

**QUALIDADE DE SOFTWARE**

Nome completo

Análise de Qualidade

Cidade

Ano

# RESUMO

Este projeto tem como objetivo aplicar na prática os conhecimentos adquiridos ao longo do curso de Teste de Software da EBAC, simulando o trabalho de um profissional de QA em um time ágil. A partir de três histórias de usuário refinadas (adicionar item ao carrinho, login na plataforma e API de cupons), foram definidos critérios de aceitação em Gherkin, criados casos de teste manuais e implementada automação utilizando Cypress para testes web e de API. Também foram aplicadas técnicas de testes funcionais, testes de performance com JMeter e integração contínua via Jenkins. Toda a estrutura do projeto foi pensada seguindo boas práticas de codificação, organização de repositório e documentação. O resultado é um projeto completo que pode ser utilizado como portfólio profissional, demonstrando domínio de ferramentas modernas e abordagem profissional em QA.

# SUMÁRIO

[1. RESUMO 2](#_Toc85541169)

[2. SUMÁRIO 3](#_Toc85541170)

[3. INTRODUÇÃO 4](#_Toc85541171)

[4. O PROJETO 5](#_Toc85541172)

[4.1 Estratégia de teste 5](#_Toc85541173)

[4.2 Critérios de aceitação 5](#_Toc85541174)

[4.2.1 História [US-0001] – Adicionar item ao carrinho 5](#_Toc85541175)

[4.2.2 História [US-0002] – Login na plataforma 6](#_Toc85541176)

[4.2.3 História [US-0003] – API de cupons 6](#_Toc85541177)

[4.3 Casos de testes 6](#_Toc85541178)

[4.3.1 História de usuário 1: 6](#_Toc85541179)

[4.3.2 História de usuário 2: 6](#_Toc85541180)

[4.3.1 História de usuário 3: API de Cupom 7](#_Toc85541181)

[4.4 Repositório no Github 7](#_Toc85541182)

[4.5 Testes automatizados 7](#_Toc85541183)

[4.6 Integração contínua 8](#_Toc85541184)

[4.7 Testes de performance 8](#_Toc85541185)

[5. CONCLUSÃO 9](#_Toc85541186)

[6. REFERÊNCIAS BIBLIOGRÁFICAS 9](#_Toc85541187)

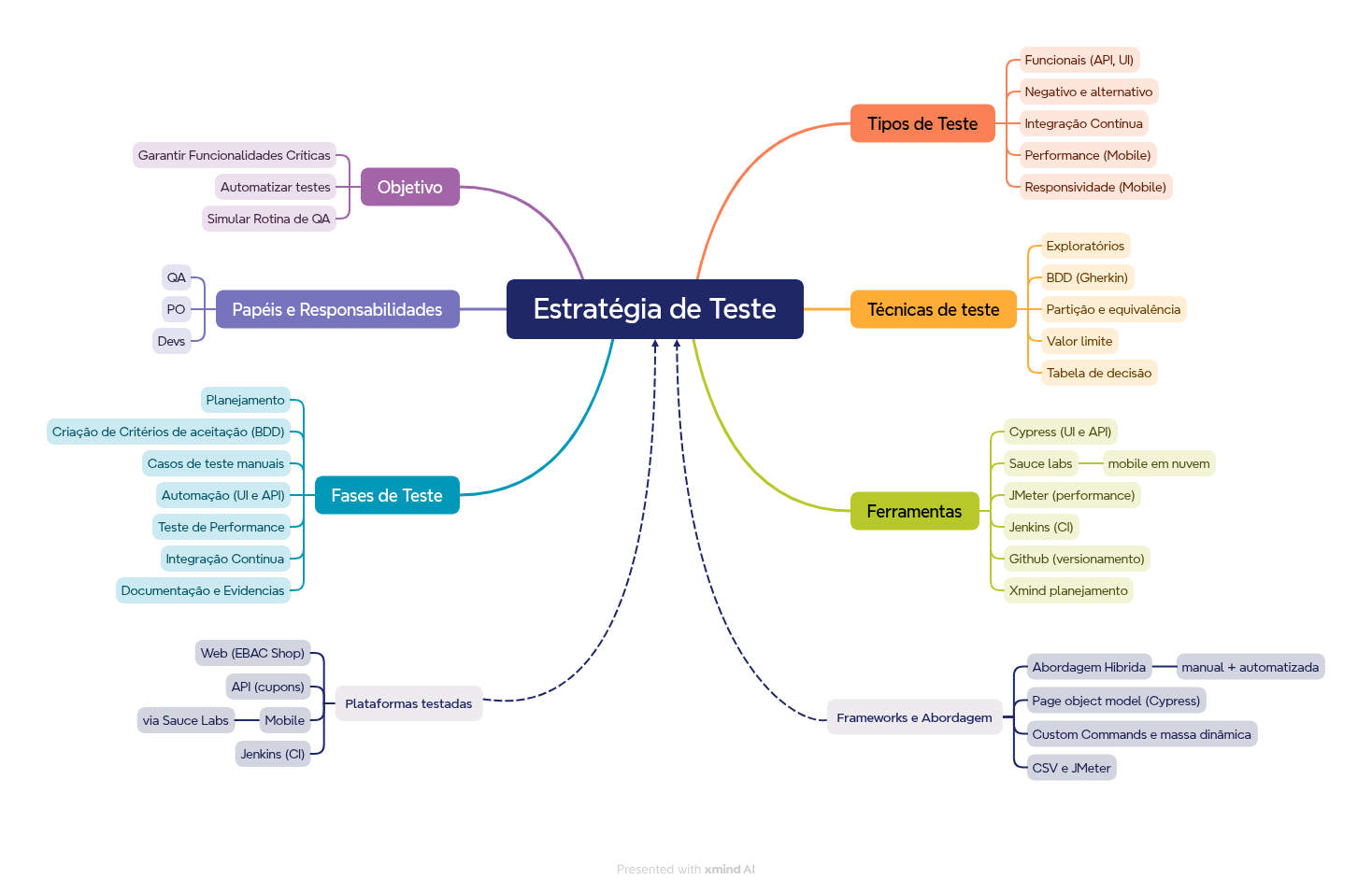
# INTRODUÇÃO

A qualidade de software desempenha um papel fundamental na entrega de produtos digitais confiáveis, seguros e eficientes. Com o crescimento da complexidade dos sistemas e a exigência por entregas contínuas, o papel do engenheiro de qualidade se tornou indispensável dentro das equipes ágeis. Este trabalho propõe a construção de um projeto completo de testes baseado em histórias reais do sistema EBAC-SHOP, com o objetivo de exercitar o ciclo completo de um QA: desde a análise e criação de critérios de aceitação até a implementação da automação, testes de performance e integração contínua.

Serão abordados testes funcionais manuais e automatizados (UI e API), uso de frameworks modernos como Cypress e JMeter, estratégias de teste bem definidas e práticas modernas de versionamento e CI/CD. O projeto também simula o ambiente de um time ágil com foco em entregas incrementais e testes orientados a comportamento (BDD). Com isso, espera-se consolidar o aprendizado técnico e fornecer uma base sólida para atuação no mercado de qualidade de software.

# O PROJETO

## Estratégia de teste



## Critérios de aceitação

## 4.2.1 História [US-0001] – Adicionar item ao carrinho

**Cenário 1**

***Adicionar um item ao carrinho com quantidade válida***

> Dado que o cliente está logado na plataforma

> E acessa a página de um produto

> Quando ele informa a quantidade "**2**"

> E clica em "**Adicionar ao carrinho**"

> Então o item deve ser adicionado com sucesso ao carrinho

> E a mensagem "**Item adicionado com sucesso!**" deve ser exibida

**Cenário 2**

***Tentar adicionar mais de 10 unidades de um produto***

> Dado que o cliente está na página de um produto

> Quando ele informa a quantidade "**11**"

> E clica em "**Adicionar ao carrinho**"

> Então uma mensagem de erro deve ser exibida dizendo "**Quantidade máxima permitida é 10**"

**Cenário 3**

**Esquema do Cenário:** Gerar cupom conforme valor total dos itens no carrinho

> Dado que o cliente adicionou produtos ao carrinho

> Quando o valor total dos itens for "**<valor\_total>**"

> Então o sistema deve aplicar o cupom "**<cupom>**" com "**<desconto>**"

**Exemplos:**

|  |  |  |
| --- | --- | --- |
| valor\_total | cupom | desconto |
| 250 | CUMPOM10 | 10% |
| 620 | CUPOM15 | 15% |
| 180 | NENHUM | 0% |

**4.2.2 História [US-0002] – Login na Plataforma**

#### **Cenário 1**

#### **Login com credenciais válidas (usuário ativo) gherkin**

> Dado que o cliente está na página de login

> E possui cadastro ativo

> Quando informar o e-mail "user1@ebac.com" e a senha "psw123"

> E clica no botão "Entrar"

> Então deve ser autenticado com sucesso

> E deve ser redirecionado para a página de pedidos

**Cenário 2**

**Login com senha inválida**

> Dado que o cliente está na página de login

> Quando informar o nome de usuário "user\_ebac" e a senha "errada123"

> E clica em "Entrar"

> Então uma mensagem de erro "Usuário ou senha inválidos" deve ser exibida

**Cenário 3**

**Bloqueio de login após 3 tentativas erradas**

> Dado que o cliente acessa a página de login

> Quando ele realiza 3 tentativas de login com os dados:

| login | senha |

| <login> | <senha> |

> Então o sistema deve bloquear o login por 15 minutos

> E exibir a mensagem "Login bloqueado por múltiplas tentativas. Tente novamente em 15 minutos."

Exemplos:

| login | senha |

| user1\_ebac | senhaErrada1 |

**4.2.3 História [US-0003] – API de Cupons**

**Cenário 1**

**Requisição GET para listar todos os cupons**

> Dado que o admin está autenticado via API

> Quando ele envia uma requisição GET para o endpoint "/coupons"

> Então a resposta deve conter a lista de cupons cadastrados

> E o status da resposta deve ser 200

**Cenário 2**

**Cadastrar cupom válido**

> Dado que o admin está autenticado

> Quando ele envia uma requisição POST com os dados:

| code | amount | discount\_type | description |

| Ganhe10 | 10.00 | fixed\_product | Cupom de teste válido |

> Então o sistema deve retornar status 201

> E o cupom cadastrado deve aparecer na listagem de cupons

**Cenário 3**

**Validar unicidade do código do cupom**

> Dado que já existe um cupom com código "<codigo>"

> Quando o admin tenta cadastrar um novo cupom com o mesmo código

> Então a resposta deve conter status 400

> E a mensagem "O código do cupom já está em uso" deve ser exibida

Exemplos:

| codigo |

| Ganhe10 |

| BLACKFRIDAY |

## Casos de testes

* Crie pelo menos 3 casos de testes para cada história de usuário, sempre que possível, usando as técnicas de testes (partição de equivalência, valor limite, tabela de decisão etc.).
* Considere sempre o caminho feliz (fluxo principal) e o caminho alternativo e negativo (fluxo alternativo). Exemplo de cenário negativo: “Ao preencher com usuário e senha inválidos deve exibir uma mensagem de alerta...”
* Referência: Módulo 4 e 5

## História de usuário 1

**CT01: Adicionar item com quantidade válida (caminho feliz)**

* 1. **Objetivo:** Verificar se um produto é adicionado corretamente ao carrinho com quantidade válida.
  2. **Pré-condição:** Usuário logado.
  3. **Dados de entrada:** Produto “Notebook”, quantidade 2.
  4. **Passos:**
     1. Acessar página do produto “Notebook”.
     2. Informar quantidade: 2.
     3. Clicar em “Adicionar ao carrinho”.
  5. **Resultado esperado:** Produto é adicionado com sucesso. Exibe a mensagem "Item adicionado com sucesso!".
  6. **Técnica:** Partição de Equivalência.

**CT02: Tentar adicionar mais de 10 itens (limite superior inválido)**

* 1. **Objetivo:** Verificar a regra de restrição de quantidade máxima.
  2. **Pré-condição:** Usuário na página de produto.
  3. **Dados de entrada:** Produto “Mouse Gamer”, quantidade 11.
  4. **Passos:**
     1. Acessar página do produto “Mouse Gamer”.
     2. Informar quantidade: 11.
  5. **Clicar** em “Adicionar ao carrinho”.
  6. **Resultado esperado:** Sistema bloqueia a ação e exibe a mensagem "Quantidade máxima permitida é 10".
  7. **Técnica:** Valor Limite (excedente)

**CT03: Validar geração de cupom com valor entre R$ 200 e R$ 600**

* 1. **Objetivo:** Verificar se o sistema aplica automaticamente o cupom de 10% para valores elegíveis.
  2. **Pré-condição:** Usuário logado.
  3. **Dados de entrada:** Produto “Fone Bluetooth”, valor R$ 300.
  4. **Passos:**
     1. Adicionar produtos que somem R$ 300.
     2. Acessar o carrinho.
  5. **Resultado esperado:** Cupom de 10% é aplicado automaticamente.
  6. **Técnica:** Tabela de decisão.

## História de usuário 2:

**CT01: Login com e-mail válido e senha correta (caminho feliz)**

* 1. **Objetivo:** Validar login bem-sucedido com e-mail válido.
  2. **Pré-condição:** Usuário ativo já cadastrado.
  3. **Dados de entrada:** E-mail: user1\_ebac@ebac.com, Senha: psw!ebac@test
  4. **Passos:**
     1. Acessar a página de login.
     2. Inserir e-mail e senha válidos.
     3. Clicar em “Entrar”.
  5. **Resultado esperado:** Login realizado com sucesso e redirecionamento para página de pedidos.
  6. **Técnica:** Partição de Equivalência.

**CT02: Login com usuário inválido (fluxo negativo)**

* 1. **Objetivo:** Verificar rejeição de login com usuário inválido.
  2. **Pré-condição:** Página de login acessível.
  3. **Dados de entrada:** Usuário: “*uzer12-ebbac.com*”,

Senha: “*senhaErrada*”

* 1. **Passos:**
     1. Inserir CPF inválido.
     2. Inserir senha inválida.
     3. Clicar em “Entrar”.
  2. **Resultado esperado:** Mensagem de erro “Usuário ou senha inválidos”.
  3. **Técnica:** Partição de Equivalência negativa.

**CT03: Bloqueio de login após 3 tentativas com senha errada**

* 1. **Objetivo:** Garantir que o sistema bloqueie o login após 3 erros consecutivos.
  2. **Pré-condição:** Usuário existente, porém insere senha errada.
  3. **Dados de entrada:** Nome de usuário: user1\_ebac, Senha: errada123 (3 vezes)
  4. **Passos:**
     1. Tentar login com senha errada (3 vezes seguidas).
  5. **Resultado esperado:** Sistema exibe mensagem de bloqueio e impede novos logins por 15 minutos.
  6. **Técnica:** Valor Limite + Fluxo alternativo.

## História de usuário 3: API de Cupom

CT01:

CT02:

CT03:

## Repositório no Github

* Crie um repositório no github com o nome TCC-EBAC;
* Deixe o repositório publico até a análise dos tutores;
* Neste repositório você deve subir este arquivo e todos os código fontes da automação WEB, API, Mobile, Performance e CI.
* Referência: Módulo 10
* Link do repositório: <cole o link aqui>

## Testes automatizados

* + 1. Automação de UI
* Crie um projeto de automação no Cypress;
* Crie uma pasta chamada UI para os testes WEB da História de Usuário [US-0001] – Adicionar item ao carrinho;
* Na automação deve adicionar pelo menos 3 produtos diferentes e validar se os itens foram adicionados com sucesso.
  + 1. Automação de API
* Crie uma pasta chamada API para os testes de API da História de usuário **“Api de cupons”**.
* Faça a automação de **listar** os cupons e **cadastrar** cupom, seguindo as regras da História de usuário.
* Exemplo da automação de Api – GET

it('Deve listar todos os cupons cadastrados', () => {

cy.request({

method: 'GET',

url: 'coupons',

headers: {

authorization: 'código\_da\_autorização\_aqui'

}

}).should((response) => {

cy.log(response)

expect(response.status).to.equal(200)

})

});

* Obs.: Considere todas as boas práticas de otimização de cenários (Page Objects, Massa de dados, Custom Commands, elementos etc.).
* Referência: Módulo 11, 12 e 14

## Integração contínua

* Coloque os testes automatizados na integração contínua com jenkins, criando um job para execução da sua automação;
* Compartilhe o *jenkinsfile* no repositório, junto ao seu projeto.
* Referência: Módulo 15

## Testes de performance

* Usando o Apache Jmeter, faça um teste de performance com o fluxo de login da História de usuário: [US-0002] – Login na plataforma
* Crie um template de gravação no jmeter (recording);
* Use massa de dados dinâmica em arquivo CSV;
* Referência: Módulo 18
* Configurações do teste de performance:

-Usuários virtuais: 20  
-Tempo de execução: 2 minutos  
-RampUp: 20 segundos  
-Massa de dados: Usuário / senha:

user1\_ebac / psw!ebac@test  
user2\_ebac / psw!ebac@test  
user3\_ebac / psw!ebac@test  
user4\_ebac / psw!ebac@test  
user5\_ebac / psw!ebac@test

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

* DICA: Em uma das requisições, após a gravação, vai aparecer os parâmetros usado. Substitua esses parâmetros pela sua massa de dados, conforme aprendido em aula:



# CONCLUSÃO

Coloque sua experiência na realização do trabalho, o que aprendeu, quais lições pode aplicar em sua vida profissional etc.

# REFERÊNCIAS BIBLIOGRÁFICAS

Seguir regras ABNT