

Relatório - Sistema Cinema

Relatório Técnico de Automação de Testes – Sistema Cinema

1. Introdução

Este documento tem como objetivo apresentar uma análise técnica detalhada do projeto de automação de testes desenvolvido para o sistema Cinema, contemplando o backend (API) e o frontend (interface web).

O projeto foi conduzido com base em práticas de garantia de qualidade (QA) e automação funcional, visando assegurar a estabilidade, confiabilidade e manutenibilidade da aplicação.

2. Objetivo do Projeto

O objetivo principal da automação foi:

- Garantir o funcionamento correto dos principais fluxos do sistema de reservas de cinema;
- Validar a integridade das comunicações entre o frontend e o backend;
- Reduzir o tempo de regressão através da automação de testes repetitivos;
- Prover uma base de testes automatizados escalável e de fácil manutenção.

3. Escopo da Automação

A automação abrangeu os módulos críticos da aplicação, divididos entre testes de API e testes de interface (UI).

Os testes foram implementados utilizando o Robot Framework, com suporte às bibliotecas RequestsLibrary (para APIs REST) e Browser (para automação de interface via Playwright).

Módulo	Tipo	Descrição	Status
Autenticação	API + UI	Login, cadastro, logout e validações de token	Concluído
Filmes	API + UI	Listagem e busca de filmes	Concluído

Sessões	API	Listagem, busca por ID e por filme	Concluído
Reservas	API + UI	Criação e validação de reservas de assentos	Concluído
Testes Negativos	API	Validação de erros e respostas inválidas	Concluído
Fluxos E2E	UI	Cadastro, login e reserva completa	Concluído
Permissões (Admin)	API	Testes administrativos (rota instável)	Não Implementado
Cinemas (Theaters)	API	Listagem e detalhes de cinemas	Não Implementado

4. Ferramentas e Tecnologias

Categoria	Ferramenta	Finalidade
Framework de automação	Robot Framework	Base estrutural dos testes
Biblioteca de API	RequestsLibrary	Comunicação HTTP e validação de endpoints
Biblioteca de UI	Browser (Playwright)	Automação de interface web
Linguagem de definição	Sintaxe Robot	Escrita dos cenários e keywords

Controle de versão	Git / GitHub	Versionamento e documentação
Relatórios	Log.html / Report.html	Geração de relatórios automáticos de execução

5. Estrutura do Projeto

A estrutura do repositório foi organizada em módulos independentes para facilitar a manutenção e a execução seletiva dos testes, essa arquitetura permite a execução modular de testes por categoria (API, UI ou E2E), além de garantir isolamento entre os ambientes.

6. Cenários Implementados (com Nomeações Oficiais)

6.1. Módulo de Autenticação (API + UI)

ID	Tipo	Nome do Teste	Descrição
API-AUTH-01	API	Login com credenciais válidas	Verifica autenticação com dados corretos
API-AUTH-02	API	Login com credenciais inválidas	Verifica retorno 401 para credenciais incorretas
API-AUTH-03	API	Acesso a rota protegida sem token	Garante que endpoints protegidos exigem autenticação
UI-AUTH-01	UI	Login via interface web	Testa login visual e exibição da mensagem de sucesso

UI-AUTH-02	UI	Registro de novo usuário	Realiza cadastro e valida mensagem “Conta criada com sucesso!”
UI-AUTH-03	UI	Logout e novo login	Realiza logout e verifica login posterior com as mesmas credenciais

6.2. Módulo de Filmes (API + UI)

ID	Tipo	Nome do Teste	Descrição
API-MOVIES-01	API	Listar todos os filmes	Verifica se o endpoint <code>/movies</code> retorna lista válida
API-MOVIES-02	API	Buscar filme por ID	Testa busca dinâmica de filme existente por ID
UI-MOVIES-01	UI	Visualizar lista de filmes na Home	Garante que os cards de filmes aparecem na página inicial
UI-MOVIES-02	UI	Clicar em filme e ver detalhes	Valida a navegação para a página de detalhes do filme selecionado

6.3. Módulo de Sessões (API)

ID	Tipo	Nome do Teste	Descrição
API-SESSIONS-01	API	Listar sessões disponíveis	Verifica resposta 200 e conteúdo da lista de sessões
API-SESSIONS-02	API	Buscar sessão por ID	Testa retorno de dados detalhados de uma sessão específica
API-SESSIONS-03	API	Listar sessões de um filme específico	Filtra sessões por <code>movieId</code> e valida consistência dos resultados

6.4. Módulo de Reservas (API + UI)

ID	Tipo	Nome do Teste	Descrição
API-RESERVES-01	API	Criar reserva com dados válidos	Testa fluxo de reserva com autenticação e assento disponível
API-RESERVES-02	API	Criar reserva com assento já ocupado	Garante retorno 400 para tentativa de reserva duplicada
API-RESERVES-03	API	Criar reserva sem autenticação	Verifica bloqueio de reservas sem token válido

API-RESERVES-04	API	Listar reservas do usuário	Garante que o endpoint <code>/reservations/me</code> retorna apenas as reservas do usuário autenticado
UI-RESERVES-01	UI	Fluxo completo de reserva (E2E)	Realiza login, seleção de filme, sessão e assento até o pagamento
UI-RESERVES-02	UI	Validação da confirmação de reserva	Garante exibição da mensagem “Reserva Confirmada!” ao final do fluxo

6.5. Testes Negativos (API)

ID	Tipo	Nome do Teste	Descrição
API-NEG-01	API	Buscar recurso inexistente (404)	Testa resposta adequada ao acessar ID inexistente
API-NEG-02	API	Enviar payload inválido (400)	Garante que a API retorna erro de validação apropriado
API-NEG-03	API	Token expirado ou inválido (401)	Verifica resposta de negação de acesso com token incorreto

6.6. Fluxos E2E (UI)

ID	Tipo	Nome do Teste	Descrição
UI-E2E-01	UI	Fluxo completo de novo usuário	Cadastro → Login → Reserva completa de ingresso
UI-E2E-02	UI	Logout pós-login e reautenticação	Verifica persistência de credenciais e fluxo de retorno
UI-E2E-03	UI	Interação com elementos da home	Valida a navegabilidade e responsividade da página inicial

7. Cobertura e Resultados

Tipo de Teste	Cobertura Estimada	Observações
API	80%	Todos os endpoints principais validados
UI	70%	Fluxos críticos de navegação e reserva cobertos
E2E	100%	Cenário completo de ponta a ponta testado
Testes Negativos	100%	Principais cenários de erro contemplados
Rotas Admin	0%	Não aplicável (instabilidade no backend e frontend)

Resultado geral: todos os testes implementados executam com sucesso e validam as principais funções de negócio.

8. Limitações Identificadas

- As rotas de administração não estão estáveis, inviabilizando testes relacionados.
-

9. Conclusão

A automação de testes do sistema Cinema apresenta uma estrutura madura, modular e de fácil manutenção.

O conjunto de testes cobre adequadamente os fluxos principais e críticos do sistema, incluindo autenticação, listagem de filmes, sessões e fluxo de reserva.

O projeto demonstra:

- Adoção de boas práticas de QA e automação;
- Validação técnica de integrações entre frontend e backend;
- Alta taxa de sucesso na execução automatizada dos testes.

Embora existam oportunidades de expansão, a automação atual atende plenamente aos objetivos propostos, servindo como base sólida para evolução contínua.

10. Considerações Finais

O projeto de automação entregue está tecnicamente consistente, bem estruturado e funcional. Ele garante que as principais jornadas do usuário e integrações de sistema estejam validadas automaticamente, contribuindo para a qualidade contínua do produto.

Com a adição futura de pipelines e testes complementares de performance e filtros, o projeto atingirá um nível de excelência corporativa em automação de QA.