

Plano de Testes

Software: Fake Store API

QA responsável: Elisama dos Santos Gregorio

1. Introdução

Este **Plano de Teste** foi elaborado para definir o objetivo, o escopo, a estratégia, o ambiente e os critérios dos testes que serão realizados para testar o sistema Fake Store API, além das informações a respeito dos entregáveis e o cronograma de atividades.

2. Objetivo dos testes

Garantir que os endpoints da FaceStore API estejam funcionando conforme o esperado, retornando os dados corretos, com os status HTTP apropriados, respeitando as regras de negócio, formatos e métodos definidos.

3. Escopo dos testes

Testar os seguintes recursos da API:

- GET /products
- GET /products/{id}
- POST /products
- PUT /products/{id}

- DELETE /products/{id}
- GET /carts
- GET /users
- GET /auth/login

4. Abordagem dos testes

Testes manuais: Postman

Testes automatizados: **JavaScript + Newman**

Verificação de:

- Status Code
- Corpo da resposta (estrutura e dados esperados)
- Validação de headers
- Comportamento com dados válidos e inválidos
- Testes de segurança básicos (ex: login com dados inválidos)

5. Ferramentas

Postman

Newman (para automatizar testes Postman)

Testes automatizados com:

JavaScript

6. Critérios dos testes

- Todos os endpoints devem retornar status HTTP 200 (ou adequado) para requisições válidas.
- Requisições inválidas devem retornar status apropriado (400, 401, 404...).
- As respostas devem ter o formato JSON esperado.
- Campos obrigatórios devem ser validados corretamente.
- Operações como **POST**, **PUT**, e **DELETE** devem refletir mudanças nos dados da API (ou simular corretamente, se for fake/mock).

7. Riscos e Mitigações

Risco	Descrição	Impacto	Ação de Mitigação
Exposição de dados sensíveis	A API não deve retornar senhas, hashes ou dados pessoais além do necessário	Alto	Garantir mascaramento e exclusão de dados sensíveis nas respostas

Falta de autenticação em endpoints críticos	Endpoints como /carts e /users podem ser acessados sem token (risco de acesso indevido)	Alto	Implementar autenticação JWT obrigatória
Inconsistência de dados entre entidades	Produtos no carrinho com IDs inválidos, usuários inexistentes etc.	Médio	Criar validação cruzada entre entidades relacionadas
Performance instável	Listagens grandes podem levar a lentidão sem paginação	Médio	Implementar paginação e cache

8. Entregáveis

- Execução de todos os casos de teste com evidências (prints ou logs).
- Relatório com resultados (sucesso/falha)
- .Relatório de Bugs
- Scripts de automação