALTERADO:

A versão mais recente das mudanças pode ser encontrada em:

[Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)](#)



07

API de Dados Cadastrais

(Customers)



API CUSTOMERS - OPEN FINANCE BRASIL



1.0.3 OAS3

API de Dados Cadastrais (Customers) de clientes do Open Finance Brasil - Fase 2. API que retorna os dados cadastrais de clientes e de seus representantes, incluindo dados de identificação, de qualificação financeira, informações sobre representantes cadastrados e sobre o relacionamento financeiro do cliente com a Instituição Transmissora dos dados.

Possui segregação entre pessoa natural e pessoa jurídica.

Requer consentimento do cliente para todos os *endpoints*.



ORIENTAÇÕES

- A Role do diretório de participantes relacionada à presente API é a DADOS.
- Para todos os *endpoints* desta API é previsto o envio de um *token* através do *header Authorization*.
- Este *token* deverá estar relacionado ao consentimento (`consentId`) mantido na Instituição Transmissora dos dados, o qual permitirá a pesquisa e retorno, na API em questão, dos dados relacionados ao `consentId` específico relacionado.
- Os dados serão devolvidos na consulta desde que o `consentId` relacionado corresponda a um consentimento válido e com o status `AUTHORISED`.
- É também necessário que o recurso em questão (conta, contrato, etc) esteja disponível na Instituição Transmissora (ou seja, sem bloqueios de qualquer natureza e com todas as autorizações/consentimentos já autorizados).
- Além disso as *permissions* necessárias deverão ter sido solicitadas quando da criação do consentimento relacionado (`consentId`).
- Relacionamos a seguir as *permissions* necessárias para a consulta de dados em cada *endpoint* da presente API.
- Os campos `dateTime` trafegados nesta API seguem o padrão da RFC3339 e formato “zulu”. E só campos do tipo data utilizam como referência o horário de Brasília

O QUE FOI ALTERADO

A versão mais recente das mudanças pode ser encontrada em:

[*Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)*](#)





08

API de Cartões de Crédito

(Credit-Cards-Accounts)



API CREDIT-CARDS-ACCOUNTS - OPEN FINANCE BRASIL



1.0.4 OAS3

API de contas de pagamento pós-pagas do Open Finance Brasil - Fase 2. API que retorna informações de contas de pagamento pós-paga mantidas nas Instituições Transmissoras por seus clientes, incluindo dados como denominação, produto, bandeira, limites de crédito, informações sobre transações de pagamento efetuadas e faturas.

Não possui segregação entre pessoa natural e pessoa jurídica.

Requer consentimento do cliente para todos os endpoints.



ORIENTAÇÕES

- A Role do diretório de participantes relacionada à presente API é a DADOS.
- Para todos os *endpoints* desta API é previsto o envio de um *token* através do *header Authorization*.
- Este *token* deverá estar relacionado ao consentimento (*consentId*) mantido na Instituição Transmissora dos dados, o qual permitirá a pesquisa e retorno, na API em questão, dos dados relacionados ao **consentId** específico relacionado.
- Os dados serão devolvidos na consulta desde que o **consentId** relacionado corresponda a um consentimento válido e com o status **AUTHORISED**.
- É também necessário que o recurso em questão (conta, contrato, etc) esteja disponível na Instituição Transmissora (ou seja, sem bloqueios de qualquer natureza e com todas as autorizações/consentimentos já autorizados).
- Além disso as *permissions* necessárias deverão ter sido solicitadas quando da criação do consentimento relacionado (*consentId*).
- Relacionamos a seguir as *permissions* necessárias para a consulta de dados em cada *endpoint* da presente API.

O QUE FOI ALTERADO

A versão mais recente das mudanças pode ser encontrada em:

[*Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)*](#)





09

API de Contas

(Accounts)



API ACCOUNTS - OPEN FINANCE BRASIL



1.0.3 OAS3

API de contas de depósito à vista, contas de poupança e contas pré-pagas do Open Finance Brasil – Fase 2. API que retorna informações de contas de depósito à vista, contas de poupança e contas de pagamento pré-pagas mantidas nas Instituições Transmissoras por seus clientes, incluindo dados de identificação da conta, saldos, limites e transações.

Não possui segregação entre pessoa natural e pessoa jurídica.

Requer consentimento do cliente para todos os *endpoints*.



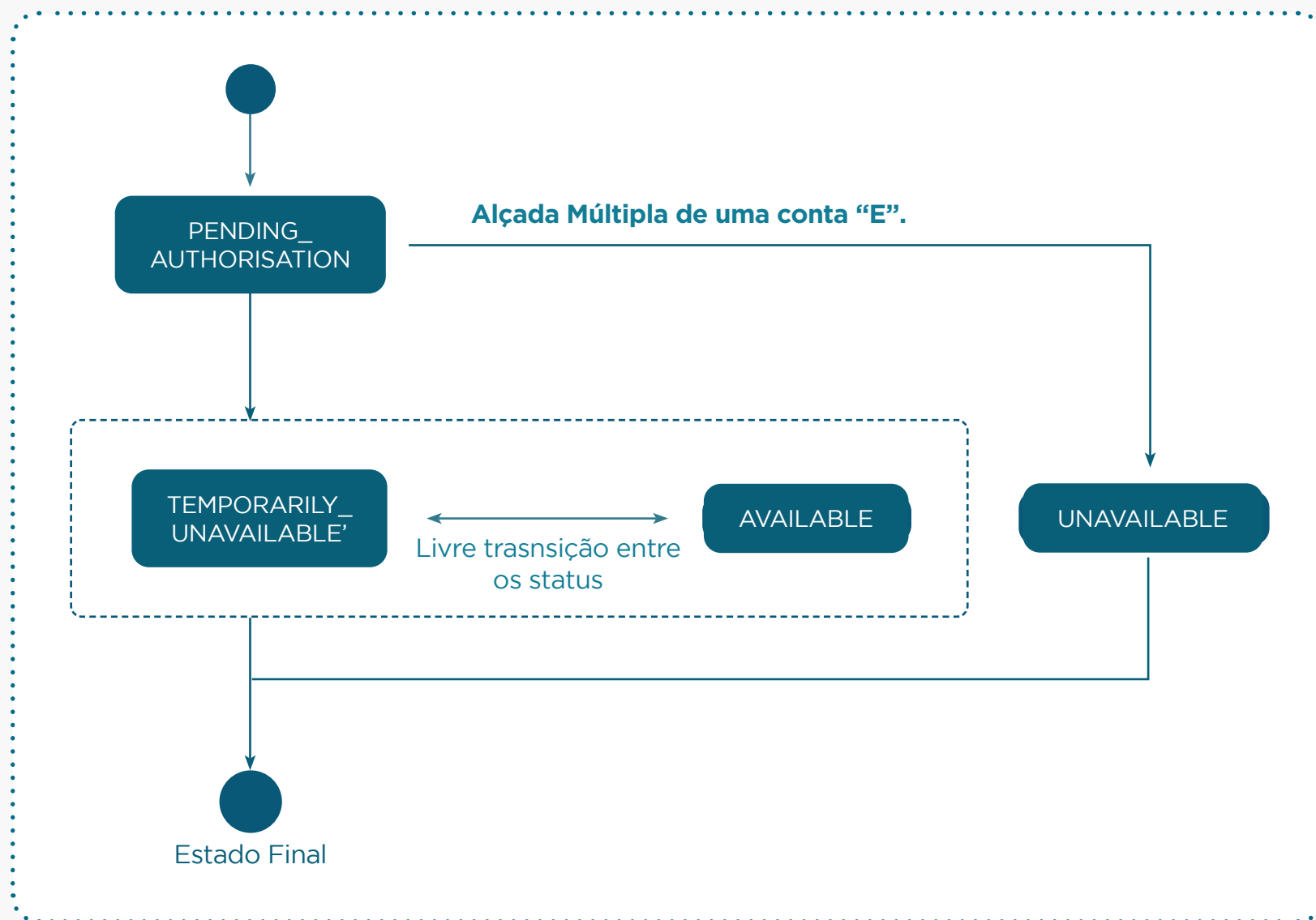
ORIENTAÇÕES

- A Role do diretório de participantes relacionada à presente API é a DADOS.
- Para todos os *endpoints* desta API é previsto o envio de um *token* através do *header Authorization*.
- Este *token* deverá estar relacionado ao consentimento (`consentId`) mantido na Instituição Transmissora dos dados, o qual permitirá a pesquisa e retorno, na API em questão, dos dados relacionados ao `consentId` específico relacionado.
- Os dados serão devolvidos na consulta desde que o `consentId` relacionado corresponda a um consentimento válido e com o status `AUTHORISED`.
- É também necessário que o recurso em questão (conta, contrato, etc) esteja disponível na Instituição Transmissora (ou seja, sem bloqueios de qualquer natureza e com todas as autorizações/consentimentos já autorizados).
- Além disso as *permissions* necessárias deverão ter sido solicitadas quando da criação do consentimento relacionado (`consentId`).
- Relacionamos a seguir as *permissions* necessárias para a consulta de dados em cada *endpoint* da presente API.



DIAGRAMA VISUAL

Diagrama de máquina de estados - API de Dados Cadastrais (Customers)



O QUE FOI ALTERADO

A versão mais recente das mudanças pode ser encontrada em:

[*Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)*](#)





10

API Operações de Crédito: Empréstimos (Loans)



API LOANS - OPEN FINANCE BRASIL



1.0.4 OAS3

API de informações de operações de empréstimos do Open Finance Brasil – Fase 2. API que retorna informações de operações de crédito do tipo empréstimo, mantidas nas Instituições Transmissoras por seus clientes, incluindo dados como denominação, modalidade, número do contrato, tarifas, prazo, prestações, pagamentos, amortizações, garantias, encargos e taxas de juros remuneratórios.

Não possui segregação entre pessoa natural e pessoa jurídica.

Requer consentimento do cliente para todos os *endpoints*.



ORIENTAÇÕES

- A Role do diretório de participantes relacionada à presente API é a DADOS.
- Para todos os *endpoints* desta API é previsto o envio de um *token* através do *header Authorization*.
- Este *token* deverá estar relacionado ao consentimento (`consentId`) mantido na Instituição Transmissora dos dados, o qual permitirá a pesquisa e retorno, na API em questão, dos dados relacionados ao `consentId` específico relacionado.
- Os dados serão devolvidos na consulta desde que o `consentId` relacionado corresponda a um consentimento válido e com o status `AUTHORISED`.
- É também necessário que o recurso em questão (conta, contrato, etc) esteja disponível na Instituição Transmissora (ou seja, sem bloqueios de qualquer natureza e com todas as autorizações/consentimentos já autorizados).
- Além disso as *permissions* necessárias deverão ter sido solicitadas quando da criação do consentimento relacionado (`consentId`).
- Relacionamos a seguir as *permissions* necessárias para a consulta de dados em cada *endpoint* da presente API.

O QUE FOI ALTERADO

A versão mais recente das mudanças pode ser encontrada em:

[*Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)*](#)





11

API Operações de Crédito: Financiamento (Financings)



API FINANCINGS - OPEN FINANCE BRASIL



1.0.4 OAS3

API de informações de operações de financiamentos do Open Finance Brasil - Fase 2. API que retorna informações de operações de crédito do tipo financiamento, mantidas nas Instituições Transmissoras por seus clientes, incluindo dados como denominação, modalidade, número do contrato, tarifas, prazo, prestações, pagamentos, amortizações, garantias, encargos e taxas de juros remuneratórios.

Não possui segregação entre pessoa natural e pessoa jurídica.

Requer consentimento do cliente para todos os *endpoints*.



ORIENTAÇÕES

- A Role do diretório de participantes relacionada à presente API é a DADOS.
- Para todos os *endpoints* desta API é previsto o envio de um *token* através do *header Authorization*.
- Este *token* deverá estar relacionado ao consentimento (`consentId`) mantido na Instituição Transmissora dos dados, o qual permitirá a pesquisa e retorno, na API em questão, dos dados relacionados ao `consentId` específico relacionado.
- Os dados serão devolvidos na consulta desde que o `consentId` relacionado corresponda a um consentimento válido e com o status `AUTHORISED`.
- É também necessário que o recurso em questão (conta, contrato, etc) esteja disponível na Instituição Transmissora (ou seja, sem bloqueios de qualquer natureza e com todas as autorizações/consentimentos já autorizados).
- Além disso as *permissions* necessárias deverão ter sido solicitadas quando da criação do consentimento relacionado (`consentId`).
- Relacionamos a seguir as *permissions* necessárias para a consulta de dados em cada *endpoint* da presente API.



O QUE FOI ALTERADO

A versão mais recente das mudanças pode ser encontrada em:

[*Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)*](#)





12

API Operações de Crédito: Adiantamento a Depositantes (Unarranged-Accounts-Overdraft)



API UNARRANGED- ACCOUNTS-OVERDRAFT - OPEN FINANCE BRASIL

1.0.4 OAS3

API de informações de operações de adiantamento a depositantes do Open Finance Brasil – Fase 2. API que retorna informações de operações de crédito do tipo adiantamento a depositantes mantidas nas Instituições Transmissoras por seus clientes, incluindo dados como denominação, modalidade, número do contrato, tarifas, prazo, prestações, pagamentos, amortizações, garantias, encargos e taxas de juros remuneratórios.

Não possui segregação entre pessoa natural e pessoa jurídica.

Requer consentimento do cliente para todos os *endpoints*.





ORIENTAÇÕES

- A Role do diretório de participantes relacionada à presente API é a DADOS.
- Para todos os *endpoints* desta API é previsto o envio de um *token* através do *header Authorization*.
- Este *token* deverá estar relacionado ao consentimento (`consentId`) mantido na Instituição Transmissora dos dados, o qual permitirá a pesquisa e retorno, na API em questão, dos dados relacionados ao `consentId` específico relacionado.
- Os dados serão devolvidos na consulta desde que o `consentId` relacionado corresponda a um consentimento válido e com o status `AUTHORISED`.
- É também necessário que o recurso em questão (conta, contrato, etc) esteja disponível na Instituição Transmissora (ou seja, sem bloqueios de qualquer natureza e com todas as autorizações/consentimentos já autorizados).
- Além disso as *permissions* necessárias deverão ter sido solicitadas quando da criação do consentimento relacionado (`consentId`).
- Relacionamos a seguir as *permissions* necessárias para a consulta de dados em cada *endpoint* da presente API.



O QUE FOI ALTERADO

A versão mais recente das mudanças pode ser encontrada em:

[*Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)*](#)





13

API Operações de Crédito: Direitos Creditórios Descontados (Invoice-Financings)



API INVOICE-FINANCINGS - OPEN FINANCE BRASIL



1.0.4 OAS3

API de informações de operações de antecipação de recebíveis do Open Finance Brasil – Fase 2. API que retorna informações de operações de crédito do tipo antecipação de recebíveis – direitos creditórios descontados - mantidas nas Instituições Transmissoras por seus clientes, incluindo dados como denominação, modalidade, número do contrato, tarifas, prazo, prestações, pagamentos, amortizações, garantias, encargos e taxas de juros remuneratórios.

Não possui segregação entre pessoa natural e pessoa jurídica.

Requer consentimento do cliente para todos os *endpoints*.



ORIENTAÇÕES

- A Role do diretório de participantes relacionada à presente API é a DADOS.
- Para todos os *endpoints* desta API é previsto o envio de um *token* através do *header Authorization*.
- Este *token* deverá estar relacionado ao consentimento (`consentId`) mantido na Instituição Transmissora dos dados, o qual permitirá a pesquisa e retorno, na API em questão, dos dados relacionados ao `consentId` específico relacionado.
- Os dados serão devolvidos na consulta desde que o `consentId` relacionado corresponda a um consentimento válido e com o status `AUTHORISED`.
- É também necessário que o recurso em questão (conta, contrato, etc) esteja disponível na Instituição Transmissora (ou seja, sem bloqueios de qualquer natureza e com todas as autorizações/consentimentos já autorizados).
- Além disso as *permissions* necessárias deverão ter sido solicitadas quando da criação do consentimento relacionado (`consentId`).
- Relacionamos a seguir as *permissions* necessárias para a consulta de dados em cada *endpoint* da presente API.



O QUE FOI ALTERADO

A versão mais recente das mudanças pode ser encontrada em:

[*Change Log - Área do Desenvolvedor - Portal do Open Banking Brasil \(atlassian.net\)*](#)





 **open**finance

The logo for Open Finance, featuring a stylized blue and white circular icon to the left of the text "openfinance".

GUIA DE IMPLEMENTAÇÃO DA
VERSÃO 2.0 DAS APIS DE DADOS
CADASTRAIS E TRANSACIONAIS