



AGÊNCIA NACIONAL DE ÁGUAS
E SANEAMENTO BÁSICO

TUTORIAL DE SERVIÇO PARA CONSUMO DE DADOS – API HidroWebService

Sumário

1 - INSTRUÇÕES PARA AUTORIZAÇÃO DE BUSCA DE DADOS VIA API	1
2 - INSTRUÇÕES SOBRE AUTENTICAÇÃO	1
3 – INSTRUÇÕES PARA CONSULTA DE DADOS.....	6
ANEXO I - Exemplo de Consultas Manuais em Alguns dos Métodos Disponíveis no Serviço.....	9
ANEXO II - Quadro mostrando os principais códigos de retorno/erro na resposta do serviço e quais as ações relacionadas.....	11
ANEXO III - Exemplo de Requisição Automatizada em Java	12

1 - INSTRUÇÕES PARA AUTORIZAÇÃO DE BUSCA DE DADOS VIA API

1.1 – Cadastro de acesso

Os usuários que desejam acessar os dados e informações da API, de forma automatizada, devem encaminhar e-mail para hidro@ana.gov.br, com o assunto “[Preencher com seu CPF ou CNPJ] - Solicitação de acesso à API HidroWebService para consumo de dados”.

No corpo do e-mail, inclua uma breve explicação (em poucas linhas) sobre a motivação da solicitação e forneça as seguintes informações para o cadastro:

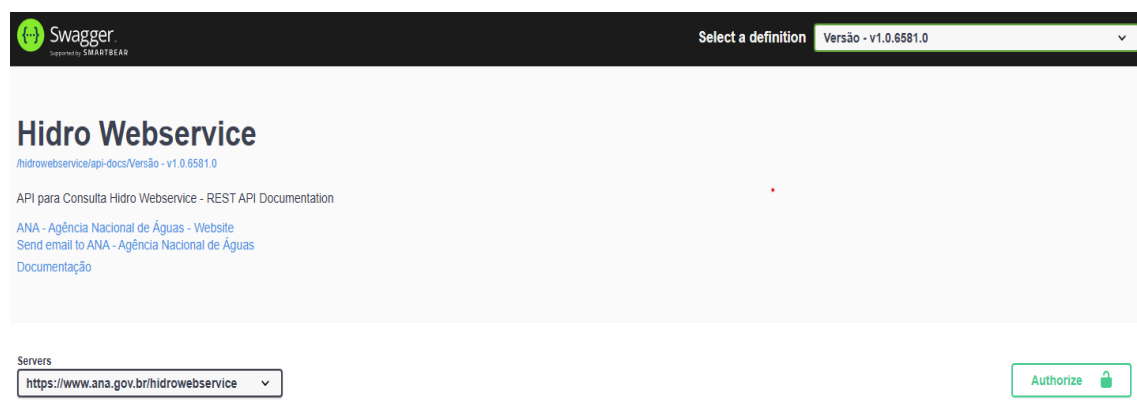
Nome completo do usuário e instituição quando for o caso.

- **CPF ou CNPJ** (que será utilizado como usuário)
- **Endereço de e-mail** (será utilizado para o recebimento da senha de acesso)

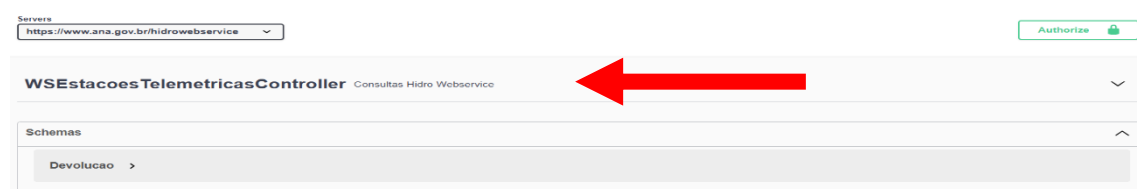
Após o recebimento do e-mail, nossa equipe irá avaliar sua solicitação e, caso todas as informações estejam corretas, o acesso à API será autorizado. Neste caso será encaminhado um e-mail informando sobre os detalhes do acesso.

2 - INSTRUÇÕES SOBRE AUTENTICAÇÃO

Acessar o link: <https://www.ana.gov.br/hidrowebservice/swagger-ui/index.html>




2.1 - Clicar em: “WS-EstacoesTelemetricasController”, para mostrar todas as rotas disponíveis.



2.2 – Visualize as rotas disponíveis para acesso e utilização.

WSEstacoesTelemetricasController Consultas Hidro Webservice			^
GET	/EstacoesTelemetricas/OAuthPermissoes/v1	Método que retorna as permissões através do accesstoken SSO	✓ 🔒
GET	/EstacoesTelemetricas/OAuth/v1	Método de autenticação do serviço, (ver manual do usuário para maiores detalhes)	✓ 🔒
GET	/EstacoesTelemetricas/HidroSatSerieDados/v1	Séries das estações virtuais (HidroSat). Deve ser informado o código de estação (consulta HidroSatInventarioEstacoes) e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSatInventarioEstacoes/v1	Inventário de estações virtuais (estimação por satélite) cadastradas na base HidroSat. Não há limitação de busca por requisição.	✓ 🔒
GET	/EstacoesTelemetricas/HidroInfoAnaSerieTelemetricaDetalhada/v1	Séries das estações telemétricas. Além dos dados adotados, são retornados também os dados brutos disponíveis. Deve ser informado o código da estação e período (limitado a 30 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroInfoAnaSerieTelemetricaAdotada/v1	Séries das estações telemétricas. Retorna os dados adotados de chuva, nível e vazão. Deve ser informado o código da estação e período (limitado a 30 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroUF/v1	Lista de unidades federativas cadastradas na base HIDRO. Não há limitação de busca por requisição.	✓ 🔒
GET	/EstacoesTelemetricas/HidroSubBacia/v1	Lista de sub-bacias hidrográficas cadastradas na base HIDRO. Não há limitação de busca por requisição.	✓ 🔒
GET	/EstacoesTelemetricas/HidroSerieVazao/v1	Séries de vazão das estações convencionais (coleta manual). Deve ser informado o código de estação e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSerieSedimentos/v1	Séries de sedimento das estações convencionais (coleta manual). Deve ser informado o código de e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSerieResumoDescarga/v1	Séries de medições de descarga líquida das estações. Deve ser informado o código de estação (consulta HidroInventarioEstacoes) e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSerieQA/v1	Séries de qualidade de água das estações convencionais (coleta manual). Deve ser informado o código de estação e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSeriePerfilTransversal/v1	Séries de medições do perfil transversal das estações. Deve ser informado o código de estação e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSerieCurvaDescarga/v1	Séries de curvas de descarga líquida traçadas para as estações. Deve ser informado o código de estação e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSerieCotas/v1	Séries de cota das estações convencionais (coleta manual). Deve ser informado o código de estação e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSerieChuva/v1	Séries de chuva das estações convencionais (coleta manual). Deve ser informado o código de estação e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroRio/v1	Lista de corpos hídricos cadastrados na base HIDRO. Não há limitação de busca por requisição.	✓ 🔒
GET	/EstacoesTelemetricas/HidroMunicipio/v1	Lista de municípios cadastrados na base HIDRO. Ressalta-se que a base Hidro usa um código diferente do IBGE. Não há limitação de busca por requisição.	✓ 🔒
GET	/EstacoesTelemetricas/HidroInventarioEstacoes/v1	Inventário completo de estações cadastradas na base Hidro. Deve ser informado, ao menos, um dos seguintes filtros: Cód. Estação, Cód. Bacia ou UF.	✓ 🔒
GET	/EstacoesTelemetricas/HidroEntidade/v1	Lista de entidades cadastradas na base Hidro. As entidades são utilizadas para caracterizar o responsável e operador das estações. Não há limitação de busca por requisição.	✓ 🔒
GET	/EstacoesTelemetricas/HidroBacia/v1	Lista de bacias hidrográficas cadastradas na base Hidro. Não há limitação de busca por requisição.	✓ 🔒

2.3 – Clicar em: **GET** [EstacoesTelemetricas/OAuth/v1](#), para acessar e visualizar os campos a serem preenchidos.

 Swagger
OPEN SOURCE, STANDARDS-BASED

Select a definition **Versão - v1.0.6581.0**

Hidro Webservice

/hidrowebservice/api-docs/Versão - v1.0.6581.0

API para Consulta Hidro Webservice - REST API Documentation

ANA - Agência Nacional de Águas - Website
Send email to ANA - Agência Nacional de Águas
Documentação

Servers
 Authorize

WSEstacoesTelemetricasController Consultas Hidro Webservice

GET	/EstacoesTelemetricas/OAuthPermissoes/v1	Método que retorna as permissões através do accesstoken SSO	✓ 🔒
GET	/EstacoesTelemetricas/OAuth/v1	Método de autenticação do serviço, (ver manual do usuário para maiores detalhes)	✓ 🔒
GET	/EstacoesTelemetricas/HidroSatSerieDados/v1	Séries das estações virtuais (HidroSat). Deve ser informado o código de estação (consulta HidroSatInventarioEstacoes) e período (limitado a 366 dias por requisição).	✓ 🔒
GET	/EstacoesTelemetricas/HidroSatInventarioEstacoes/v1	Inventário de estações virtuais (estimação por satélite) cadastradas na base HidroSat. Não há limitação de busca por requisição.	✓ 🔒

2.4 – Será mostrado um formulário para preenchimento das credenciais de acesso (identificador e senha) do usuário (ver item 1.1 para informações de como obtê-los).

GET /EstacoesTelemetricas/OAuth/v1 Método de autenticação do serviço, (ver manual do usuário para maiores detalhes)

Parameters Try it out

Name	Description
Identificador * required string (header)	IDENTIFICADOR/USUÁRIO
Senha * required string (header)	SENHA

Identificador

Senha

2.5 – Clique em “Try it out” para permitir a inserção dos parâmetros.

WSEstacoesTelemetricasController Consultas Hidro Webservice

GET /EstacoesTelemetricas/OAuthPermissoes/v1 Método que retorna as permissões através do accesstoken SSO

GET /EstacoesTelemetricas/OAuth/v1 Método de autenticação do serviço, (ver manual do usuário para maiores detalhes)

Parameters Try it out

Name	Description
Identificador * required string (header)	IDENTIFICADOR/USUÁRIO
Senha * required string (header)	SENHA

Identificador

Senha

2.6 – Após inserir os parâmetros clicar em “Execute” para submeter.

GET /EstacoesTelemetricas/OAuth/v1 Método de autenticação do serviço, (ver manual do usuário para maiores detalhes)

Parameters Cancel

Name	Description
Identificador * required string (header)	IDENTIFICADOR/USUÁRIO
Senha * required string (header)	SENHA

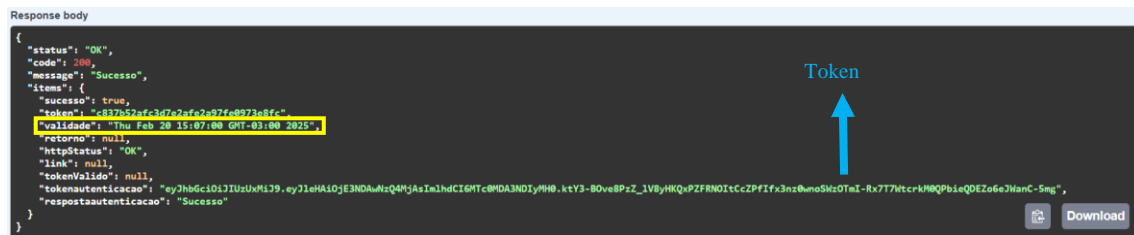
Identificador

Senha

Preencha os dois campos obrigatórios

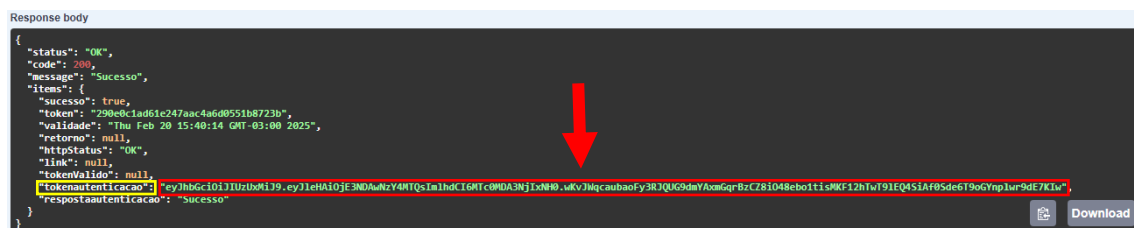
Execute

2.7 - O serviço deverá retornar as informações necessárias para autenticação (ver imagem abaixo).

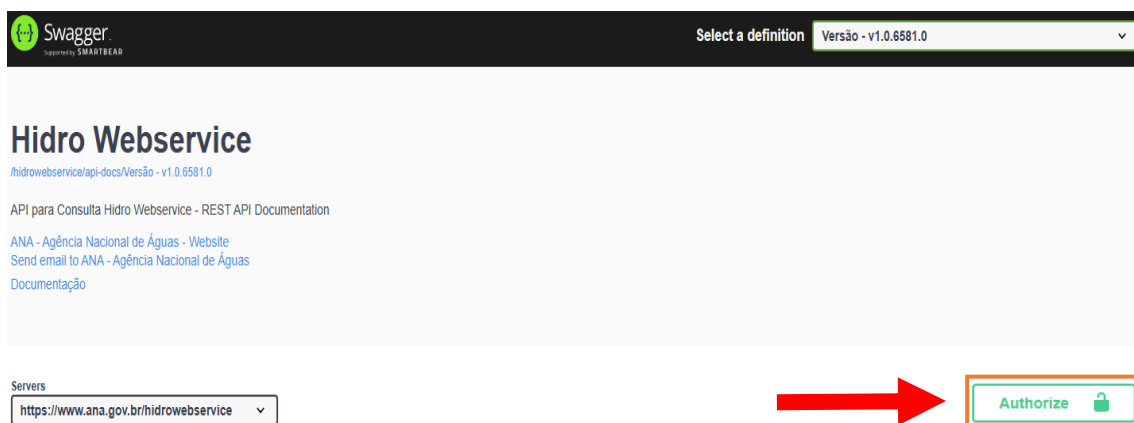


Nesta tela deve ser observado o campo “tokenautenticacao”, que terá uma validade de 15 minutos. Após este tempo um novo tokenautenticacao” deverá ser gerado.

2.8 – O conteúdo do campo 'tokenautenticacao' deverá ser copiado, sem aspas, para utilização na próxima etapa.



2.9 – Em seguida, navegar até o topo da página e clicar em “Authorize”.



2.10 – Colar o “tokenautenticacao” copiado e, em seguida, clique em “Authorize” para concluir o processo de autenticação.

Available authorizations x

Authorization (http, Bearer)

Value:

Cole o token de autenticação aqui

Authorize Close

2.11 – Após inserir o token, clicar em “Close” para finalizar o processo. A partir deste momento o serviço estará pronto para a realização das consultas e interação com os dados conforme necessário.

Available authorizations x

Authorization (http, Bearer)

Authorized

Value: *****

Logout Close

2.12 – Para gerar um novo token de autenticação clicar primeiro em “Clear”, reinserir as credenciais e clicar em “Execute”. Repetir os passos 2.5 a 2.11 para completar o processo.

GET /EstacoesTelemetricas/OAuth/v1 Método de autenticação do serviço. (ver manual do usuário para maiores detalhes)

Parameters Cancel

Name	Description
Identificador * required string (header)	IDENTIFICADOR/USUÁRIO
Senha * required string (header)	SENHA

Identificador

Senha

2º passo

1º passo

Execute Clear

3 – INSTRUÇÕES PARA CONSULTA DE DADOS

3.1 – Para realizar a consulta primeiro deve ser selecionado uma rota, como por exemplo a rota “Series Telemétricas Detalhadas”:
[/EstacoesTelemetricasController/gethidroinfoanaserietelemetricaDetalhada/v1](#)

WSEstacoesTelemetricasController Consultas Hidro Webservice		
GET	/EstacoesTelemetricas/OAuthPermissoes/v1	Método que retorna as permissões através do accesstoken SSO
GET	/EstacoesTelemetricas/OAuth/v1	Método de autenticação do serviço, (ver manual do usuário para maiores detalhes)
GET	/EstacoesTelemetricas/HidroSatSerieDados/v1	Séries das estações virtuais (HidroSat). Deve ser informado o código de estação (consulta HidroSatInventarioEstacoes) e período (limitado a 365 dias por requisição).
GET	/EstacoesTelemetricas/HidroSatInventarioEstacoes/v1	Inventário de estações virtuais (estimação por satélite) cadastradas na base HidroSat. Não há limitação de busca por requisição.
GET	/EstacoesTelemetricas/HidroInfoanaSerieTelemetricaDetalhada/v1	Séries das estações telemétricas. Além dos dados adotados, são retornados também os dados brutos disponíveis. Deve ser informado o código da estação e período (limitado a 30 dias por requisição).
GET	/EstacoesTelemetricas/HidroInfoanaSerieTelemetricaAdotada/v1	Séries das estações telemétricas. Retorna os dados adotados de chuva, nível e vazão. Deve ser informado o código da estação e período (limitado a 30 dias por requisição).

3.2 – Clicar em “Try it out” para habilitar a opção de preenchimento dos campos e iniciar o teste da funcionalidade.

GET	/EstacoesTelemetricas/HidroInfoanaSerieTelemetricaDetalhada/v1	Séries das estações telemétricas. Além dos dados adotados, são retornados também os dados brutos disponíveis. Deve ser informado o código da estação e período (limitado a 30 dias por requisição).
-----	--	---

Parameters

Name	Description
Código da Estação * required integer(\$int32) (query)	<input type="text" value="Código da Estação"/>
Tipo Filtro Data * required string (query)	Available values : DATA_LEITURA, DATA_ULTIMA_ATUALIZACAO <input type="text" value="DATA_LEITURA"/>
Data de Busca (yyyy-MM-dd) string(\$date-time) (query)	<input type="text" value="Data de Busca (yyyy-MM-dd)"/>
Range Intervalo de busca * required string (query)	Available values : MINUTO_5, MINUTO_10, MINUTO_15, MINUTO_30, HORA_1, HORA_2, HORA_3, HORA_4, HORA_5, HORA_6, HORA_7, HORA_8, HORA_9, HORA_10, HORA_11, HORA_12, HORA_13, HORA_14, HORA_15, HORA_16, HORA_17, HORA_18, HORA_19, HORA_20, HORA_21, HORA_22, HORA_23, HORA_24, DIAS_2, DIAS_7, DIAS_14, DIAS_21, DIAS_30 <input type="text" value="MINUTO_5"/>

Try it out

3.3 – Preencher os campos “Código da Estação” e “Data de Busca” para realizar a consulta. Os demais campos devem ser preenchidos conforme a necessidade.

GET /EstacoesTelemetricas/HidroinfoanaSerieTelemetricaDetalhada/v1 Séries das estações telemétricas. Além dos dados adotados, são retornados também os dados brutos disponíveis. Deve ser informado o código da estação e período (limitado a 30 dias por requisição).

Parameters Try it out

Name	Description
Código da Estação * required integer(\$int32) (query)	<input type="text" value="Código da Estação"/> → Campo obrigatório
Tipo Filtro Data * required string (query)	Available values : DATA_LEITURA, DATA_ULTIMA_ATUALIZACAO <input type="text" value="DATA_LEITURA"/>
Data de Busca (yyyy-MM-dd) string(\$date-time) (query)	<input type="text" value="Data de Busca (yyyy-MM-dd)"/> → Campo obrigatório
Range Intervalo de busca * required string (query)	Available values : MINUTO_5, MINUTO_10, MINUTO_15, MINUTO_30, HORA_1, HORA_2, HORA_3, HORA_4, HORA_5, HORA_6, HORA_7, HORA_8, HORA_9, HORA_10, HORA_11, HORA_12, HORA_13, HORA_14, HORA_15, HORA_16, HORA_17, HORA_18, HORA_19, HORA_20, HORA_21, HORA_22, HORA_23, HORA_24, DIAS_2, DIAS_7, DIAS_14, DIAS_21, DIAS_30 <input type="text" value="MINUTO_5"/>

Atenção!!! Se os campos não forem preenchidos, ou houver algum erro no seu preenchimento, não haverá retorno de registros.

3.4 – Após preencher os campos obrigatórios, clicar em “Execute” para realizar a busca e visualizar os dados da estação.

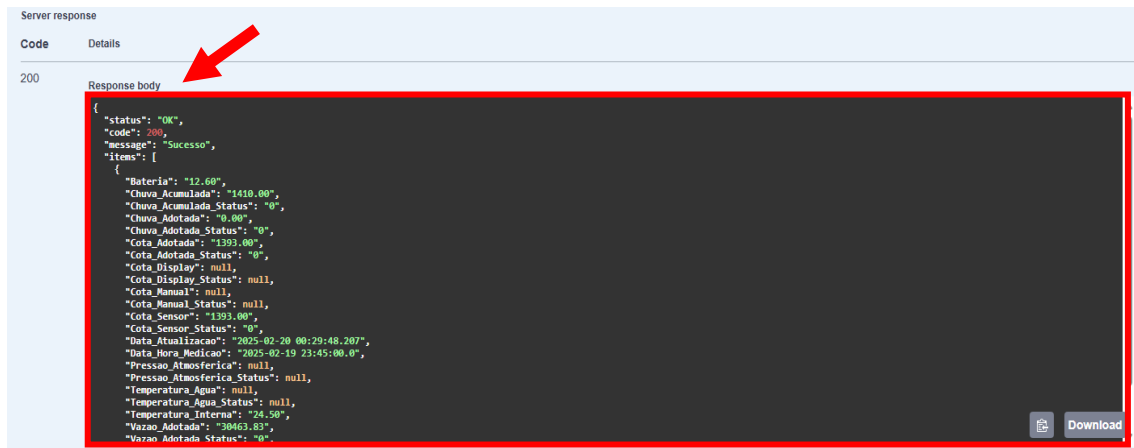
GET /EstacoesTelemetricas/HidroinfoanaSerieTelemetricaDetalhada/v1 Séries das estações telemétricas. Além dos dados adotados, são retornados também os dados brutos disponíveis. Deve ser informado o código da estação e período (limitado a 30 dias por requisição).

Parameters Cancel

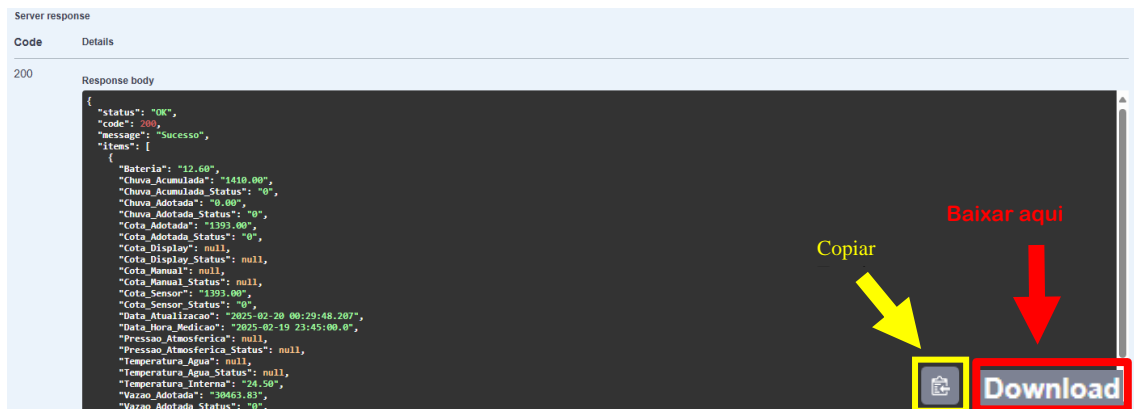
Name	Description
Código da Estação * required integer(\$int32) (query)	<input type="text" value="15400000"/>
Tipo Filtro Data * required string (query)	<input type="text" value="DATA_LEITURA"/>
Data de Busca (yyyy-MM-dd) string(\$date-time) (query)	<input type="text" value="2025-02-19"/>
Range Intervalo de busca * required string (query)	<input type="text" value="MINUTO_15"/>

Execute

3.5 – O resultado da busca será exibido na seção “Response body”, onde poderá visualizar todos os detalhes e informações da estação consultada.



3.6 - Os dados podem ser copiados ou baixados no formato JSON.



ANEXO I - Exemplo de Consultas Manuais em Alguns dos Métodos Disponíveis no Serviço.

Consultando a rota HidroinfoanaSerieTelemetricaAdotada

A seguir é apresentado um exemplo de realização de uma consulta as séries de dados das estações telemétricas, que transmitem em tempo quase-real.

Selecionar `/EstacoesTelemetricas/HidroinfoanaSerieTelemetricaAdotada/v1` e clicar no botão “Try it out”.

GET `/EstacoesTelemetricas/HidroinfoanaSerieTelemetricaAdotada/v1` Séries das estações telemétricas. Retorna os dados adotados de chuva, nível e vazão. Deve ser informado o código da estação e período (limitado a 24h por requisição).

Parameters

Name	Description
Código da Estação * required integer(\$int32) (query)	Código da Estação
Tipo Filtro Data * required string (query)	Available values : DATA_LEITURA, DATA_ULTIMA_ATUALIZACAO DATA_LEITURA
Data de Busca (yyyy-MM-dd) * required string(\$date-time) (query)	Data de Busca (yyyy-MM-dd)
Range Intervalo de busca * required string (query)	Available values : MINUTO_5, MINUTO_10, MINUTO_15, MINUTO_30, HORA_1, HORA_2, HORA_3, HORA_4, HORA_5, HORA_6, HORA_7, HORA_8, HORA_9, HORA_10, HORA_11, HORA_12, HORA_13, HORA_14, HORA_15, HORA_16, HORA_17, HORA_18, HORA_19, HORA_20, HORA_21, HORA_22, HORA_23, HORA_24 MINUTO_5

Try it out

Segue o resultado da consulta. Em **amarelo** é mostrado o significado dos campos.

```
{
  "status": "OK",
  "code": 200,
  "message": "Sucesso",
  "items": [
    {
      "Chuva_Adotada": "0.00", [Precipitação (mm)]
      "Chuva_Adotada_Status": "0", [Precipitação QC (0 = ok, 1 = suspeito, 2 = ruim)]
      "Cota_Adotada": "781.00", [Cota (cm)]
      "Cota_Adotada_Status": "0", [Cota QC (0 = ok, 1 = suspeito, 2 = ruim)]
      "Data_Atualizacao": "2024-01-02 00:28:03.307", [DataHora da atualização do dado na base]
      "Data_Hora_Medicao": "2024-01-01 23:00:00.0", [DataHora da medição/coleta do dado]
      "Vazao_Adotada": "13225.42", [Vazão (m3/s)]
      "Vazao_Adotada_Status": "0", [Vazão QC (0 = ok, 1 = suspeito, 2 = ruim)]
      "codigoestacao": "15400000" [Código da Estação]
    },
    {
      "Chuva_Adotada": "0.00",
      "Chuva_Adotada_Status": "0",
      "Cota_Adotada": "781.00",
      "Cota_Adotada_Status": "0",
      "Data_Atualizacao": "2024-01-02 00:28:03.317",
      "Data_Hora_Medicao": "2024-01-01 23:15:00.0",
      "Vazao_Adotada": "13225.42",
      "Vazao_Adotada_Status": "0",
      "codigoestacao": "15400000"
    }
  ]
}
```

Consultando a rota *HidroInventarioEstacoes*

Selecione a rota `/EstacoesTelemetricas/HidroInventarioEstacoes/v1` e clique no botão “Try it out”.

Name	Description
Código da Estação <small>integer(\$int32) (query)</small>	Código da Estação
Data Atualização Inicial (yyyy-MM-dd) <small>string(\$date-time) (query)</small>	Data Atualização Inicial (yyyy-MM-dd)
Data Atualização Final (yyyy-MM-dd) <small>string(\$date-time) (query)</small>	Data Atualização Final (yyyy-MM-dd)
Unidade Federativa <small>string (query)</small>	Available values : AC, AL, AM, AP, BA, CE, DF, ES, GO, MA, MT, MS, MG, PA, PB, PR, PE, PI, RJ, RN, RS, RO, RR, SC, SP, SE, TO
Código da Bacia <small>integer(\$int32) (query)</small>	Código da Bacia

Segue o resultado da consulta. Em **amarelo** é mostrado o significado dos campos.

```
{
  "status": "OK",
  "code": 200,
  "message": "Sucesso",
  "items": [
    {
      "Altitude": "42.88",
      "Area_Drenagem": "976000.0",
      "Bacia_Nome": "RIO AMAZONAS",
      "Codigo_Adicional": "ANA",
      "Codigo_Operadora_Unidade_UF": "1",
      "Data_Periodo_Climatologica_Fim": null,
      "Data_Periodo_Climatologica_Inicio": null,
      "Data_Periodo_Desc_Liquida_Fim": null,
      "Data_Periodo_Desc_liquida_Inicio": "1964-01-01 00:00:00.0",
      "Data_Periodo_Telemetrica_Inicio": "2001-06-01 00:00:00.0",
      ...
      "Data_Ultima_Atualizacao": "2023-12-19 00:00:00.0 [DataHora da atualização na base]
      "Estacao_Nome": "PORTO VELHO", [Nome da Estação]
      "Latitude": "-8.7483", [Latitude da Estação]
      "Longitude": "-63.9169", [Longitude da Estação]
      "Municipio_Codigo": "1010000", [Código hidro do município]
      "Municipio_Nome": "PORTO VELHO", [Nome do município]
      "Operadora_Codigo": "82", [Código da operadora da estação]
      "Operadora_Sigla": "CPRM", [Entidade que Opera (faz a manutenção) da estação]
      "Responsavel_Sigla": "ANA", [Entidade Responsável pela Estação]
      "UF_Estacao": "RO", [UF da Estação]
      "UF_Nome_Estacao": "RONDÔNIA", [Nome da Estação]
      "codigobacia": "1", [Código da bacia hidrográfica (1 – 9)]
      "codigoestacao": "15400000 [Nome da Estação]
      "Operando": "1", [Ativo (SIM = 1)]
      "Tipo_Estacao": "Fluviometrica", [Tipo da Estação (Fluviométrica, Pluviométrica)]
    }
  ]
}
```

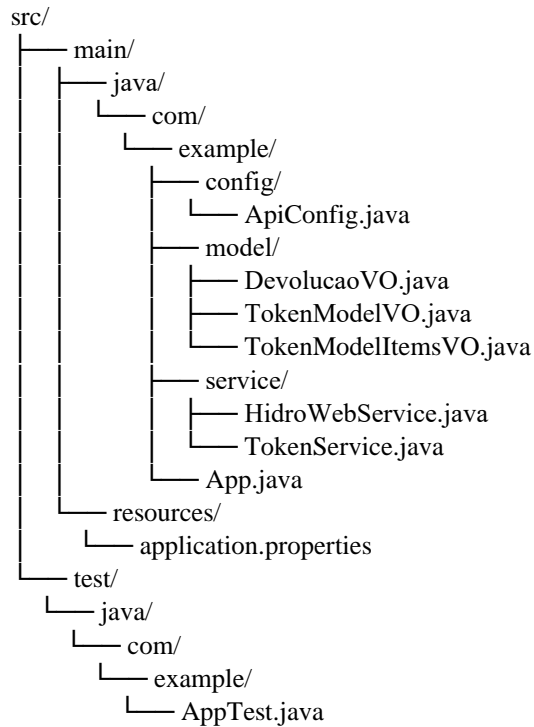
ANEXO II - Quadro mostrando os principais códigos de retorno/erro na resposta do serviço e quais as ações relacionadas.

Código de resposta	Erro	Descrição	Exemplo
200	Sucesso	Sua requisição foi processada com sucesso e os dados podem ser visualizados.	<pre>{ "status": "OK", "code": 200, "message": "Sucesso", }</pre>
400	Requisição Inválida	O servidor não consegue ou não irá processar a solicitação devido a um erro do lado do cliente (por exemplo, sintaxe de solicitação incorreta, estrutura de mensagem inválida, dados inválidos).	<pre>{ "timestamp": "2025-03-06T14:40:51.194+00:00", "status": 400, "error": "Bad Request", "path": "/api/solicitacao" }</pre>
401	Não Autorizado	Significa que você não tem permissão para acessar algo porque não fez login ou as informações que forneceu estão erradas. É necessária autorização para o acessar o recurso solicitado.	<pre>{ "timestamp": "2025-03-06T14:40:51.194+00:00", "status": 401, "error": "Unauthorized", "message": "Credenciais inválidas ou ausência de autenticação.", "path": "/api/recurso-protetido" }</pre>
500	Erro Interno do Servidor	Significa que algo deu errado no servidor ao tentar processar a sua solicitação, mas o servidor não sabe exatamente o que foi o problema.	<pre>{ "timestamp": "2025-03-06T14:40:51.194+00:00", "status": 500, "error": "Internal Server Error", "message": "Ocorreu um erro inesperado no servidor.", "path": "/api/processamento" }</pre>

Se, mesmo seguindo todos os passos corretamente, as mensagens de erro acima forem exibidas, verifique os dados inseridos e repita o processo. Caso o problema persista, envie um e-mail para **hidro@ana.gov.br** relatando seu erro e com o título “[Seu CPF ou CNPJ] - Solicitação de suporte API HIDRO WEBSERVICE.

ANEXO III - Exemplo de Requisição Automatizada em Java

Estrutura de pastas



Classes e Conteúdo

1. *ApiConfig.java* - Responsável por manter configurações como URLs de API e outros parâmetros.

```
package com.example.config;

public class ApiConfig {
    public static final String HIDRO_WEBSERVICE_URL =
        "https://www.ana.gov.br/hidrowebservice/EstacoesTelemetricas";
}
```

2. *DevolucaoVO.java* - Modelo de resposta.

```
package com.example.model;

import org.springframework.http.HttpStatus;

public class DevolucaoVO {
    private HttpStatus status;
    private Integer code;
    private String message;
    private Object items;

    // Getters e Setters
}
```

3. TokenModelVO.java - Modelo de token.

```
package com.example.model;

public class TokenModelVO {
    private String status;
    private int code;
    private String message;
    private TokenModelItemsVO items;

    // Getters e Setters
}
```

4. TokenModelItemsVO.java - Modelo para itens do token.

```
package com.example.model;

public class TokenModelItemsVO {
    private String sucesso;
    private String token;
    private String validade;
    private String retorno;
    private String httpStatus;
    private String link;
    private String tokenValido;
    private String tokenautenticacao;
    private String respostaautenticacao;

    // Getters e Setters
}
```

5. TokenService.java - Serviço para manipulação de tokens.

```
package com.example.service;

import com.example.config.ApiConfig;
import com.example.model.TokenModelVO;
import com.google.gson.Gson;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;

public class TokenService {
    public TokenModelVO getToken(String identificador, String senha) {
        try {
            var obj = new URL(ApiConfig.HIDRO_WEBSERVICE_URL + "/OAuth/v1");
            HttpURLConnection con = (HttpURLConnection) obj.openConnection();
            con.setRequestMethod("GET");
            con.setRequestProperty("Identificador", identificador);
            con.setRequestProperty("Senha", senha);

            var in = new BufferedReader(new InputStreamReader(con.getInputStream()));
            var response = new StringBuilder();
            String inputLine;
            while ((inputLine = in.readLine()) != null) {
                response.append(inputLine);
            }
            in.close();
        }
    }
}
```

```

        return new Gson().fromJson(response.toString(), TokenModelVO.class);
    } catch (Exception e) {
        return new TokenModelVO();
    }
}
}

```

6. HidroWebService.java - Serviço principal para consumir os dados.

```

package com.example.service;

import com.example.config.ApiConfig;
import com.example.model.DevolucaoVO;
import com.google.gson.Gson;
import org.apache.http.HttpHeaders;
import org.apache.http.client.methods.CloseableHttpResponse;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.impl.client.CloseableHttpClient;
import org.apache.http.impl.client.HttpClients;
import org.apache.http.util.EntityUtils;

import java.util.concurrent.CountDownLatch;
import java.util.concurrent.ExecutorService;
import java.util.concurrent.Executors;
import java.util.concurrent.TimeUnit;

public class HidroWebService {
    private final TokenService tokenService = new TokenService();

    public void orchestra(String[] estacoes, int threadCount, String identificador, String senha) {
        if (estacoes == null || estacoes.length == 0) {
            System.out.println("Nenhuma estação para processar.");
            return;
        }

        ExecutorService executor = Executors.newFixedThreadPool(threadCount);
        var token = tokenService.getToken(identificador, senha);

        if (token != null && token.getItems().getTokenAutenticacao() != null) {
            Instant tokenCreationTime = token.getCreationTime();
            CountDownLatch latch = new CountDownLatch(estacoes.length);

            for (String estacao : estacoes) {
                executor.submit(() -> {
                    try {
                        if (isTokenValid(tokenCreationTime)) {
                            DevolucaoVO response = executeRoute(token.getItems().getTokenAutenticacao(),
                                estacao);
                            System.out.println("Response: " + response);
                        } else {
                            System.out.println("Token expirado. Interrompendo execução.");
                        }
                    } catch (Exception e) {
                        System.out.println("Erro ao processar: " + e.getMessage());
                    } finally {
                        latch.countDown();
                    }
                });
            }
        }
    }
}

```



```

    try {
        latch.await();
    } catch (InterruptedException e) {
        Thread.currentThread().interrupt();
    } finally {
        executor.shutdown();
        try {
            if (!executor.awaitTermination(120, TimeUnit.SECONDS)) {
                executor.shutdownNow();
            }
        } catch (InterruptedException e) {
            executor.shutdownNow();
            Thread.currentThread().interrupt();
        }
    }
} else {
    System.out.println("Token inválido ou ausente.");
}
}

private boolean isValidToken(Instant tokenCreationTime) {
    // Verifica se o token foi criado há menos de 15 minutos
    return Instant.now().isBefore(tokenCreationTime.plus(15, ChronoUnit.MINUTES));
}

private DevolucaoVO executeRoute(String token, String codigoEstacao) {
    var gson = new Gson();
    var devolucaoVO = new DevolucaoVO();
    var url = ApiConfig.HIDRO_WEBSERVICE_URL +
        "/HidroinfoanaSerieTelemetricaAdotada/v1?CodigoDaEstacao=" +
        codigoEstacao +
        "&TipoFiltroData=DATA_LEITURA&RangeIntervaloDeBusca=DIAS_30";

    try (CloseableHttpClient httpClient = HttpClients.createDefault()) {
        HttpGet httpGet = new HttpGet(url);
        httpGet.setHeader(HttpHeaders.AUTHORIZATION, "Bearer " + token);

        try (CloseableHttpResponse response = httpClient.execute(httpGet)) {
            int statusCode = response.getStatusLine().getStatusCode();
            if (statusCode == 200) {
                if (response.getEntity() != null) {
                    return gson.fromJson(EntityUtils.toString(response.getEntity()), DevolucaoVO.class);
                }
            }
        }
    } catch (Exception e) {
        System.out.println("Erro na requisição: " + e.getMessage());
    }
    return devolucaoVO;
}
}

```

7. App.java - Classe principal para executar a aplicação.

```

package com.example;

import com.example.service.HidroWebService;

public class App {

```

```
public static void main(String[] args) {  
    HidroWebService service = new HidroWebService();  
    String[] estacoes = {"123", "456", "789"};  
    service.orquestra(estacoes, 5, "CNPJ", "SENHA");  
}  
}
```

8. *application.properties* - Arquivo para parâmetros configuráveis (caso necessário no futuro).

Configuração da URL da API

hidro.webservice.url=https://www.ana.gov.br/hidrowebservice/EstacoesTelemetricas

9. *Teste Unitário*

Adicione testes em AppTest.java para garantir o funcionamento.

Pronto!