

Contexto do teste:

Abaixo, estão as instruções para a realização da avaliação técnica.

O objetivo é conhecermos um pouco mais da sua entrega:

- 1-Escolha um site de sua preferência, os fluxos de sua preferência e realize os passos:
 - 2-Escreva a história do usuário e critérios de aceite
 - 3-Escreva os casos de testes
 - 4-Estime o tempo de teste e demonstre como fez o cálculo
 - 5-[Cypress] Desenvolva os testes automatizados dos fluxos que entender - demonstre a sua escolha
 - 6-[github actions] Crie o arquivo de pipeline para a execução dos seus testes
 - 7-[Github] Disponibilize o link do github com seu código versionado
-

1. Escolher o site e os fluxos (16:20 – 17:00)

Site escolhido: [Sauce Demo](#)

Fluxos escolhidos:

1. Login
 - 1.1. Login sucesso
 - 1.2. Login erro
2. Adicionar produtos ao carrinho
3. Remover produtos do carrinho
4. Checkout:
 - 4.1. finalização de compra – sucesso
 - 4.2. finalização de compra – erro
5. Ordenação de produtos

2. Escrever a história do usuário e critérios de aceite (17:00 – 17:15)

2.1. RN1: Login no Sauce Demo

- **História do Usuário:**

"Como um usuário do Sauce Demo, quero fazer login no sistema para visualizar o catálogo e comprar produtos."

- **Critérios de Aceite:**

- ✓ O usuário deve ser autenticado e redirecionado para a página de produtos após login bem-sucedido.
- ✗ Caso as credenciais sejam inválidas, deve ser exibida uma mensagem de erro: **"Epic sadface: Username and password do not match any user in this service"**
- ✗ Usuários bloqueados não devem conseguir acessar a aplicação e exibirá a mensagem: **"Epic sadface: Sorry, this user has been locked out."**

2.2. RN2: Adicionar produtos ao carrinho

- **História do Usuário:**

"Como um usuário logado, quero adicionar produtos ao carrinho para finalizar uma compra."

- **Critérios de Aceite:**

- ✓ O usuário pode adicionar qualquer produto ao carrinho.
- ✓ O número de itens no ícone do carrinho deve ser atualizado corretamente.

2.3. RN3: Remover produtos do carrinho

- ✓ **Critérios de Aceite**

- O usuário deve conseguir remover um ou mais itens do carrinho individualmente.
- O carrinho deve atualizar a quantidade de itens corretamente após a remoção.
- O sistema deve exibir um carrinho vazio caso todos os itens sejam removidos.
- Se um item for removido, ele não deve reaparecer ao recarregar a página.

- ✗ **Critérios de Rejeição**

- O botão "Remover" não deve estar desativado ou ausente.
- O produto não pode continuar aparecendo no carrinho após ser removido.

2.4. RN4: Checkout (Finalização de Compra)

2.4.1 Finalização de Compra – Sucesso

- **História do Usuário**

"Como um usuário do Sauce Demo, quero finalizar minha compra para receber os produtos em meu endereço."

- ✓ **Critérios de Aceite**

- O usuário deve conseguir acessar o checkout apenas se houver produtos no carrinho.
- O checkout deve exigir que o usuário preencha os campos obrigatórios: **Nome, Sobrenome e CEP**.
- Após preencher os dados corretamente, o usuário deve conseguir concluir a compra.
- O sistema deve exibir uma mensagem de sucesso ao finalizar a compra.
- O carrinho deve ser esvaziado após a compra.

- ✗ **Critérios de Rejeição**

- O usuário não pode finalizar a compra sem preencher todos os campos obrigatórios.
- O sistema não pode permitir que um usuário finalize a compra sem produtos no carrinho.
- O preço total deve ser calculado corretamente no resumo da compra.
- Se o pedido for finalizado, os itens não podem permanecer no carrinho ao recarregar a página.

2.4.2 Finalização de Compra – Erro

- **História do Usuário**

"Como um usuário do Sauce Demo, quero ser informado sobre erros ao tentar finalizar a compra para corrigir os dados antes de concluir o pedido."

- ✓ **Critérios de Aceite**

- O sistema deve exibir mensagens de erro se algum campo obrigatório não for preenchido.
- O botão "Continue" deve permanecer desativado até que todos os campos obrigatórios sejam preenchidos.
- Se o usuário tentar prosseguir sem preencher o **Nome, Sobrenome** ou **CEP**, uma mensagem de erro deve ser exibida.
- O sistema deve permitir a correção dos dados e permitir nova tentativa de finalização.

- ✗ **Critérios de Rejeição**

- O sistema não pode permitir checkout sem produtos no carrinho.
- O sistema não pode finalizar uma compra com campos obrigatórios em branco.
- Se houver erro, o usuário não pode ser redirecionado para a tela de sucesso.
- O sistema não deve perder os dados preenchidos ao exibir a mensagem de erro.

2.5. RN5: Ordenação de Produtos

- **História do Usuário**

"Como um usuário do Sauce Demo, quero ordenar os produtos de diferentes maneiras para facilitar a escolha do item desejado."

- ✓ **Critérios de Aceite**

- O usuário deve conseguir ordenar produtos por Nome (A-Z) e Nome (Z-A).
- O usuário deve conseguir ordenar produtos por Preço (Menor para Maior) e Preço (Maior para Menor).
- O sistema deve atualizar a exibição dos produtos imediatamente ao selecionar um critério de ordenação.

- ✗ **Critérios de Rejeição**

- O sistema não pode exibir produtos fora da ordem definida.
- O sistema não pode redefinir a ordenação ao recarregar a página.

3. Escrever os casos de teste (17:15 –18:12)

RN1-CT1: Login com usuário válido

- **Pré-condição:** Acessar a página de login do Sauce Demo.
- **Passos:**
 1. Inserir `standard_user` no campo de usuário.
 2. Inserir `secret_sauce` no campo de senha.
 3. Clicar no botão de login.
- **Resultado esperado:** Sistema exibe o catálogo de itens a venda.

RN1-CT2: Login com usuário inválido

- **Pré-condição:** Acessar a página de login do Sauce Demo.
- **Passos:**
 1. Inserir invalid_user no campo de usuário.
 2. Inserir secret_sauce no campo de senha.
 3. Clicar no botão de login.
- **Resultado esperado:** Sistema exibe a mensagem: “Epic sadface: Username and password do not match any user in this service”.

RN2-CT1: Adicionar produtos ao carrinho

- **Pré-condição:** Usuário precisa estar logado.
- **Passos:**
 1. No produto ‘Sauce Labs Backpack’, clicar em ‘Add to cart’.
 2. No produto ‘Sauce Labs Bike Light’, clicar em ‘Add to cart’.
- **Resultado esperado:** Os produtos escolhidos exibem botão ‘Remove’ e o ícone do carrinho no canto direito-superior atualiza o total de itens selecionados.

RN3-CT1: Remover produtos do carrinho

- **Pré-condição:** Usuário precisa estar logado e com itens no carrinho.
- **Passos:**
 1. No produto ‘Sauce Labs Backpack’, clicar em ‘Remove’.
 2. No produto ‘Sauce Labs Bike Light’, clicar em ‘Remove’.
 3. Clicar no ícone do carrinho.
- **Resultado esperado:** Os produtos escolhidos exibem botão ‘Add to cart’, o ícone do carrinho no canto direito-superior e a lista de itens do carrinho ficam zerados.

RN4-CT1: Finalização de Compra – Sucesso

- **Pré-condição:** Usuário precisa estar logado e com itens no carrinho.
- **Passos:**
 1. Clicar no ícone do carrinho.
 2. Clicar no botão ‘Checkout’
 3. Preencher os campos: ‘First Name’, ‘Last Name’, ‘Zip/Postal Code’
 4. Clicar no botão ‘Continue’
 5. Clicar no botão ‘Finish’
- **Resultado esperado:** Sistema redireciona para a tela de ‘Checkout: Complete!’.

RN4-CT2: Finalização de Compra – Erro

- **Pré-condição:** Usuário precisa estar logado e com itens no carrinho.
- **Passos:**
 1. Clicar no ícone do carrinho.
 2. Clicar no botão ‘Checkout’
 3. Clicar no botão ‘Continue’
- **Resultado esperado:** Sistema exibe a mensagem ‘Error: First Name is required’.

RN5-CT1: Ordenação de Produtos

- **Pré-condição:** Usuário precisa estar logado.
- **Passos:**
 1. Clicar no seletor de ordenação abaixo e selecionar a opção ‘Price (high to low)’.
- **Resultado esperado:** Sistema exibe a ordenação dos produtos verticalmente do mais caro, para o mais barato.

4. Estimar o tempo de teste e demonstrar o cálculo (18:12 – 20:00)

4.1 – Testes Manuais (18:12 – 18:55)

RN1-CT1: Login com usuário válido (10 segundos)

RN1-CT2: Login com usuário inválido (15 segundos)

RN2-CT1: Adicionar produtos ao carrinho (12 segundos)

RN3-CT1: Remover produtos do carrinho (10 segundos)

RN4-CT1: Finalização de Compra – Sucesso (25 segundos)

RN4-CT2: Finalização de Compra – Erro (12 segundos)

RN5-CT1: Ordenação de Produtos (10 segundos)

4.2 – Testes Automatizados (18:55 – 20:00)

```
(Run Finished)

Spec                                Tests  Passing  Failing  Pending  Skipped
✓ RN1-CT1.spec.js                  00:02    1      1      -      -      -
✓ RN1-CT2.spec.js                  00:01    1      1      -      -      -
✓ RN2-CT1.spec.js                  00:01    1      1      -      -      -
✓ RN3-CT1.spec.js                  00:01    1      1      -      -      -
✓ RN4-CT1.spec.js                  00:02    1      1      -      -      -
✓ RN4-CT2.spec.js                  00:01    1      1      -      -      -
✓ RN5-CT1.spec.js                  00:01    1      1      -      -      -
✓ All specs passed!                00:13    7      7      -      -      -

PS C:\Users\JOB\Documents\projetoDOT\automaTests> 
```

Como os tempos foram calculados?

- **Testes Manuais:** O tempo foi estimado utilizando a extensão [TestCase Studio](#), que registrou o tempo de cada interação, vide vídeos na pasta ‘evidencias\testes manuais’.
- **Testes Automatizados:** O tempo foi capturado diretamente do log do Cypress após a execução dos testes.
- **Automação reduziu o tempo dos testes em aproximadamente 86%.**

Comparação entre Testes Manuais e Automatizados

Caso de Teste	Tempo Manual	Tempo Automatizado
RN1-CT1: Login Válido	10s	2s
RN1-CT2: Login Inválido	15s	1s
RN2-CT1: Adicionar Produto ao Carrinho	12s	1s
RN3-CT1: Remover Produto	10s	1s
RN4-CT1: Checkout Sucesso	25s	2s
RN4-CT2: Checkout Erro	12s	1s
RN5-CT1: Ordenação de Produtos	10s	2s
Tempo Total	1m34s	13 segundos

5. Criar a pipeline no GitHub Actions (20:00 – 22:00)

A pipeline foi configurada para atualizar automaticamente sempre que houver um push/pull request pra Branch master.