|  |  |
| --- | --- |
| DOCUMENTAÇÃO DE TESTES -DESAFIO JUSTA |  |
| João Victor de Oliveira Santos | A garantia da qualidade do software deve ser uma preocupação de toda a equipe de desenvolvimento durante todo o ciclo de vida do software.  Mesmo que todos entrem em concordância com esta afirmação, dentro de um projeto de software, a etapa de testes é o momento em que nos certificamos de que o software foi construído de maneira correta e que atende aos requisitos de software levantados durante o processo de análise e especificação de requisitos. |

**OBJETIVOS DO DOCUMENTO**

Este documento tem como objetivo identificar os componentes de software e requisitos a serem testados e descrever as estratégias de testes a serem utilizadas. Neste plano de testes também será fornecida uma estimativa dos esforços e recursos empregados e, será descrito o planejamento dos testes, a fim de que a execução destes seja acompanhada em detalhes e alcance de maneira prática e organizada os seus objetivos.

**ESCOPO DO PRODUTO**

O sistema a ser testado é e-commerce onde foi automatizado a rotina de compra do site https://www.saucedemo.com/.

Os testes lidarão com diversos campos com seletores diferentes para selecionar e irão avaliar o funcionamento e desempenho do sistema como um todo.

O teste de aceitação consistirá na utilização do sistema final pelo usuário a fim de testar cada requisito implementado, para que finalmente o produto seja validado de três cenários que foram solicitados, que são:

1. Compra de uma camisa vermelha;
2. Compra de uma bolsa e ordenação de preços do menor para o maior;
3. Seleção de dois produtos com valores menores.

**TESTE FUNCIONAL**

**Descrição:**

Os testes funcionais permitem que os testes ocorram de uma forma mais eficiente e rápida, possibilitando encontrar as não conformidades do software.

**Testes executados no Projeto:**

**Teste 1**

* Fazer o login utilizando os dados disponíveis;
* Clicar no botão login;
* Selecionar a camisa vermelha para o carrinho de compras;
* Timer de 2 segundos de segurança caso não carregue os produtos a tempo;
* Selecionar o botão de carrinho de compras;
* Página de carrinho de compras - clicar no botão de checkout;
* Página de inserir 3 informações do comprador utilizando type;
* Clicar em continuar;
* Página final de método de pagamento, onde clicamos em finish.

**Teste 2**

* Fazer o login utilizando os dados disponíveis;
* Clicar no botão login;
* Selecionar a bolsa para o carrinho de compras;
* Timer de 2 segundos de segurança caso não carregue os produtos a tempo;
* Selecionar o botão de carrinho de compras;
* Página de carrinho de compras - clicar no botão de checkout;
* Página de inserir 3 informações do comprador utilizando type;
* Clicar em continuar;
* Página final de método de pagamento, onde clicamos em finish.

**Teste 3**

* Fazer o login utilizando os dados disponíveis;
* Clicar no botão login;
* Utilizar comando select para alterar o filtro para seleção de menor para maior preço;
* A partir da nova seleção, selecionar os dois primeiros resultados que são os menores preços;
* Timer de 2 segundos de segurança caso não carregue os produtos a tempo;
* Selecionar o botão de carrinho de compras;
* Página de carrinho de compras - clicar no botão de checkout;
* Página de inserir 3 informações do comprador utilizando type;
* Clicar em continuar;
* Página final de método de pagamento onde clicamos em finish.

**PLANEJAMENTO**

Cada teste foi rodado 20 vezes para carga e busca por erro, onde restaram testes sem erros e objetivos.