

SigaBem – api

Endereço api = <http://alefe.vpskinghost.net:8080/calcula-frete>

Endereço Swagger = <http://alefe.vps-kinghost.net:8080/calcula-frete/swagger-ui/>

A api foi deackerizada e está em um container em um vps (virtual private server).

Descrição = o objetivo dessa api é consultar quanto fica um frete. Ao inserir o cep de origem e cep de destino é feito um calculo a partir do DDD e estado.

Caso exista os ceps ou os mesmos não for vaziao ou inválidos, o micro serviço faz um cálculo, retorna para quem está solicitando esse cálculo, o valor total do frete, data prevista da entrega, o cep de origem que foi inserido e o cep de destino.

Path do End-point <http://alefe.vpskinghost.net:8080/calcula-frete/>frete

End-points

Calcular um frete

Post

<http://alefe.vpskinghost.net:8080/calcula-frete/frete>

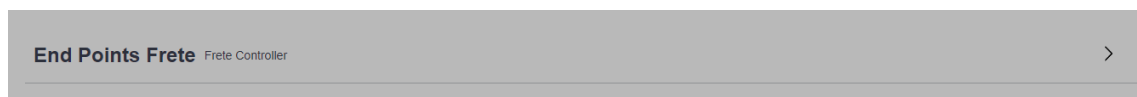
Para que haja retorno do frete calculado é necessário enviar um corpo para essa requisição, caso esteja fora do Swagger, e esteja em um programa de requisição como insomnia ou postman, envie o corpo do tipo JSON da seguinte forma.

Exemplo:

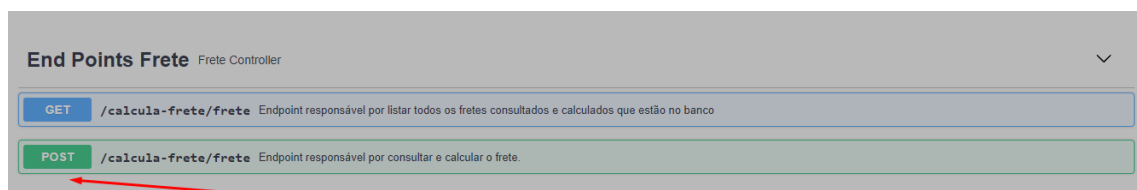
```
{  
  "peso": "450",  
  "cepOrigem": "78143-312",  
  "cepDestino": "78005-060",  
  "nomeDestinatario": "Alefe Patrick"  
}
```

Caso esteja dentro do Swagger

Clique em End Points Frete



Clique no post



Irá abrir uma tela, clique em try it ou(experimentar)



Após isso o quadrado abaixo ficara editável.

The screenshot shows a Swagger UI editor interface. At the top, there are two tabs: 'Name' and 'Description'. Below the 'Name' tab, the text 'freteInputDto' is displayed with a red asterisk and the word 'required' next to it. Below the 'Description' tab, the text 'freteInputDto' is displayed. Underneath, the text 'object' and '(body)' are shown. To the right of these labels, there is a text input field containing the JSON schema:

```
{
  "cepDestino": "string",
  "cepOrigem": "string",
  "nomeDestinatario": "string",
  "peso": 0
}
```

. Below the input field, there are two buttons: 'Edit Value' and 'Model'. At the bottom of the editor, there is a 'Cancel' button.

Onde está escrito string, renomeie e digite o cep de origem, cep de destino, nome do destinatário e o peso. Coloquei o exemplo que fiz em um programa de requisição as mesmas informações, após isso clique no botão azul escrito execute.

— Deixo alguns testes abaixo —

1° condição: Ceps com DDDs iguais tem 50% de desconto no valor do frete e entrega prevista de 1 dia,

2° condição: Ceps de estados iguais tem 75% de desconto no valor do frete e entrega prevista de 3 dias

3° condição: Ceps de estados diferentes não deve ser aplicado o desconto no valor do frete e entrega prevista de 10 dias

Teste DDD igual e estado igual

input cep origem: 78143312

input cep destino: 78005060

resultado esperado: 1° condição.

Pois DDDs iguais e estados iguais são cidades perto então pode entrar na primeira condição do frete ser menos.

Teste estados igual

input cep origem: 78143312

input cep destino: 78556318

resultado esperado: 2º condição

Pois estados iguais e DDDs diferentes entra na segunda condição

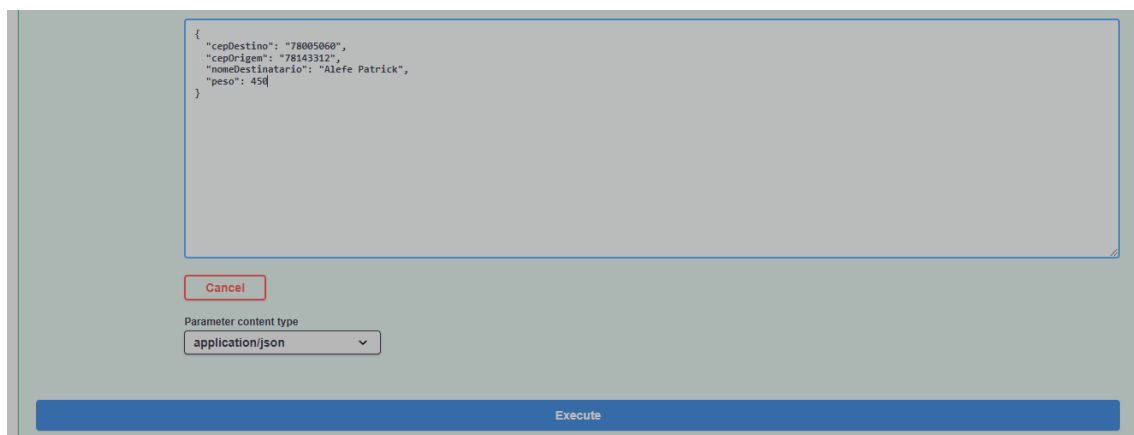
Teste estados diferentes

input cep origem: 78143312

input cep destino: 50030460

resultado esperado: 3º condição

Pois estados diferentes entra na terceira condição



The screenshot shows a REST client interface. A large text area contains a JSON object:

```
{  "cepDestino": "78005060",  "cepOrigem": "78143312",  "nomeDestinatario": "Aléfe Patrick",  "peso": 45}
```

. Below the text area is a "Cancel" button. Underneath that is a label "Parameter content type" followed by a dropdown menu currently set to "application/json". At the bottom of the interface is a blue "Execute" button.

O resultado será igual o da imagem abaixo, vai retornar o valor total do frete, data prevista de entrega, o cep de origem que foi digitado e o cep de destino.



The screenshot shows the "Response body" section of a REST client. It displays a JSON object:

```
{  "vlTotalFrete": 225,  "dataPrevistaEntrega": "05/02/2022",  "cepOrigem": "78143-312",  "cepDestino": "78005-060"}
```

. To the right of the JSON is a "Download" button. The status "201" is visible in the top left corner of the response area.

A api está tratada, caso digite alguma informação vazia, a api irá retornar um aviso, exemplo do nome de proprietário.

```
{
  "cepDestino": "78143312",
  "cepOrigem": "78143312",
  "nomeDestinatario": "",
  "peso": "1"
}
```

O aviso será como na imagem abaixo, assim também serve para outros campos.

Response body

```
{
  "erro": "Campo nome do destinatario não pode ser Vazio!"
}
```

Response headers

Cep inválido

A api não aceita cep com menos de 8 caracteres ou mais, mas aceita o traço, o traço é um caractere, mas na api ela trata isso, então continua com a condição dos 8 caracteres.

Exemplo. Foi digitado no cep de destino 7 caracteres.

```
{
  "cepDestino": "7814331",
  "cepOrigem": "78143312",
  "nomeDestinatario": "Alefe Patrick",
  "peso": "1"
}
```

O retorno será, isso acontecerá também se digitar mais do que 8 caracteres.

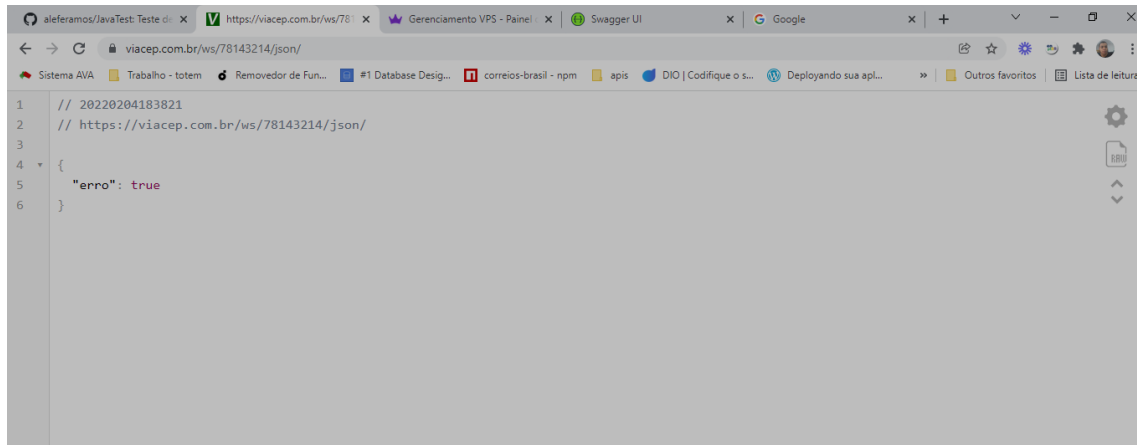
Response body

```
{
  "erro": "Algum Cep é Inválido ou Não Existe, Obs: cep possui 8 digitos!"
}
```

Caso o cep não exista a api retornará o mesmo erro acima, exemplo como sei que não existe um cep.

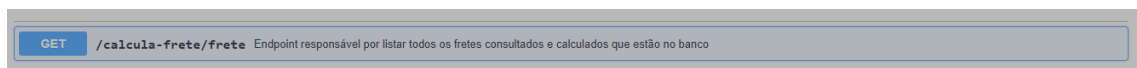
Caso você está digitando um cep e as validações estão corretas, mas se aparecer a mensagem, para testar se é real mesmo que ele não existe, basta entrar no site abaixo e inserir os possíveis ceps do cep de origem e cep de destino.

Exemplo, se você colocar esse cep que está abaixo, na hora de calcular o frete e retornar a mensagem de inválido, basta entrar no site abaixo e inserir o cep, verá que retornará a seguinte página, quer dizer que o cep não existe.

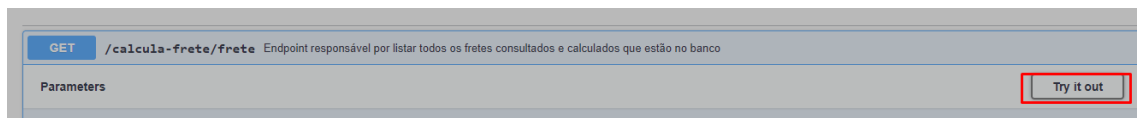


<https://viacep.com.br/ws/78143214/json/>

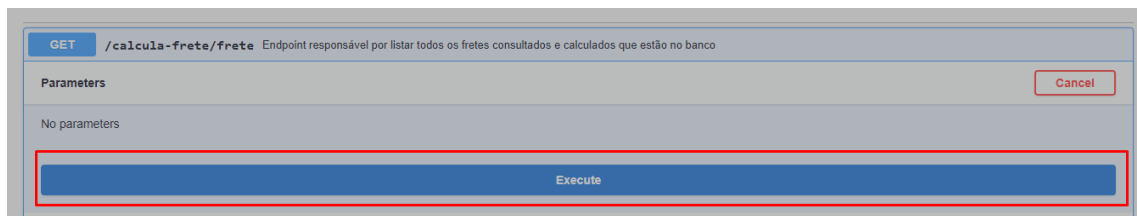
Por fim, caso queira ver quais cálculos de fretes que foram feitos, basta entrar no primeiro End point.



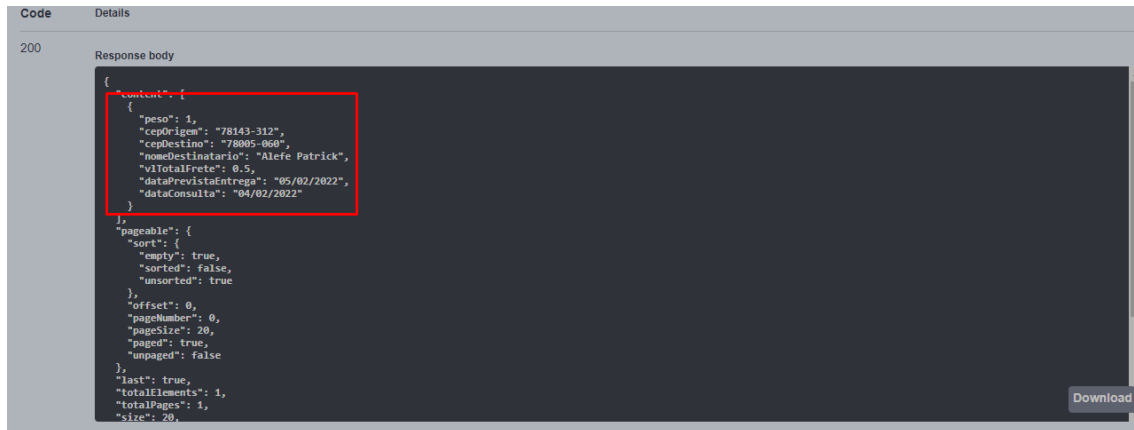
Clique em try out it



Clique em execute



Abaixo irá retornar um objeto com o calculo feito que está gravado no banco de dados.



```
Code    Details
200
Response body
{
  "content": {
    {
      "peso": 1,
      "cepOrigem": "78143-312",
      "cepDestino": "78005-060",
      "nomeDestinatario": "Alefe Patrick",
      "vTotalFrete": 0.5,
      "dataPrevistaEntrega": "05/02/2022",
      "dataConsulta": "04/02/2022"
    }
  },
  "pageable": {
    "sort": {
      "empty": true,
      "sorted": false,
      "unsorted": true
    },
    "offset": 0,
    "pageNumber": 0,
    "pageSize": 20,
    "paged": true,
    "unpaged": false
  },
  "last": true,
  "totalElements": 1,
  "totalPages": 1,
  "size": 20
}
```

O banco de dados está também dockerizado.

Caso queira conectar: ip: 177.153.59.31

Porta: 3306

Senha: alefek159

Sou grato pela oportunidade, pois quero muito ser júnior e também encarar desafios maiores, até breve.