



Automação de Testes (QD-15)

 [QD-5] Como QA, quero automatizar os testes de login utilizando Selenium para aumentar a eficiência. Criado: 24/ago/25 Atualizado(a): 24/ago/25

Status:	Tarefas pendentes		
Projeto:	QA-dio		
Componentes:	Nenhum		
Versões afetadas:	Nenhum		
Versões corrigidas:	Nenhum		
Pai:	Automação de Testes		

Tipo:	Tarefa	Prioridade:	Medium
Relator:	Giuseppe Mota	Responsável:	Não atribuído
Resolução:	Não resolvido(s)	Votos:	0
Categorias:	Nenhum		
Estimativa de trabalho restante:	Desconhecido		
Tempo gasto:	Desconhecido		
Estimativa original:	Desconhecido		

Anexos:

 ACg8ocLb93MNU43jRZoRZXrhMR8TAod47qTs6Trz_XQkcDMW60mNOA=s64-c-mo

Rank:

0|i00023:

Sprint:

Descrição

Valor de Negócio

A automação dos testes de login e autenticação é uma medida estratégica que garante a estabilidade e a confiabilidade de uma das funcionalidades mais críticas do sistema. Ao automatizar, **reduzimos a necessidade de testes manuais repetitivos**, permitindo que a equipe de QA se concentre em cenários mais complexos e exploratórios. Isso **acelera o ciclo de desenvolvimento**, garante que novas versões não introduzam regressões no fluxo de login e, em última instância, **protege a experiência do usuário**, impedindo que ele tenha problemas para acessar sua conta.

Narrativa do Usuário

- **Como** um QA,
- **Quero** automatizar os testes para os fluxos de login, cadastro e recuperação de senha,
- **Para** garantir que as funcionalidades de autenticação estejam sempre funcionando corretamente e para agilizar o processo de validação em cada nova versão do software.

Requisitos

1. Atores e Cenários de Teste

- **Atores:** Usuários (validando login e cadastro) e o próprio sistema (validando o comportamento de senhas e tokens).
- **Cenários Cobertos:**
 - **Login:**
 - Login com credenciais válidas.
 - Login com credenciais inválidas (usuário inexistente, senha incorreta).
 - Login com campos em branco (usuário, senha ou ambos).
 - **Cadastro:**
 - Cadastro com dados válidos.
 - Cadastro com e-mail já existente.
 - Cadastro com senhas que não atendem aos critérios de segurança (se houver).
 - **Recuperação de Senha:**
 - Fluxo completo de recuperação de senha para e-mail válido.
 - Tentativa de recuperação de senha com e-mail inválido/não cadastrado.

2. Ferramentas e Ambiente

- **Ferramentas de Automação:** Selenium, Cypress, Playwright ou outra ferramenta de automação de testes de UI. A escolha da ferramenta deve ser baseada na compatibilidade com o projeto e nas habilidades da equipe.

- **Linguagem de Programação:** JavaScript/TypeScript (para Cypress/Playwright) ou Python/Java (para Selenium), dependendo da escolha da ferramenta.
- **Ambientes de Teste:** O suite de testes automatizados deve ser executado em ambientes de QA e Staging para simular o ambiente de produção.
- **Navegadores:** A automação deve ser configurada para rodar em diferentes navegadores para garantir compatibilidade (ex: Chrome, Firefox, Edge).

3. Estrutura e Código

- **Framework:** Implementar um framework de testes com Page Object Model (POM) para garantir a organização e a manutenibilidade do código.
- **Dados de Teste:** Utilizar um conjunto de dados de teste (data-driven testing) que inclua credenciais válidas e inválidas, e-mails de teste, etc. Os dados sensíveis, como senhas, devem ser armazenados de forma segura (ex: variáveis de ambiente).
- **Execução:** O suite de testes deve ser facilmente executável via linha de comando (ex: `npm run test:login` ou `pytest`).

Critérios de Aceitação (QA)

- **Execução Múltipla:** Os testes devem ser executados com sucesso em pelo menos dois navegadores diferentes (ex: Chrome e Firefox).
- **Relatórios Automatizados:** A execução dos testes deve gerar um relatório detalhado (preferencialmente em HTML) que mostre quais testes passaram, quais falharam e por quê. O relatório deve ser gerado automaticamente ao final da execução.
- **Cenários de Falha:** Para cada cenário de falha (ex: login com senha incorreta), o teste deve validar a mensagem de erro correta exibida na interface do usuário.
- **Integração:** O suite de testes automatizados deve ser integrado ao pipeline de CI/CD para que os testes de login sejam executados automaticamente em cada nova commit ou pull request na branch principal.
- **Estabilidade:** Os testes devem ser estáveis e não ter falhas intermitentes (flaky tests).
- **Performance:** O tempo de execução do suite de testes deve ser otimizado para não impactar negativamente o pipeline de CI/CD.

Unable to embed resource: ACg8ocLb93MNU43jRZoRZXrhMR8TAod47qTs6Trz_XQkcDMW60mNOA=s64-c-mo of type application/octet-stream