



IURA KURTZ FILHO

PLANO DE TESTES

ServeRest

SUMÁRIO

INTRODUÇÃO.....	1
OBJETIVOS.....	2
ESCOPO.....	3
MAPA MENTAL.....	4
SUÍTE DE CASOS DE TESTE.....	5
ESTRATÉGIA DE TESTE.....	6
PRIORIDADES.....	7
CANDIDATOS PARA AUTOMAÇÃO.....	7
FERRAMENTAS.....	8

INTRODUÇÃO

O seguinte plano de teste delineou a estratégia para efetivar as verificações da API ServeRest de teste.

Esta plataforma REST oferece suporte a todos os principiantes no domínio de QA, facilitando a aprendizagem, compreensão e avaliação da quantidade e qualidade de testes viáveis para uma API.

Dentro deste aplicativo, encontramos uma representação simplificada de um ambiente de compras que nos habilita a executar o login com um token, bem como criar e remover usuários e produtos, além de conduzir testes de fluxo de compras.

OBJETIVOS

Este documento tem como objetivo descrever um plano de teste para a aplicação da API ServeRest.

- Identificar as informações do projeto e o que deve ser testado de acordo com a documentação do Swagger.
- Verificar se a documentação esta de acordo com as regras do negocio bem como seus requisitos.
- Descrever as estratégias que deverão ser utilizadas
- Identificar e fornecer os recursos e estimativa de esforços de teste

ESCOPO

O plano de testes para o site ServRest incluirá os testes prioritários para os fluxos principais das operações, que envolvem as seguintes etapas:

Cadastro – (Usuários , Produtos e Carrinho)

Login – (Usuários e Administradores)

Listar – (Usuários , Produtos e carrinho)

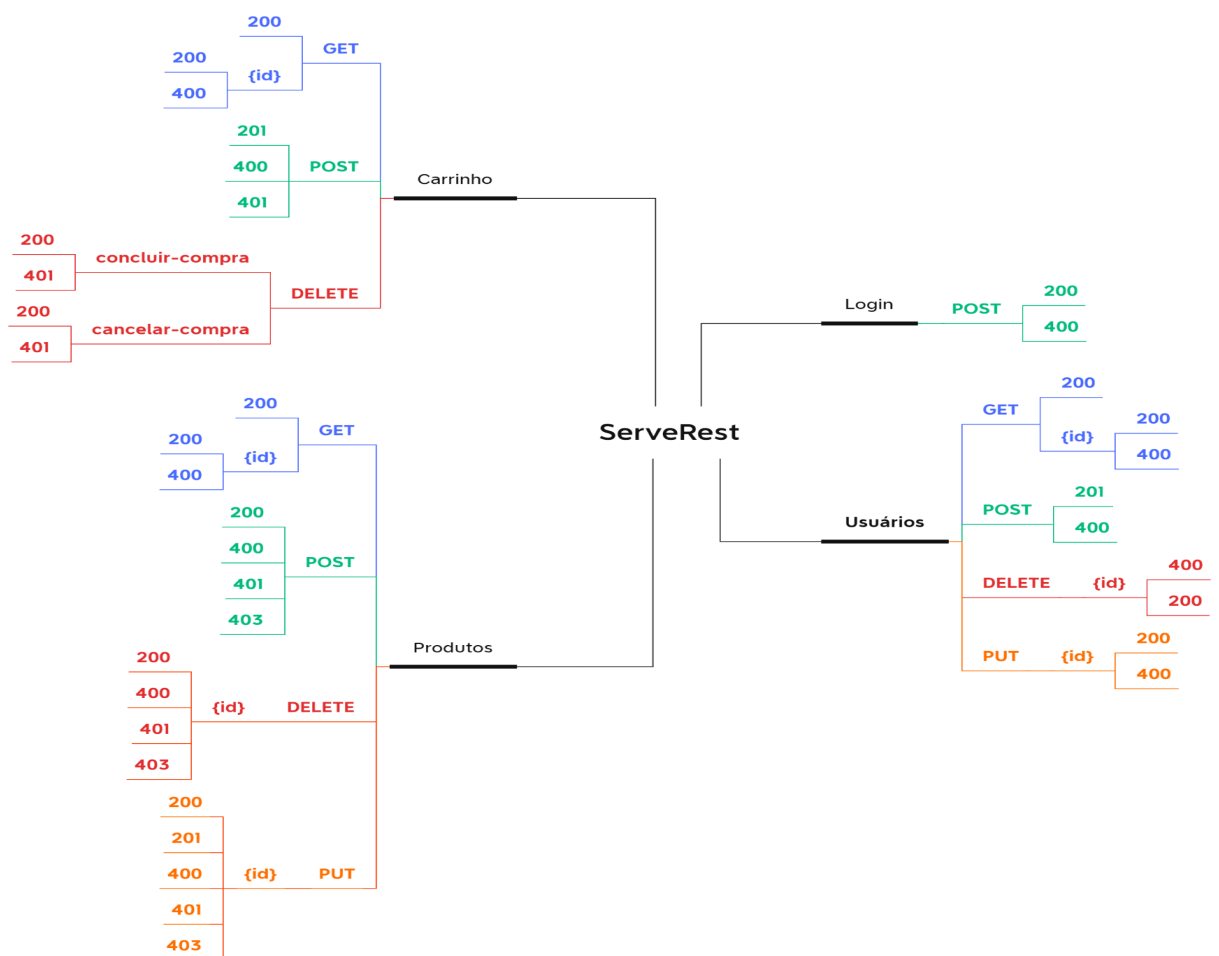
Modificar – (Usuários , Produtos e carrinho)

Deletar – (Usuários , Produtos e carrinho)

Verificar Integração de ID gerados – (Usuários , Produtos e Carrinho)

Verificar Integração de Token gerado – (Usuários)

MAPA MENTAL



SUÍTE DE CASOS DE TESTE

Caso de Teste CT1: Login

- CT 101: Login válido;
- CT 102: Validar login com email inválido;
- CT 103: Validar login com usuário não cadastrado;

Caso de Teste CT2: Usuários

- CT 201: Validar cadastro de usuário;
- CT 202: Validar cadastro com email já cadastrado;
- CT 203: Validar busca por todos usuários cadastrados;
- CT 204: Validar alteração de dados de usuário cadastrado;
- CT 205 Validar alteração de dados de usuário não encontrado;
- CT 206 Validar exclusão de usuário cadastrado
- CT 207 validar exclusão de usuário com carrinho;

Caso de Teste CT03: Produtos

- CT 301: Validar cadastro de produto;
- CT 302: Validar cadastro de produtos com mesmo nome;
- CT 303: Validar busca por todos produtos cadastrados;
- CT 304: Validar busca por determinado produto;
- CT 305: Validar edição de determinado produto;
- CT 306: Validar Exclusão de determinado produto;
- CT 307: Validar exclusão de produto que faz parte de um carrinho;

Caso de Teste CT 04: Carrinho

- CT 401: Validar criação de um carrinho;
- CT 402: Validar busca por todos carrinhos existentes;
- CT 403: Validar busca por determinado carrinho;
- CT 404: Validar edição de determinado carrinho;
- CT 405: Validar exclusão de determinado carrinho;
- CT 406 Validar cancelamento de compra onde o carrinho da compra é excluído;
- CT 407: Validar conclusão de compra do carrinho;

ESTRATÉGIA DE TESTE

Em relação à estratégia de teste adotada para a API ServerREST, observamos e consideramos que a API tem um número limitado de endpoints. Portanto, podemos abordá-la de forma abrangente, garantindo a inclusão de token e ID em todos os processos. Dessa forma, podemos executar testes de:

- Funcionalidade;
- Integração;
- Validação de rotas;

Utilizando a ferramenta Postman. Isso é feito enquanto seguimos e comparamos com a documentação Swagger e o mapa mental.

PRIORIDADES

Os testes escolhidos para priorizar foram os que se incluem dentro de um fluxo de compras:

CT 201- validar rota Cadastro de Usuário

CT 301- validar rota de Cadastro de Produto

CT 401- validar rota de Cadastro de Carrinho

CT 101- validar rota de Login de Usuário

CT 407- validar rota de conclusão de compra do carrinho

CT 406- validar rota de cancelamento de compra do carrinho

CANDIDATOS PARA AUTOMAÇÃO

Como candidatos para automação foi escolhido o Fluxo de Compras – Verificado através do Mapa Mental

- Cadastro de Usuário – criação de _idUser
- Autenticação convencional – criação de Bearer token
- Adição produto ao carrinho criação de _idCart
- Finalizar Processo de Compra

FERRAMENTAS

Para realização do plano de teste em questão as ferramentas utilizadas foram:

- Postman, ferramenta para interagir com as rotas da API, através do Postman é possível realizar a execução do testes;
- Xmind, para mapeamento mental, capturar ideias e gerenciar informações complexas para melhor entendimento da aplicação;
- Git e GitHub. para realizar commits e poder trabalhar em diferentes branches sem atrapalhar o trabalho de outros colegas.

CRONOGRAMA

ATIVIDADES	22/09	26/09	28/09	01/10	04/10	06/10	08/10
	25/09	27/09	30/09	03/10	05/10	07/10	08/10
Tirar dúvidas							
Testes exploratórios							
Revisão dos requisitos e preparação do ambiente de teste							
Execução dos testes							
Análise dos resultados dos testes e relatório de bugs							
Revisão final e preparação do relatório de teste							