

**QUALIDADE DE SOFTWARE**

Viviane Viana Monteiro Peres

Análise de Qualidade

Curitiba

2023

# RESUMO

Projeto de conclusão do curso de Qualidade de Software. Este projeto visa apresentar o processo de qualidade de software, desde o planejamento, incluindo a estratégia de testes até a implementação de testes manuais e automatizados.

# SUMÁRIO

[1. RESUMO 2](#_Toc85541188)

[2. SUMÁRIO 3](#_Toc85541189)

[3. INTRODUÇÃO 4](#_Toc85541190)

[4. O PROJETO 5](#_Toc85541191)

[4.1 Estratégia de teste 5](#_Toc85541192)

[4.2 Critérios de aceitação 5](#_Toc85541193)

[4.2.1 História de usuário 1: [US-0001] – Adicionar item ao carrinho 5](#_Toc85541194)

[4.2.2 História de usuário 2: [US-0002] – Login na plataforma 6](#_Toc85541195)

[4.2.3 História de usuário 2: [US-0003] – API de cupons 6](#_Toc85541196)

[4.3 Casos de testes 6](#_Toc85541197)

[4.3.1 História de usuário 1: 6](#_Toc85541198)

[4.3.2 História de usuário 2: 6](#_Toc85541199)

[4.3.1 História de usuário 3: API de Cupom 7](#_Toc85541200)

[4.4 Repositório no Github 7](#_Toc85541201)

[4.5 Testes automatizados 7](#_Toc85541202)

[4.6 Integração contínua 8](#_Toc85541203)

[4.7 Testes de performance 8](#_Toc85541204)

[5. CONCLUSÃO 9](#_Toc85541205)

[6. REFERÊNCIAS BIBLIOGRÁFICAS 9](#_Toc85541206)

# INTRODUÇÃO

Esse documento apresenta um projeto de qualidade de software para a loja virtual da EBAC, considerando as necessidades do cliente, adotando metodologias eficientes através de um processo robusto e utilizando as técnicas e ferramentas apresentadas durante todo o curso.

# O PROJETO

Para o Projeto mestre da primeira parte do curso Profissão: Engenheiro de Qualidade, você deve considerar as histórias de usuário já refinadas e como se você estivesse participando de um time ágil. As funcionalidades devem seguir todo o fluxo de trabalho de um QA, desde o planejamento até a entrega. Siga as etapas dos subtópicos para te orientar no trabalho. Todas as boas práticas, tanto de documentação, escrita e desenvolvimento, serão consideradas na nota. Portanto caprichem, pois além de trabalho servir como nota para o curso, vai servir como Portfólio em seu GitHub.

## Estratégia de teste

* Faça uma estratégia de testes em um mapa mental, seguindo algumas diretrizes como objetivos, papeis e responsabilidades, fases de testes, padrões, tipos de testes, técnicas de testes, ambientes, ferramentas, abordagem (manual ou automatizado), framework ou ferramenta usados, plataformas (web, api, mobile), etc.;
* Referência: Módulo 5
* Após fazer sua estratégia de teste, tire um print e cole aqui:

Uma imagem contendo Diagrama

Descrição gerada automaticamente

## Critérios de aceitação

* Considere as histórias de usuário: [US-0001] – Adicionar item ao carrinho, [US-0002] – Login na plataforma e [US-0003] – API de cupons
* Para cada uma delas crie pelo menos 2 critérios de aceitação usando a linguagem Gherkin;
* Em pelo menos um dos critérios, usar tabela de exemplos ( Esquema do Cenário / Scenario Outline);
* Referência: Módulo 8

## História de usuário 1: [US-0001] – Adicionar item ao carrinho

Critérios de aceitação:

1. Não é permitido inserir mais de 10 itens de um mesmo produto ao carrinho;
2. Os valores não podem ultrapassar a R$ 990,00;
3. Valores entre R$ 200 e R$ 600 ganham cupom de 10%
4. Valores acima de R$ 600 ganham cupom de 15%

Cenário 1: Adicionar um item ao carrinho

Dado que o usuário está no EBAC Shop  
Quando seleciona um produto

E clica em comprar

Então o item deve ser adicionado ao carrinho

Cenário 2: Adicionar mais de 10 itens ao carrinho

Dado que o usuário está no EBAC Shop  
Quando seleciona um produto

E altera a quantidade do item para 12

E clica em comprar

Então os itens não devem ser adicionados ao carrinho

## História de usuário 2: [US-0002] – Login na plataforma

Critérios de aceitação:

1. Somente usuários ativos podem fazer login;
2. Deve exibir uma mensagem de erro caso o usuário erre o login e senha;
3. Login deve permitir e-mail, nome de usuário ou CPF;
4. Se o usuário errar por 3 vezes a senha, deve travar por 15 minutos o login

Cenário 1: Login com usuário e senha válidos

Dado que o usuário está no EBAC Shop  
Quando clicar em login

E informar os dados de login válidos

Então deve ter acesso a página do usuário

Cenário 2: Mais de 3 tentativas com login e senha inválidos

Quando eu inserir um usuário e uma senha inválidos

E clicar em login

E tentar novamente por mais 2 vezes

Então o usuário deve ficar bloqueado por 15 minutos

## História de usuário 2: [US-0003] – Lista de desejos (favoritos)

Critérios de aceitação:

GET:

1. Deve listar todos os cupons cadastrado ou listar buscando por ID do cupom

POST:

1. Deve cadastrar os cupons com os campos obrigatórios abaixo:

1. - Código do cupom: Exemplo: “Ganhe10”   
   - Valor: “10.00”  
   - Tipo do desconto: “fixed\_product”  
   -Descrição: “Cupom de teste”
2. Nome do cupom não pode ser repetido;
3. Os outros campos são opcionais.

Cenário 1: Consultar todos os cupons cadastrados

Dado que o usuário é um administrador da EBAC Shop

E quer consultar os cupons todos os cupons cadastrados  
Quando enviar uma requisição do tipo GET

Então todos os cupons devem ser apresentados no response body da requisição

Cenário 2: Cadastrar um cupom com o nome já existente

Dado que o usuário é um administrador da EBAC Shop

E quer cadastrar um novo cupom de desconto com um nome já existente  
Quando preencher todos os dados e enviar a requisição POST

Então o response deve retornar a mensagem que já existe um cupom com esse nome

## 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: Incluir mais de 5 produtos no carrinho

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar a adição da quantidade máxima de 5 itens no carrinho |
| **Pré Condições:** | Estar no site do EBAC Shop |
| **Massa de Dados:** | Livingston All-Purpose Tight |
| **Prioridade** | Alta |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Clicar no item | Página do item deve ser exibida |
| 1. Selecionar o tamanho e a cor do produto | Tamanho e cor deve ser selecionados |
| 1. Alterar a quantidade para 6 | Usuário não deve conseguir incluir mais de 5 itens no carrinho |

CT02: Excluir um item do carrinho

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar a exclusão de itens no carrinho |
| **Pré Condições:** | Estar no site do EBAC Shop Possuir mais de 1 produto no carrinho  Possuir mais de 1 item do mesmo produto no carrinho |
| **Massa de Dados:** | Livingston All-Purpose Tight  Thorpe Track Pant |
| **Prioridade** | Alta |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Acessar a página do carrinho | Página de carrinho (checkout) deve ser exibida |
| 1. Clicar no ícone de exclusão (lixeira) | Item deve ser removido da lista |
| 1. Alterar a quantidade através do ícone “-“até o valor 0 | Ao reduzir até 0 o item deve ser removido da lista |

CT03: Alterar quantidade de itens do carrinho

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar a alteração de quantidade de itens no carrinho |
| **Pré Condições:** | Estar no site do EBAC Shop  Possuir mais de 1 item do mesmo produto no carrinho |
| **Massa de Dados:** | Thorpe Track Pant |
| **Prioridade** | Alta |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Acessar a página do carrinho | Página de carrinho (checkout) deve ser exibida |
| 1. Alterar a quantidade através do ícone de adição de itens (+) | A quantidade deve ser alterada e o valor total deve ser atualizado |
| 1. Alterar a quantidade através do ícone redução de itens (-) | A quantidade deve ser alterada e o valor total deve ser atualizado |

## História de usuário 2:

CT01: Login com Sucesso

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar o acesso a página do usuário com login e senha válidos |
| **Pré Condições:** | Estar no site do EBAC Shop |
| **Massa de Dados:** | Username: [test@mailinator.com](mailto:test@mailinator.com) Senha: teste@123 |
| **Prioridade** | Alta |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Acessar a página de login | Página de login deve ser exibida |
| 1. Informar o username e a senha em seus respectivos campos | Dados devem ser preenchidos corretamente |
| 1. Clicar no botão “Login” | A página do usuário deve ser acessada com sucesso |

CT02: Login com usuário válido e senha inválida

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar o acesso a página do usuário com login válido e senha inválida |
| **Pré Condições:** | Estar no site do EBAC Shop |
| **Massa de Dados:** | Username: [test@mailinator.com](mailto:test@mailinator.com) Senha: qualquersenhaquenaofuncione |
| **Prioridade** | Alta |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Acessar a página de login | Página de login deve ser exibida |
| 1. Informar o username e a senha em seus respectivos campos | Dados devem ser preenchidos corretamente |
| 1. Clicar no botão “Login” | Mensagem de alerta deve ser apresentada informando que a senha informada está incorreta |

CT03: Login com usuário e senha inválidos

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar o acesso a página do usuário com login e senha inválidos |
| **Pré Condições:** | Estar no site do EBAC Shop |
| **Massa de Dados:** | Username: [test&mailinator.com](mailto:test@mailinator.com) Senha: qualquersenhainvalida |
| **Prioridade** | Alta |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Acessar a página de login | Página de login deve ser exibida |
| 1. Informar o username e a senha em seus respectivos campos | Dados devem ser preenchidos corretamente |
| 1. Clicar no botão “Login” | Mensagem de alerta deve ser apresentada informando que o usuário não está cadastrado no site |

## História de usuário 3: API de Cupom

CT01: Consultar todos os cupons cadastrados

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar o retorno da requisição GET com todos os cupons cadastrados |
| **Pré Condições:** | Ter acesso a API (<http://lojaebac.ebaconline.art.br/rest-api/docs/#/coupons/get_wc_v3_coupons>  ) como administrador com os seguintes dados: Usuário: admin\_ebac Senha: @admin!&b@c!2022 |
| **Prioridade** | Média |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Acessar a API | API deve ser acessada |
| 1. Realizar o envio da requisição GET | Response body deve retornar todos os cupons cadastrados |

CT02: Consultar um cupom com um ID válido

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar o retorno da requisição GET com a informação do cupom solicitado via ID |
| **Pré Condições:** | Ter acesso a API (http://lojaebac.ebaconline.art.br/rest-api/docs/#/coupons/get\_wc\_v3\_coupons\_\_id\_ ) como administrador com os seguintes dados: Usuário: admin\_ebac Senha: @admin!&b@c!2022 |
| **Massa de Dados:** | 11775 |
| **Prioridade** | Média |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Acessar a API | API deve ser acessada |
| 1. Realizar o envio da requisição GET informando o ID | Response body deve retornar os dados do cupom do ID informado |

CT03: Consultar um cupom com um ID inválido

|  |  |
| --- | --- |
| **Descrição:** | Teste para validar o retorno da requisição GET com a informação do cupom solicitado via ID |
| **Pré Condições:** | Ter acesso a API (http://lojaebac.ebaconline.art.br/rest-api/docs/#/coupons/get\_wc\_v3\_coupons\_\_id\_ ) como administrador com os seguintes dados: Usuário: admin\_ebac Senha: @admin!&b@c!2022 |
| **Massa de Dados:** | 1209 |
| **Prioridade** | Média |

|  |  |
| --- | --- |
| **Ação** | **Resultado Esperado** |
| 1. Acessar a API | API deve ser acessada |
| 1. Realizar o envio da requisição GET informando o ID | Response body deve retornar mensagem informando “ID inválido’ |

## 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: https://github.com/vivianemonteiro/TCC-EBAC

## 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 *Jenkins file* 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

Com esse trabalho foi possível recapitular tudo que foi aprendido durante o curso, tanto o teórico quanto o prático, esse projeto veio a agregar conhecimento que deverá ser utilizado no dia a dia de trabalho. O portifólio foi atualizado com a entrega desse projeto, sendo um dos repositórios mais completos em relação a variedade de testes.

# REFERÊNCIAS BIBLIOGRÁFICAS

EBAC, Qualidade de Software, Materiais de Estudos.