**Apresentação dos detalhes técnicos das APIs**

Sumário

[Introdução 2](#_Toc323298350)

[Objetivo 3](#_Toc319135756)

[API Quotes 3](#_Toc1948612717)

[Endpoint de Notificação 3](#_Toc1819077428)

[Endpoint de Requisição de Consentimento 5](#_Toc1128802528)

[Schemas 7](#_Toc2068515713)

[Cabeçalhos HTTP de Requisição e Resposta 8](#_Toc1187774493)

[Cabeçalho de Requisição 8](#_Toc2000646274)

[Cabeçalho de Resposta 9](#_Toc1607466334)

[Convenções de Nomenclatura 9](#_Toc544486504)

[Códigos de Resposta HTTP 9](#_Toc350228109)

[Tipos de Dados Comuns 11](#_Toc2035395390)

[Segurança 14](#_Toc2083344630)

[Changelog 15](#_Toc644846253)

# **Introdução**

O Open Insurance é um conceito que visa criar uma arquitetura aberta para que seguradoras venham a trabalhar em conjunto para melhorar os produtos entregues aos seus clientes. Para tal, Open Insurance se vale do conceito de compartilhamento de dados de seus clientes, isso quando há explícita permissão do cliente para tal compartilhamento. Desta forma os segurados receberiam o melhor produto conforme o seu perfil e suas necessidades.

Neste documento discutimos a forma de viabilizar este compartilhamento de recursos consentidos durante a cotação e contratação de produtos de seguros, de capitalização ou de previdência complementar aberta, junto as sociedades participantes.

# **Objetivo**

Este documento visa orientar o uso dos paths (endpoints) da API Quotes que compõe o grupo de APIs destinadas ao envio de LEADs e a geração de cotações e contratações de produtos de seguros, de capitalização ou de previdência complementar aberta, por meio de uma sociedade iniciadora, junto com as seguradoras participantes do Open Insurance.

\*Utilize ALT+left ou ALT+right para navegar entre os links

# **API Quotes**

A API Quotes é composta pelos métodos responsáveis pelo início e controle do fluxo de geração de cotações e contratações de produtos de seguro, de capitalização ou de previdência complementar aberta, bem como a geração de LEADs para as seguradoras participantes do Open Insurance.

## Endpoint de Notificação

POST /open-insurance/quotes/v1/notifications

**Visão Geral**

Cria notificação sobre oportunidade de negócio. Por meio deste path as seguradoras receberão a “mensagem” de que há um cliente potencial interessado na contratação de um produto de seguro, de capitalização ou de previdência complementar aberta, o qual estará especificado no corpo da requisição.

**Parâmetros**

As requisições direcionadas para o path de notificação deverão enviar os seguintes parâmetros no *header*.

[Cabeçalho de Requisição](#_Cabeçalho_de_Requisição)

**Resposta**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status** | **Significado** | **Descrição** | **Schema** |
| 202 | Accepted | Notificação aceita |  |
| 422 | Unprocessable Entity | A solicitação foi bem formada, mas não pôde ser processada devido à lógica de negócios específica da solicitação. | 422ResponseErrorCreateNotification |

**Exemplo de request:**

POST https://api.seguradora.com.br/open-insurance/quotes/v1/notifications HTTP/1.1

Host: api.seguradora.com.br

Accept: application/json

Content-Type: application/json

var req = new XMLHttpRequest();

req.setRequestHeader("Accept", "application/json");

req.open("POST", "https://api.seguradora.com.br/open-insurance/quotes/v1/notifications", true);

req.send(JSON.stringify(

    {

        "data": {"type": "AUTO"}

    }

));

**Exemplo de Requisição (Payload de criação de notificação)**

O *payload* deverá conter a indicação do grupo do produto de interesse do cliente seguindo a estrutura abaixo.

{

    "data": {

        "type": "AUTO"

    }

}

## Endpoint de Requisição de Consentimento

POST /open-insurance/quotes/v1/consent-requests

**Visão Geral**

Cria a solicitação de pedido de consentimento. O request enviado para esse path indicará para a receptora que o pedido de consentimento para busca dos dados, já pode ser feito sobre um id específico.

**Resposta**

|  |  |  |  |
| --- | --- | --- | --- |
| **Status** | **Significado** | **Descrição** | **Schema** |
| 202 | Accepted | Request de criação de consentimento aceita. |  |

**Exemplo de request:**

POST https://api.seguradora.com.br/open-insurance/quotes/v1/consent-requests HTTP/1.1

Host: api.seguradora.com.br

Accept: application/json

Content-Type: application/json

var req = new XMLHttpRequest();

req.setRequestHeader("Accept", "application/json");

req.open("POST", "https://api.seguradora.com.br/open-insurance/quotes/v1/consent-requests", true);

req.send(JSON.stringify(  
  {

    "data": {

      "correlationId": "ab342c34",

      "loggedUser": {

        "document": {

          "identification": "11111111111",

          "rel": "CPF"

        }

      },

      "productType": "AUTO",

      "permissions": [

        "DAMAGES\_AND\_PEOPLE\_AUTO\_CLAIM\_READ"

      ]

    }

  }  
));

**Exemplo de Requisição (Payload de criação de uma requisição de consentimento)**

O *payload* deverá conter o correlationId que será gerado e enviado pela sociedade iniciadora e repassado para a mesma quando a seguradora enviar o request de solicitação de consentimento.

{

"data": {

    "correlationId": "ab342c34",

    "loggedUser": {

      "document": {

        "identification": "11111111111",

        "rel": "CPF"

      }

    },

    "productType": "AUTO",

    "permissions": [

      "DAMAGES\_AND\_PEOPLE\_AUTO\_CLAIM\_READ"

    ]

  }

}

# **Schemas**

**422ResponseErrorCreateNotification**

|  |  |  |  |
| --- | --- | --- | --- |
| **Nome** | **Tipo** | **Obrigatório** | **Definição** |
| errors | [Errors](#Error) | Sim |  |
| meta | [MetaPaginated](#2s8eyo1) | Sim |  |

**MetaPaginated**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nome** | **Tipo** | **Definição** | **Obrigatório** | **Restrição** |
| totalRecords | integer | Número total de registros no resultado | Sim |  |
| totalPages | integer | Número total de páginas no resultado | Sim |  |
| requestDateTime | string |  | Sim | conforme especificação RFC-3339, formato UTC. *exemplo: 2021-05-21T08:30:00Z* |

**Errors**

|  |  |  |
| --- | --- | --- |
| **Propriedade** | **Código** | **Definição** |
| code | String | Código específico do endpoint. |
| title | String | Título do erro legível por humanos. |
| detail | String | Descrição do erro legível por humanos. |
| requestDateTime | String | Data e hora. |

**Enum Error (422ResponseErrorCreateNotification)**

|  |  |  |
| --- | --- | --- |
| **Propriedade** | **Código** | **Definição** |
| Code | PRODUTO\_NAO\_COMERCIALIZADO | Produto não comercializado. |
| Code | RISCO\_NEGADO | Risco em questão está fora das políticas de aceitação da Seguradora. |
| Code | DETALHE\_COTACAO\_INVALIDO | Detalhe da cotação inválido. |
| Code | NAO\_INFORMADO | Não informado |

# **Cabeçalhos HTTP de Requisição e Resposta**

## Cabeçalho de Requisição

|  |  |  |
| --- | --- | --- |
| **Nome do cabeçalho** | **Descrição** | **Obrigatório** |
| Content-Type | Representa o formato do payload de requisição, por padrão/default definido como application/json;charset UTF-8. Obrigatório para chamadas PUT e POST. Os transmissores poderão implementar tratamento para outros padrões, sendo obrigatório apenas o suporte ao padrão. | Não |
| Accept | Especifica o tipo de resposta. Se especificado, deve ser definido como application/json, a menos que o endpoint explicitamente suporte outro formato. Se for definido um valor não suportado pelo endpoint, será retornado o código HTTP 406. Se não especificado, o padrão será application/json. | Não |
| Accept-Encoding | Especifica os tipos de encoding (geralmente algoritmo de compressão) que são suportados pelo cliente, com previsão de suporte ao gzip por parte dos transmissores, sendo que o padrão é a transmissão dos dados não compactados e esta orientação aplica-se aos Dados Abertos. | Não |
| x-fapi-auth-date | Data em que o usuário logou pela última vez com o receptor | Condicional |
| x-fapi-customer-ip-address | O endereço IP do usuário se estiver atualmente logado com o receptor | Condicional |
| x-fapi-interaction-id | Um UUID RFC4122 usado como um ID de correlação. Se fornecido, o transmissor deve "reproduzir" esse valor no cabeçalho de resposta | Não |
| Authorization | Cabeçalho HTTP padrão. Permite que as credenciais sejam fornecidas dependendo do tipo de recurso solicitado. Não obrigatório para APIs públicas. | Sim |
| x-idempotency-key | Cabeçalho HTTP personalizado. Identificador de solicitação exclusivo para suportar a idempotência | Condicional |
| x-jws-signature | Cabeçalho contendo uma assinatura JWS separada do corpo do payload | Condicional |
| x-customer-user-agent | Indica o user agent que o usuário utiliza | Condicional |

## Cabeçalho de Resposta

|  |  |  |
| --- | --- | --- |
| **Nome do cabeçalho** | **Descrição** | **Obrigatório** |
| Content-Encoding | Cabeçalho que indica o tipo de encoding (geralmente algoritmo de compressão) que foi utilizado para envio da resposta. | Não |
| Content-Type | Representa o formato do payload de resposta. Deverá ser application/json a menos que o endpoint requisitado suporte outro formato e este formato tenha sido solicitado através do cabeçalho Accept no momento da requisição. | Condicional |
| x-v | Cabeçalho que indica a versão implementada da API pela sociedade participante. Deve ser preenchido de forma completa, por exemplo: x-v: 1.0.2 | Sim |
| x-jws-signature | Cabeçalho contendo uma assinatura JWS separada do corpo do payload. | Condicional |
| x-fapi-interaction-id | Um UUID RFC4122 usado como um ID de correlação. O transmissor deve usar o mesmo valor recebido na requisição para o cabeçalho de resposta recebido na requisição, caso não tenha sido fornecido, deve se usar um UUID RFC4122 | Não |

# **Convenções de Nomenclatura**

Como padrão é adota padrão Lower Camel Case. Além de seguir o padrão para evitar redundância de termos e utilizar nomes autoexplicativos.

Ex.:

“userEmailAddress”

“userTelephoneNumber”

# **Códigos de Resposta HTTP**

Os códigos de resposta HTTP devem ser usados conforme a tabela abaixo.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Situação** | **Código HTTP** | **Notas** | **POST** | **GET** | **DELETE** |
| A resposta não foi modificada desde a última chamada | 304 Not Modified |  | Não | Sim | Não |
| A requisição foi malformada, omitindo atributos obrigatórios, seja no payload ou através de atributos na URL. | 400 Bad Request. | A operação solicitada não será realizada. | Sim | Sim | Sim |
| Cabeçalho de autenticação ausente/inválido ou token inválido. | 401 Unauthorized. | A operação foi recusada devido a um problema de autenticação. | Sim | Sim | Sim |
| O token tem escopo incorreto ou uma política de segurança foi violada. | 403 Forbidden. | A operação foi recusada devido à falta de permissão para execução. | Sim | Sim | Sim |
| O recurso solicitado não existe ou não foi implementado. | 404 Not Found. |  | Sim | Sim | Sim |
| O consumidor tentou acessar o recurso com um método não suportado. | 405 Method Not Allowed. |  | Sim | Sim | Sim |
| A solicitação continha um cabeçalho Accept diferente dos tipos de mídia permitidos ou um conjunto de caracteres diferente de UTF-8. | 406 Not Acceptable. |  | Sim | Sim | Sim |
| Indica que o recurso não está mais disponível. | 410 Gone. |  | Sim | Sim | Sim |
| A operação foi recusada porque o payload está em um formato não suportado pelo endpoint. | 415 Unsupported Media Type. |  | Sim | Não | Não |
| A solicitação foi bem formada, mas não pôde ser processada devido à lógica de negócios específica da solicitação. | 422 Unprocessable Entity. | Se aplicável ao endpoint, espera-se que esse erro resulte em um payload de erro. | Sim | Sim | Não |
| A operação foi recusada, pois muitas solicitações foram feitas dentro de um determinado período ou o limite global de requisições concorrentes foi atingido. | 429 Too Many Requests. |  | Sim | Sim | Sim |
| Ocorreu um erro no gateway da API ou no microsserviço. | 500 Internal Server Error. | A operação falhou. | Sim | Sim | Sim |
| O serviço está indisponível no momento. | 503 Service Unavailable. |  | Sim | Sim | Sim |
| O servidor não pôde responder em tempo hábil. | 504 Gateway Timeout. | Retornado se ocorreu um tempo limite, mas um reenvio da solicitação original é viável (caso contrário, use 500 Internal Server Error). | Sim | Sim | Sim |

# **Tipos de Dados Comuns**

**Propriedades**

|  |  |  |
| --- | --- | --- |
| **Tipo** | **Descrição** | **Exemplos válidos** |
| AmountString | - Uma string que representa um valor monetário. | "1.37" |
| - Um número positivo, zero ou negativo. | "54.85" |
| - Sem o símbolo da moeda. | "3456928.98" |
| - Com pelo menos 1 e no máximo 16 dígitos antes do ponto decimal. | "-2387.02" |
| - Com no mínimo 2 dígitos (mais dígitos são permitidos, porém não obrigatórios). |  |
| - Sem formatação adicional. Ex.: Separador de milhar. |  |
| Boolean | - Valor booleano padrão. | TRUE |
| FALSE |
| CurrencyString | - Uma string que representa a abreviação da moeda conforme especificação ISO-4217. | "BRL" |
| "USD" |
| "EUR" |
| DateTimeString | - Uma string com data e hora conforme especificação RFC-3339, sempre com a utilização de timezone UTC(UTC time format). | "2020-07-21T08:30:00Z" |
| DurationString | - Uma string que representa um período de duração conforme especificação ISO-8601. | "P23DT23H" |
| "PT2H30M" |
| Enum | - Uma string que representa um domínio de valores | "PRIMEIRA\_OPCAO" |
| - Todos os possíveis valores são definidos. | "OUTRA\_OPCAO\_EXISTENTE" |
| - Os valores devem estar em letras maiúsculas. |  |
| - Espaços em branco devem ser substituídos por \_. |  |
| - Artigos e preposições devem ser removidos. |  |
| - Não devem possuir caracteres acentuados. |  |
| Integer | - Números inteiros. | -1, 0, 1 |
| RateString | - Uma string que representa um valor percentual, tendo como referência que 100% é igual ao valor 1. | "0.01" |
| - Com pelo menos 1 e no máximo 16 dígitos antes do ponto decimal. | "0.1" |
| - Com no máximo 16 dígitos após o ponto decimal. | "-0.05" |
| - Sem formatação adicional. Ex.: Separador de milhar. | "-0.98365" |
| String | - Padrão de texto UTF-8 sem restrição de conteúdo. | "Uma string qualquer." |
| TimeString | - Uma string que representa a hora conforme especificação RFC-3339, sempre com a utilização de timezone UTC(UTC time format). | "00:39:57Z" |
| URIString | - Uma string que representa URI válida. | "http://www.google.com.br" |
| CountryCode | - Código do país de acordo com o código “alpha3” do ISO-3166. | "BRA" |
| IbgeCode | - Código IBGE de Município. A Tabela de Códigos de Municípios do IBGE apresenta a lista dos municípios brasileiros associados a um código composto de 7 dígitos, sendo os dois primeiros referentes ao código da Unidade da Federação. | "3550308" |
| DateString | - Uma string com data conforme especificação RFC-3339 | "2014-03-19" |

# **Paginação**

|  |  |  |
| --- | --- | --- |
| **Nome do Campo** | **Descrição** | **Exemplo de Valor** |
| page | A numeração da página que está sendo requisitada. | 3 |
| page-size | Número de registros por página | 10 |

Como padrão das API elas podem conter o recurso paginação. Este recurso é usado em caso a quantidade de registros justifique a paginação.

Na paginação deverão ser usados parâmetros de query no seguinte formato:

GET {uri}?**page**=3&**page-size**=10

**Regras de Paginação**

Para o funcionamento adequado da paginação é requerido um tamanho máximo de página de 1000 registros. Em caso de requisição com uma quantidade maior que o suportado será retornado o código *422 Unprocessable Entity*.

# **Requisitos não funcionais**

**Disponibilidade das APIs**

A disponibilidade é checada no endpoint GET /discovery/status, conforme foi documentada no item API de Status.

A API de status receberá a requisição a cada 30 segundos com timout de 1s. Serão considerados como *uptime* se o retorno for “OKAY” e Downtime para os retornos:

* PARTIAL\_FAILURE
* SCHEDULED\_OUTAGE

1. Se a requisição for realizada entre o período de 01h e 07h, o contador de SCHEDULED\_OUTAGE é iniciado com 30 segundos acrescidos;
2. Cada nova requisição vai adicionando 30 segundos mais ao contador de SCHEDULED\_OUTAGE, até que uma requisição volte outro valor ou a requisição for feita depois das 07h.

* UNAVAILABLE

1. Se a requisição for realizada entre o período de 07h e 01h;
2. Se serviço não responder a requisição;
3. O contador de downtime é iniciado com 30 segundos acrescidos;
4. Cada nova requisição adicionará 30 segundos a mais ao contador de downtime, até que uma requisição retorne OK.

Regras para cálculo do downtime

* O downtime se refere período de indisponibilidade dentro de 24h, começando e terminando à meia-noite. Sendo contabilizado qualquer endpoint indisponível.
* O downtime é contabilizado com o total de segundos simultâneos por requisição.
* A porcentagem de downtime é calculada dividindo a total de segundos contabilizados de indisponibilidade dividido por 86.400 (24 horas em segundos).
* A disponibilidade é resultado de 100% menos a porcentagem de indisponibilidade.
* Erros da família de erro 5xx http são contabilizados como indisponibilidades por serem considerados erros do servidor de APIs.

**Não contabilizado como downtime**

* Erros da família de erro 4xx http não contabilizados, por normalmente se tratar de falhas dos receptores.
* Uma indisponibilidade por mês, por 3h entre 01h e 07h, sendo necessário reportar com 7 dias de antecedência ao diretório.
* Em caso de manutenção de segurança, previamente aprovado pelo diretório.

**Classificação dos Níveis de Desempenho**

A desempenho será medido em milissegundos, sendo mensurado desde o recebimento da solicitação até a resposta para solicitação.

As APIs são classificadas como:

1. APIs de alta prioridade dentro do percentil 95 em no máximo 1000ms. (status/outages).
2. APIs de média prioridade dentro do percentil mínimo de 95 em no máximo 1500ms. (Channels/Service Products).
3. APIs de baixa prioridade dentro do percentil 95 em no máximo 4000ms. (Admin metrics).

**Nível de serviço (SLA)**

Para garantir a disponibilidade do das APIs e do sistema Open Insurance, as seguintes métricas de disponibilidade são adotadas:

1. 85% do tempo a cada 24 horas.
2. 95% do tempo a cada 1 mês.
3. 99,5% do tempo a cada 3 meses.

**Limite de Requisições**

Para garantir a usabilidade do sistema e estabelecer métricas de requerimentos de disponibilidade, as instituições participantes deverão garantir os limites mínimos de 500 requisições por minuto por receptora, sendo o controle via IP, e 300 transações por minuto.

Caso as requisições excedam os limites, estas requisições poderão ser enfileiradas ou rejeitadas (Uso do código HTTP 429 Too Many Requests). Não impactando os requisitos de disponibilidade.

# **Segurança**

Conforme os GT de segurança foram requeridos headers para aumentar segurança na chamada das APIs. Segue a identificação dos headers e suas respectivas funções.

Cache-Control: Controle de cache para evitar que informações confidenciais sejam armazenadas em cache.

Content-Security-Policy: Campo para proteção contra ataques clickjack do estilo - drag and drop.

Content-Type: Especificar o tipo de conteúdo da resposta.

Strict-CivilLiability-Security: Campo para exigir conexões por HTTPS e proteger contra certificados falsificados.

X-Content-Type-Options: Campo para evitar que navegadores executem a detecção de MIME e interpretem respostas como HTML de forma inadequada.

X-Frame-Options: Campo indica se o navegador deve ou não renderizar um frame.

# **Changelog**

|  |  |  |  |
| --- | --- | --- | --- |
| **CHANGELOG** | | | |
| Data | Página Alterada | Alteração (De > para) | Responsável |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |