

# Como usar a API Pipo Saúde

## Iniciando a aplicação

A aplicação se encontra nesse [repositório](#) do GitHub.

Baixando o projeto do GitHub, você será capaz de navegar pelo código e executá-lo.

Existem duas maneiras, pela IDE ou pelo .jar da aplicação.

Para gerar o .jar da aplicação você deve rodar o comando:

```
mvn clean install
```

Na raiz do projeto. E então o .jar estará no diretório: [target/pipo-2.0.0.jar](#)

Para rodar o .jar você deve executar o seguinte comando dentro do diretório do .jar

```
java -jar pipo-2.0.0.jar
```

Feito isso, você terá sua aplicação rodando no endereço:

```
http://localhost:8080
```

## Se autenticar na API

O primeiro passo é você conseguir o Bearer token que será o que habilitará acesso às request.

Para conseguir seu token de acesso, você deve fazer um request de método POST:

```
curl -v --request POST \
  --url 'http://localhost:8080/api/authenticate?username=user&password=pass'
```

Passando seu usuário e senha nos campos destacados.

Dentro dessa aplicação, para facilitar o teste, deixei como padrão a autenticação usando os valores:

```
username=user password=pass
```

A resposta desse curl será assim:

```
* Trying ::1...
* TCP_NODELAY set
* Connected to localhost (::1) port 8080 (#0)
> POST /api/authenticate?username=user&password=pass HTTP/1.1
> Host: localhost:8080
> User-Agent: curl/7.58.0
> Accept: */*
>
< HTTP/1.1 200
< Vary: Origin
< Vary: Access-Control-Request-Method
< Vary: Access-Control-Request-Headers
< Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW51cmUtYXBpIiwiaXVkiOiJoic2VjdXJlLWFWcCIsInN1YiI6InVzZXIiLCJleHAiOiJ
< X-Content-Type-Options: nosniff
< X-XSS-Protection: 1; mode=block
```

```
< Cache-Control: no-cache, no-store, max-age=0, must-revalidate
< Pragma: no-cache
< Expires: 0
< X-Frame-Options: DENY
< Content-Length: 0
< Date: Sat, 26 Mar 2022 02:44:21 GMT
<
* Connection #0 to host localhost left intact
```

O token que iremos usar está destacado em vermelho na resposta. Olhando para o campo Authorization temos o token:

```
Bearer eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzUxMiJ9.eyJpc3MiOiJzZW51cmUtYXBpIiwiaXVkiOiJ0eXVjdXJLLWFwcCI6InN1YiI6InVzZXIiLCJleHAiOiJlE2NDgyNjMyNjEsInJ
```

É importante destacar que a duração desse token é de 10 minutos, passando disso deve-se solicitar novamente um token através dessa request.

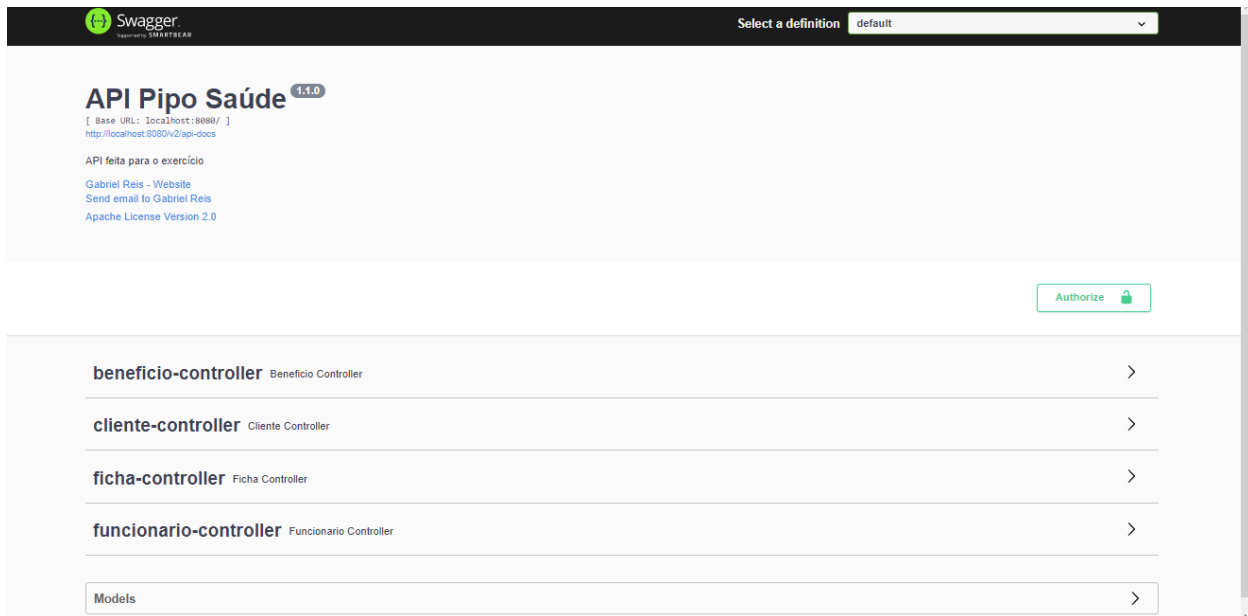
## Utilizando a API

Para utilizar a API agora que já temos o token, podemos utilizar o Swagger.

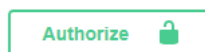
O swagger pode ser acessado após iniciarmos a aplicação Java no seguinte endereço:

```
http://localhost:8080/swagger-ui/index.html#/
```

Exemplo da tela do Swagger:



Veja que temos várias request disponíveis no swagger, para realizar as requests devemos clicar no botão :



E então irá aparecer esse pop-up para realizar a autorização usando o token:

**Available authorizations** ✕

**Bearer token (apiKey)**  
Name: Authorization  
In: header  
Value:  

Bearer eyJ0eXAiOiJKV1QiLC

Authorize

Close

Após clicar em "Authorize", você poderá realizar qualquer request da API:

beneficio-controller <small>Beneficio Controller</small>			▼
GET	/api/private/beneficio	Retorna todos os beneficios	🔒
POST	/api/private/beneficio	Cria um Beneficio	🔒
GET	/api/private/beneficio/{id}	Busca um beneficio pelo seu id	🔒
PUT	/api/private/beneficio/{id}	Atualiza um beneficio por id	🔒
DELETE	/api/private/beneficio/{id}	Deleta Beneficio por id	🔒

Além disso, também é disponibilizado uma collection para ser importada no Postman ou outra aplicação que execute requests http. A collection é disponibilizada após iniciar a aplicação, no endereço:

```
http://localhost:8080/v2/api-docs
```