



# MENTORIA TESTES DE API COM POSTMAN

Aula 3 – Prática com Server Rest

Gustavo Machado

<https://linktr.ee/qagesmachado>

<https://www.youtube.com/@qagesmachado>



# SUMÁRIO

- AMBIENTE DE TESTE
- APIS DO SERVER REST
  - *LOGIN*
  - *USUÁRIOS*
  - *PRODUTOS*
  - *CARRINHOS*
- ORIENTAÇÃO PARA OS TESTES
- PRÁTICA: API USUÁRIOS
- PRÁTICA: APIS LOGIN
- PRÁTICA: APIS PRODUTOS
- PRÁTICA: APIS CARRINHOS

# AMBIENTE DE TESTE

- SERVER REST
  - <https://serverest.dev/>

# APIS DO SERVER REST

## LOGIN

- Utilizado para autenticar no servidor
- Anteriormente será necessário criar um usuário

**Login** Autentique o seu usuário para montar um carrinho e, se for administrador, gerenciar os produtos

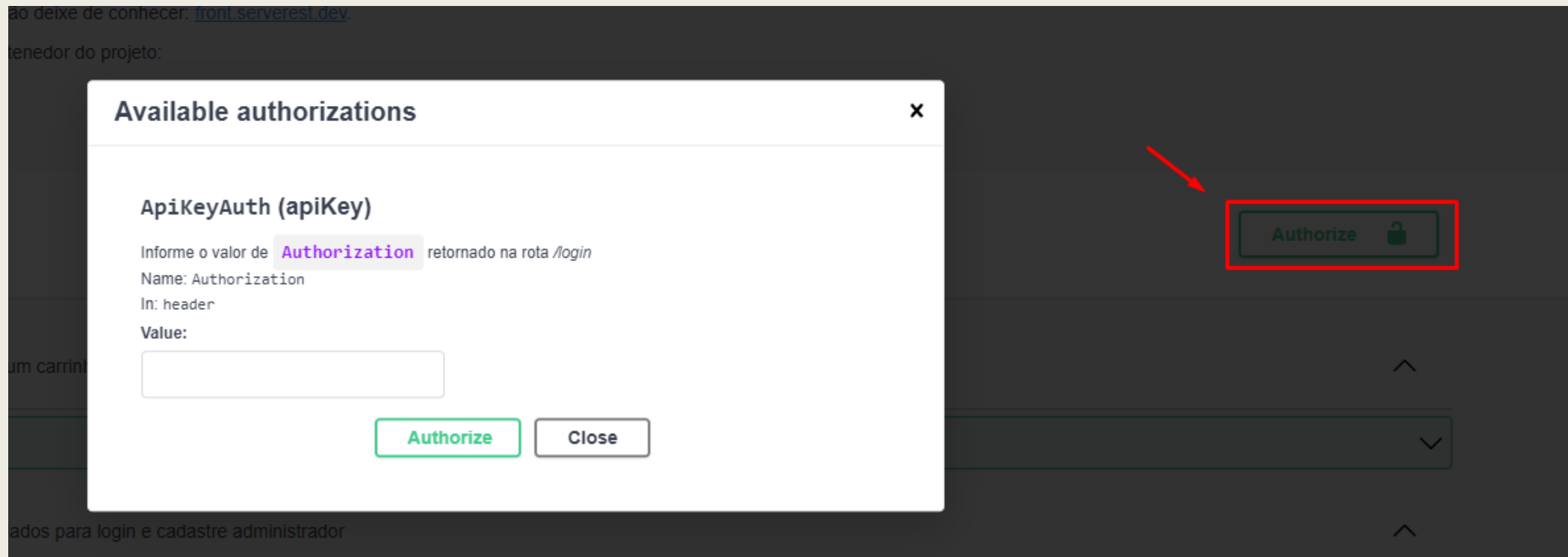
**POST**

**/login** Realizar login

# APIS DO SERVER REST

## LOGIN

- Para estar autenticado pelo Swagger precisamos colocar o token de autenticação neste local indicado



# APIS DO SERVER REST

## USUÁRIOS

- API utilizada para criar, editar, excluir e gerenciamento geral dos usuários
- Não é necessário nenhuma autenticação para utilizar essa API

Usuários <small>Gerencie os usuários, consulte dados para login e cadastre administrador</small>		
GET	/usuarios	Listar usuários cadastrados
POST	/usuarios	Cadastrar usuário
GET	/usuarios/{_id}	Buscar usuário por ID
DELETE	/usuarios/{_id}	Excluir usuário
PUT	/usuarios/{_id}	Editar usuário

# APIS DO SERVER REST

## PRODUTOS

- Todo processo de criação e gerenciamento dos produtos é feito por meio dessas APIs
- É necessário estar autenticado para utilizá-los

<b>Produtos</b> Consulte produtos cadastrados ou, como administrador, gerencie os produtos da loja		
GET	/produtos	Listar produtos cadastrados
POST	/produtos	Cadastrar produto
GET	/produtos/{_id}	Buscar produto por ID
DELETE	/produtos/{_id}	Excluir produto
PUT	/produtos/{_id}	Editar produto

# APIS DO SERVER REST

## CARRINHOS

- Nessa API podemos: Colocar um produto previamente cadastrado no carrinho, finalizar a compra ou apenas desistir desta comprar

**Carrinhos** Consulte os carrinhos cadastrados, faça um para o seu usuário e, no final, desista da compra pois estava só dando uma olhadinha

GET

/carrinhos Listar carrinhos cadastrados

POST

/carrinhos Cadastrar carrinho

GET

/carrinhos/{\_id} Buscar carrinho por ID

DELETE

/carrinhos/concluir-compra Excluir carrinho

DELETE

/carrinhos/cancelar-compra Excluir carrinho e retornar produtos para estoque



# ORIENTAÇÃO PARA OS TESTES

- Você deverá criar os casos de testes que JULGAR NECESSÁRIO para validar os resultados de cada requisição conforme as orientações de cada API
- Caso exista mais de um cenário de teste possível crie-os
- Parametrize tudo que for necessário em um environment
- Um caso de teste deve sempre ser independente dos demais
- Observe as possíveis respostas esperadas para cada end-point vão lhe ajudar a planejar os testes
- Você pode optar por organizar seus teste da forma que achar melhor
  - *Por comportamento*
  - *Por validações end-point por end-point*

CT 1 - USUÁRIO ADMIN TRUE (SUCESSO)

1 - Usuários

- > Create
- > Read
- > Update
- > Delete

3 - Produtos

- > Test Setup
- > Validações API Produto
  - > Create
  - > Read
  - > Update
  - > Delete

# PRÁTICA: API USUÁRIOS

- CENÁRIO 1: Faça um CRUD com sucesso de um usuário
  - *Tipo de usuário: true e false*
- CENÁRIO 2: Faça cenários de FALHA para cada endpoint dessa API
  - *Usuário já existente*
  - *Requisição sem todos os parâmetros*
  - *Falha na atualização de usuário*

# PRÁTICA: API LOGIN

- CENÁRIO 1: Faça um LOGIN com sucesso
- CENÁRIO 2: Faça cenários de FALHA
  - *Usuário e senha incorretos*
  - *Requisição com informações faltando*

# PRÁTICA: API PRODUTOS

- CENÁRIO 1: Faça um CRUD com sucesso da API PRODUTOS
- CENÁRIO 2: Faça cenários de FALHA
  - *Falta de token de autenticação na criação e atualização de produtos*
  - *Utilizar usuário não admin*
  - *2 produtos com o mesmo nome*
  - *Buscar produto que não existe*
  - *Erro na atualização de produtos*

# PRÁTICA: API CARRINHOS

- CENÁRIO 1: Faça um CRUD com sucesso da API CARRINHOS
  - *Usuário Admin true e admin false*
- CENÁRIO 2: Faça cenários de FALHA
  - *Falta de token de autenticação na utilização dos endpoints*
  - *Criar mais de 1 carrinho*
  - *Duplicar produto*
  - *Colocar produto no carrinho sem quantidade suficiente*
  - *Buscar carrinho que não existe*
  - *Tentar concluir-compra sem um carrinho*

# ESTUDO COMPLEMENTAR

- Playlist Gratuita no youtube

- <https://www.youtube.com/playlist?list=PLVQdoQTyi01iRnGqolawDKAoo3RIXSoQg>

- Curso Udemy

- <https://www.udemy.com/course/postman-do-basico-ao-avancado-com-projetos/?referralCode=3ED49D8F294C92F0B8A1>

# REFERÊNCIAS

- <https://aws.amazon.com/pt/what-is/api/>