# Tutorial Swagger

Equipe: Felipe Braga, Raquel Moreira, Artur Alves, Matheus Motta e Caio Daniel.

## Introdução -

Olá, tudo bem? Este é um tutorial de como se deve usar o swagger editor para documentar uma API e uma breve documentação da documentação que nossa equipe fez de uma API.

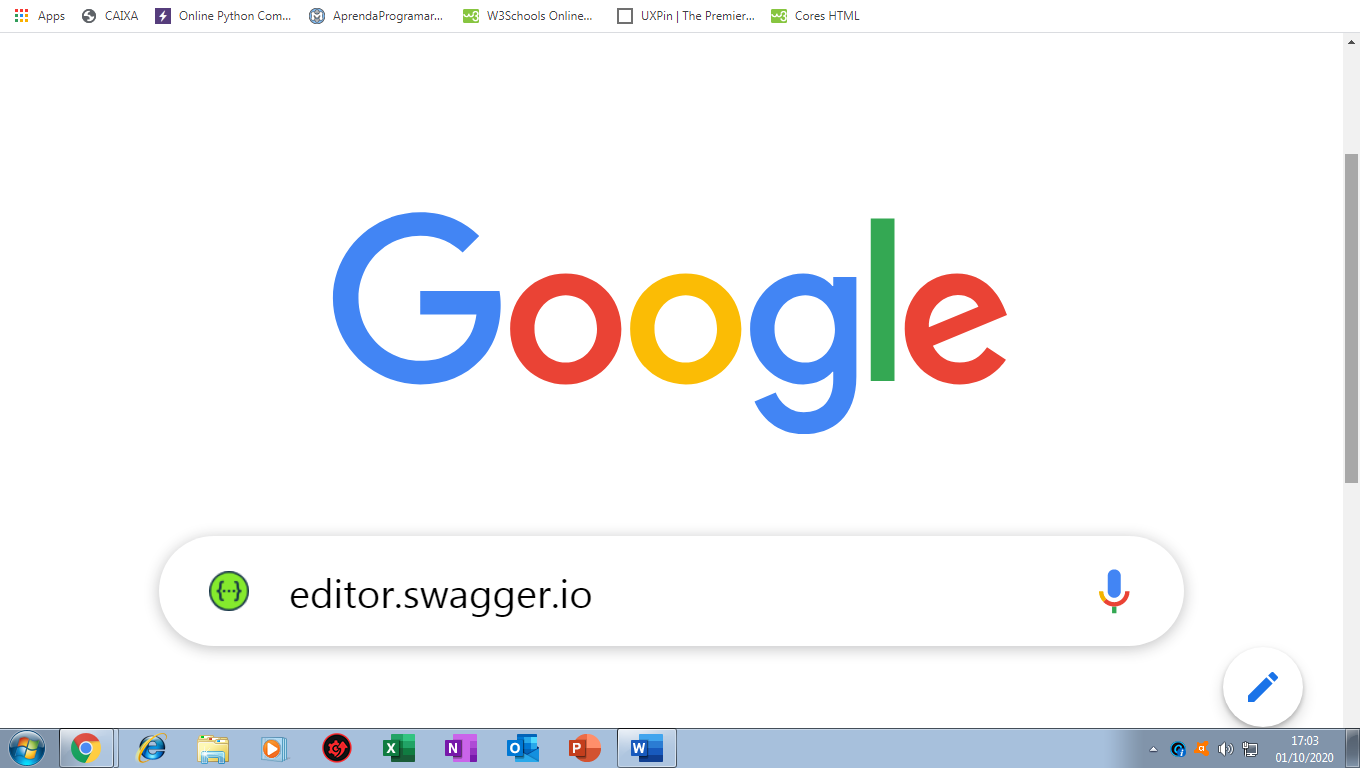
Neste tutorial lhe será apresentado o passo a passo para você documentar sua API. Você verá diversas imagens de exemplo para conseguir acompanhar todo o processo.

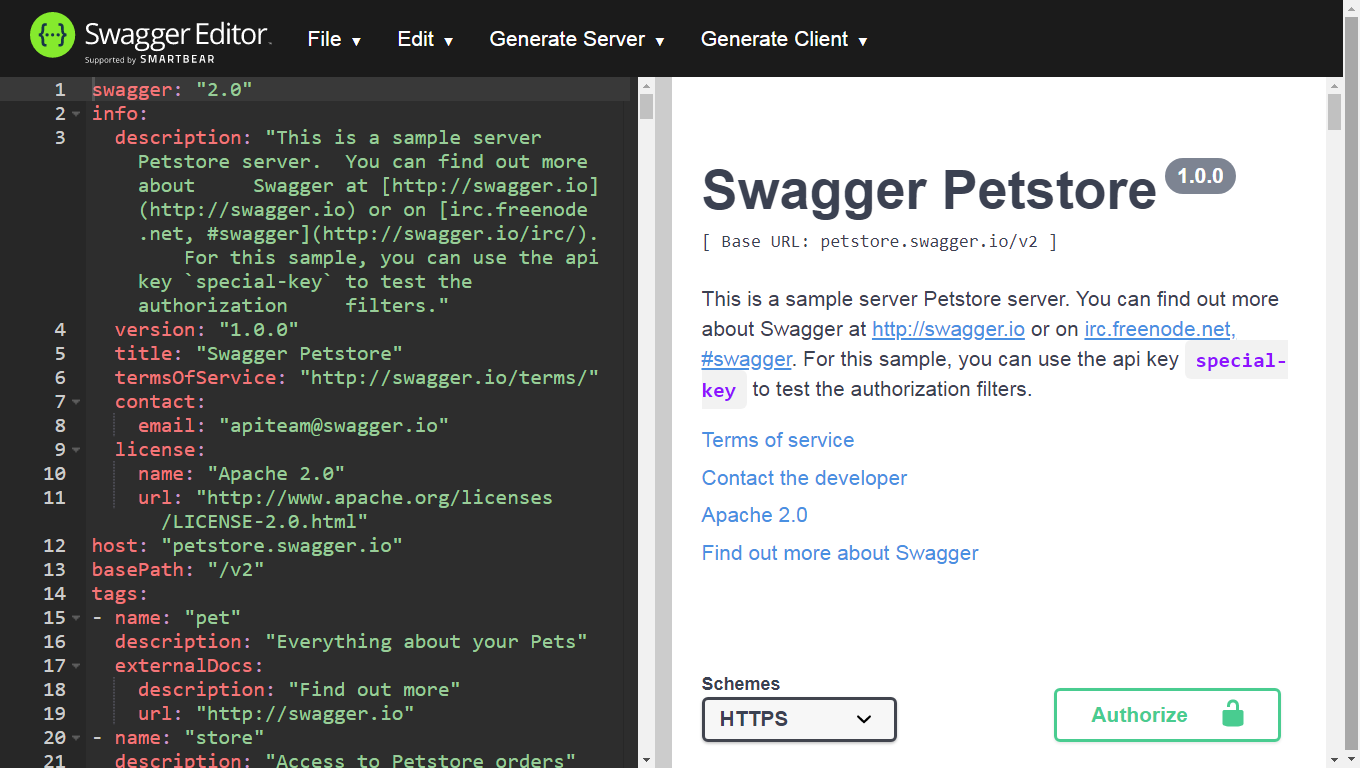
Aviso importante -

Neste tutorial não será necessário a instalação de nenhum recurso para o acompanhamento do tutorial a seguir. Mas, caso você deseje ter a ferramenta em sua máquina, também é possível.

## Primeiro-passo

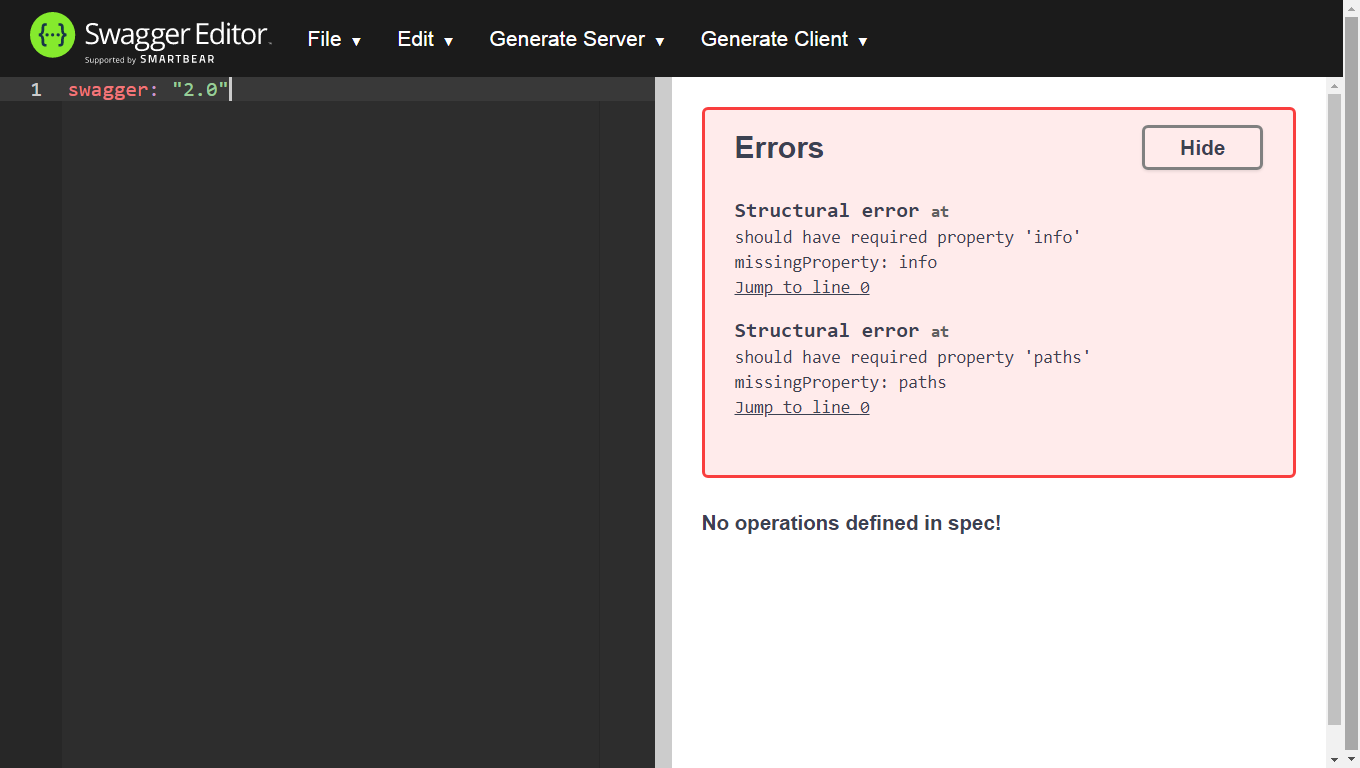
* Primeira coisa que vamos fazer é acessar o “Swagger editor”, então vamos acessar qualquer navegador e inserir o caminho “editor.swagger.io” na barra de navegação, reparem que já de cara ele mostra para gente um exemplo de Swagger, o “Petstore”, do lado esquerdo da tela nós temos um script Swagger onde fala de versão, info, várias coisas e aqui do lado direito nós temos a documentação interativa que é gerada a partir desse script que nós estamos usando.





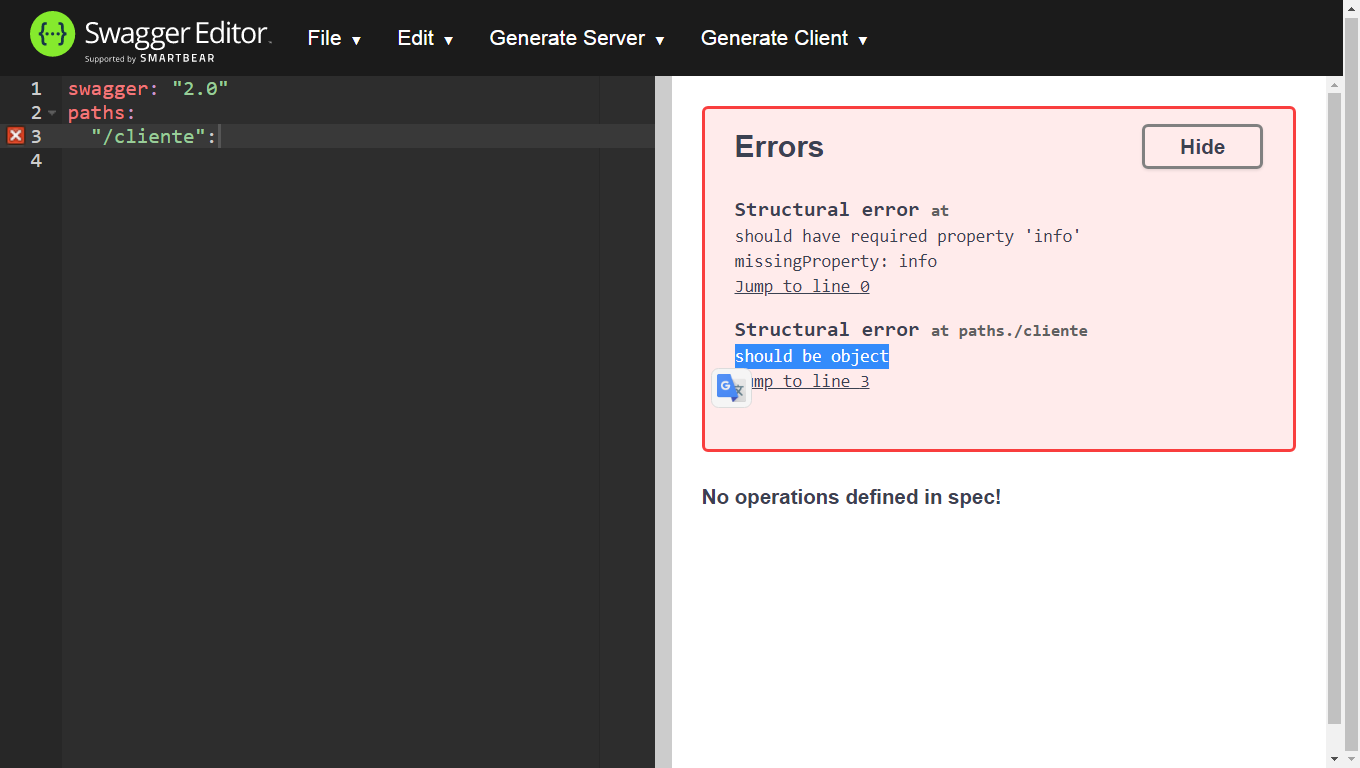
## Segundo-passo

* Agora, no lado esquerdo da tela, nós iremos zerar o script do “Petstore” e inserir nosso próprio script. Para isso, selecione o script todo pressionando “Ctrl + A” e depois pressione“Delete” para deletar o script ainda visível em sua tela. Após deletar o antigo script, vá no campo vazio do lado esquerdo e insira seu script que deseja utilizar. É necessário que no início você digite “swagger: ‘2.0’ ou ‘3.0’” para definir a versão do swagger que você irá utilizar. Do lado diteiro da tela já é possível ver alguns erros, isso tudo acontece de forma interativa. Com isso, será necessário introduzir também a “info” e os “paths”.



## Terceiro-passo

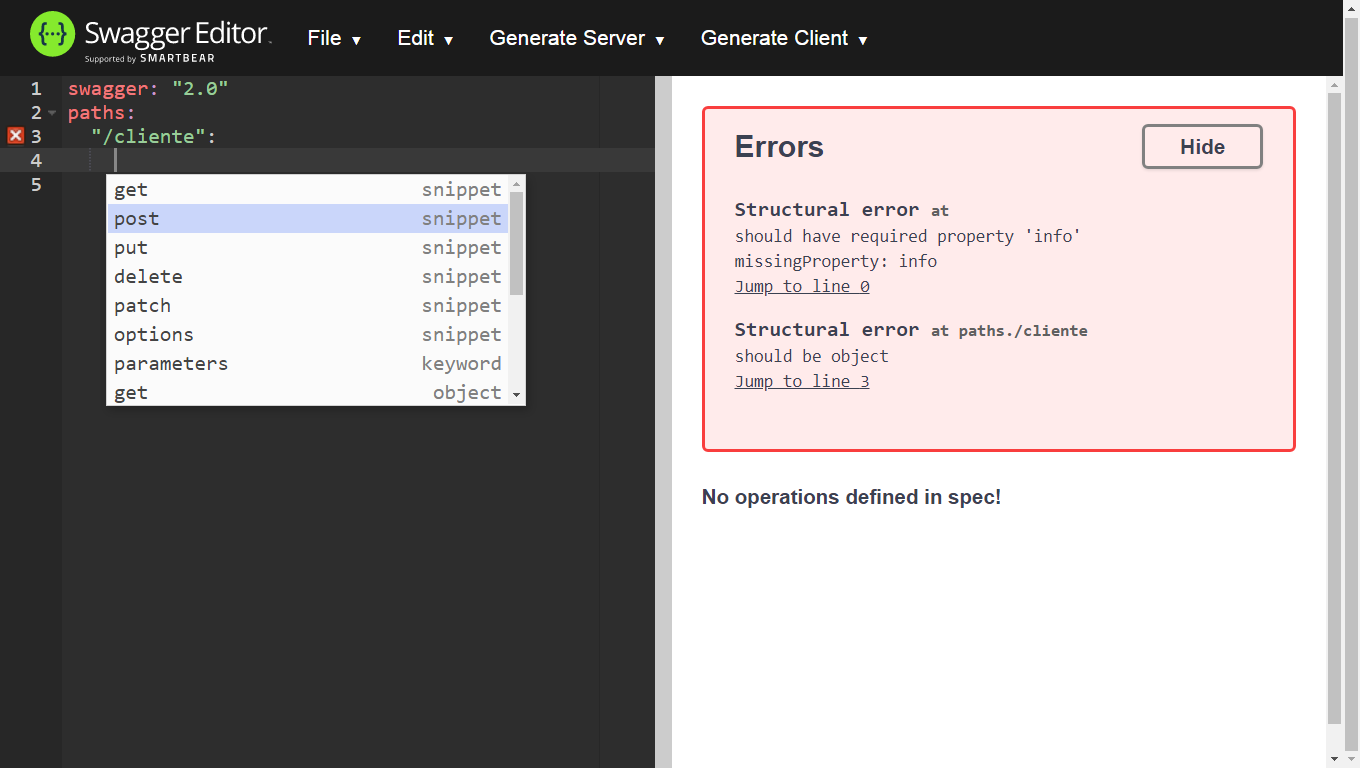
* O que é o path? São as operações que a nossa API vai oferecer, então digite “paths:” pressione “Enter” para próxima linha e um “Tab”, então vá dando “Tab” para incluir uma nova propriedade dessa palavra chave. Para fazer essa primeira operação, vamos fazer primeiro o endereço, nós vamos fazer o cadastro de um cliente então aqui coloque “/cliente”;

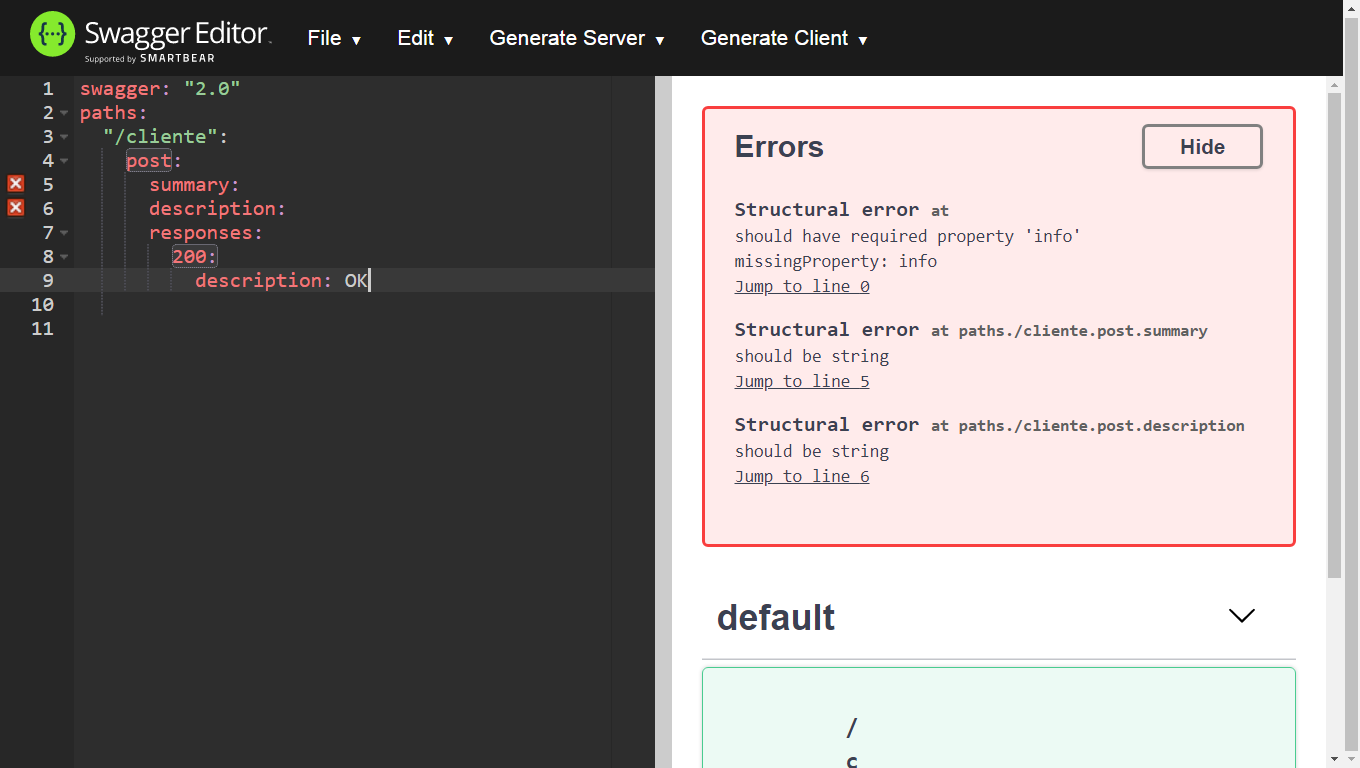


* Agora ele já deu um outro erro aqui se observamos do lado direito, ele está dizendo que a palavra /cliente, não é, e não pode ser um objeto, realmente, ele é só o nosso endereço, então vamos saltar mais uma linha dar um Tab, e agora o que vamos fazer? Vamos indicar qual é o método Rest que vamos utilizar, se vai ser o post, o get ou o put, no nosso caso vai ser o put, sempre que estivermos trabalhando com Rest vamos utilizar o post para fazer o cadastro de qualquer coisa que seja.

## Quarto-passo

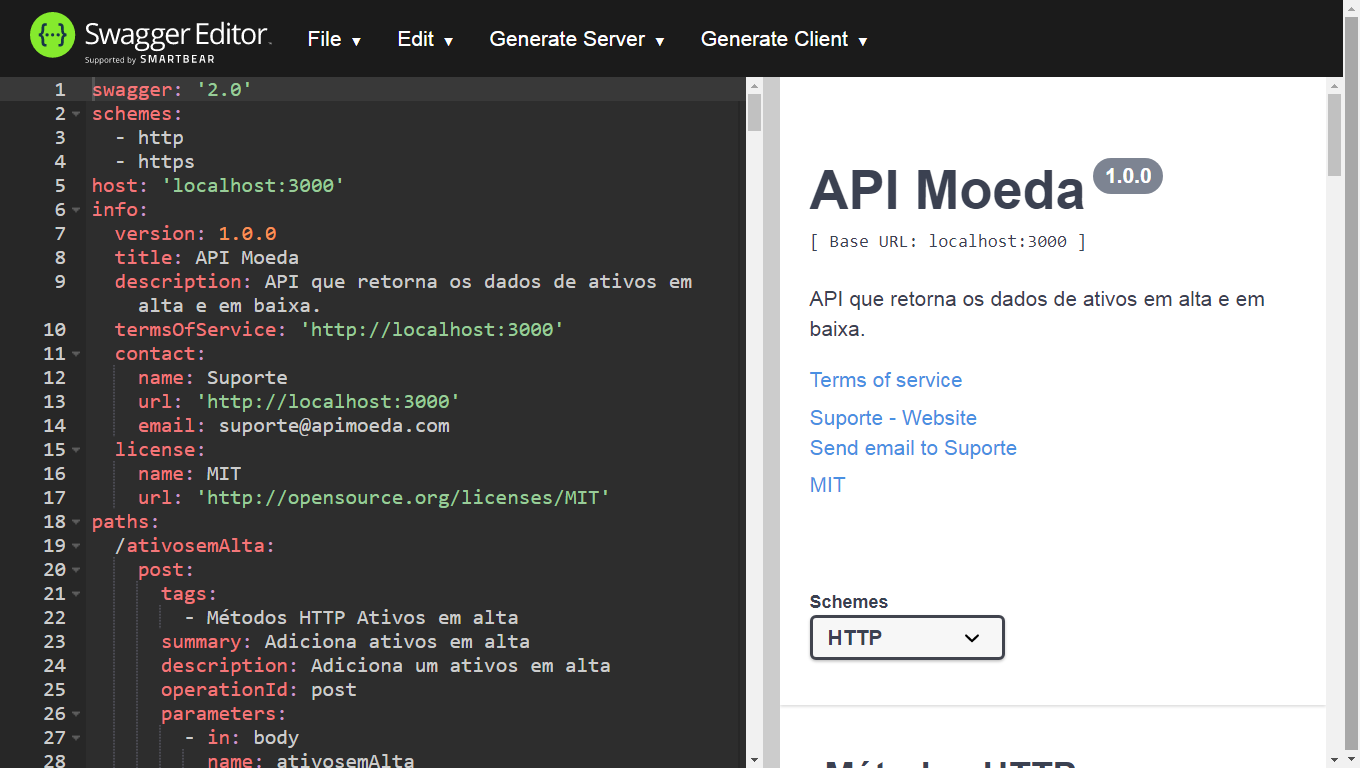
* Se for fazer uma alteração já será outro método, então no editor aqui do swagger há a possibilidade também do ‘auto complete’, dê um “Ctrl + Espaço + Enter”, repare que do lado esquerdo ele já indicou alguns métodos que podem ser usados, vamos clicar no “post” e ver como ele auto completa para nós. Da mesma forma é possível utilizar outros métodos como “git”, “put”, “delete” e alguns outros.



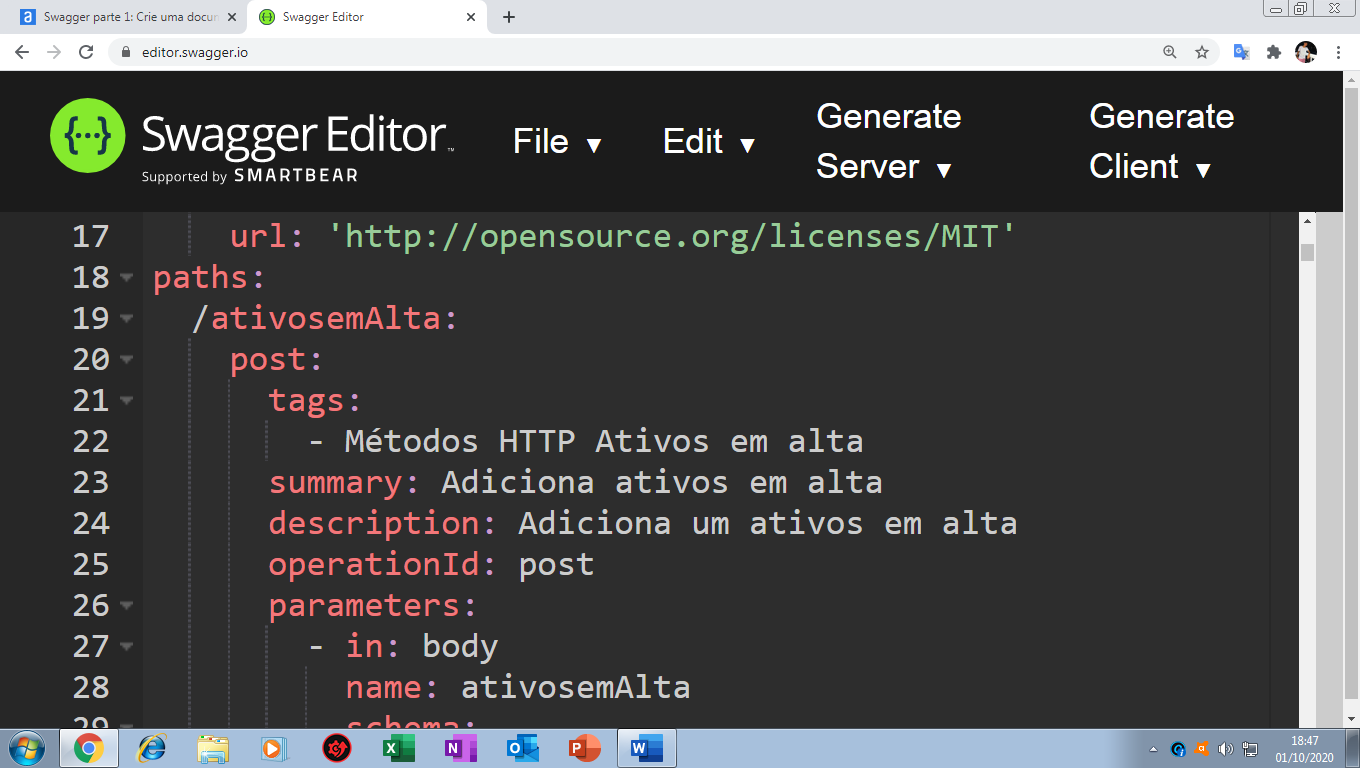


## API de exemplo –

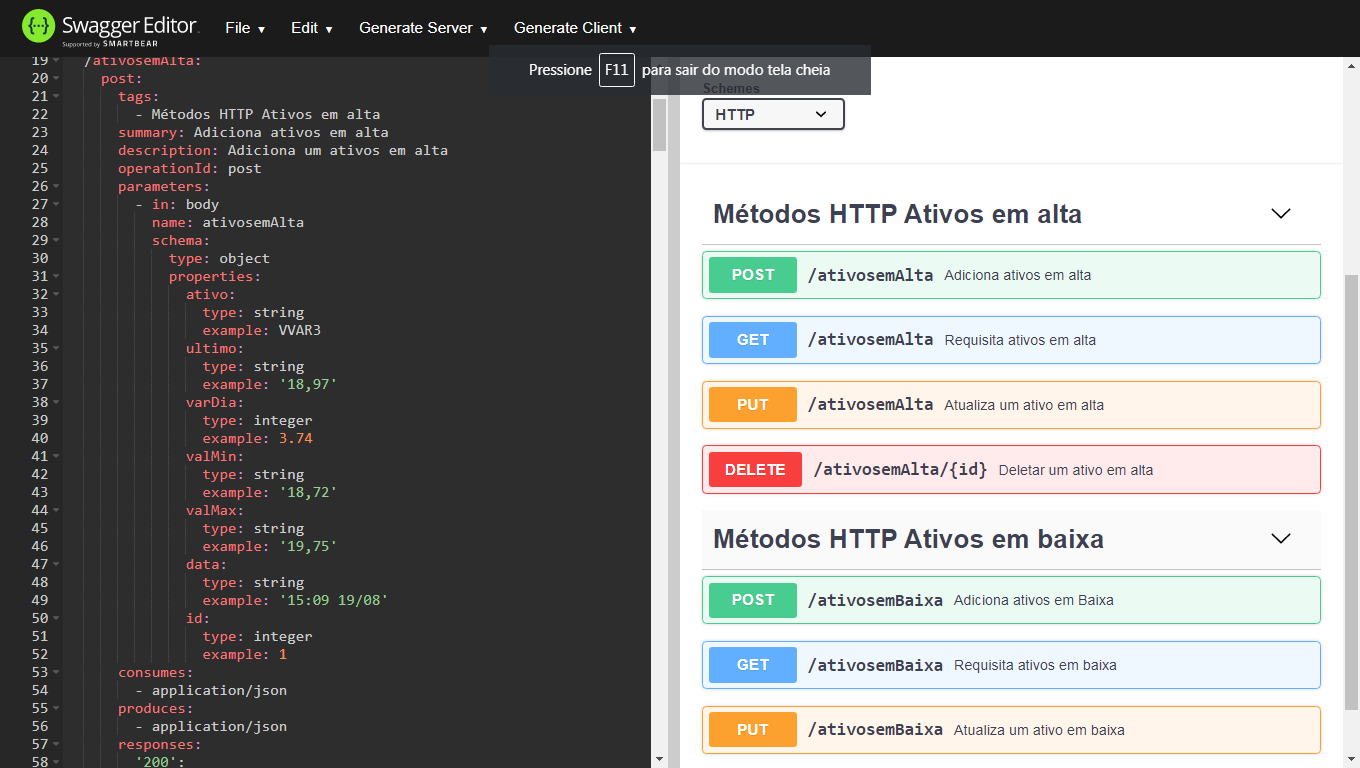
* Acima lhe foi mostrado como iniciar um script no Swagger Editor. Agora irei lhe mostrar uma API que nós (equipe de Devs que montou este tutorial) consumimos e documentamos utilizando esta mesma ferramenta, inicialmente da mesma forma que foi apresentada anteriormente. Esta é uma API de moedas que exibe “ativos em alta” e “ativos em baixa”. Vamos averiguar como foi feita essa documentação.
* Olhando o script no mesmo momento que abrir, é possível ver algumas coisas, começando pela versão (Que é a mesma utilizada mais acima), schemes, host, info e algumas outras coisas.

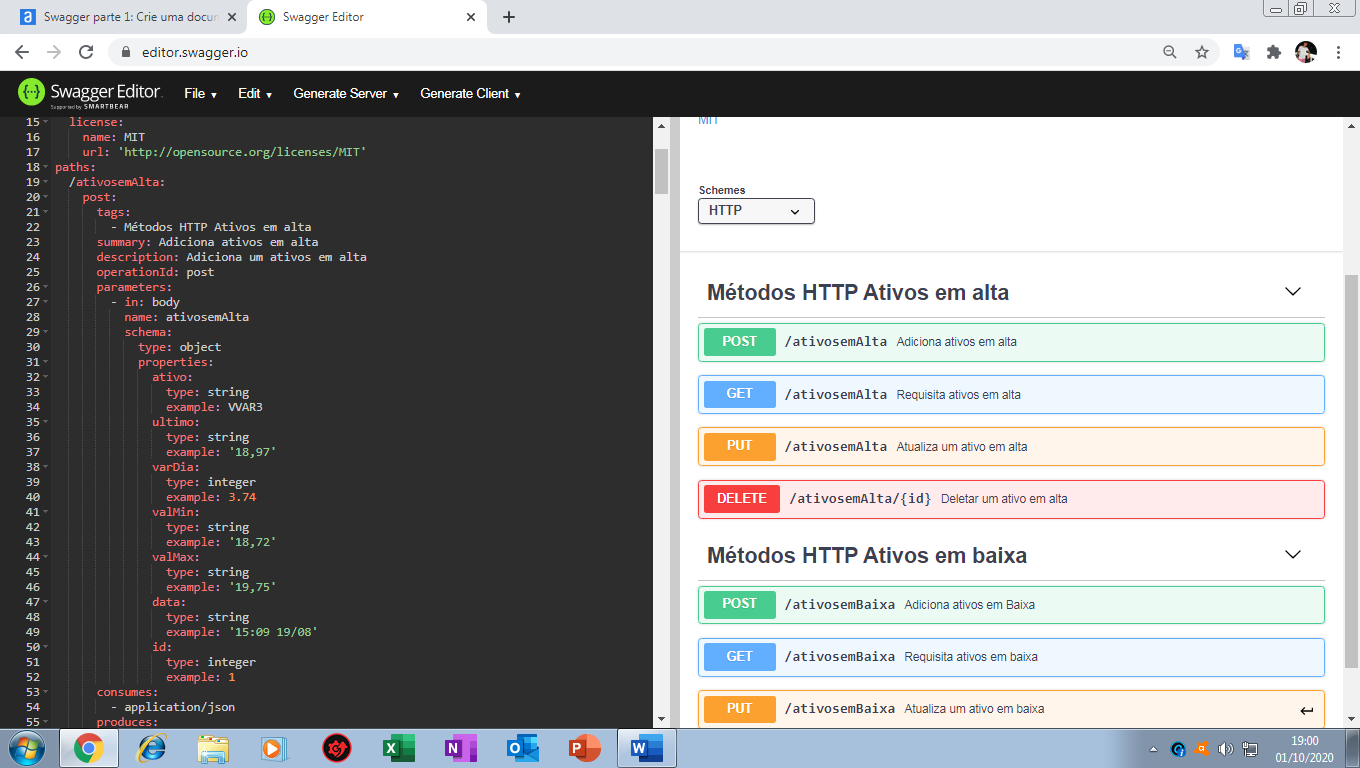


* Em “info” vão as informações importantes do script de consumo da nossa API, como por exemplo: O título que é exibido no lado direito da tela, a descrição e outras informações que são importantes para nosso script.
* Também é possível ver os “paths”, é onde vai todo o resto, como os métodos e as informações do consumo da API dentro deles.
* De primeira mão é possível ver o método “post” sendo inicializado na linha 20.



* Olhe como o swagger responde de modo interativo no lado direito da tela:

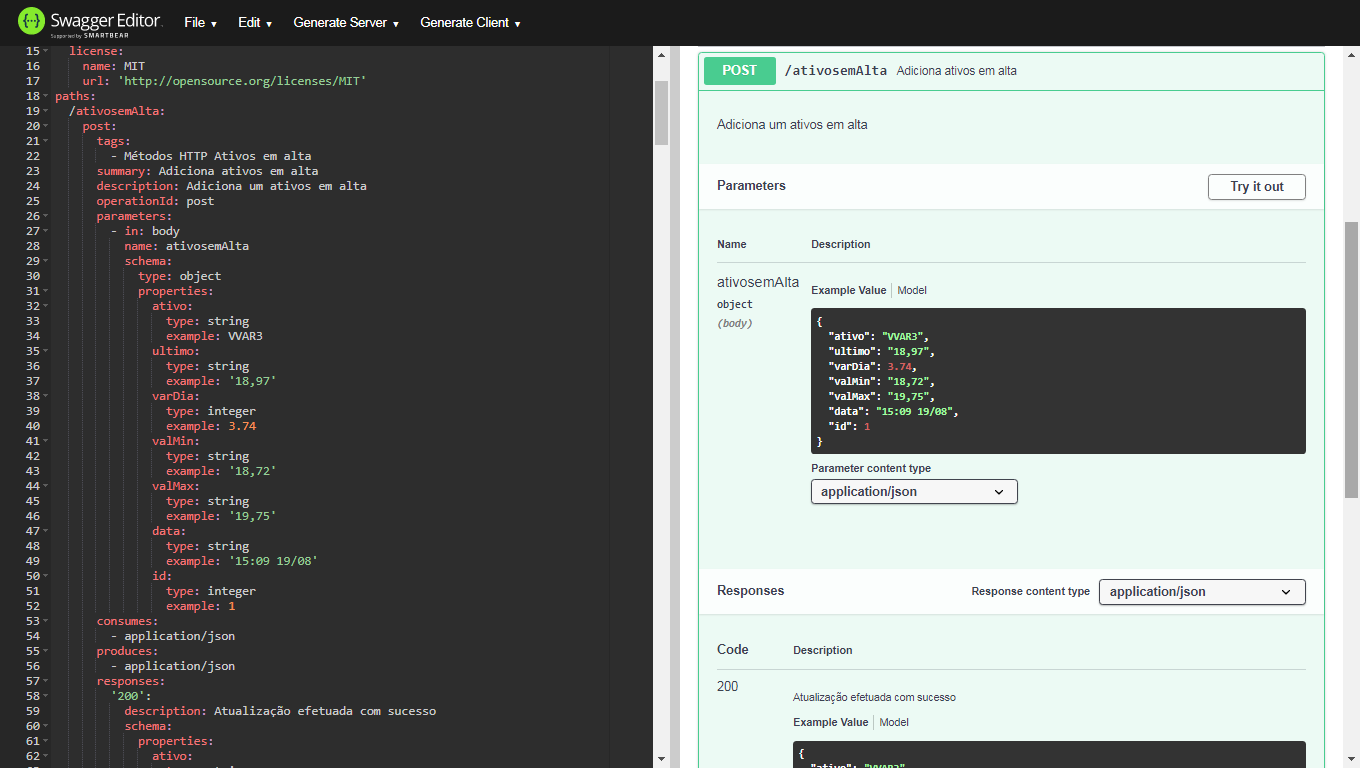




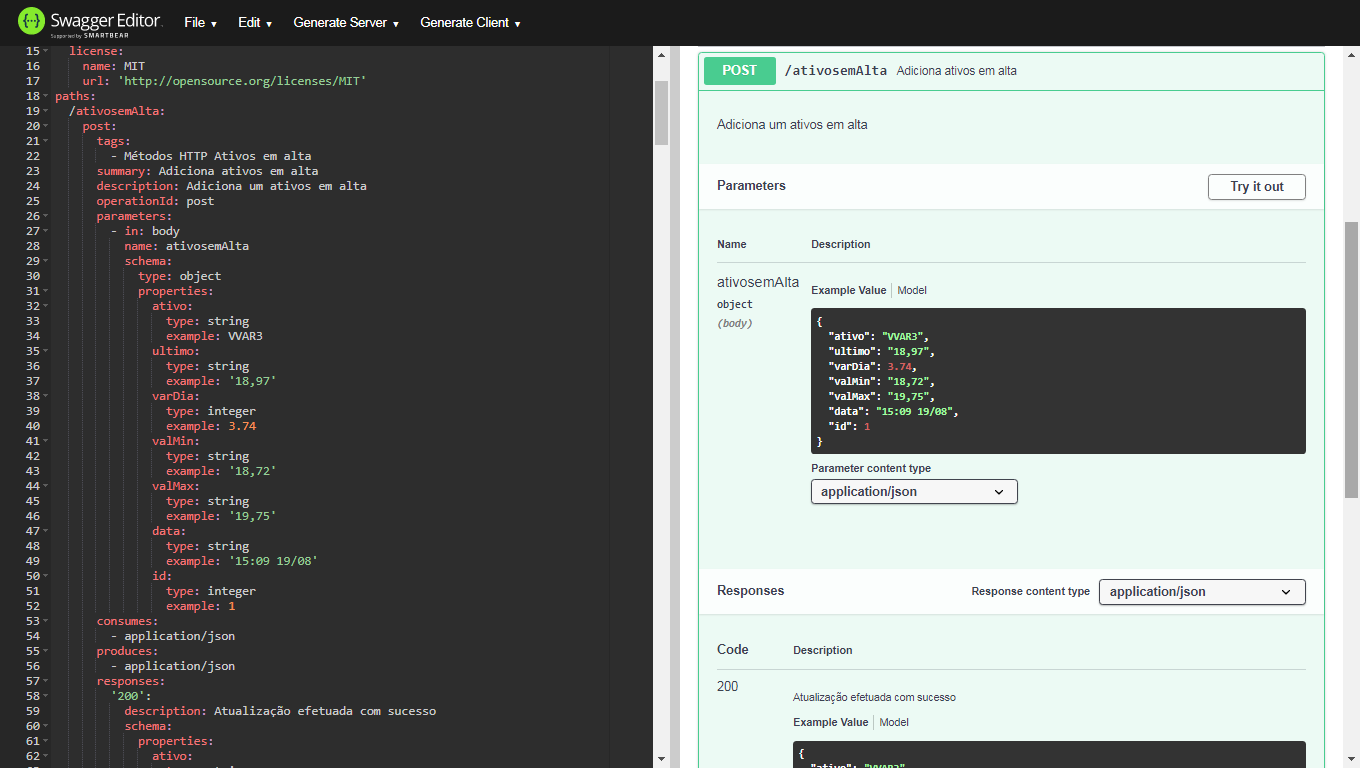
* Interessante, não é mesmo? Agora, mais abaixo eu irei mostrar como cada método foi chamado na documentação e como o swagger exibiu do lado direito da tela.

### Post –

Esquerda:

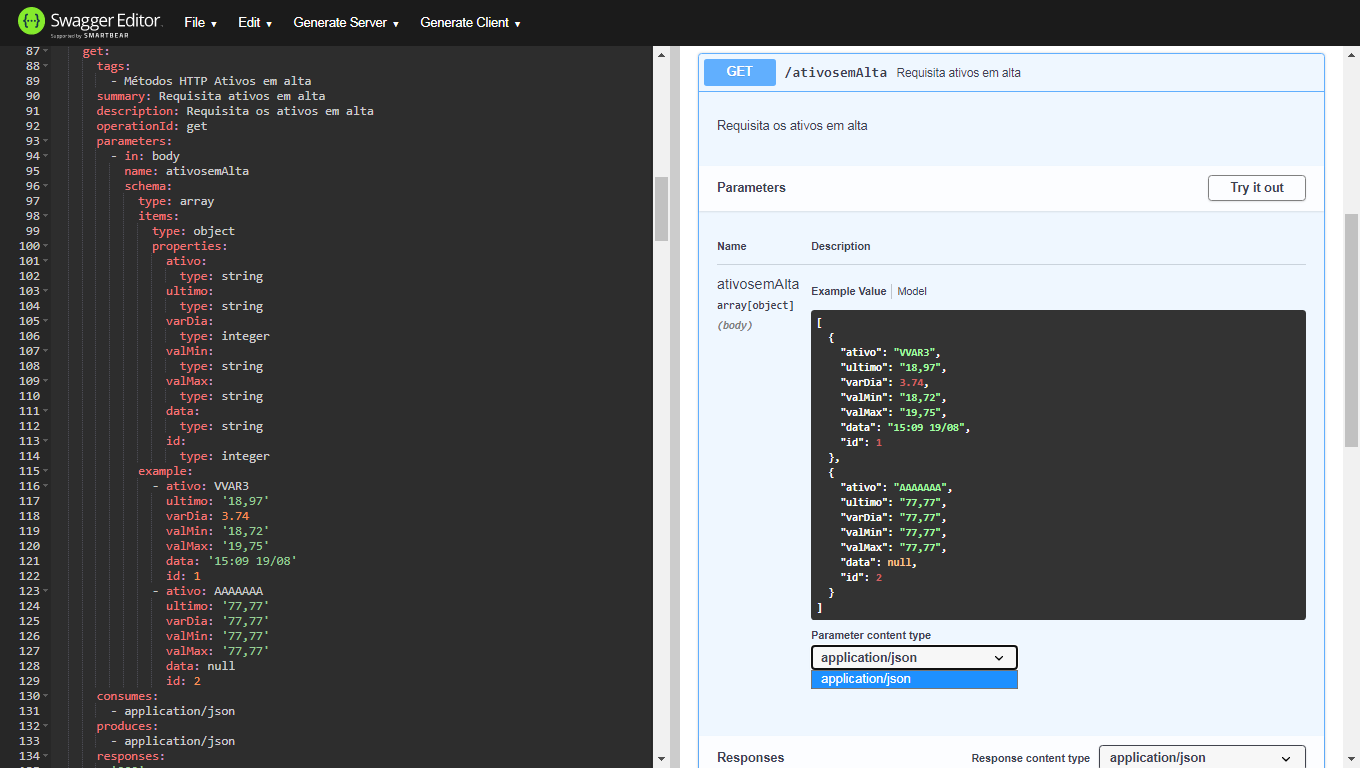


Direita:

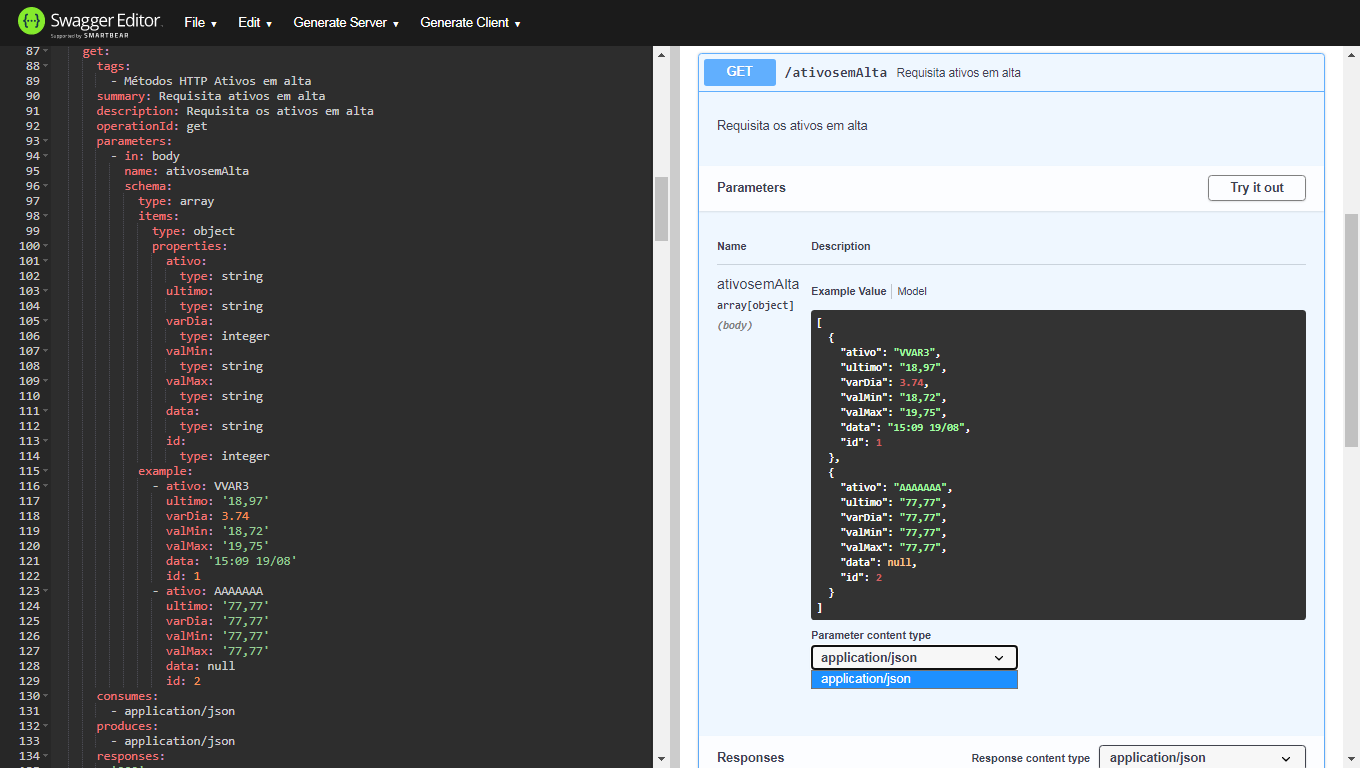


### Get –

Esquerda:

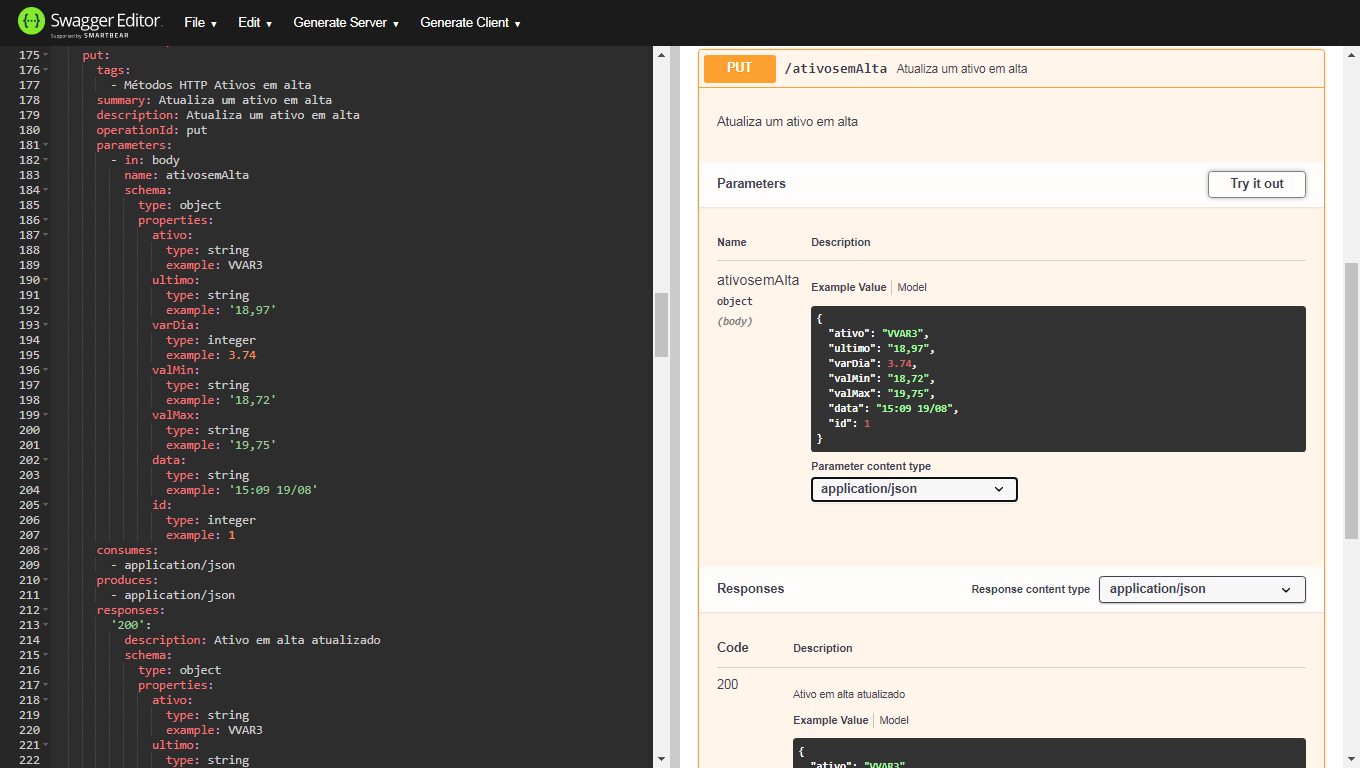


Direita:

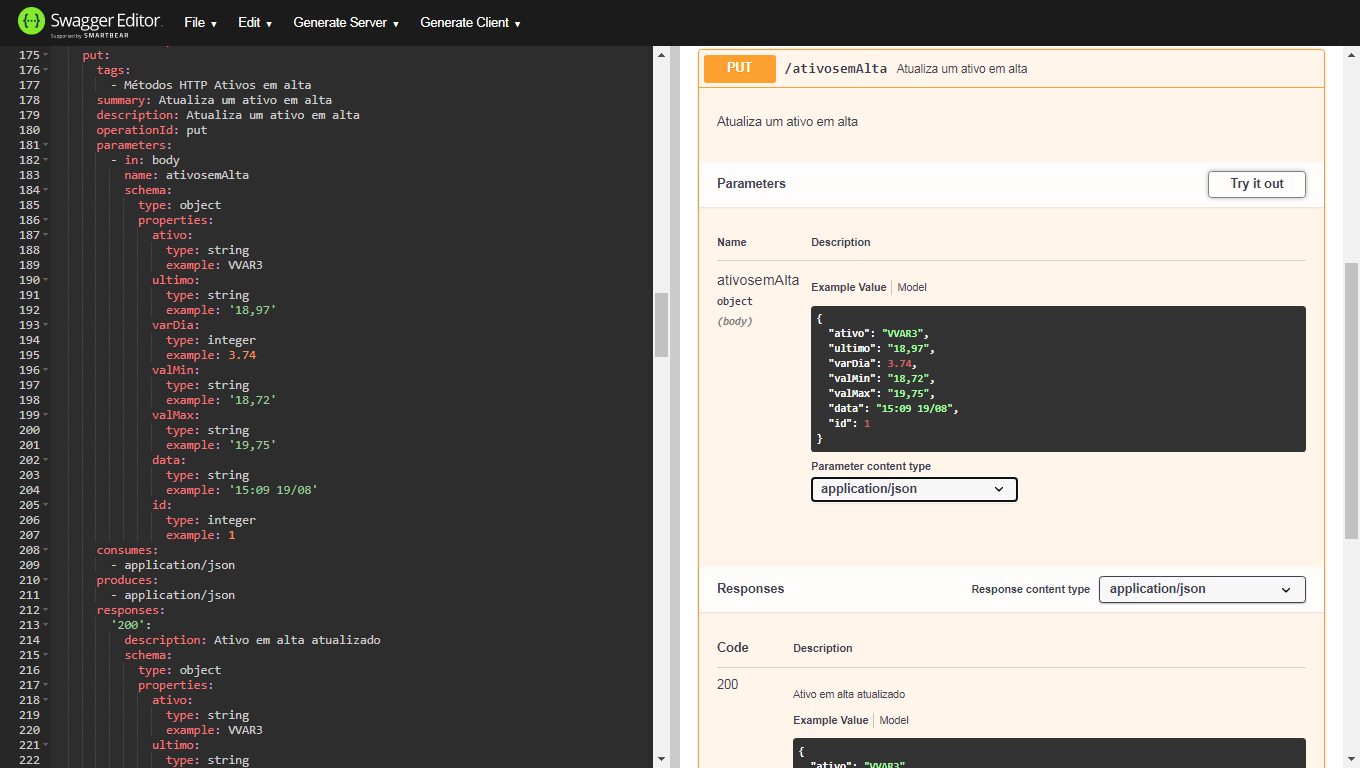


### Put –

Esquerda:

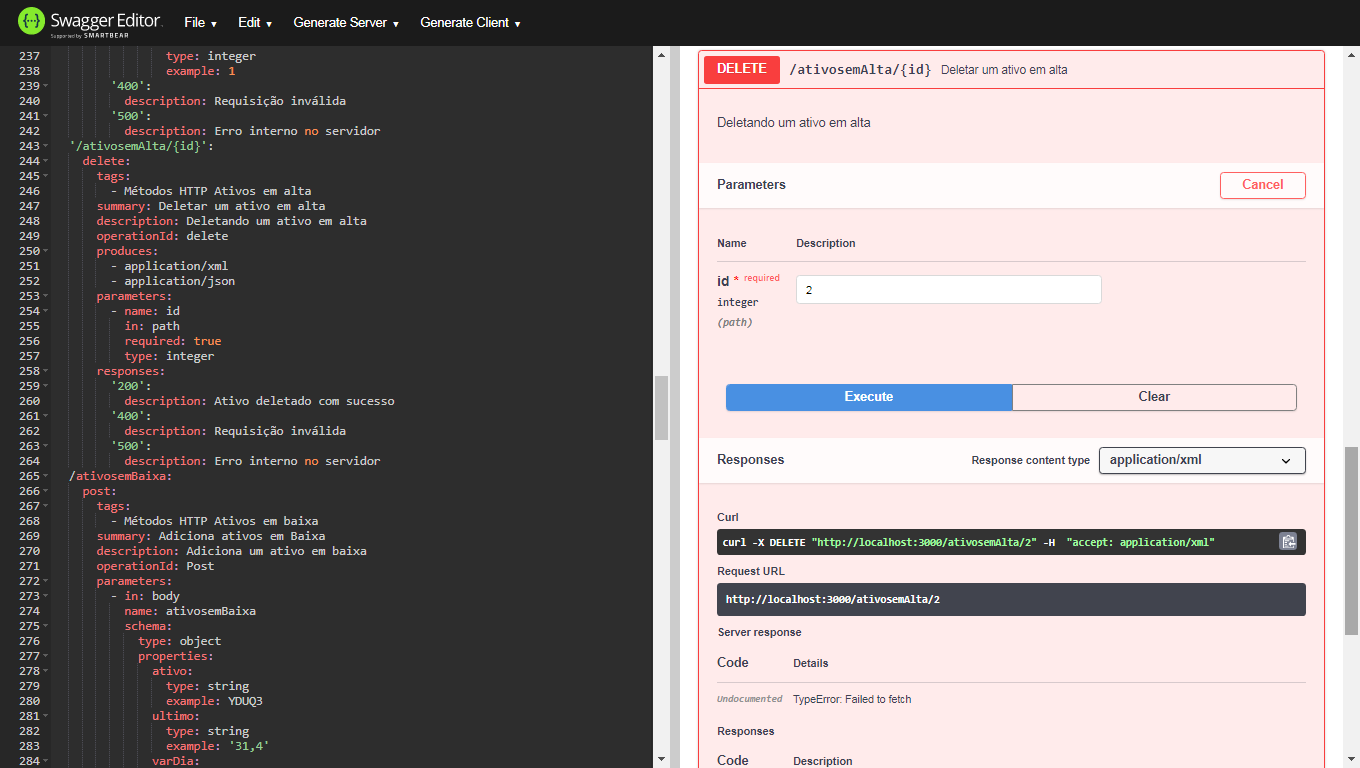


Direita:

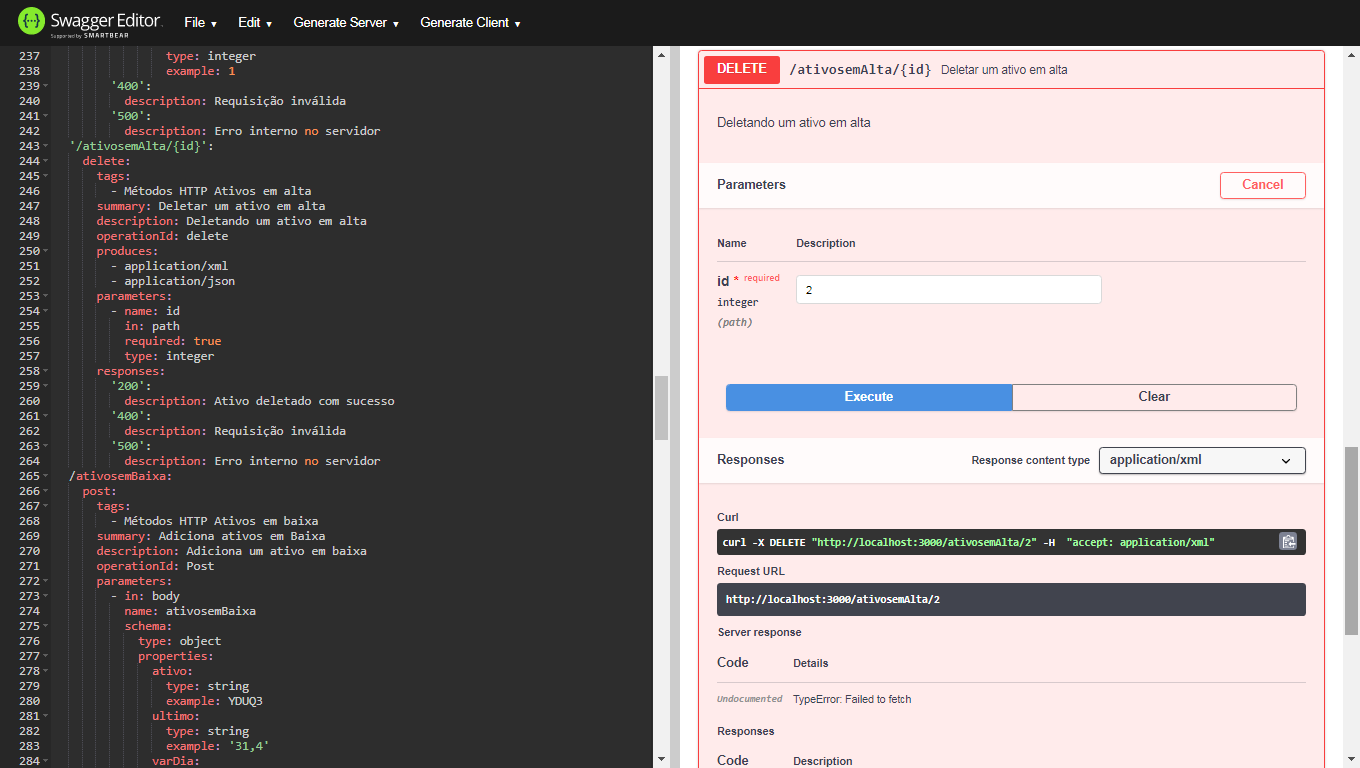


### Delete –

Esquerda:

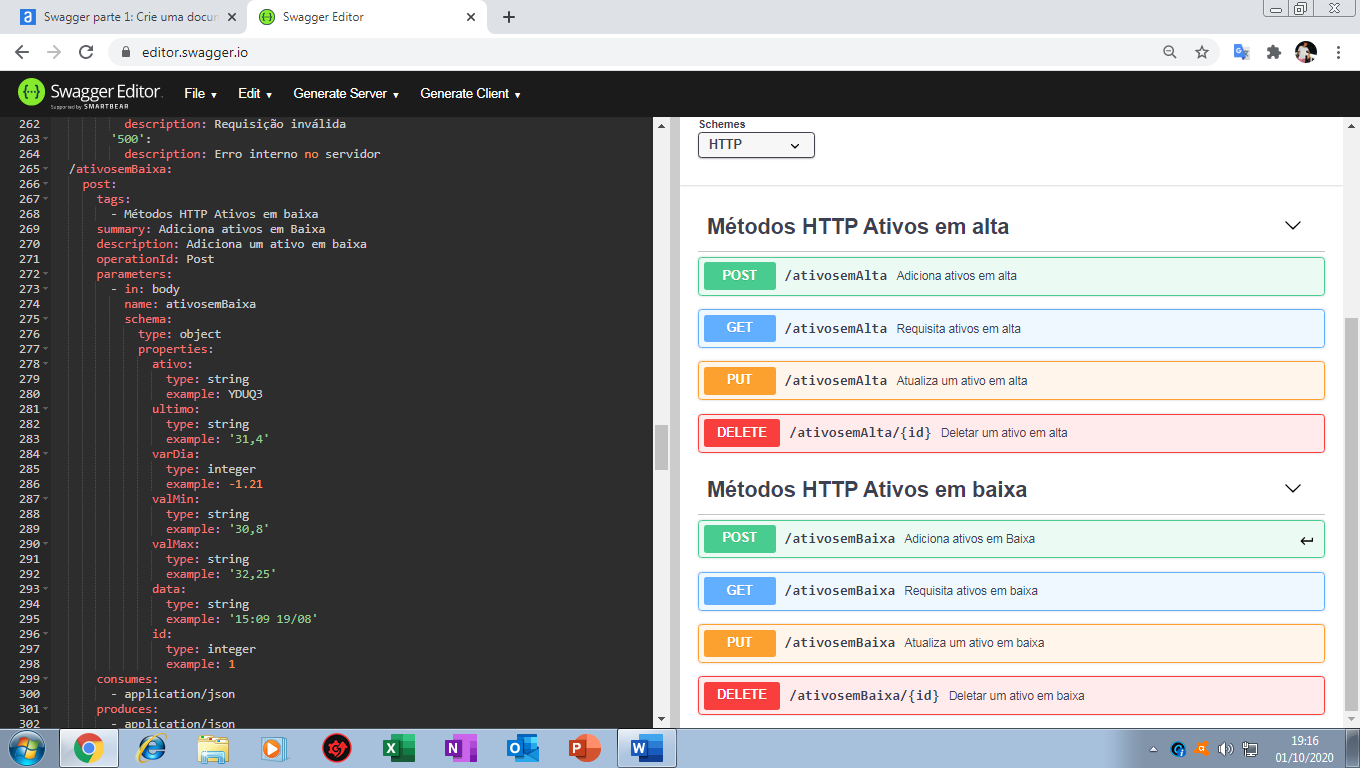


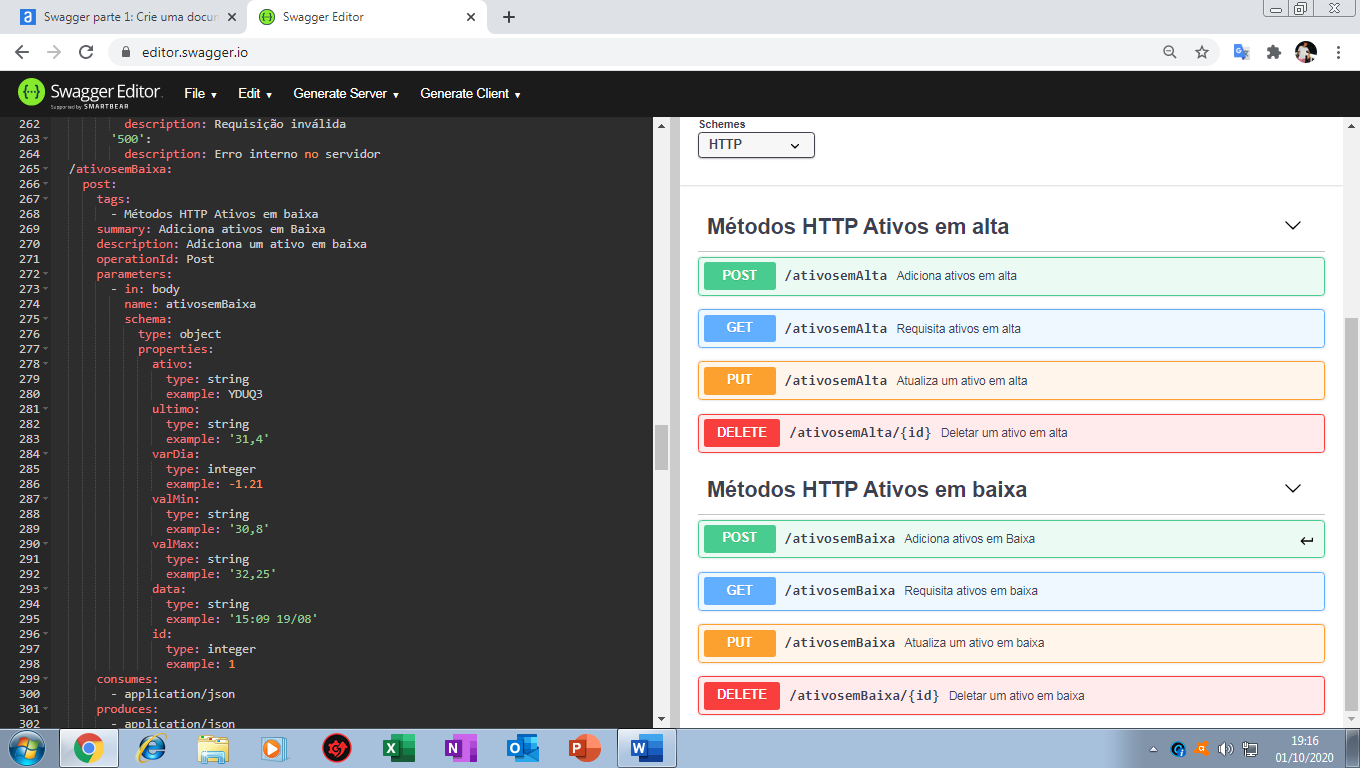
Direita:



### Último exemplo –

* Nos exemplos dados anteriormente, os prints exibiram o resultado do path “ativosemAlta” como você pode perceber. Há apenas dois paths, o “ativosemAlta” e o “ativosemBaixa”, abaixo eu irei mostrar como ficou o resultado do path “ativosemBaixa”, é basicamente a mesma coisa que o outro path, só é alterado coisas como o nome da tag e etc.





## Conclusão –

* Neste tutorial nós aprendemos a acessar o swagger, aprendemos um pouco sobre como criar um script, documentá-lo e como nosso script deverá aparecer na tela caso não seja encontrado nenhum erro.

## Referências –

* <https://www.alura.com.br/conteudo/swagger-crie-uma-documentacao-rest>
* <http://www.matera.com/blog/post/swagger-como-gerar-uma-documentacao-interativa-para-api-rest>