Automação de API Rest

Jornada Learning Online - Qualidade
Out/20

Mentores

Ketlin Pedron

Paulo Gonçalves

João Clineu

Agenda

Introdução
- O que é API?

Requisições HTTP
- Estrutura
  - Endpoint
  - Body
  - Response
  - Header

Postman 

Métodos HTTP
- GET
- POST
- PUT
- DELETE

Executando os testes via Newman 

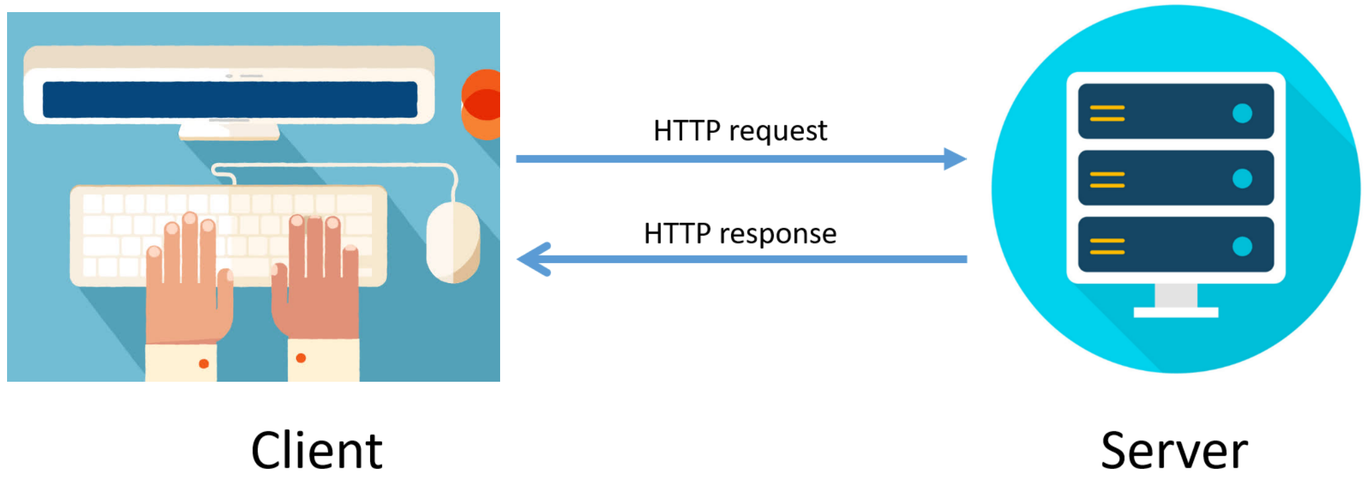
Relação com o Front End

Outros exemplos

O que é API?

API é uma interface para um usuário que precisa fazer uso dele para resolver algo.
Existem diferentes tipos de API, o mais clássico são API's fornecidas por Sistemas Operacionais.

Uma **WEB API** também é uma interface, porém utilizada para comunicação entre clientes e servidores na web. Essa comunicação é realizada utilizando **protocolo HTTP**.



Requisições HTTP

HTTPS://API.SERVEREST.DEV/USUARIOS

Onde minha API está hospedada

Serviço disponível

HTTP Method

GET	**Obter**
POST	**Enviar**
DEL	**Deletar**
PUT	**Atualizar**

HTTP Status Code

2XX	**SUCCESS**
4XX	**CLIENT ERROR**
5XX	**SERVER ERROR**

ServeRest


Overview
Inicializar o ServeRest
Pré-requisito
Execução
Autenticação
Resource Group
Realizar login
Usuários
Listar usuários cadastrados
Cadastrar usuário
ServeRest
O ServeRest é um servidor REST que simula uma loja virtual com intuito de servir de material de estudos de testes de API.
Criado com ♥ por Paulo Gonçalves
Gostou do projeto? Deixe uma estrelinha no repositório. É gratuito, gasta poucos segundos e motiva a contribuir com a comunidade cada vez mais.
Precisa de ajuda? Abra uma issue aqui ou envie e-mail para paulorochag@hotmail.com. Se estiver em algum processo seletivo limitarei a ajudar apenas sobre como funciona o ServeRest e as rotas (e estarei torcendo por você 😄).
Inicializar o ServeRest ¶
PRÉ-REQUISITO ¶
É preciso ter Node.js instalado com versão igual ou superior à 10.0.0. Para verificar sua versão execute o comando node -v no terminal.
EXECUÇÃO ¶

Body

Response

Headers

Criando Requests

Newman

Python

```python
    def test_find_unknown_user(self):
        full_url = BASE_URL + "/usuarios"
        query= {"_id": "123"}

        # Send HTTP Request
        response = requests.get(url=full_url, params=query)

        # Check the response from ServeRest
        self.assertEqual(response.status_code, 200)

        response_json = json.loads(response.text)
        self.assertEqual(response_json["quantidade"], 0, "Quantidade invalida de items para usuario nao cadastrado")
```

Javascript

```
it('Find unknown user', async () => {
  const url = +'https://api.serverest.dev/usuarios'

  # Send HTTP Request
  const { body } = await request
    .get(url)
    .query({ _id: '123' })
    .expect(200)

  # Check the response from ServeRest
  chai.assert.deepEqual(body, { quantidade: 0, usuarios: [] })
})
```

Estratégias de testes

Teste de serviço

Teste de contrato

Teste de performance

Teste de segurança

Teste E2E

Repositório

Dúvidas?

Muito obrigado :D

Ketlin Pedron

Paulo Gonçalves

João Clineu