**Documentação do Projeto de Automação de Testes – Sauce Demo**

1. **Introdução**

Esse documento descreve o desenvolvimento e implementação dos testes automatizados na aplicação web [**Sauce Demo**](https://www.saucedemo.com/). Os testes foram desenvolvidos utilizando Playwright, Python e pytest.

1. **Estrutura do Projeto**

O projeto segue o padrão de desgin Page Object Model (POM) para organizar o código de forma que facilite a manutenção e ajude na escalabilidade.

1. **Page Objects (Diretório pages/)**
   1. **login\_page.py**

**Propósito:** Modelar a página de login da aplicação Sauce demo.

**Funções:**

* **goto** – Navegar para a página de login
* **login** – Preencher os campos de usuário e senha
* **get\_error\_message** - Capturar mensagem de erro
  1. **inventory\_page.py**

**Propósito:** Modelar a página de produtos da aplicação

**Funções:**

* **get\_inventory\_items\_count –** Conta o total de itens exibidos na pagina
* **add\_item\_to\_cart** – Adiciona um item específico ao carrinho com base em seu índice na lista.
* **go\_to\_cart** – Vai para página de compras ao clicar no ícone do carrinho
* **sort\_products** – Ordena os produtos na página de acordo com a opção selecionada no menu dropdown
* **get\_product\_names** – Retorna uma lista com os nomes de todos os produtos exibidos na página

1. **Arquivos de Teste (Diretório tests/)**
   1. **authentication/test\_login.py**

**Propósito:** Testar todas as funcionalidades da página de login

**Testes Implementados:**

* **test\_login\_valid\_user** – Login com credenciais válidas
* **test\_login\_invalid\_user** – Login com credenciais inválidas
* **test\_error\_message\_on\_empty\_login** – Validação do campo de usuário vazio
* **test\_error\_message\_on\_empty\_password** – Validação do campo de senha vazio
* **test\_login\_locked\_user** – Login com usuário bloqueado
  1. **authentication/test\_logout.py**

**Propósito:** Testar a funcionalidade de logout

**Testes Implementados:**

* **test\_logout\_functionality** – Verificação do processo completo de logout após login
  1. **cart/test\_cart.py**

**Propósito:** Testar as funcionalidades relacionadas ao carrinho de compras

**Testes implementados:**

* **test\_cart\_icon –** Verificação da atualização do ícone do carrinho
* **test\_add\_item\_to\_cart –** Adição de item no carrinho
* **test\_correct\_item –** Verificação do item correto no carrinho
* **test\_add\_multiple\_items\_to\_cart –** Adição de múltiplos itens no carrinho
* **test\_cart\_remove\_item –** Remove item do carrinho
  1. **checkout/test\_checkout.py**

**Propósito:** Testar o processo de finalização de compra

**Testes implementados:**

* **test\_checkout\_process** – Verificação do fluxo completo de checkout
* **test\_checkout\_required\_fields** – Validação de campos obrigatórios
* **test\_checkout\_success\_message** – Verificação da mensagem de sucesso após finalização
  1. **inventory/test\_inventory.py**

**Propósito: Testar as funcionalidades da página de listagem de produtos**

**Testes Implementados:**

* **test\_inventory\_items\_count –** Verifica a existência de produtos na pagina
* **test\_sort\_inventory\_items –** Ordenação de produtos por preço
* **test\_sort\_products\_a\_z –** Ordenação de produtos de A-Z
* **test\_sort\_products\_z\_a –** Ordenação de produtos Z-A

1. **Arquivos de Configuração**
   1. **conftest.py**

**Proposito:** Definir configurações e fixtures para os testes

**Funções:**

* setup – Fornecera fixture setup que é utilizada por todos os testes, e configurar o ambiente de teste (browser)
  1. **requirements.txt**

**Proposito:** Listar todas as dependências para o projeto

* 1. **README.md**

**Proposito:** Fornecer instruções de configuração e execução dos testes

1. **Considerações Finais**

A estrutura do projeto segue boas práticas de automação de testes, sendo elas:

* **Page Object Model**
* **Oraganização de testes por funcionalidades**
* **Testes independentes**
* **Fixture reutilizável**

Com essa abordagem o projeto se torna escalável, permitindo adição de novos testes e manutenção dos existentes