

Documentação do QA

Prática de User Stories para o Swag Labs (saucedemo.com)

1. Introdução

Este documento apresenta duas User Stories (Histórias de Usuário) para o site Swag Labs (<https://www.saucedemo.com/>), conforme solicitado na prática de escrita de histórias de usuário. As histórias seguem o formato padrão e incluem épicos, Critérios de Aceitação detalhados, Casos de Teste Step-by-Step, Mapas Mentais e o fluxo de trabalho de desenvolvimento e bug.

2. Épico: Gestão de Acesso

Este épico engloba todas as funcionalidades relacionadas ao acesso e autenticação do usuário na loja virtual.

2.1. User Story 1: Login com Credenciais Inválidas

Item	Descrição
Título	Validação de login com nome de usuário ou senha incorretos
Narrativa de Usuário	Como um usuário da loja virtual, desejo receber uma mensagem de erro clara ao tentar fazer login com nome de usuário ou senha que não correspondam a um usuário registrado, para que eu saiba que as credenciais estão incorretas e possa tentar novamente.

2.1.1. Critérios de Aceitação (BDD)

- Dado que estou na página de login da loja virtual (<https://www.saucedemo.com/>)
- Quando eu insiro um nome de usuário (ex: usuario_invalido)
- E insiro uma senha (ex: senha_invalida)
- E clico no botão "Login"
- Então devo permanecer na página de login

- E devo ver a mensagem de erro "Epic sadface: Username and password do not match any user in this service".

2.1.2. Caso de Teste Step-by-Step (CT 01)

- Pré-condição: A página de login está acessível (<https://www.saucedemo.com/>).
- Passos:
 - i. Acessar a URL <https://www.saucedemo.com/>.
 - ii. Inserir um nome de usuário inválido (ex: usuario_teste) no campo de usuário.
 - iii. Inserir uma senha inválida (ex: 123456) no campo de senha.
 - iv. Clicar no botão "Login".
- Resultado Esperado: A página não deve ser redirecionada, e deve ser exibida a mensagem de erro "Epic sadface: Username and password do not match any user in this service".

2.2. Mapa Mental: Login com Credenciais Inválidas (US 1)

NÓ CENTRAL: Login (<https://www.saucedemo.com/>)

- 1. CREDENCIAIS INVÁLIDAS (Testes Negativos)
 - Usuário/Senha Incorretos.
 - Usuário Válido, Senha Incorreta.
 - Usuário Bloqueado (Ex: locked_out_user).
- 2. CAMPOS VAZIOS (Validação de Frontend)
 - Nome de Usuário Vazio.
 - Senha Vazia.
- 3. INTERFACE / UX
 - Botão "Login": Estado e Múltiplos Cliques.
 - Mensagem de Erro: Validação da posição e cor.
- 4. SEGURANÇA

- Tentar Injeção de SQL ou XSS.
 - Limitar tentativas de login.
- 5. REDIRECIONAMENTO
 - Com Falha: Permanecer em /.
 - Com Sucesso (após correção): Redirecionar para /inventory.html.
-

3. Épico: Experiência de Compra

Este épico engloba as funcionalidades de navegação de produtos, visualização de detalhes, e gerenciamento de itens no carrinho.

3.1. User Story 2: Remoção de Produto da Página de Produtos

Item	Descrição
Título	Remoção de item da página de Produtos
Narrativa de Usuário	Como um cliente, desejo poder remover um produto diretamente da página de Produtos ("Products"), depois de adicioná-lo ao carrinho, para desfazer rapidamente a ação sem precisar navegar até a página do carrinho.

3.1.1. Critérios de Aceitação (BDD)

- Dado que o usuário está logado no sistema Swag Labs com o usuário standard_user
- E o usuário está na página de Produtos (/inventory.html)
- E um produto já foi adicionado ao carrinho (o botão mostra "Remove" e o ícone do carrinho mostra um número)
- Quando eu clico no botão "Remove" para esse produto
- Então o botão deve voltar a exibir "Add to cart"
- E o contador de itens no ícone do carrinho deve diminuir em um
- E o item não deve mais constar na página do Carrinho ao ser acessada.

3.1.2. Caso de Teste Step-by-Step (CT 02)

- Pré-condição: O usuário está logado com sucesso e na página de Produtos (/inventory.html).
- Passos:
 - i. Clicar no botão "Add to cart" em um produto.
 - Verificação: O botão muda para "Remove" e o ícone do carrinho mostra "1".
 - ii. Clicar no botão "Remove" que apareceu no produto.
 - iii. Verificar o contador do carrinho.
 - iv. Acessar a página do Carrinho (/cart.html).
- Resultado Esperado:
 - O botão do produto na página de Produtos deve voltar a exibir "Add to cart".
 - O ícone do carrinho deve mostrar o número "0".
 - O produto removido não deve constar na tela do carrinho.

3.2. Mapa Mental: Remoção de Produto (US 2)

NÓ CENTRAL: Gerenciamento do Carrinho (Página de Produtos)

- 1. FLUXO PRINCIPAL
 - Adicionar 1 item -> Remover.
 - Adicionar N itens -> Remover 1 item específico.
- 2. BOTÕES / UX
 - Validação da troca de texto: "Add to cart" vs "Remove".
 - Validação da cor do botão "Remove" (vermelho).
- 3. COERÊNCIA (Impacto)
 - Ação na Página de Produtos deve refletir na Página do Carrinho.
 - O ícone do carrinho deve ser atualizado instantaneamente.
- 4. TESTES DE EXCEÇÃO
 - Remover item enquanto está na página de detalhes do item.

- Remover o último item (o contador deve zerar/sumir).
-

4. Fluxos de Trabalho (Development & Bug)

Os fluxos a seguir, adaptados dos seus documentos de referência, definem o ciclo de vida de uma nova funcionalidade (Task) e o tratamento de um problema (Bug).

4.1. Plano de Fluxo de Trabalho de Desenvolvimento da Task

Este fluxo é usado para entregar as User Stories (US 1 e US 2).

Etapa	Status	Descrição
01	COMEÇAR / NEW	A User Story é criada e entra no backlog de desenvolvimento.
02	TO DO	A US é priorizada e aguarda para ser iniciada pelo time de desenvolvimento.
03	IN PROGRESS / DEVELOPMENT	O desenvolvedor está implementando a funcionalidade.
04	CODE REVIEW	O código é revisado por outro desenvolvedor para garantir qualidade e padrões.
05	TESTING / VALIDATION	A funcionalidade é testada pelo QA (execução dos Casos de Teste e Mapas Mentais).
06	DONE / CLOSED	O código foi aprovado, testado e a funcionalidade está pronta para produção.

4.2. Ciclo de Vida do Bug (Identificado em qualquer etapa)

Este ciclo é acionado se um Casos de Teste (CT 01 ou CT 02) falhar ou se um bug for encontrado.

Etapa	Status	Descrição
01	COMEÇAR / NEW	O QA ou usuário reporta o problema, e um ticket de bug é aberto.
02	ASSIGN	O bug é atribuído a um líder técnico ou gerente.
03	ROOT CAUSE ANALYSIS	A equipe investiga a origem (causa raiz) do defeito.
04	DEVELOPMENT	O desenvolvedor trabalha na correção do código.
05	TESTS	O QA realiza testes unitários e de regressão na correção (Avaliação da correção).
06	VALIDATION / RESOLVED	O bug é confirmado como corrigido e o status é definido como "RESOLVED".
07	CLOSED	A correção de bug é feita com sucesso, o código está na produção e o ticket é fechado.