

**PLANO DE TESTES PARA A API DA SERVEREST**

**03/10/2023**

**Sumário**

**Introdução3**

**Objetivos3**

**Escopo3**

**Mapa Mental4**

**Suíte de Caso de Testes4**

**Estratégia de Testes7**

**Prioridades7**

**Testes Candidatos a Automação7**

**Ferramentas8**

**Cronograma8**

1. **Introdução**

A API da ServeRest (serverest.dev) é basicamente um CRUD de um ecommerce, com login, gerenciamento de usuários, produtos e carrinho. A API também serve de backend para a front da mesma empresa, que pode ser visitado em “[front.serverest.dev](https://front.serverest.dev/)”.

# Objetivos

Este plano de testes tem como objetivo estabelecer um conjunto de testes para validar as principais funcionalidades da API. Os testes em questão ajudarão a garantir que funcionalidades como fazer login, gerenciar um usuário, gerenciar um produto ou mesmo adicionar produtos ao carrinho e concluir compra estejam funcionando conforme o esperado.

# Escopo

O plano de testes para o site da ParaBank incluirá as princiais funcionalidades da API, que são login, gerenciar usuário, gerenciar produtos e gerenciar carrinho de compra.

Além das funcionaldades, serão explorados as diversas respostas para cada um destas funcionalidades, explorando o Swagger da API e buscando ainda outras requisições que não estão ali informadas.

# Mapa Mental

# 

# Suíte de Casos

Suíte de Teste CT01: Login

* CT101: Validar login com campos vazios;
* CT102: Validar login com uma conta válida;
* CT103: Validar login com uma conta inválida;
* CT104: Validar login com email inválido;

Suíte de Teste CT02: Usuários

* CT201: Validar listar todos;
* CT202: Validar cadastro de usuário válido;
* CT203: Validar cadastro de usuário email inválido;
* CT204: Validar cadastro de usuário com campos vazios;
* CT205: Validar cadastro de usuário com email já utilizado;
* CT206: Validar cadastro de usuário com email do provedor Gmail;
* CT207: Validar cadastro de usuário com email do provedor Hotmail;
* CT208: Validar cadastro de usuário com senha menor que 5 caracteres;
* CT209: Validar cadastro de usuário com senha maior que 10 caracteres;
* CT210: Validar listagem com id cadastrado;
* CT211: Validar listagem com id não cadastrado;
* CT212: Validar delete usuário com id cadastrado;
* CT213: Validar delete usuário com id não cadastrado;
* CT214: Validar delete usuário com id vazio;
* CT215: Validar editar usuário com id vazio;
* CT216: Validar editar usuário com campos vazios;
* CT217: Validar editar usuário com dados válidos;
* CT218: Validar editar usuário com novo id;
* CT219: Validar editar usuário com email já utilizado;

Suíte de Teste CT03: Produtos

* CT301: Validar listar todos;
* CT302: Validar cadastro de produto válido;
* CT303: Validar cadastro de produto válido sem token;
* CT304: Validar cadastro de produto nome já utilizado;
* CT305: Validar cadastro de produto campos vazios;
* CT306: Validar cadastro de produto sem ser administrador;
* CT307: Validar listagem com id cadastrado;
* CT308: Validar listagem com id não cadastrado;
* CT309: Validar delete produto com id cadastrado;
* CT310: Validar delete produto com id não cadastrado;
* CT311: Validar delete produto com que está em um carrinho;
* CT312: Validar delete produto com id vazio;
* CT313: Validar editar produto com id vazio;
* CT314: Validar editar produto com campos vazios;
* CT315: Validar editar produto com dados válidos;
* CT316: Validar editar produto com dados válidos sem token;
* CT317: Validar editar produto com novo id;
* CT318: Validar editar produto com nome já utilizado;

Suíte de Teste CT04: Carrinhos

* CT401: Validar listar todos;
* CT402: Validar cadastro de carrinho;
* CT403: Validar cadastro de carrinho sem token;
* CT404: Validar cadastro de carrinho, já tendo um carrinho;
* CT405: Validar cadastro de carrinho sem informar produto;
* CT406: Validar listagem com id cadastrado;
* CT407: Validar listagem com id não cadastrado;
* CT408: Validar delete concluir compra com carrinho aberto;
* CT409: Validar delete concluir compra sem carrinho aberto;
* CT410: Validar delete concluir compra com carrinho aberto e sem token;
* CT411: Validar delete cancelar compra com carrinho aberto;
* CT412: Validar delete cancelar compra sem carrinho aberto;
* CT413: Validar delete cancelar compra com carrinho aberto e sem token;

1. **Estratégia de teste**

Neste plano de teste para a API da ServeRest, os testes foram divididos pelas funcionalidades e criamos suítes de caso de testes, priorizando as funcionalidades necessárias para que de fato a aplicação cumpra com eu papel como ecommerce.

Para cada funcionalidade deverão ser explorados os métodos utilizados (GET, POST, PUT e DELETE) e também as possibilidades dentro de cada método, para assim obtermos uma maior cobertura dos testes.

Os defeitos e melhorias encontrados serão reportados em uma ferramenta de gerenciamento de defeitos, com as evidências necessárias para a reprodução e a correção dos mesmos.

1. **Prioridades**

Como itens prioritários de testes foram adotados os casos:

* Suíte de Teste CT01: CT102 e CT103
* Suíte de Teste CT02: CT201, CT202, CT203, CT204 e CT205;
* Suíte de Teste CT03: CT301, CT302, CT303, CT304, CT305 e CT306;
* Suíte de Teste CT04: CT401, CT402, CT403, CT404, CT405, CT408, CT409 e CT410;

Foram definidos como prioritários as suítes de testes acima, visto que em caso de mal funcionamento ou a não execução, impedem o principal objetivo da API, que é ser servir ecommerce para realizar vendas.

1. **Testes Candidatos à automação**

Todos os casos de testes elencados neste plano são candidatos à automação e deverão após serem testados manualmente, serem automatizados.

1. **Ferramentas**

Para a realização do plano de testes em questão as ferramentas utilizadas serão:

* Testes exploratórios e manuais;
* Postman;
* Swagger;
* Linguagem de programação JavaScript

1. **Cronograma**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **ATIVIDADES** |  | **25/set** | **28/set** | **02/out** | **04/out** | **06/out** |
| **Planejamento** |  |  |  |  |  |  |
| **Execução do Plano de Testes** |  |  |  |  |  |  |
| **Testes exploratórios** |  |  |  |  |  |  |
| **Relatório de Bugs encontrados** |  |  |  |  |  |  |
| **Apresentação relatório de testes** |  |  |  |  |  |  |
|  |  |  |  |  |  |  |