

PLANO DE TESTE PARA A API SERVEREST

Edgard Ernesto D. Bezerra

Plano de Teste

Para a

API ServeRest

Introdução

Esse é o documento de plano de teste completo referente a API ServeRest, para auxílio do entendimento dos procedimentos de testes realizados na mesma, na tentativa de simplificar e organizar os testes de forma intuitiva.

Objetivos

Objetivo

ServeRest é uma API REST pública baseada em E-commerce, criada com o objetivo de servir como material de estudos de testes de API.

Escopo

Este documento descreve o plano de teste para todas as funcionalidades da API ServeRest. Tendo como objetivo:

- Apresentar o mapa mental completo criado para a API.
- Descrever as estratégias de teste a serem empregadas.
- Identificar e listar os testes prioritários.
- Identificar os testes que devem ser automatizados.

O objetivo é testar o funcionamento e desempenho da API, esperando-se que todos os endpoints sejam testados de forma individual ao menos uma vez.

Mapa mental

(segue em anexo juntamente com o plano de testes, o mapa mental completo da API)

Suíte de casos de teste

Login

CT-1: Realizar login válido	
Pré-requisitos: ter conta válida cadastrada na API	
Passos:	Descrição:

1	inserir email e senha validos	
Resultado esperado:		
status code:		mensagem:
200		OK
Pós-Condição:		
Retornar mensagem / token de autorizaçao do usuário		
CT-2: Realizar Login dados vazios		
Pré-requisitos: Nao possui		
Passos:	Descrição:	
1	mandar dados vazios	
Resultado esperado:		
status code:		mensagem:
400		Bad Request
Pós-Condição:		
Retornar mensagem de email e senha em branco		
CT-3: Realizar Login dados inválidos		
Pré-requisitos: mandar email e/ou senha nao cadastrado ou incorreto		
Passos:	Descrição:	
1	mandar dados inválidos	
Resultado esperado:		
status code:		mensagem:
401		Unauthorized
Pós-Condição:		
Retornar mensagem de email e senha inválidos		

Usuários

- Cadastro

CT-4: Cadastrar usuário	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	mandar dados válidos para cadastro
Resultado esperado:	
status code:	mensagem:
201	Created
Pós-Condição:	
Retornar mensagem / id do usuário	
CT-5: Cadastrar usuário já existente	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	enviar email ja cadastrado

Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de Email já utilizado	
CT-6: Cadastrar usuário com campos vazios	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	enviar objeto vazio para cadastro
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de campos obrigatórios	

- **Buscar usuário por ID**

CT-7: Buscar usuário por ID	
Pré-requisitos: Obter ID do usuário escolhido	
Passos:	Descrição:
1	Mandar a requisição com o ID do usuário
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar dados do usuário	
CT-8: Buscar usuário não existente ou inválido	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	Mandar a requisição com o ID inválido ou sem ID
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de usuário não encontrado	

- **Excluir usuário**

CT-9: Excluir usuário	
Pré-requisitos: Obter ID de um usuário cadastrado	
Passos:	Descrição:

1	Mandar a requisição com o ID do usuário
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem de usuário excluído com sucesso	
CT-10: Excluir usuário inexistente ou inválido	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	Mandar a requisição com o ID inválido ou sem ID
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem nenhum registro excluído	

- **Editar usuário**

CT-11: Editar usuário	
Pré-requisitos: Obter ID de um usuário cadastrado	
Passos:	Descrição:
1	Alterar dado do usuário
2	Mandar a requisição com o ID do usuário que quer alterar
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem de registro alterado com sucesso	
CT-12: Usuário não existente	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	Mandar body completo com dados válidos
2	Mandar a requisição com o ID inválido ou sem ID
Resultado esperado:	
status code:	mensagem:
201	Created
Pós-Condição:	
Retornar mensagem de cadastro / id do usuário	
CT-13: E-mail alterado já existente	
Pré-requisitos: ter conta válida cadastrada	
Passos:	Descrição:

1	Trocar e-mail de um usuário por outro que já está cadastrado	
Resultado esperado:		
status code:		mensagem:
400		Bad Request
Pós-Condição:		
Retornar mensagem este e-mail já está sendo usado		

- Listar usuários cadastrados

CT-14: Listar usuários cadastrados	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	Mandar requisição
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar lista de usuários cadastrados	

Produtos

- Listar produtos cadastrados

CT-15: Listar produtos cadastrados	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	Mandar requisição
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar lista de produtos cadastrados	

- Cadastrar produto

CT-16: Cadastrar produto	
Pré-requisitos: Estar logado como administrador	
Passos:	Descrição:
1	inserir token do usuário
2	Mandar requisição com os dados completos
Resultado esperado:	
status code:	mensagem:

201	Created
Pós-Condição:	
Retornar mensagem / ID do produto cadastrado	
CT-17: token ausente	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	mandar dados sem o token de autenticação
Resultado esperado:	
status code:	mensagem:
401	Unauthorized
Pós-Condição:	
Retornar mensagem de token de acesso invalido, expirado, ou usuário não existe mais.	
CT-18: Cadastrar produtos com mesmo nome	
Pré-requisitos: ter ao menos um produto cadastrado	
Passos:	Descrição:
1	mandar dados iguais ao produto já existente
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de já existe produto com esse nome	
CT-19: Cadastrar produto sem ADM	
Pré-requisitos: Estar logado numa conta de usuário sem administrador	
Passos:	Descrição:
1	mandar dados válidos para cadastro
Resultado esperado:	
status code:	mensagem:
403	Forbidden
Pós-Condição:	
Retornar mensagem de rota exclusiva para administradores.	

- **Buscar produto por ID**

CT-20: Buscar produto por ID	
Pré-requisitos: obter ID de um produto cadastrado	
Passos:	Descrição:
1	Mandar requisição com o id do produto
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	

Retornar dados do produto escolhido	
CT-21: Produto não existente / não encontrado	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	Enviar requisição vazia ou com id inválido
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de produto não encontrado	

- **Excluir produto**

CT-22: Excluir produto	
Pré-requisitos: Estar logado como administrador, recuperar token de usuário e obter id do produto	
Passos:	Descrição:
1	inserir id do produto na requisição e enviar
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem de registro excluído com sucesso	
CT-23: Token ausente	
Pré-requisitos: Não ter token válido na requisição	
Passos:	Descrição:
1	Enviar requisição com id valido
Resultado esperado:	
status code:	mensagem:
401	Unauthorized
Pós-Condição:	
Retornar mensagem de token ausente, invalido, expirado, ou usuário do token não existe mais.	
CT-24: Produto inexistente	
Pré-requisitos: Estar logado como administrador e recuperar token de usuário	
Passos:	Descrição:
1	mandar requisição com id inválido
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem de nenhum registro excluído	

CT-25: Excluir produto sem ser ADM	
Pré-requisitos: Estar logado em conta de usuário sem privilégios de administrador	
Passos:	Descrição:
1	mandar requisição com id válido
Resultado esperado:	
status code:	mensagem:
403	forbidden
Pós-Condição:	
Retornar mensagem de rota exclusiva para administradores	
CT-26: Excluir produto que está em um carrinho	
Pré-requisitos: Estar logado como administrador e recuperar token de usuário E ter carrinho criado com ao menos um produto	
Passos:	Descrição:
1	Mandar requisição com o id do produto que está em um carrinho
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem / id do carrinho em que o produto se encontra	

- **Editar produto**

CT-27: Editar produto	
Pré-requisitos: Estar logado como administrador	
Passos:	Descrição:
1	inserir body completo da requisição com alterações de sua preferencia
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem Registro alterado com sucesso	
CT-28: Produto vazio	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	mandar body com dados vazios
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar body com mensagens de campo obrigatório	

CT-29: Nome já existente	
Pré-requisitos: ter um produto cadastrado	
Passos:	Descrição:
1	Mudar nome do produto para um já existente e enviar
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de já existe produto com esse nome	
CT-30: Produto inexistente	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	inserir body completo de um produto ainda não cadastrado
Resultado esperado:	
status code:	mensagem:
201	Created
Pós-Condição:	
Retornar mensagem / id do produto	
CT-31: Editar produto sem ser ADM	
Pré-requisitos: fazer login com conta sem privilégios de administrador	
Passos:	Descrição:
1	mandar requisição com id válido
Resultado esperado:	
status code:	mensagem:
403	Forbidden
Pós-Condição:	
Retornar mensagem de rota exclusiva para administradores	

Carrinho

- Listar carrinhos cadastrados

CT-32: Listar carrinhos	
Pré-requisitos: Estar logado como administrador	
Passos:	Descrição:
1	Enviar requisição
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar lista de carrinhos criados	

- Cadastrar carrinhos

CT-33: Cadastrar carrinho	
Pré-requisitos: Estar logado	
Passos:	Descrição:
1	inserir body com produtos cadastrados
Resultado esperado:	
status code:	mensagem:
201	Created
Pós-Condição:	
Retornar mensagem / ID do carrinho cadastrado	
CT-34: Criar carrinho sem produtos	
Pré-requisitos: Estar logado	
Passos:	Descrição:
1	mandar dados vazios
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de produtos é obrigatório	
CT-35: Carrinho com produto inexistente	
Pré-requisitos: Estar logado	
Passos:	Descrição:
1	mandar dados inválidos
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de produto não encontrado	
CT-36: Cadastrar mais de um carrinho	
Pré-requisitos: Ter criado um carrinho com a conta logada	
Passos:	Descrição:
1	inserir body com produtos cadastrados
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de não é permitido mais de um carrinho	
CT-37: Token ausente ou expirado	
Pré-requisitos: Nao possui	

Passos:	Descrição:
1	enviar dados válidos
Resultado esperado:	
status code:	mensagem:
401	Unauthorized
Pós-Condição:	
Retornar mensagem de token de acesso ausente, inválido ou expirado	

- **Buscar carrinho por ID**

CT-38: Buscar carrinho por ID	
Pré-requisitos: Estar logado como administrador	
Passos:	Descrição:
1	Enviar requisição com ID de um carrinho válido
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar dados do carrinho e do usuário no qual ele pertence	
CT-39: Token	
Pré-requisitos: Não possui	
Passos:	Descrição:
1	Mandar requisição com ID inválido
Resultado esperado:	
status code:	mensagem:
400	Bad Request
Pós-Condição:	
Retornar mensagem de Carrinho não encontrado	

- **Excluir carrinho e concluir compra**

CT-40: Excluir carrinho concluir compra	
Pré-requisitos: Estar logado e ter um carrinho criado	
Passos:	Descrição:
1	Enviar requisição
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem de registro excluído com sucesso	
CT-41: Token ausente	

Pré-requisitos: Ter criado um carrinho	
Passos:	Descrição:
1	Não ter o token do usuário logado
Resultado esperado:	
status code:	mensagem:
401	Unauthorized
Pós-Condição:	
Retornar mensagem de token de acesso ausente, inválido ou expirado	

- Excluir carrinho e retornar produtos para o estoque

CT-42: Excluir carrinho e cancelar compra	
Pré-requisitos: Ter criado um carrinho	
Passos:	Descrição:
1	Enviar requisição
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem de registro excluído com sucesso	
CT-43: Carrinho inexistente	
Pré-requisitos: não ter criado um carrinho	
Passos:	Descrição:
1	mandar requisição
Resultado esperado:	
status code:	mensagem:
200	OK
Pós-Condição:	
Retornar mensagem de não foi encontrado um carrinho para esse usuário	
CT-44: Token inválido	
Pré-requisitos: Ter criado um carrinho	
Passos:	Descrição:
1	mandar requisição sem token do usuário
Resultado esperado:	
status code:	mensagem:
401	Unauthorized
Pós-Condição:	
Retornar mensagem de token de acesso ausente, inválido ou expirado	

Estratégia de testes

- **Teste de funcionalidade**

A estratégia usada para os testes da API ServeRest consiste no teste individual de todas as suas requisições, ou em forma de fluxo para verificar se os resultados esperados corresponderão aos dados enviados e se os dados inválidos oferecem mensagens de erro.

Os critérios para a conclusão dos testes será se todos os testes propostos foram executados e verificados.

- **Teste de segurança e controle de acesso**

O objetivo é garantir que determinados usuários utilizem apenas as funções que o mesmo possua acesso

A técnica consiste em fazer as requisições com e sem um usuário administrador.

Os critérios para a conclusão são as permissões das funções estarem de acordo para cada tipo de usuário.

Prioridades

Na API ServerRest vamos priorizar tudo aquilo que é necessário para exercer sua função de E-Commerce, então temos fluxos criados com o obrigatório para a API se tornar funcional.

Login:

Sem a rota login não é possível interagir com a API, sendo assim, é a rota mais importante

Produtos

Para o funcionamento é necessário que as rotas de produto estejam todas funcionais

Carrinho

Só é possível finalizar transações se a rota de carrinhos estiver funcionando, pois é onde estão os únicos endpoints que fazem essa função.

Usuários

Rota menos importante, o único endpoint que realmente importa para o funcionamento é o de criar conta.

Candidatos a automação

O critério para definição dos pontos de automação foi a importância das funcionalidades envolvidas, e o risco ao negócio que o mal funcionamento desses componentes podem causar.

- **Utilização do token de usuário**
- **Login automático para todas as rotas**

Ferramentas

FERRAMENTA	FINALIDADE	VERSÃO
Microsoft Word	Gerenciamento do projeto	1804 (16.0.9226.2156)
Microsoft Excel	Gerenciamento do projeto	1804 (16.0.9226.2156)
Postman	Teste de API	v9.24.1
Xmind	Desenvolvimento de mapa mental	64bit-12.0.3-202206241736

Observações complementares:

O cadastro na plataforma não é opcional para utilização da API, sendo também um requisito necessário para obtenção do token. Vale lembrar que o token de acesso só é válido por uma hora, após isso será necessário gerar novamente