Plano de Teste - Challenge Final



sumário | ☑ Apresentação | ⊚ Objetivo | ⊚ Escopo | 🦟 Ferramentas utilizadas | ◎ Mapa mental da aplicação | Q Análise | № Requisitos | ☐ Cenários de teste planejados | № Erros e Melhorias | ⚠ Matriz de risco | ் Testes automatizados | № Cobertura de testes (Manuais e Automatizados) | ☐ Resultados finais | ☑ Conclusão

📝 Apresentação 🛭

A aplicação Cinema App foi desenvolvida para simular um sistema de reserva de ingressos de cinema com o principal objetivo de ser usada como material de estudo em testes de API e de interface web durante este Challenge Final. Em sua documentação, conta com os endpoints /auth, /movies, /reservations, /sessions, /theaters e /users. Esses recursos permitem que sejam realizadas explorações e análises em busca de possíveis erros para que os estagiários na área de QA na Compass UOL apliquem seus conhecimentos de forma prática quanto à qualidade de software e testes de APIs REST e interface web.

A aplicação Cinema App é composta por uma interface web (front-end) e uma API (back-end), cujos repositórios estão disponíveis no GitHub:

- Front-end: GitHub juniorschmitz/cinema-challenge-front
- Back-end: GitHub juniorschmitz/cinema-challenge-back

⊚ Objetivo

O objetivo principal da realização dos testes realizados na API Cinema App é garantir a qualidade da aplicação por meio da validação das regras de negócio e da identificação de erros e possíveis oportunidades de melhoria.



Dentro do escopo *∂*

Serão testadas as seguintes rotas e suas respectivas requisições, conforme documentado no Swagger da API Cinema App:

- /auth
 - Realizar login
 - o Obter perfil do usuário atual
 - Cadastrar novo usuário
 - o Atualizar perfil do usuário
- /movies
 - o Obter todos os filmes
 - o Obter detalhes de um filme por ID
 - o Cadastrar novo filme (apenas administradores)
 - o Atualizar filme (apenas administradores)
 - Excluir filme (apenas administradores)

/reservations

- Obter todas as reservas (apenas administradores)
- o Obter reservas do usuário atual
- o Obter detalhes da reserva por ID
- o Cadastrar nova reserva
- Atualizar status da reserva (apenas administradores)
- Excluir reserva (apenas administradores)

/sessions

- o Obter todas as sessões
- o Obter detalhes da sessão por ID
- o Cadastrar nova sessão (apenas administradores)
- Atualizar sessão (apenas administradores)
- Excluir sessão (apenas administradores)
- o Resetar todos os assentos de uma sessão para status "disponível" (apenas administradores)

/theaters

- o Obter todos os cinemas
- o Obter detalhes do cinema por ID
- o Cadastrar novo cinema (apenas administradores)
- Atualizar cinema (apenas administradores)
- Excluir cinema (apenas administradores)

/users

- o Obter todos os usuários (apenas administradores)
- o Obter detalhes do usuário por ID (apenas administradores)
- o Atualizar usuário (apenas administradores)
- Excluir usuário (apenas administradores)

Além das rotas, também estão incluídos no escopo:

- Testes manuais e automatizados das funcionalidades disponíveis
- Testes de interface web (front-end), com foco na usabilidade e experiência do usuário
- Validação de responsividade da interface em diferentes resoluções

Fora do escopo @

Não serão testadas as seguintes rotas e suas respectivas requisições, conforme documentado no Swagger da API Cinema App:

- /setup
- / (API Info)

Além das rotas, também não estão incluídos no escopo:

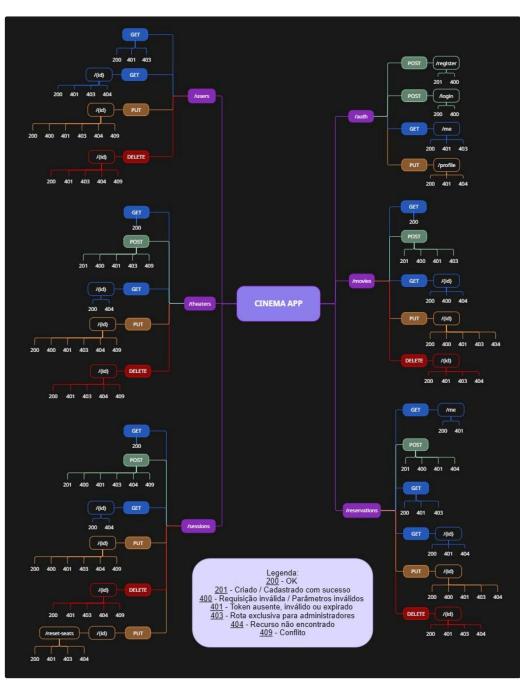
- Testes de desempenho e carga
- Testes de segurança avançados
- Verificações de acessibilidade

🛠 Ferramentas utilizadas 🛭

- Aplicação Cinema App:
 - Front-end: GitHub juniorschmitz/cinema-challenge-front

- Back-end: GitHub juniorschmitz/cinema-challenge-back
- Postman
- Jira
- Confluence
- Miro
- VS Code + Robot Framework
- QAlity no Jira

🧠 Mapa mental da aplicação 🔊





Cenários de teste @

Técnicas aplicadas 🕖



Materiais utilizados para definição de requisitos:

- Histórias de usuário
 - o Utilizadas como fonte principal, representando a visão e os objetivos do usuário final.
- <u>Documentação Swagger</u>
 - Utilizada como base para extrair os comportamentos esperados dos endpoints, especialmente quando as histórias de usuário estão ausentes ou não fornecem informações suficientes sobre as requisições e respostas esperadas.

/auth ∂

- Base principal: US-AUTH-001, US-AUTH-002, US-AUTH-003 e US-AUTH-004.
- Complementado com requisitos técnicos extraídos do Swagger.

ID	Descrição	Origem
R1.1	Sistema deve permitir que usuário visitante cadastre uma conta informando nome, e-mail e senha.	US-AUTH-001
R1.2	Sistema deve validar o formato do e-mail, exigindo que ele contenha o caractere "@" e termine com ".com" após o domínio.	US-AUTH-001
R1.3	Sistema deve validar o formato da senha, exigindo que ela contenha 6 ou mais caracteres.	US-AUTH-001
R1.4	Sistema deve impedir cadastro de e-mail duplicados.	US-AUTH-001
R1.5	Após cadastro bem-sucedido, a interface deve redirecionar o usuário para a página de login e o sistema deve autenticar o acesso automaticamente.	US-AUTH-001
R1.6	Sistema deve permitir que usuário registrado faça login em sua conta informando e-mail e senha.	US-AUTH-002
R1.7	Sistema deve autenticar credenciais válidas no login.	US-AUTH-002
R1.8	Sistema deve manter sessão do usuário através de token JWT.	US-AUTH-002
R1.9	Após login bem-sucedido, a interface deve redirecionar o usuário para a página inicial.	US-AUTH-002
R1.10	Interface deve permitir que usuário logado possa fazer logout através do menu de navegação.	US-AUTH-003

R1.11	Após logout bem-sucedido, o sistema não deve permitir que rotas protegidas sejam acessadas.	US-AUTH-003
R1.12	Após logout bem-sucedido, sistema deve remover Token JWT do localStorage.	US-AUTH-003
R1.13	Sistema deve exibir nome do usuário, e-mail e função da conta no perfil do usuário logado.	US-AUTH- 004
R1.14	Sistema deve permitir que usuário logado edite seu nome completo.	US-AUTH- 004
R1.15	Interface deve indicar visualmente os campos que foram alterados.	US-AUTH- 004
R1.16	Sistema deve confirmar sucesso após salvar alterações.	US-AUTH- 004
R1.17	Interface deve exibir mensagem de confirmação após atualização bemsucedida.	US-AUTH- 004
R1.18	Página de perfil deve ser separada da página de reservas para melhor organização.	US-AUTH- 004
R1.19	Sistema deve permitir que usuário atualize sua senha informando nome, senha atual e nova senha.	Swagger

A Observação: O formato que e-mail e senha devem ter informados nos requisitos R1.2 e R1.3 foram registrados de acordo com teste exploratório no front-end da aplicação, uma vez que não estavam especificados nas histórias de usuário.

/movies ∂



- Base principal: US-HOME-001, US-MOVIE-001 e US-MOVIE-002.
- Complementado com requisitos técnicos extraídos do Swagger.

ID	Descrição	Origem
R2.1	Interface deve exibir banner com informações sobre o cinema na página inicial.	US-HOME-001
R2.2	Interface deve exibir uma seção destacada de "Filmes em Cartaz" com pôsteres em tamanho adequado.	US-HOME-001
R2.3	Layout da página inicial deve ser responsivo, se adaptando a diferentes tamanhos de tela.	US-HOME-001
R2.4	Interface deve ter links rápidos para principais áreas da aplicação na página inicial.	US-HOME-001
R2.5	Interface deve fornecer acesso fácil à navegação principal por meio do cabeçalho.	US-HOME-001
R2.6	Interface deve mostrar opções personalizadas no menu para usuários autenticados.	US-HOME-001

R2.7	Interface deve permitir que usuário visualize uma lista dos filmes em exibição com layout em grid.	US-MOVIE-001
R2.8	Interface deve permitir que filmes sejam exibidos com pôster grande e de alta qualidade.	US-MOVIE-001
R2.9	Interface deve permitir que cards de filmes mostrem título, classificação e gêneros.	US-MOVIE-001
R2.10	Interface deve permitir que cards incluam duração do filme e data de lançamento.	US-MOVIE-001
R2.11	Sistema deve adaptar layout para diferentes tamanhos de tela (responsivo) na lista de filmes.	US-MOVIE-001
R2.12	Interface permitir que usuário consiga acessar detalhes do filme com um clique.	US-MOVIE-001
R2.13	Detalhes do filme devem incluir sinopse, elenco, diretor, data de lançamento, duração.	US-MOVIE-002
R2.14	Interface deve mostrar pôster do filme na página de detalhes.	US-MOVIE-002
R2.15	Interface deve exibir horários de sessões disponíveis na página de detalhes.	US-MOVIE-002
R2.16	Interface deve permitir que usuário navegue para reserva a partir dos horários disponíveis.	US-MOVIE-002
R2.17	Sistema deve retornar todos os filmes, cadastrar novo filme, atualizar filme e excluir filme corretamente.	Swagger

/reservations @

- Base principal: US-RESERVE-001, US-RESERVE-002 e US-RESERVE-003.
- Complementado com requisitos técnicos extraídos do Swagger.

ID	Descrição	Origem
R3.1	Interface deve permitir que usuário logado possa visualizar o layout de assentos do cinema.	US-RESERVE-001
R3.2	Assentos devem ser codificados por cores conforme disponibilidade.	US-RESERVE-001
R3.3	Sistema deve permitir que usuário selecione múltiplos assentos disponíveis.	US-RESERVE-001
R3.4	Sistema não deve permitir que usuário selecione assentos já reservados.	US-RESERVE-001
R3.5	Sistema deve mostrar o subtotal à medida que os assentos são selecionados.	US-RESERVE-001
R3.6	Interface deve redirecionar usuário para a página de checkout após selecionar os assentos.	US-RESERVE- 002

R3.7	Interface deve permitir que resumo dos assentos selecionados sejam exibido na página de checkout.	US-RESERVE- 002
R3.8	Sistema deve permitir que usuário visualize o valor total da compra.	US-RESERVE- 002
R3.9	Sistema deve permitir que usuário selecione um método de pagamento (cartão de crédito, débito, PIX, transferência).	US-RESERVE- 002
R3.10	Sistema deve processar o pagamento (simulado) e confirmar a reserva.	US-RESERVE- 002
R3.11	Interface deve permitir que usuário receba confirmação visual do sucesso da reserva.	US-RESERVE- 002
R3.12	Assentos selecionados devem ser marcados como ocupados.	US-RESERVE- 002
R3.13	Interface deve permitir que usuário acesse a lista de suas reservas através do link "Minhas Reservas" no menu.	US-RESERVE- 003
R3.14	Interface deve exibir reserva em formato de card com informações visuais claras.	US-RESERVE- 003
R3.15	Detalhes devem incluir filme, data, horário, cinema, assentos, status e método de pagamento para cada reserva.	US-RESERVE- 003
R3.16	Interface deve permitir que usuário visualize o pôster do filme associado à reserva.	US-RESERVE- 003
R3.17	Sistema deve exibir indicadores visuais de status da reserva (confirmada, pendente, cancelada).	US-RESERVE- 003
R3.18	Interface deve permitir que usuário acesse página dedicada de reservas separada das informações de perfil.	US-RESERVE- 003
R3.19	Sistema deve retornar todas as reservas, retornar reservas do usuário logado, cadastrar nova reserva, atualizar status da reserva e excluir reserva corretamente.	Swagger

/sessions ∂

- Base principal: US-SESSION-001.
- Complementado com requisitos técnicos extraídos do Swagger.

ID	Descrição	Origem
R4.1	Sistema deve permitir que usuário veja horários disponíveis para um filme selecionado.	US-SESSION- 001
R4.2	Horários devem exibir data, hora, cinema e disponibilidade.	US-SESSION- 001
R4.3	Interface deve permitir que usuário navegue para a seleção de assentos de um horário.	US-SESSION- 001

R4.4	Sistema deve retornar todas as sessões de filmes, cadastrar nova sessão, atualizar sessão e excluir sessão corretamente.	Swagger
R4.5	Sistema deve resetar todos os assentos em uma sessão para atualizar status para "disponível".	Swagger

/theaters ∂

• Base principal: Documentação Swagger, como não há histórias de usuário.

ID	Descrição	Origem
R5.1	Detalhes do cinema deve incluir ID, nome, capacidade, tipo e data do cadastro.	Swagger
R5.2	Sistema deve retornar todos os cinemas, cadastrar novo cinema, atualizar cinema e excluir cinema corretamente.	Swagger

/users ⊘

• Base principal: Documentação Swagger, como não há histórias de usuário.

ID	Descrição	Origem
R6.1	Detalhes do usuário devem incluir ID, nome, e-mail, função e data do cadastro.	Swagger
R6.2	Sistema deve retornar todos os usuários, atualizar usuário e excluir usuário corretamente.	Swagger

Experiência do Usuário 🔗

• Base principal: US-NAV-001.

ID	Descrição	Origem
R7.1	Cabeçalho deve estar presente em todas as páginas com links para áreas principais.	US-NAV-001
R7.2	Menu deve ser responsivo e se adaptar a diferentes tamanhos de tela (versão móvel).	US-NAV-001
R7.3	Usuário logado deve ter acesso à seções personalizadas como "Minhas Reservas" e "Perfil".	US-NAV-001
R7.4	Breadcrumbs ou elementos de navegação devem indicar o caminho atual do usuário.	US-NAV-001
R7.5	Interface deve ter links diretos para retornar à página anterior quando apropriado.	US-NAV-001
R7.6	Feedback visual deve indicar a página atual no menu de navegação.	US-NAV-001

📋 Cenários de teste planejados 🛭

As tabelas abaixo, separadas por rotas e experiência do usuário, apresentam os cenários de teste planejados com base nos requisitos definidos a partir das histórias de usuário e da documentação Swagger. Também estão incluídas as informações sobre a execução dos testes manuais, indicando se foram realizados via Postman (para validação de requisições da API) ou via navegador (para testes de interface e experiência do usuário).

/auth ∂

ID	Descrição	Pré- condições	Ferrame nta utilizada	Passo a passo	Resultad os esperad os	Resultad os obtidos	Status	Requisi tos que cobre	Priorida de 1
CT01	Login com dados válidos	Usuário cadastrado e dados válidos	Postman	1. Criar requisição POST para /auth/login 2. Incluir e-mail e senha válidos no corpo da requisição 3. Verificar autenticação de credenciais válidas	Status code 200	Status code 200	Passou	R1.6 e R1.7	Alta
CT02	Verificar redirecionament o para página inicial após login bem-sucedido	Usuário cadastrado e login bem- sucedido	Navegad or	1. Realizar login em /login com dados válidos 2. Analisar redirecionament o para página inicial com conta logada	Redirecio namento de login para página inicial	Redirecio namento de login para página inicial	Passou	R1.9	Média
CT03	Logout através do menu de navegação e acesso à áreas protegidas após logout	Usuário logado ou administra dor	Navegad or	1. Estar logado em uma conta 2. Clicar em "Sair" no canto direito do menu de navegação 3. Tentar acessar rotas protegidas	Logout com sucesso e acesso não permitido à rotas protegida s após desconex ão			R1.10 e R1.11	Alta

CT04	Verificar se sessão do usuário se mantém através de token JWT	Usuário logado ou administra dor	Postman	1. Criar requisição POST para /auth/login 2. Incluir e-mail e senha válidos no corpo da requisição 3. Verificar token JWT retornado na resposta da requisição	Presença de Token JWT durante sessão ativa			R1.8	Alta
CT05	Verificar remoção do token JWT do localStorage após logout bem-sucedido	Usuário desconect ado	Navegad or	Realizar logout Verificar remoção de token JWT no localStorage	Sem presença do tolen JWT após logout			R1.12	Alta
CT06	Verificar dados do perfil	Usuário logado ou administra dor	Postman	1. Criar requisição GET para /auth/me 2. Adicionar token de autenticação no cabeçalho 3. Verificar dados retornados na resposta da requisição	Status code 200 e presença de nome do usuário, e-mail e função da conta nos detalhes	Status code 200 e presença de nome do usuário, e-mail e função da conta nos detalhes	Passou	R1.13	Média
CT07	Cadastrar usuário com dados válidos	Usuário visitante e dados válidos	Postman	1. Criar requisição POST para /auth/registe r 2. Incluir nome, e- mail e senha válidos no corpo da requisição	Status code 201	Status code 201	Passou	R1.1	Alta
CT08	Cadastrar usuário com formato de e- mail inválido	Usuário visitante e e-mail inválido	Postman	1. Criar requisição POST para /auth/registe r 2. Incluir nome e senha válidos e e-mail inválido no corpo da requisição	Status code 400	Status code 400	Passou	R1.2	Média

СТ09	Cadastrar usuário com formato de senha inválido	Usuário visitante e senha inválida	Postman	 Criar requisição POST para /auth/registe r Incluir nome e e-mail válidos e senha inválida no corpo da requisição 	Status code 400	Status code 400	Passou	R1.3	Alta
CT10	Cadastrar usuário com e- mail já utilizado	Usuário visitante e e-mail já utilizado	Postman	1. Criar requisição POST para /auth/registe r 2. Incluir nome e senha válidos e e-mail já utilizado no corpo da requisição	Status code 400	Status code 400	Passou	R1.4	Alta
CT11	Verificar redirecionament o para página de login e autenticação automática após cadastro bem- sucedido	Usuário cadastrado	Navegad or	1. Realizar cadastro em /register com dados válidos 2. Analisar redirecionament o para página inicial com login em conta recém- cadastrada	Redirecio namento de cadastro para página inicial			R1.5	Baixa
CT12	Atualizar nome completo do usuário e verificar visualmente campos alterados	Usuário logado ou administra dor	Navegad or	1. Acessar "Perfil" no menu de navegação 2. Atualizar nome completo no campo "Nome Completo" em /profile 3. Verificar marcação visual dos campos alterados 4. Clicar em "Salvar Alterações"	Alteração do nome bem- sucedida e marcaçã o visual de campos alterados			R1.14 e R1.15	Média
CT13	Verificar mensagem de sucesso após	Usuário logado ou administra	Navegad or	Acessar "Perfil" no menu de navegação	Presença de uma mensage			R1.17	Baixa

	atualização bem-sucedida	dor e alteração bem - sucedida		 Atualizar nome completo no campo "Nome Completo" em /profile Clicar em "Salvar Alterações" Verificar mensagem de sucesso 	m de sucesso após salvar alteraçõe s				
CT14	Atualizar senha com dados válidos	Usuário logado ou administra dor	Postman	1. Criar requisição PUT para /auth/profile 2. Adicionar token de autenticação no cabeçalho 3. Incluir dados válidos para serem atualizados no corpo da requisição 4. Verificar confirmação de sucesso após salvar alterações no sistema	Status code 200	Status code 500	X Falhou	R1.19 e R1.16	Média
CT15	Verificar separação da página de perfil e da página de reservas	Usuário logado ou administra dor	Navegad or	 Realizar login em /login com dados válidos Acessar página de perfil pelo menu de navegação Acessar página de reservas pelo menu de navegação 	Página de perfil e página de reservas separada s			R1.18 e R3.18	Baixa

/movies @

ID	Descrição	Pré- condições	Ferrame nta	Passo a passo	Resultad os	Resultad os	Status	Requisi tos que	Priorida de 1
			utilizada		esperado	obtidos		cobre	
					s				

CT16	Obter todos os filmes	Usuário visitante, logado ou administrad or	Postman	1. Criar requisição GET para /movies 2. Incluir parâmetros válidos para filtrar filmes por título, gênero, ordem, limite ou página	Status code 200	Status code 200	Passou	R2.17	Média
CT17	Verificar detalhes do filme	Usuário visitante, logado ou administrad or	Postman	 Criar requisição GET para /movies/{id} Incluir ID do filme nos parâmetros Verificar dados retornados na resposta da requisição 	Status code 200 e presença de sinopse, elenco, diretor, data de lançamen to, duração nos detalhes	Status code 200, porém não há elenco nos detalhes do filme	X Falhou	R2.13	Médic
CT18	Verificar detalhes dos cards de filmes	Usuário visitante, logado ou administrad or	Navegad or	1. Acessar página principal ou seção "Filmes em Cartaz" 2. Clicar em "Ver Detalhes" de um dos filmes 3. Verificar detalhes	Cards de filmes devem ter título, classifica ção, gênero, duração do filme e data de lançamen to e devem ser acessado s facilment e			R2.9, R2.10 e R2.12	Média
CT19	Verificar presença de horários de sessões disponíveis na página de detalhes	Usuário visitante, logado ou administrad or	Navegad or	1. Acessar página principal ou seção "Filmes em Cartaz" 2. Clicar em "Ver Detalhes" de	Presença de horários de sessões disponívei s na			R2.15	Alta

				um dos filmes 3. Verificar horários de sessões	página de detalhes				
CT20	Cadastrar filme com dados válidos	Usuário administrad or e dados válidos	Postman	1. Criar requisição POST para /movies 2. Adicionar token de autenticação no cabeçalho 3. Incluir dados válidos no corpo da requisição	Status code 201	Status code 201	Passou	R2.17	Alta
CT21	Atualizar filme com dados válidos	Usuário administrad or e dados válidos	Postman	1. Criar requisição PUT para /movies/{id} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID do filme nos parâmetros 4. Incluir dados válidos para serem atualizados no corpo da requisição	Status code 200	Status code 200	Passou	R2.17	Alta
CT22	Excluir filme com dados válidos	Usuário administrad or e dados válidos	Postman	 Criar requisição DELETE para /movies/{id} Adicionar token de autenticação no cabeçalho Incluir ID do filme nos parâmetros 	Status code 200	Status code 200	Passou	R2.17	Média

CT23	Verificar navegação para reserva a partir de horários disponíveis	Usuário visitante, logado ou administrad or	Navegad or	1. Acessar página principal ou seção "Filmes em Cartaz" 2. Clicar em "Ver Detalhes" de um dos filmes 3. Escolher horário de sessão em "Selecionar Assentos" 4. Acessar página de reserva	Ir para reserva por meio de horários disponívei s com sucesso		R2.16	Média
CT24	Verificar responsividade da página inicial	Usuário visitante, logado ou administrad or	Navegad or	Testar aplicação em diferentes tamanhos de tela	Tela da aplicação se adapta a diferentes formatos		R2.3	Alta
CT25	Verificar banner com informações sobre o cinema na página inicial	Usuário visitante, logado ou administrad or	Navegad or	Acessar página inicial Analisar banner com informações sobre cinema	Presença de banner sobre cinema na página inicial		R2.1	Média
CT26	Verificar seção de "Filmes em Cartaz"	Usuário visitante, logado ou administrad or	Navegad or	1. Acessar página inicial 2. Clicar em "Filmes em Cartaz" no menu de navegação	Acessar "Filmes em Cartaz" com sucesso		R2.2	Alta
CT27	Verificar exibição dos pôsteres dos filmes	Usuário visitante, logado ou administrad or	Navegad or	1. Acessar página principal ou seção "Filmes em Cartaz" 2. Clicar em "Ver Detalhes" de um dos filmes 3. Analisar pôster	Pôster do filme deve ser grande e com alta qualidade		R2.8 e R2.14	Média
CT28	Verificar layout de lista de filmes em exibição	Usuário visitante, logado ou	Navegad or	1. Acessar página principal ou	Filmes devem ser exibidos		R2.7	Média

		administrad or		seção "Filmes em Cartaz" 2. Analisar layout da lista de filmes	com layout em grid			
CT29	Verificar responsividade da lista de filmes	Usuário visitante, logado ou administrad or	Navegad or	1. Acessar página principal ou seção "Filmes em Cartaz" 2. Testar página em diferentes tamanhos de tela	Tela da página de filmes se adapta a diferentes formatos		R2.11	Alta
CT30	Verificar opções personalizados no menu para usuários autenticados	Usuário logado ou administrad or	Navegad or	1. Acessar página principal 2. Analisar opções no menu para usuário logado ou administrador	Presença de opções personali zadas no menu		R2.6	Média

/reservations ∂

ID	Descrição	Pré- condições	Ferrame nta utilizada	Passo a passo	Resultad os esperad os	Resultad os obtidos	Status	Requisi tos que cobre	Priorida de <u>1</u>
CT31	Obter todas as reservas	Usuário administra dor	Postman	1. Criar requisição GET para /reservation s 2. Adicionar token de autenticação no cabeçalho 3. Incluir parâmetros válidos para filtrar por página e/ou limite	Status code 200	Status code 200	Passou	R3.19	Média
СТ32	Obter reservas do usuário atual	Usuário logado ou	Postman	1. Criar requisição GET para	Status code 200	Status code 200	V Passou	R3.19	Alta

CT33	Verificar detalhes da reserva	Usuário logado ou administra dor	Postman	/reservations /me 2. Adicionar token de autenticação no cabeçalho 1. Criar requisição GET para /reservations /{id} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID da reserva nos parâmetros 4. Verificar dados retornados na resposta da requisição	Status code 200 e presença de filme, data, horário, cinema, assentos, status e método de pagamen to nos detalhes	Status code 200 e presença de filme, data, horário, cinema, assentos , status e método de pagamen to nos detalhes	Passou	R3.15	Alta
CT34	Cadastrar reserva com dados válidos	Usuário logado ou administra dor e	Postman	1. Criar requisição POST para /reservation	Status code 201	Status code 201	✓ Passou	R3.19	Alta
		dados válidos		s 2. Adicionar token de autenticação no cabeçalho 3. Incluir dados válidos no corpo da requisição					

				3. Incluir dados de mais de um assento no corpo da requisição					
CT36	Cadastrar assentos já reservados em uma reserva	Usuário logado ou administra dor e assento já reservado	Postman	1. Criar requisição POST para /reservation s 2. Adicionar token de autenticação no cabeçalho 3. Incluir dados de assento já reservado no corpo da requisição	Status code 400	Status code 400	Passou	R3.4	Alta
CT37	Verificar layout de assentos do cinema com usuário logado	Usuário logado ou administra dor	Navegad or	 Escolher horário em sessão de um filme Analisar layout de assentos 	Visualiza ção correta do layout de assentos do cinema			R3.1	Média
CT38	Verificar cores por disponibilidade e seleção dos assentos	Usuário logado ou administra dor	Navegad or	 Escolher horário em sessão de um filme Analisar cores por disponibilidad e e seleção em layout de assentos 	Assentos precisam estar com coloraçã o correspo ndente a disponibil idade e seleção			R3.2 e 3.12	Média
СТ39	Verificar subtotal ao que assentos são selecionados	Usuário logado ou administra dor	Navegad or	 Escolher horário em sessão de um filme Verificar subtotal ao escolher assentos 	Presença do subtotal ao escolher assentos			R3.5	Média

CT40	Verificar redirecionamento para página de checkout após seleção de assentos	Usuário visitante ou logado	Navegad or	1. Selecionar assentos em sessão de um filme 2. Clicar em "Continuar para Pagamento" 3. Analisar redirecioname nto para página de checkout em /checkout	Redirecio namento da página de escolha de assentos para página de checkout	R3.6	Alta
CT41	Verificar resumo dos assentos selecionados e valor total da compra na página de checkout	Usuário visitante, logado ou administra dor	Navegad or	1. Selecionar assentos em sessão de um filme 2. Clicar em "Continuar para Pagamento" 3. Verificar resumo dos assentos e total da compra na página de checkout em /checkout	Presença do resumo dos assentos seleciona dos e valor total da compra no checkout	R3.7 e R3.8	Alta
CT42	Verificar seleção do método de pagamento, processamento do pagamento e confirmação da reserva	Usuário visitante, logado ou administra dor	Navegad or	 Selecionar assentos em sessão de um filme Clicar em "Continuar para Pagamento" Escolher método de pagamento na página de checkout em /checkout Clicar em "Finalizar Compra" Verificar confirmação 	Presença do método de pagamen to na página de checkout, processa mento de pagamen to e confirma ção da reserva mostrado s com sucesso	R3.9, R3.10 e R3.11	Alta

				visual da reserva					
CT43	Verificar reservas no link "Minhas Reservas" e verificar informações visuais de cada uma	Usuário logado ou administra dor	Navegad or	1. Clicar em "Minhas Reservas" no menu de navegação 2. Analisar se reservas estão presentes, em formato de card e possuem o pôster do filme	Acessar minhas reservas por meio do menu de navegaç ão e visualizálas em formato de card e com o pôster do filme			R3.13, R3.14 e R3.16	Média
CT44	Verificar indicadores visuais de status da reserva	Usuário logado ou administra dor	Navegad or	1. Clicar em "Minhas Reservas" no menu de navegação 2. Analisar se card possui indicar visual com status da reserva	Presença de indicador visual com status da reserva			R3.17	Baixa
CT45	Atualizar status da reserva com dados válidos	Usuário administra dor	Postman	1. Criar requisição PUT para /reservations /{id} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID da reserva nos parâmetros 4. Incluir dados válidos para serem atualizados no corpo da requisição	Status code 200	Status code 200	Passou	R3.19	Alta
CT46	Excluir reserva com dados válidos	Usuário administra dor e	Postman	1. Criar requisição DELETE para	Status code 200	Status code 200	✓ Passou	R3.19	Média

dados	/reservations
válidos	/{id}
	2. Adicionar
	token de
	autenticação
	no cabeçalho
	3. Incluir ID da
	reserva nos
	parâmetros

/sessions ∂

ID	Descrição	Pré- condições	Ferrame nta utilizada	Passo a passo	Resultad os esperad os	Resultad os obtidos	Status	Requisi tos que cobre	Priorida de 1
CT47	Obter todas as sessões	Usuário visitante, logado ou administra dor	Postman	 Criar requisição GET para /sessions Incluir parâmetros válidos para filtrar por filme, cinema, data, limite e/ou página 	Status code 200	Status code 200	Passou	R4.4	Média
CT48	Verificar detalhes dos horários	Usuário visitante, logado ou administra dor	Postman	 Criar requisição GET para /sessions/{i d} Incluir ID da sessão nos parâmetros Verificar dados retornados na resposta da requisição 	Status code 200 e presença de data, hora, cinema e disponibil idade nos detalhes	Status code 200 e presença de data, hora, cinema e disponibi lidade nos detalhes	V Passou	R4.2	Média
CT49	Verificar horários disponíveis para um filme e seleção de assentos de um horário	Usuário visitante, logado ou administra dor	Navegad or	 Acessar "Filmes em Cartaz" Clicar em "Ver Detalhes" de um filme Analisar horários 	Presença de horários disponíve is para um filme e navegaç			R4.1 e R4.3	Alta

				disponíveis para sessões do filme 4. Clicar em "Selecionar Assentos" para escolher um horário	ão até seleção de assentos bem- sucedida				
CT50	Cadastrar sessão com dados válidos	Usuário administra dor e dados válidos	Postman	1. Criar requisição POST para /sessions 2. Adicionar token de autenticação no cabeçalho 3. Incluir dados válidos no corpo da requisição	Status code 201	Status code 201	Passou	R4.4	Alta
CT51	Atualizar sessão com dados válidos	Usuário administra dor e dados válidos	Postman	1. Criar requisição PUT para /sessions/{i d} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID da sessão nos parâmetros 4. Incluir dados válidos para serem atualizados no corpo da requisição	Status code 200	Status code 200	Passou	R4.4	Alta
CT52	Excluir sessão com dados válidos	Usuário administra dor e dados válidos	Postman	1. Criar requisição DELETE para /sessions/{i d} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID da sessão nos	Status code 200	Status code 200	Passou	R4.4	Média

				parâmetros					
CT53	Resetar assentos e verificar status dos assentos	Usuário administra dor	Postman	1. Criar requisição PUT para /sessions/{i d}/reset- seats 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID da sessão nos parâmetros	Status code 200	Status code 200	Passou	R4.5	Média

/theaters ∂

ID	Descrição	Pré- condiçõe s	Ferrame nta utilizada	Passo a passo	Resultad os esperad os	Resultad os obtidos	Status	Requisit os que cobre	Priorida de 1
CT54	Obter todos os cinemas	Dados válidos	Postman	1. Criar requisição GET para /theaters 2. Incluir parâmetros válidos para filtrar por tipo do cinema, ordem, limite e/ou página	Status code 200	Status code 200	Passou	R5.2	Média
CT55	Verificar detalhes do cinema	Dados válidos	Postman	1. Criar requisição GET para /theaters/{ id} 2. Incluir ID do cinema nos parâmetros 3. Verificar dados retornados na resposta da requisição	Status code 200 e presença de ID, nome, capacida de, tipo e data do cadastro nos detalhes	Status code 200 e presença de ID, nome, capacida de, tipo e data do cadastro nos detalhes	Passou	R5.1	Média
CT56	Cadastrar cinema com dados válidos	Usuário administra dor e	Postman	1. Criar requisição	Status code 201	Status code 201	V Passou	R5.2	Alta

		dados válidos		POST para /theaters 2. Adicionar token de autenticação no cabeçalho 3. Incluir dados válidos no corpo da requisição					
CT57	Atualizar cinema com dados válidos	Usuário administra dor e dados válidos	Postman	1. Criar requisição PUT para /theaters/{ id} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID do cinema nos parâmetros 4. Incluir dados válidos para serem atualizados no corpo da requisição	Status code 200	Status code 200	Passou	R5.2	Alta
CT58	Excluir cinema com dados válidos	Usuário administra dor e dados válidos	Postman	1. Criar requisição DELETE para /theaters/{ id} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID do cinema nos parâmetros	Status code 200	Status code 200	Passou	R5.2	Média

/users ⊘

ID	Descrição	Pré- condições	Ferrame nta utilizada	Passo a passo	Resultad os esperad	Resultad os obtidos	Status	Requisi tos que cobre	Priorida de <u>1</u>
					os				

CT59	Obter todos os usuários	Usuário administra dor	Postman	1. Criar requisição GET para /users 2. Adicionar token de autenticação no cabeçalho 3. Incluir parâmetros válidos para filtrar por página, limite e/ou função	Status code 200	Status code 200	Passou	R6.2	Média
CT60	Verificar detalhes do usuário	Usuário administra dor	Postman	1. Criar requisição GET para /users/{id} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID do usuário nos parâmetros 4. Verificar dados retornados na resposta da requisição	Status code 200 e presença de ID, nome, e- mail, função e data do cadastro nos detalhes	Status code 200 e presença de ID, nome, e- mail, função e data do cadastro nos detalhes	Passou	R6.1	Médic
CT61	Atualizar usuário com dados válidos	Usuário administra dor e dados válidos	Postman	1. Criar requisição PUT para /users/{id} 2. Adicionar token de autenticação no cabeçalho 3. Incluir ID do usuário nos parâmetros 4. Incluir dados válidos para serem atualizados no corpo da requisição	Status code 200	Status code 200	Passou	R6.2	Médic
CT62	Excluir usuário com dados válidos	Usuário administra	Postman	1. Criar requisição	Status code 200	Status code 200	✓ Passou	R6.2	Médio

dor e	DELETE para
dados	/users/{id}
válidos	2. Adicionar
	token de
	autenticação
	no cabeçalho
	3. Incluir ID do
	usuário nos
	parâmetros

Experiência do Usuário 🔗

ID	Descrição	Pré- condições	Ferrame nta utilizada	Passo a passo	Resultad os esperado s	Resulta dos obtidos	Status	Requisi tos que cobre	Priorida de <u>1</u>
CT63	Verificar presença do cabeçalho contendo links rápidos para áreas principais nas páginas da aplicação	Usuário visitante, logado ou administrad or	Navegad or	1. Analisar links rápidos para áreas principais no cabeçalho da aplicação em qualquer página	Presença de de cabeçalh o e links diretos			R2.4, R2.5 e R7.1	Alta
CT64	Verificar responsividade do menu (versão móvel)	Usuário visitante, logado ou administrad or	Navegad or	Testar menu da aplicação na versão móvel	Tela do menu se adapta a versão móvel			R7.2	Alta
CT65	Verificar exibição de seções personalizadas para usuário logado	Usuário logado ou administrad or	Navegad or	 Acessar página principal após login Observar o menu de navegação 	Seções "Minhas Reservas " e "Perfil" devem estar visíveis no menu de navegaçã o para usuários autentica dos			R7.3	Média
CT66	Verificar caminho atual do usuário pelos elementos de navegação	Usuário logado ou administrad or	Navegad or	Acessar alguma página da aplicação	Presença de elemento s de			R7.4	Baixa

				2. Verificar elementos de navegação (breadcrumb s, título da página, destaque no menu)	navegaçã o que mostrem caminho atual do usuário			
CT67	Verificar retorno para página anterior por links diretos	Usuário visitante, logado ou administrad or	Navegad or	1. Acessar alguma página da aplicação 2. Verificar se possui botão para voltar à página anterior	Presença de retorno à página anterior		R7.5	Média
CT68	Verificar indicação da página atual no menu por feedback visual	Usuário visitante, logado ou administrad or	Navegad or	1. Acessar alguma página da aplicação 2. Verificar indicação da página atual no menu	Presença de indicação da página atual no menu		R7.6	Baixa



ID	Título	Probabilidade (1-3)	Impacto (1-3)	Risco
Risco 001	Falha na autenticação de usuários permite acessos não autorizados	2	3	6
Risco 002	Ausência de validação dos dados inseridos	2	3	6
Risco 003	Falha da API ou indisponibilidade do back- end	2	3	6
Risco 004	Tempo insuficiente para execução de todos os testes planejados	3	2	6

Risco 005	Interface e back-end exibem informações inconsistentes	2	3	6
Risco 006	Interface exibe elementos quebrados em telas menores (responsividade)	2	2	4
Risco 007	Botões ou links importantes não funcionam na interface	2	2	4
Risco 008	Sistema não fornece feedback visual após ações importantes	2	2	4
Risco 009	Documentação da API desatualizada ou incompleta	2	2	4
Risco 010	Token JWT expirado permite continuar acessando a API	1	3	3



∠ Cobertura de testes (Manuais e Automatizados)
_e

Path Coverage (input) ${\mathscr Q}$

Operator Coverage (input) \mathscr{O}

Parameter Coverage (input) 🕖

Content-Type Coverage (input e output) ∅

Status Code Coverage (Output) ${\mathscr O}$



Testes manuais 🕖

Testes automatizados 🕖

