



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/hidrowebserice/swagger-ui/index.html>

The screenshot shows the Swagger UI interface for the Hidro Webservice API. At the top, there's a navigation bar with the 'Swagger' logo and a dropdown menu 'Select a definition' set to 'Versão - v1.0.6581.0'. Below the header, the title 'Hidro Webservice' is displayed, along with the URL '/hidrowebserice/api-docs/Versão - v1.0.6581.0'. A brief description follows: 'API para Consulta Hidro Webservice - REST API Documentation'. It also includes links to 'ANA - Agência Nacional de Águas - Website', 'Send email to ANA - Agência Nacional de Águas', and 'Documentação'. On the right side of the main content area, there are buttons for 'Servers' (set to 'https://www.ana.gov.br/hidrowebserice') and 'Authorize' (with a lock icon). A red arrow points to the 'WS-EstacoesTelemetricasController' section in the left sidebar.

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

This screenshot shows the 'WS-EstacoesTelemetricasController' section of the Swagger UI. The left sidebar lists 'Schemas' and 'Devolucao'. The main content area shows the controller's documentation with a red arrow pointing to it. At the bottom right, there are 'Authorize' and 'Logout' buttons.

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 estação 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 hidrícos 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. Powered by SMARTBEAR

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

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

Try it out

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

WSEstacoesTelemetricasController Consultas Hidro Webservice

GET /EstacoesTelemetricas/OAuthPermissões/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

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

Try it out

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

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

Preencha os dois campos obrigatórios

Execute

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

The screenshot shows a JSON response body with a 'Token' field highlighted by a blue arrow pointing upwards. The JSON structure is as follows:

```
{  
  "status": "OK",  
  "code": 200,  
  "message": "Sucesso",  
  "items": {  
    "sucesso": true,  
    "token": "e837b52aefc3d7e2aef2a97fe0973e08fc",  
    "validade": "Thu Feb 20 15:07:00 GMT-03:00 2025",  
    "retorno": null,  
    "httpStatus": "OK",  
    "link": null,  
    "tokenValido": null,  
    "tokenautenticacao": "eyJhbGciOiJIUzIwMiJ9.eyJleHAiOiJE3NDAwNzQ4MjAsImhdCI6MTc0MDA3NDIyMjY0.LtY3-B0ve8Pz_1VByHKQxPZFRNOItCcZPfifx3nz0moShzOTmI-Rx7T7HtcrkM0QPbieQDEz6eJWnC-5mg",  
    "respostaautenticacao": "Sucesso"  
}
```

Download

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

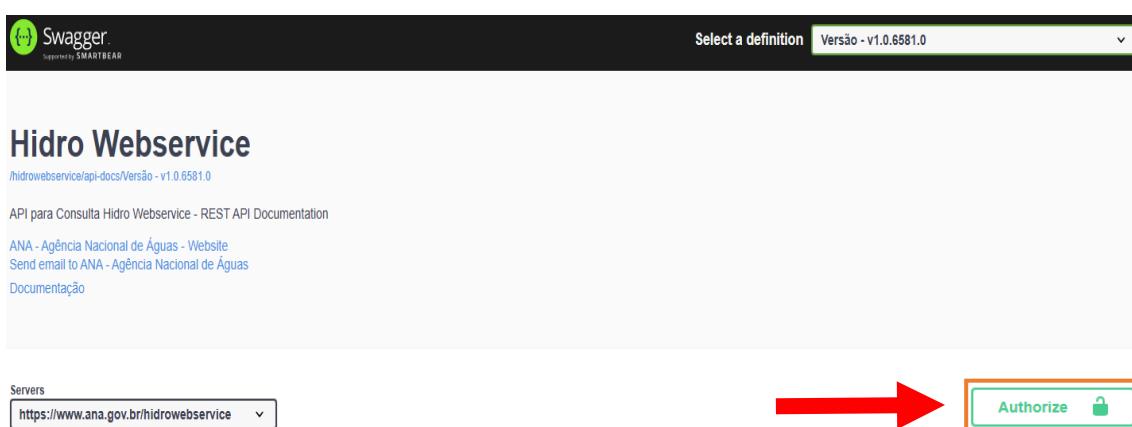
2.8 – O conteúdo do campo 'tokenautenticacao' deverá ser copiado, sem aspas, para utilização na próxima etapa.

The screenshot shows a JSON response body with the 'tokenautenticacao' field highlighted by a red arrow pointing downwards. The JSON structure is identical to the one in the previous screenshot.

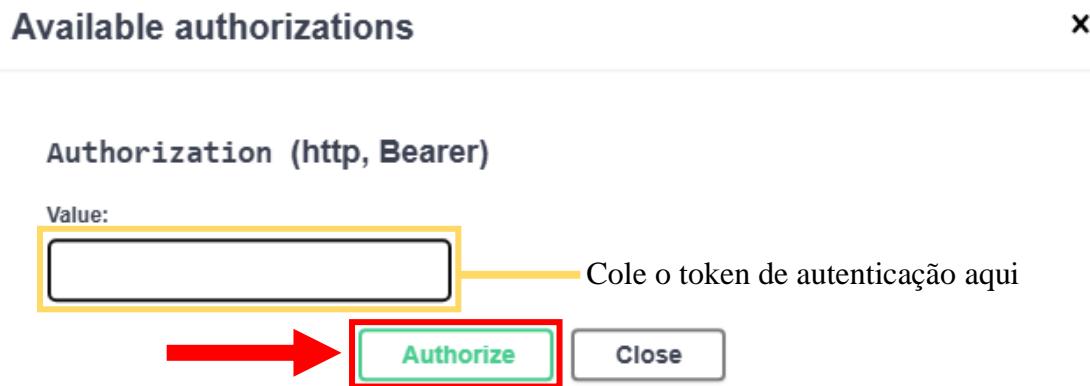
```
{  
  "status": "OK",  
  "code": 200,  
  "message": "Sucesso",  
  "items": {  
    "sucesso": true,  
    "token": "e837b52aefc3d7e2aef2a97fe0973e08fc",  
    "validade": "Thu Feb 20 15:40:14 GMT-03:00 2025",  
    "retorno": null,  
    "httpStatus": "OK",  
    "link": null,  
    "tokenValido": null,  
    "tokenautenticacao": "eyJhbGciOiJIUzIwMiJ9.eyJleHAiOiJE3NDAwNzQ4MjAsImhdCI6MTc0MDA3NDIyMjY0.LtY3-B0ve8Pz_1VByHKQxPZFRNOItCcZPfifx3nz0moShzOTmI-Rx7T7HtcrkM0QPbieQDEz6eJWnC-5mg",  
    "respostaautenticacao": "Sucesso"  
}
```

Download

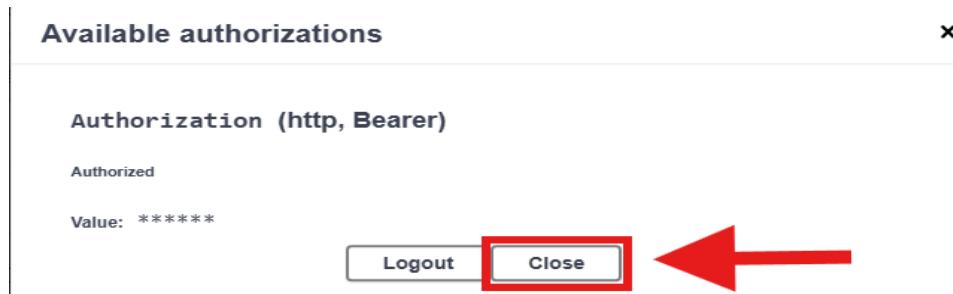
2.9 – Em seguida, navegar até o topo da página e clicar em “Authorize”.



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



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.



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

The screenshot shows a dialog for "EstacoesTelemetricas/OAuth/v1". At the top, it says "GET /EstacoesTelemetricas/OAuth/v1 Método de autenticação do serviço, (ver manual do usuário para maiores detalhes)". Below that is a "Parameters" section with two fields: "Identificador * required" and "Senha * required". Each field has a corresponding input box below it. At the bottom are two buttons: "Execute" (blue) and "Clear" (grey). Two red arrows point downwards from the text "1º passo" and "2º passo" to the "Clear" and "Execute" buttons respectively. The "Clear" button is highlighted with a red border.

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/OAuthPermisssoes/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 da estação (consulta HidroSatInventarioEstacoes) e período (ilimitado 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 (ilimitado 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 (ilimitado 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 (ilimitado a 30 dias por requisição).

Parameters

Name	Description
Código da Estação * required	Código da Estação integer(\$int32) (query)
Tipo Filtro Data * required	Available values : DATA_LEITURA, DATA_ULTIMA_ATUALIZACAO string (query)
Data de Busca (yyyy-MM-dd)	Data de Busca (yyyy-MM-dd) string(date-time) (query)
Range Intervalo de busca * required	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 string (query)

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

Name	Description
Código da Estação * required	Código da Estação
Tipo Filtro Data * required	Available values : DATA_LEITURA, DATA_ULTIMA_ATUALIZACAO
Data	DATA_LEITURA
Data de Busca (yyyy-MM-dd)	Data de Busca (yyyy-MM-dd)
Range Intervalo de busca * required	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
Intervalo	MINUTO_5

Try it out

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

Name	Description
Código da Estação * required	15400000
Tipo Filtro Data * required	DATA_LEITURA
Data de Busca (yyyy-MM-dd)	2025-02-19
Range Intervalo de busca * required	MINUTO_15

Cancel

Execute

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

Server response

Code Details

200 Response body

```
{ "status": "OK", "code": 200, "message": "Sucesso", "items": [ { "Bateria": "12.60", "Chuva_Accumulada": "1410.00", "Chuva_Accumulada_Status": "0", "Chuva_Adotada": "0.00", "Cota_Adotada_Status": "0", "Cota_Adotada": "1393.00", "Cota_Display_Status": "0", "Cota_Display": null, "Cota_Manual_Status": null, "Cota_Sensor_Status": null, "Cota_Sensor": "1393.00", "Cota_Sensor_Status": "0", "Data_Atualizacao": "2025-02-28 00:29:48.207", "Data_Hora_Medicao": "2025-02-19 23:45:00.0", "Pressao_Atmosferica": null, "Pressao_Atmosferica_Status": null, "Temperatura_Agua_Status": null, "Temperatura_Agua": null, "Temperatura_Interior_Status": null, "Temperatura_Interior": "24.50", "Vazao_Adotada": "30463.83", "Vazao_Adotada_Status": "0" } ] }
```

Download

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

Server response

Code Details

200 Response body

```
{ "status": "OK", "code": 200, "message": "Sucesso", "items": [ { "Bateria": "12.60", "Chuva_Accumulada": "1410.00", "Chuva_Accumulada_Status": "0", "Chuva_Adotada": "0.00", "Cota_Adotada_Status": "0", "Cota_Adotada": "1393.00", "Cota_Display_Status": "0", "Cota_Display": null, "Cota_Manual_Status": null, "Cota_Sensor_Status": null, "Cota_Sensor": "1393.00", "Cota_Sensor_Status": "0", "Data_Atualizacao": "2025-02-28 00:29:48.207", "Data_Hora_Medicao": "2025-02-19 23:45:00.0", "Pressao_Atmosferica": null, "Pressao_Atmosferica_Status": null, "Temperatura_Agua_Status": null, "Temperatura_Interior_Status": null, "Temperatura_Interior": "24.50", "Vazao_Adotada": "30463.83", "Vazao_Adotada_Status": "0" } ] }
```

Copiar

Baixar aqui

Download

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

Código da Estação * required Código da Estação integer(\$int32) (query)

Tipo Filtro Data * required Available values : DATA_LEITURA, DATA_ULTIMA_ATUALIZACAO string (query) DATA_LEITURA

Data de Busca (yyyy-MM-dd) * required Data de Busca (yyyy-MM-dd) string(\$date-time) (query)

Range Intervalo de busca * required 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 string (query) 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", [Data/Hora da atualização do dado na base]
      "Data_Hora_Medicao": "2024-01-01 23:00:00.0", [Data/Hora 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”.

The screenshot shows the configuration interface for a GET request to the endpoint /EstacoesTelemetricas/HidroInventarioEstacoes/v1. The interface includes fields for parameters such as Código da Estação, Data Atualização Inicial, Data Atualização Final, Unidade Federativa, and Código da Bacia. A note at the top right indicates that the inventory is complete and filters like Código da Estação or Bacia/UF are required. A 'Try it out' button is visible in the top right corner.

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_Perodo_Climatologica_Fim": null,  
      "Data_Perodo_Climatologica_Inicio": null,  
      "Data_Perodo_Desc_Liquida_Fim": null,  
      "Data_Perodo_Desc_liquida_Inicio": "1964-01-01 00:00:00.0",  
      "Data_Perodo_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)]  
    }  
  ]  
}
```

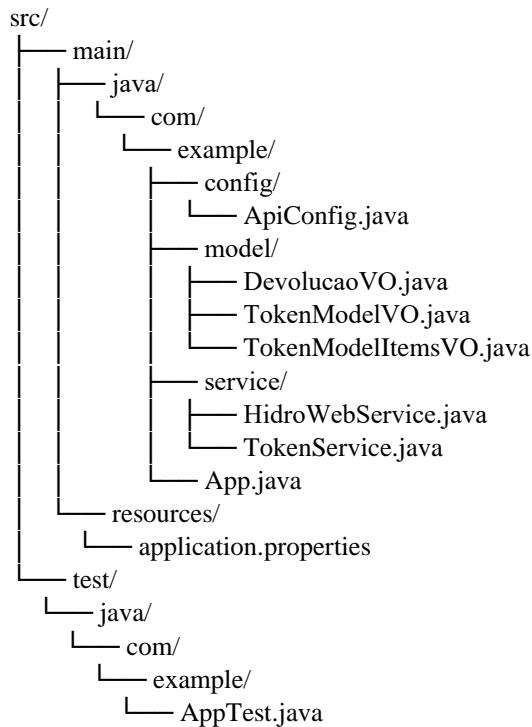
ANEXO II - Quadro mostrando os principais códigos de retorno/erro na resposta do serviço e quais as ações relacionadas.

Código de resposta	Erro	Descrição	Exemplo
200	Sucesso	Sua requisição foi processada com sucesso e os dados podem ser visualizados.	{ "status": "OK", "code": 200, "message": "Sucesso", ... }
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).	{ "timestamp": "2025-03-06T14:40:51.194+00:00", "status": 400, "error": "Bad Request", "path": "/api/solicitacao" }
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.	{ "timestamp": "2025-03-06T14:40:51.194+00:00", "status": 401, "error": "Unauthorized", "message": "Credenciais inválidas ou ausência de autenticação.", "path": "/api/recurso-protégido" }
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.	{ "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" }

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/hidrowebbservice/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 orquestra(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 isTokenValid(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!