

Plano de Testes - Automação Playwright

Informações Gerais

Aplicação: Serverest (<https://front.serverest.dev/>)
Framework: Playwright + JavaScript
Padrões: Page Object Model (POM) + Arrange-Act-Assert (AAA)

Casos de Teste Implementados

1. Cadastro de Usuário

Arquivo: `cadastro.spec.js`

ID	Cenário	Dados	Resultado Esperado
CT001	Cadastrar novo usuário	Dados dinâmicos únicos	Mensagem "Cadastro realizado com sucesso"
CT002	Cadastrar com email existente	Reutilizar email do CT001	Mensagem "Este email já está sendo usado"
CT003	Cadastrar com dados de fixture	Dados da fixture + email único	Mensagem de sucesso

2. Login de Usuário

Arquivo: `login.spec.js`

ID	Cenário	Dados	Resultado Esperado
CT004	Setup - Criar usuário para testes	Usuário dinâmico	Usuário criado com sucesso
CT005	Login com credenciais válidas	Usuário criado no setup	Mensagem "Bem Vindo" + redirecionamento
CT006	Login com email inválido	Email inexistente + senha válida	Mensagem "Email e/ou senha inválidos"
CT007	Login com senha inválida	Email válido + senha incorreta	Mensagem "Email e/ou senha inválidos"
CT008	Login com email vazio	Email vazio + senha válida	Mensagem "Email é obrigatório"

3. Cadastro de Produto

Arquivo: `produto-final.spec.js`

ID	Cenário	Pré-requisito	Resultado Esperado
CT009	Setup - Criar usuário para produtos-	Usuário admin criado	
CT010	Cadastrar produto dinâmico	Login realizado	Redirecionamento para "Lista dos Produtos"
CT011	Cadastrar smartphone premium	Login realizado	Produto cadastrado na lista
CT012	Cadastrar produto categoria casa	Login realizado	Produto cadastrado na lista

Estrutura dos Testes

Organização por Módulos

- **Cadastro:** Testes independentes com dados únicos
- **Login:** Testes sequenciais com usuário compartilhado
- **Produto:** Testes sequenciais com login obrigatório

Geração de Dados

- **Dinâmicos:** Timestamp + número aleatório para unicidade
- **Fixtures:** Dados pré-definidos para cenários específicos
- **Compartilhados:** Reutilização entre testes relacionados

Validações Implementadas

Funcionais

- Mensagens de sucesso e erro
- Redirecionamentos corretos
- Campos obrigatórios
- Regras de negócio (email único)

Interface

- Visibilidade de elementos
- Habilitação de botões
- Navegação entre páginas
- Timeout adequados

Cobertura de Testes

Total de Casos: 13

Fluxos Principais: 3

Cenários Positivos: 7

Cenários Negativos: 6

Por Funcionalidade

- Cadastro: 3 casos
- Login: 6 casos
- Produto: 4 casos

Execução

```
# Todos os testes
npx playwright test

# Por módulo
npx playwright test cadastro.spec.js
npx playwright test login.spec.js
npx playwright test produto-final.spec.js

# Com interface
npx playwright test --headed
```

Observações Técnicas

- Testes de produto dependem de login prévio
- Dados dinâmicos evitam conflitos entre execuções
- Timeouts ajustados para estabilidade
- Screenshots automáticos em falhas