

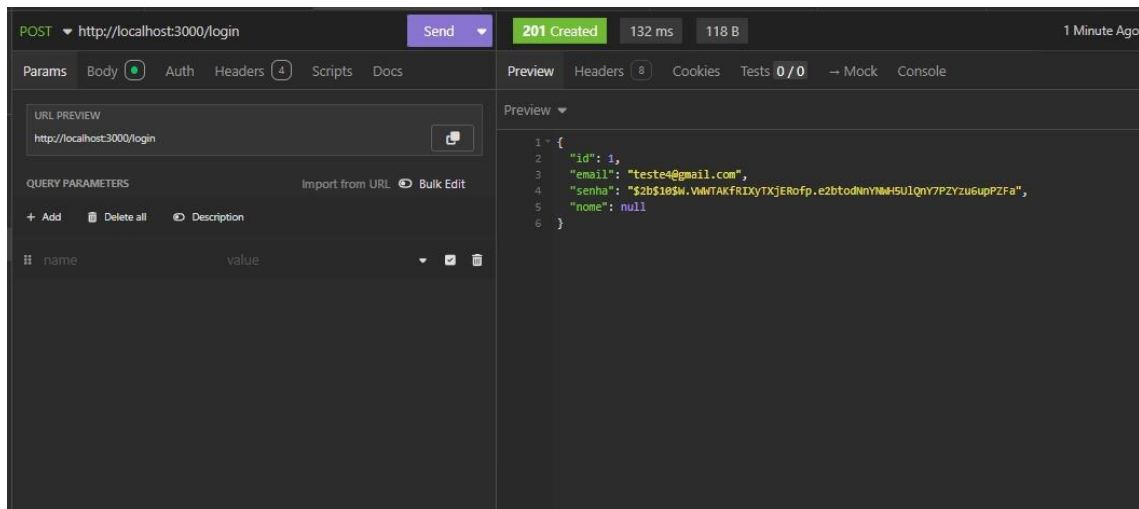
- Projeto: Pet shop
- Versão: 1.2.0
- Data: 27/08/2025

O que é um relatório de testes manuais

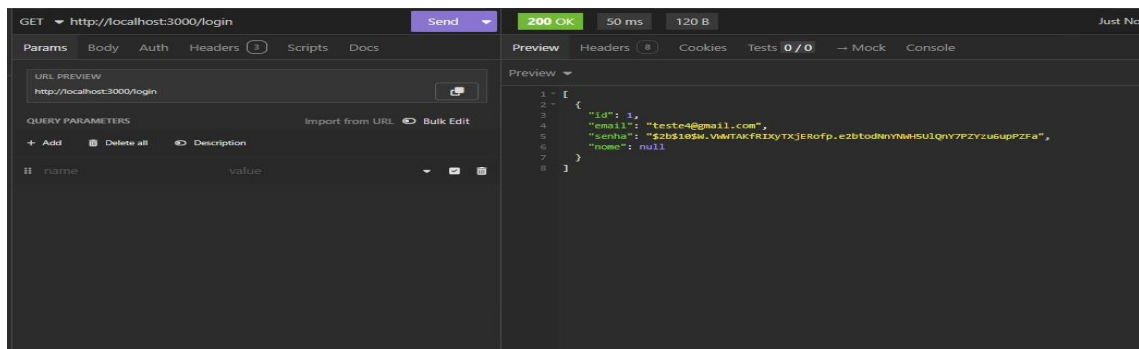
É um documento que mostra como você testou seu sistema, quais funcionalidades funcionam, quais não funcionam e se o sistema atende aos requisitos. Ele inclui três tipos de testes:

1. Unitários: testam uma função ou parte do sistema isoladamente.
2. Integração: testam se diferentes módulos do sistema funcionam juntos.
3. Ponto a ponto: testam o fluxo completo do usuário.

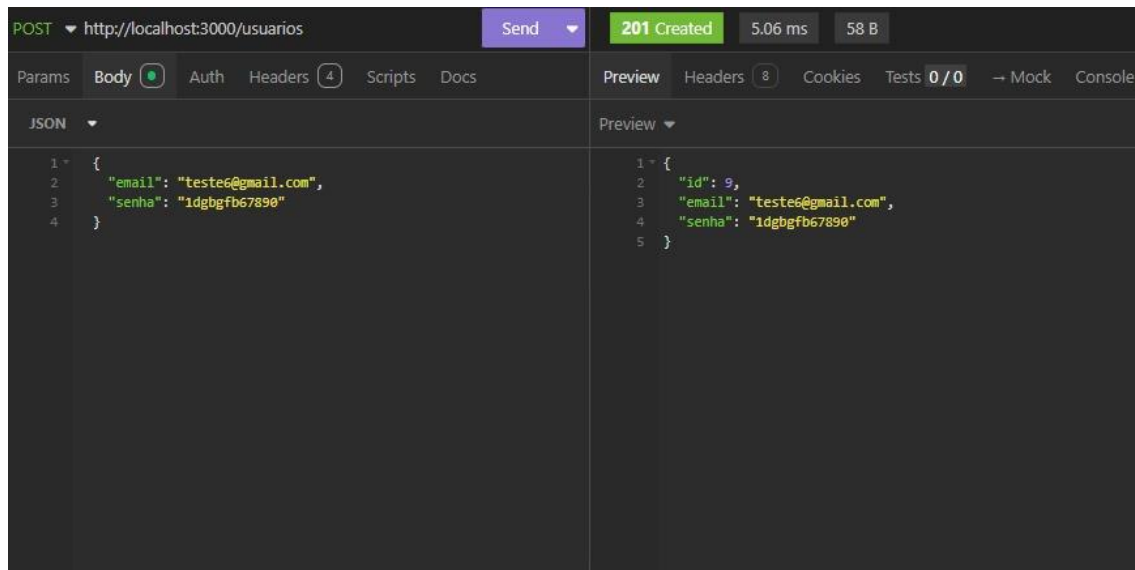
Teste do login: O login foi criado corretamente e as informações foram retornadas como esperado.



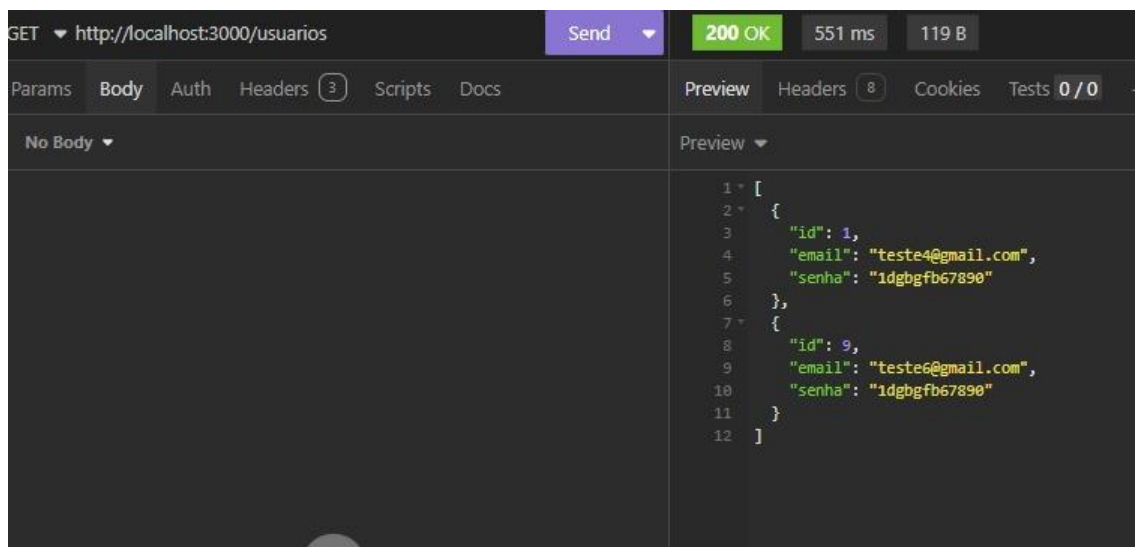
Retorno do login criado: O login foi criado corretamente e as informações foram retornadas como esperado.



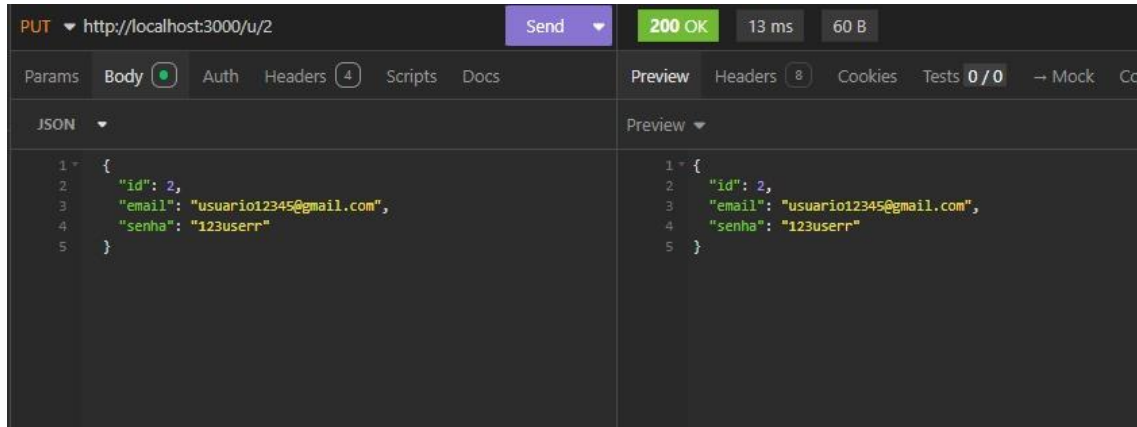
Teste de usuário: O usuário foi criado corretamente e as informações foram retornadas.



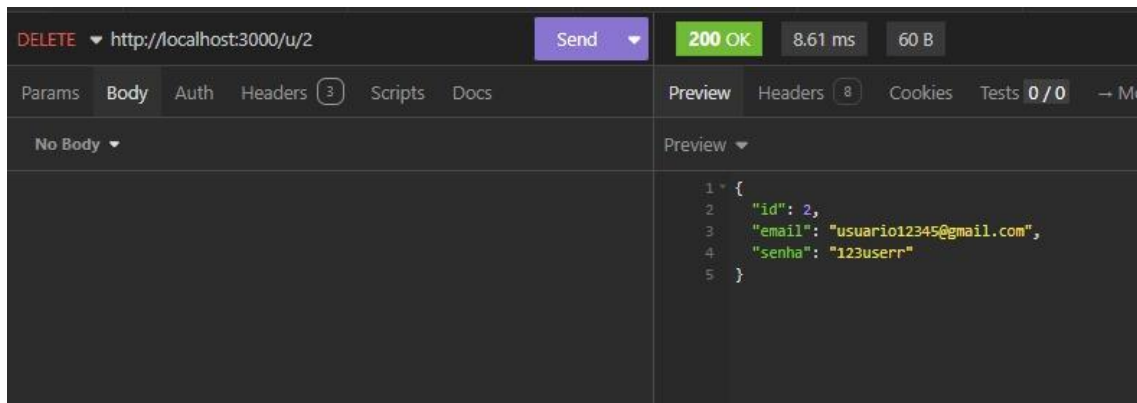
Retorno de usuários criado: : Os usuários foram criados corretamente e as informações foram retornadas com sucesso de acordo com cada usuário criado.



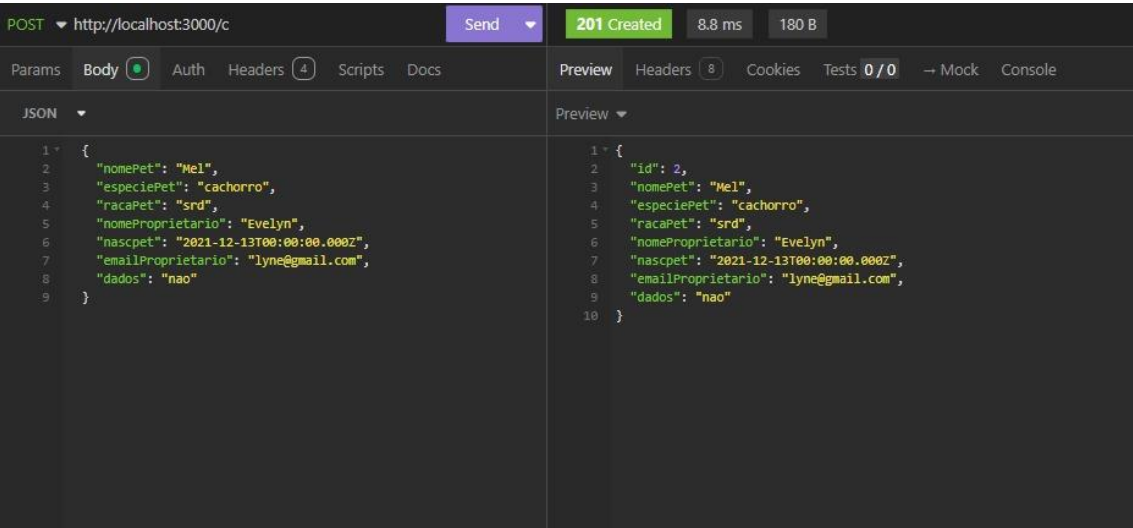
Atualizar os dados do usuário: Foi atualizado corretamente como podemos ver na imagem



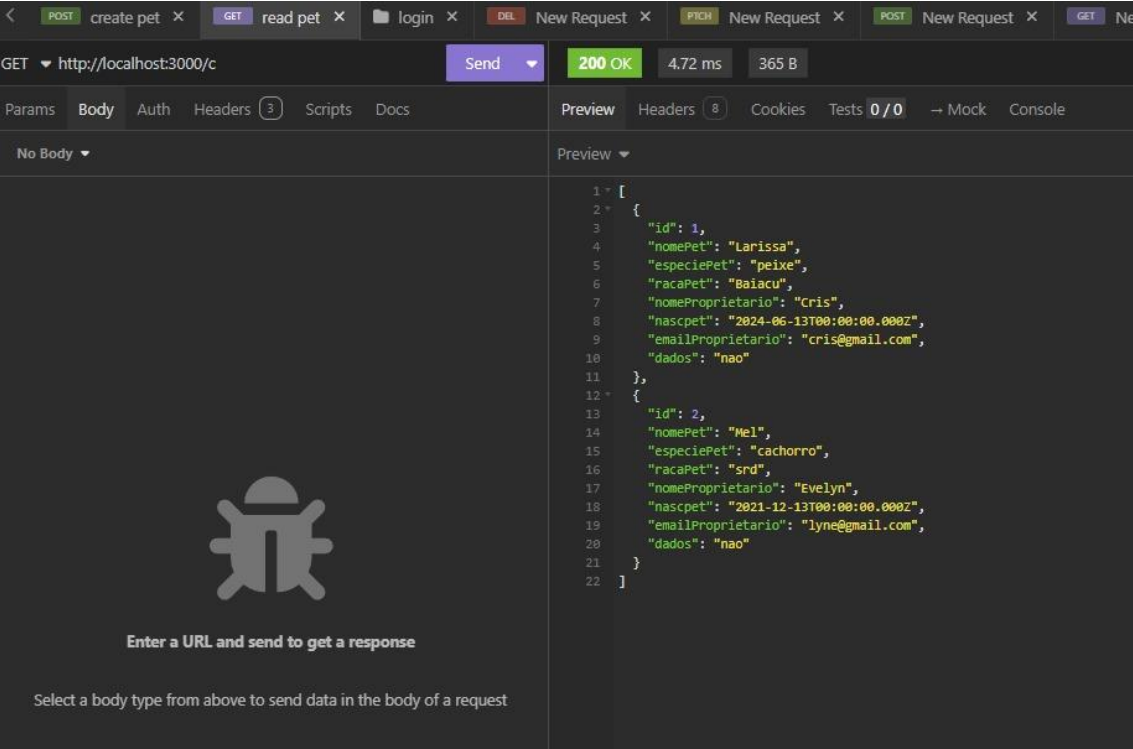
Deletar um usuário: O usuário com ID 2 foi deletado corretamente, e a resposta retornou os dados do usuário deletado corretamente.



Cadastrar um pet: O pet foi criado com sucesso, e a resposta retornou os dados do novo pet cadastrado.



Listar pets cadastrados: Retornou a lista completa dos pets, com os detalhes de cada um, como nome, espécie, raça e os dados do proprietário, está funcionando corretamente.



Atualizar pet cadastrado: O pet com id: 2 foi atualizado com sucesso, todos os dados foram mantidos ou modificados.

The screenshot shows a REST client interface with a PUT request to `http://localhost:3000/c/2`. The request body is a JSON object representing a pet with id 2. The response is a 200 OK status with a response time of 8.06 ms and a body size of 194 B. The response body is a JSON object identical to the request body.

```
PUT http://localhost:3000/c/2
```

Send

200 OK 8.06 ms 194 B

Params Body Auth Headers 4 Scripts Docs

JSON

```
1 {
2   "id": 2,
3   "nomePet": "Lulu",
4   "especiePet": "cachorro",
5   "racaPet": "srd",
6   "nomeProprietario": "Larissa Santos",
7   "nascpet": "2020-02-25T00:00:00.000Z",
8   "emailProprietario": "larissas@gmail.com",
9   "dados": "gato"
10 }
```

Preview

```
1 {
2   "id": 2,
3   "nomePet": "Lulu",
4   "especiePet": "cachorro",
5   "racaPet": "srd",
6   "nomeProprietario": "Larissa Santos",
7   "nascpet": "2020-02-25T00:00:00.000Z",
8   "emailProprietario": "larissas@gmail.com",
9   "dados": "gato"
10 }
```

Deletar um pet cadastrado: O pet com id: 2 chamado Lulu, foi deletado com sucesso.

The screenshot shows a REST client interface with a DELETE request to `http://localhost:3000/c/2`. The response is a 200 OK status with a response time of 10.3 ms and a body size of 239 B. The response body is a JSON object containing a success message and the details of the deleted pet.

```
DELETE http://localhost:3000/c/2
```

Send

200 OK 10.3 ms 239 B

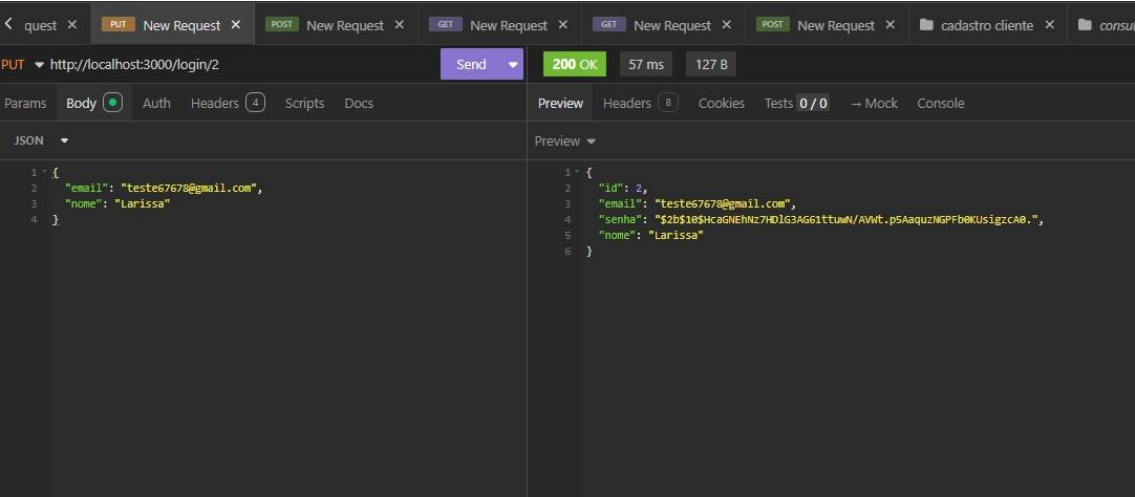
Params Body Auth Headers 3 Scripts Docs

No Body

Preview

```
1 {
2   "message": "Pet deletado com sucesso",
3   "pet": {
4     "id": 2,
5     "nomePet": "Lulu",
6     "especiePet": "cachorro",
7     "racaPet": "srd",
8     "nomeProprietario": "Larissa Santos",
9     "nascpet": "2020-02-25T00:00:00.000Z",
10    "emailProprietario": "larissas@gmail.com",
11    "dados": "gato"
12  }
13 }
```

Atualizar dados de um login (usuário): A senha foi mantida e é exibida criptografada.



Deletar um login cadastrado: O login com ID 3 foi deletado com sucesso.

