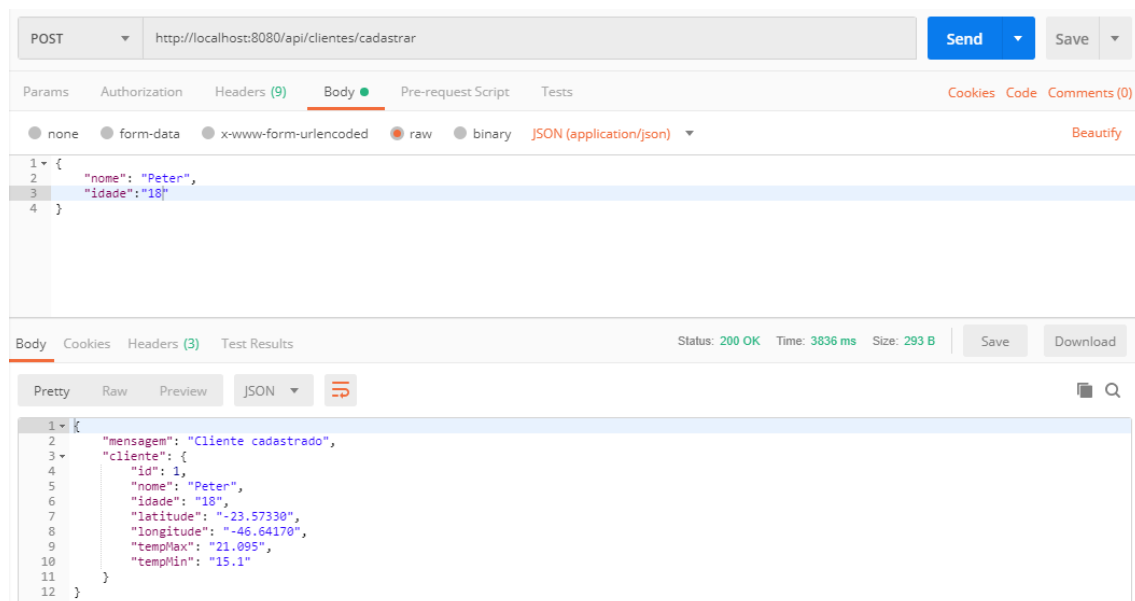


Para utilização desse serviço você precisa executá-lo em sua máquina ou em um ambiente de produção.

Para usar em produção precisará de um servidor web com uma jvm. Uma maneira de iniciar o serviço é gerando uma versão de WAR do projeto e apontando um servidor de aplicação para ele em produção.

Para realização de cadastro de cliente será necessário fazer uma requisição utilizando o endereço do servidor em que o serviço foi alocado, além disso deve informar o nome e a idade no padrão JSON, como o exemplo:

```
{
  "nome": "Peter",
  "idade": "18"
}
```



### 1. Simulação de cadastro de cliente utilizando o software Postman

Após a passagem dessas informações, o serviço irá usar o IP da máquina que realizou o cadastro para obter as informações de localização e condições climáticas do lugar e vincular ao usuário cadastrado.

*Essas informações são voláteis, pois o serviço utiliza o banco H2 nesse exemplo, que salva as informações em memória, isso significa que, caso o servidor seja reiniciado, as informações cadastradas serão perdidas. Para um ambiente de produção outros tipos de bancos poderiam ser utilizados, como Oracle ou MySQL, precisando apenas configurar os data sources e adicionar a dependência ao pom.xml do projeto.*

Após realização do cadastro, é possível consultar as informações vinculadas ao cliente da seguinte maneira:

*url do servidor + "/api/clientes/buscar/" + id do cliente (informação retornada no momento do cadastro)*

GET http://localhost:8080/api/clientes/buscar/1

Send Save

Params Authorization Headers (7) Body Pre-request Script Tests Cookies Code Comments (0)

Query Params

KEY	VALUE	DESCRIPTION	...	Bulk Edit
Key	Value	Description		

Body Cookies Headers (3) Test Results Status: 200 OK Time: 193 ms Size: 304 B Save Download

Pretty Raw Preview JSON

```
1 {
2   "mensagem": "Pessoa encontrada com sucesso",
3   "cliente": {
4     "id": 1,
5     "nome": "Peter",
6     "idade": "18",
7     "latitude": "-23.57330",
8     "longitude": "-46.64170",
9     "tempMax": "21.095",
10    "tempMin": "15.1"
11  }
12 }
```

## 2. Realização de busca de cliente por id

Para retornar todos os clientes cadastrados o processo é mais simples:

*url do servidor + "/api/clientes/"*

Estão disponíveis alteração do **nome e idade** do cliente, como as informações de localização e clima foram retornadas de APIs auxiliares, não será permitida a alterações desses valores após o cadastro. Para realizar esse procedimento será necessário chamar o método *alterar* do serviço passando um JSON com o id do cliente, nome e idade, já aplicando a mudança desejada no valor.

A requisição nesse caso é do tipo *PUT*.

PUT http://localhost:8080/api/clientes/alterar

Send Save

none form-data x-www-form-urlencoded raw binary JSON (application/json) Beautify

```
1 {
2   "id": 1,
3   "nome": "Peter",
4   "idade": "20"
5 }
```

Body Cookies Headers (3) Test Results Status: 200 OK Time: 134 ms Size: 283 B Save Download

Pretty Raw Preview JSON

```
1 {
2   "mensagem": "Cliente alterado com sucesso",
3   "cliente": {
4     "id": 1,
5     "nome": "Peter",
6     "idade": "20",
7     "latitude": null,
8     "longitude": null,
9     "tempMax": null,
10    "tempMin": null
11  }
12 }
```

### 3. Alteração da idade do usuário de 18 para 20

The screenshot shows a Postman interface with a GET request to `http://localhost:8080/api/clientes/buscar/1`. The response status is 200 OK. The response body is a JSON object:

```
1 {
2   "mensagem": "Pessoa encontrada com sucesso",
3   "cliente": {
4     "id": 1,
5     "nome": "Peter",
6     "idade": "20",
7     "latitude": "-23.57330",
8     "longitude": "-46.64170",
9     "tempMax": "21.095",
10    "tempMin": "15.1"
11  }
12 }
```

### 4. Busca do usuário Peter após a alteração da idade

Para deletar um cliente a requisição deve ser do tipo *DELETE*, o procedimento é apenas passar o id na url, como numa busca, segue exemplo:

*url do servidor + "/api/clientes/delete/" + id do cliente (informação retornada no momento do cadastro)*

As ferramentas utilizadas no desenvolvimento desse serviço foram:

- Eclipse  
IDE escolhida por ser uma das mais utilizadas no mercado e aliar simplicidade no uso com robustez no desempenho.
- Postman  
Ferramenta simples para realização de requisições.