

# Planejamento de Testes – Challenge Final

**Relator:** João Vitor Facklamm

---

## Apresentação

Este documento apresenta o **planejamento de testes** realizado para o *Challenge Final* do programa **PB AWS & AI for QE**, abordando o processo de análise, escopo, definição de cenários e critérios de automação.

A aplicação testada é o **Cinema App**, composta por duas camadas: **Front-end** e **Back-end**.

---

## Objetivo

Planejar, estruturar e implementar o projeto de automação de testes aplicando os conceitos estudados ao longo da trilha, garantindo a qualidade das entregas e a cobertura adequada dos principais fluxos do sistema.

---

## Escopo

### Testes dentro do escopo

- Testes exploratórios voltados para **funcionalidades críticas**, como login, reserva de assentos, atualização de perfil e navegação nas páginas principais (início, filmes e perfil).
- Validação dos fluxos **CRUD** para usuários, reservas, filmes, sessões e teatros (criação, atualização, busca e deleção).
- Testes de **autenticação e autorização**, assegurando que apenas usuários válidos acessem rotas protegidas.
- Testes de fluxo de visualização e controle de reservas, garantindo acesso correto a dados do usuário autenticado.

- Verificação de **campos obrigatórios**, formatos e mensagens de erro nos payloads das requisições.

## ✗ Testes fora do escopo

- Funcionalidades não implementadas ou inacessíveis no front-end (ex.: página de administradores e gerenciamento de teatros).
- Recursos bloqueados ou ausentes devido a restrições técnicas no sistema atual.

---

## Análise Inicial do Projeto

### Visão geral

Usuários autenticados podem buscar, reservar, criar, atualizar e deletar filmes.

Visitantes têm acesso apenas à visualização, enquanto administradores possuem privilégios adicionais, como gerenciamento de reservas e usuários.

### Visão do Back-end

Alguns endpoints de **reservas** e **sessões** apresentaram documentação incompleta, o que dificultou os testes e exigiu exploração manual para identificação dos parâmetros obrigatórios.

Após ajustes na documentação, a execução dos testes tornou-se mais fluida e precisa.

### Visão do Front-end

Foram observadas dificuldades nos fluxos de **cadastro** e **login**, bem como indisponibilidade de certas áreas administrativas e funcionalidades relacionadas a teatros.

### Visão das User Stories

As histórias de usuário estão bem estruturadas e segmentadas por papéis (visitante, usuário autenticado e administrador), abordando autenticação, navegação, visualização e reserva de filmes.

O sistema busca oferecer uma interface **intuitiva, responsiva e de fácil uso**, garantindo boa experiência em diferentes dispositivos.

---

## Testes Exploratórios

Foram realizados testes manuais nos fluxos principais, incluindo autenticação, reserva, navegação e atualização de perfil.  
Esses testes ajudaram a identificar inconsistências em usabilidade e comportamento do sistema.

Principais observações:

- Falhas na validação de campos e mensagens de erro em alguns formulários;
- Inconsistências visuais e interativas em pop-ups e botões de navegação;
- Problemas de autenticação e cadastro solucionados durante o processo.

Todos os **bugs identificados** foram documentados e rastreados em ferramenta de controle de issues.

---

## Cenários de Teste

Os cenários foram divididos por módulos principais do sistema, cobrindo autenticação, administração de usuários, filmes, sessões, teatros e reservas.

Cada cenário foi classificado conforme prioridade, impacto e criticidade, abrangendo fluxos de sucesso, falhas e restrições de acesso.

---

### Autenticação (auth\_tests)

ID	Cenário	Método	Prioridade	Resultado Esperado
CT-001	Cadastro de usuário válido	POST	Alta	201 – Sucesso
CT-002	Atualizar perfil com dados válidos	PUT	Média	200 – Sucesso
CT-003	Buscar perfil do usuário autenticado	GET	Média	200 – Sucesso

CT-00 4	Cadastro com e-mail duplicado	POST	Baixa	400 – Inválido
CT-00 5	Login com credenciais inválidas	POST	Baixa	400 – Inválido
CT-00 6	Acessar /auth/me sem token	GET	Alta	401 – Unauthorized
CT-00 7	Atualizar perfil sem token	PUT	Alta	401 – Unauthorized
CT-00 8	Logout simulando exclusão de token	GET	Média	401 – Unauthorized
CT-00 9	Token expirado ou inválido	GET	Alta	401 – Token Inválido

---

### Usuário Administrador (users\_admin\_tests)

ID	Cenário	Método	Prioridade	Resultado Esperado
CT-010	Listar todos os usuários	GET	Alta	200 – Lista de usuários
CT-011	Buscar usuário por ID	GET	Média	200 – Dados do usuário
CT-012	Atualizar um usuário existente	PUT	Média	200 – Atualização ok

CT-013	Deletar um usuário existente	DELETE	Alta	200 – Remoção ok
--------	------------------------------	--------	------	------------------

---

## Filmes (movie\_tests)

ID	Cenário	Método	Prioridade	Resultado Esperado
CT-014	Criar novo filme (admin)	POST	Média	201 – Filme criado
CT-015	Atualizar filme existente	PUT	Média	200 – Atualizado com sucesso
CT-016	Buscar filme por ID	GET	Média	200 – Detalhes do filme
CT-017	Deletar filme por ID	DELETE	Média	200 – Filme deletado
CT-018	Listar todos os filmes	GET	Alta	200 – Lista de filmes
CT-019	Buscar filme inexistente	GET	Baixa	404 – Não encontrado
CT-020	Usuário comum tenta criar filme	POST	Alta	403 – Forbidden

---

## Sessões (session\_tests)

ID	Cenário	Método	Prioridade	Resultado Esperado
CT-02 1	Criar nova sessão (admin)	POST	Média	201 – Criado com sucesso
CT-02 2	Atualizar sessão	PUT	Média	200 – Atualizado
CT-02 3	Buscar sessão por ID	GET	Média	200 – Detalhes da sessão
CT-02 4	Deletar sessão	DELETE	Média	200 – Deletado
CT-02 5	Resetar assentos da sessão	PUT	Média	200 – Assentos resetados
CT-02 6	Usuário comum tenta deletar sessão	DELETE	Alta	403 – Forbidden
CT-02 7	Criar sessão sem data/hora	POST	Média	400 – Campo obrigatório
CT-02 8	Criar sessão com IDs inválidos	POST	Alta	404 – Entidade inválida

---

### Teatros (theater\_tests)

ID	Cenário	Método	Prioridade	Resultado Esperado
----	---------	--------	------------	--------------------

CT-02 9	Criar teatro válido (admin)	POST	Baixa	201 – Teatro criado
CT-03 0	Atualizar teatro	PUT	Baixa	200 – Atualizado
CT-03 1	Buscar teatro por ID	GET	Baixa	200 – Detalhes do teatro
CT-03 2	Deletar teatro	DELETE	Baixa	200 – Teatro deletado
CT-03 3	Listar todos os teatros	GET	Média	200 – Lista de teatros
CT-03 4	Criar teatro sem capacidade	POST	Média	400 – Campo obrigatório
CT-03 5	Usuário comum tenta atualizar teatro	PUT	Alta	403 – Forbidden
CT-03 6	Criar teatro com capacidade negativa	POST	Média	400 – Capacidade inválida

---

## Reservas (reservation\_tests)

ID	Cenário	Método	Prioridade	Resultado Esperado
CT-03 7	Criar nova reserva	POST	Alta	201 – Reserva criada

CT-03 8	Buscar reserva por ID	GET	Média	200 – Detalhes da reserva
CT-03 9	Atualizar reserva (admin)	PUT	Média	200 – Atualizado
CT-04 0	Deletar reserva (admin)	DELETE	Média	200 – Deletado
CT-04 1	Buscar reservas do próprio usuário	GET	Alta	200 – Lista de reservas
CT-04 2	Buscar todas reservas (admin)	GET	Alta	200 – Lista de todas as reservas
CT-04 3	Criar reserva com dados inválidos	POST	Média	400 – Erro de validação
CT-04 4	Buscar reserva com ID inválido	GET	Baixa	404 – Não encontrado
CT-04 5	Criar reserva sem autenticação	POST	Alta	401 – Unauthorized
CT-04 6	Usuário comum tenta deletar reserva	DELETE	Alta	403 – Forbidden
CT-04 7	Criar reserva sem assentos	POST	Alta	400 – Campo obrigatório

---



## Cobertura de Testes – Back-end

Módulo	Endpoint s	Coberto s	%
Authentication	4	4	100%
Movies	5	5	100%
Reservations	6	6	100%
Sessions	6	5	83%
Theaters	5	5	100%
Users (Admin)	4	4	100%
<b>Total</b>	<b>30</b>	<b>29</b>	<b>~97%</b>

---

## Bugs Encontrados

Durante a execução dos testes manuais e exploratórios, foram identificadas e registradas as seguintes falhas:

- **✗ BUG-001:** Erro ao tentar realizar login após cadastro – sistema não reconhece credenciais recém-criadas.
- **✗ BUG-002:** Pop-up de debug (“API Debug Tool”) não fecha ao clicar fora da área.
- **✗ BUG-003:** Campo de nome no perfil aceita caracteres inválidos sem validação.
- **✗ BUG-004:** Link “Minhas Reservas” na página de perfil não possui ação associada.

- **✗ BUG-005:** Opção de meia-entrada ausente no fluxo de pagamento.
- **✗ BUG-006:** Elementos na seção “Available Features” aparentam ser clicáveis, mas não possuem ação.

Todos os bugs foram documentados e encaminhados para acompanhamento no sistema de controle de issues.

---

## Teste de Responsividade

Página	Resultado
Login	Totalmente responsiva, com excelente adaptação a diferentes telas.
Página Principal	Responsiva e funcional em múltiplas resoluções.
Filmes em Cartaz	Parcialmente responsiva – em telas menores, o layout se desorganiza.
Minhas Reservas	Responsiva e estável.
Perfil	Boa responsividade, mantendo legibilidade e usabilidade.

---

## Testes Candidatos à Automação

### Critérios de Seleção

1. Alta prioridade no plano de testes;
2. Funcionalidades críticas (autenticação, reservas e controle de acesso);
3. Validação de segurança (token, perfis e restrições de acesso);

4. Estabilidade e ausência de bugs bloqueadores;
5. Frequência de uso pelos usuários.

## Cenários Sugeridos para Automação (CI/CD)

Total recomendado: **16 cenários principais**

- **Autenticação:** CT-001, CT-003, CT-006, CT-009
- **Admin (Usuários):** CT-010, CT-011, CT-012
- **Filmes:** CT-014, CT-018, CT-020
- **Sessões:** CT-021, CT-026, CT-028
- **Teatros:** CT-033, CT-035
- **Reservas:** CT-041, CT-046, CT-047

Esses cenários cobrem fluxos de **segurança**, **CRUDs principais** e **restrições de acesso**, garantindo estabilidade e qualidade contínua por meio da integração com o pipeline de **CI/CD**.

---

## Conclusão

O planejamento de testes do **Cinema App** abrangeu de forma sólida os principais fluxos do sistema, com cobertura robusta e priorização eficiente.

Os testes exploratórios e automatizados evidenciaram a maturidade da aplicação e destacaram pontos de melhoria — especialmente em **responsividade**, **validação de campos** e **experiência do usuário**.

Recomenda-se continuidade da automação e acompanhamento dos módulos pendentes (área administrativa e gerenciamento de teatros) para garantir a completude da validação.

---

## Recomendações Finais

Para a evolução contínua da qualidade do projeto, recomenda-se:

- **Expandir a cobertura de testes automatizados** para módulos secundários, como sessões e teatros;
- **Implementar testes de UI com Playwright** ou ferramentas equivalentes, validando comportamento e responsividade;
- **Executar testes de carga e desempenho**, medindo tempo de resposta e estabilidade em múltiplas requisições simultâneas;
- **Reforçar testes de segurança**, simulando tentativas de acesso não autorizado e injeção de dados inválidos;
- **Integrar a automação ao pipeline CI/CD**, garantindo execução contínua dos testes a cada commit;
- **Revisar e atualizar a documentação técnica** sempre que novas features forem adicionadas.

Essas ações fortalecem a confiabilidade e a manutenção do projeto, assegurando que a aplicação evolua de forma escalável e estável.