

**ENGENHEIRO DE QUALIDADE DE SOFTWARE**

Alex Barbeiro Gonçalves Rocha

Análise de Qualidade

Americana-SP

2024

# RESUMO

O projeto de conclusão de curso em Engenharia de Qualidade de Software da EBAC – Escola Britânica de Artes Criativas e Tecnologia foca na otimização dos processos de desenvolvimento de *software,* com ênfase na qualidade. Utilizando abordagens de engenharia de *software,* utilizando técnicas de teste*,* buscando aprimorar a eficiência nos teste de *software.* Durante o projeto adotamos metodologias ágeis, como o Scrum, para assegurar uma comunicação eficaz entre as equipes de desenvolvimento. Além disso, concentra-se na implementação de estratégias de automação de testes e na utilização de métricas de qualidade para avaliar o desempenho do software.

O objetivo é demonstrar como a aplicação na prática de engenharia de qualidade de software pode resultar em produtos mais robustos, reduzindo custos e tempo de desenvolvimento. Como isso temos o crescente do projeto e mostramos a importância da qualidade de software no mercado atual, onde a satisfação do cliente e a competitividade são cruciais para o sucesso das empresas.

# SUMÁRIO

[1. RESUMO 2](#__RefHeading___Toc398_4040013964)

[2. SUMÁRIO 3](#__RefHeading___Toc400_4040013964)

[3. INTRODUÇÃO 4](#__RefHeading___Toc402_4040013964)

[4. O PROJETO 5](#__RefHeading___Toc404_4040013964)

[4.1 Estratégia de teste 5](#__RefHeading___Toc406_4040013964)

[4.2 Critérios de aceitação 6](#__RefHeading___Toc408_4040013964)

[4.3 Casos de testes 10](#__RefHeading___Toc410_4040013964)

[4.4 Repositório no Github 14](#__RefHeading___Toc412_4040013964)

[4.5 Testes automatizados 14](#__RefHeading___Toc414_4040013964)

[4.6 Integração contínua 16](#__RefHeading___Toc416_4040013964)

[4.7 Testes de performance 16](#__RefHeading___Toc418_4040013964)

[5. CONCLUSÃO 17](#__RefHeading___Toc420_4040013964)

[6. REFERÊNCIAS BIBLIOGRÁFICAS 18](#__RefHeading___Toc422_4040013964)

# INTRODUÇÃO

A importância do *software* de alta qualidade é cada vez mais vital em nosso mundo. Com o avanço da tecnologia a sua aderência em todos os aspectos de nossas vidas, a demanda por sistemas confiáveis e de alto desempenho se torna ainda mais evidente.

Considere usar um aplicativo frequentemente trava é lento, apresenta falhas ou não corresponde da forma esperada. Isso pode causar uma perda financeira e prejudicar empresas, causar um processo e entre outros problemas. É aqui que a Engenharia de Qualidade de Software desempenha um papel crucial.

*Quality Assurance* (QA) é o profissional dedicado a assegurar que o software seja desenvolvido, testado e entregue com os mais altos níveis de qualidade. Isso inclui estabelecer requisitos claros, adotar metodologias eficientes de desenvolvimento e aplicar rigorosas práticas de teste. Além disso, a automatização de testes e a análise de métricas são fundamentais na busca pela excelência.

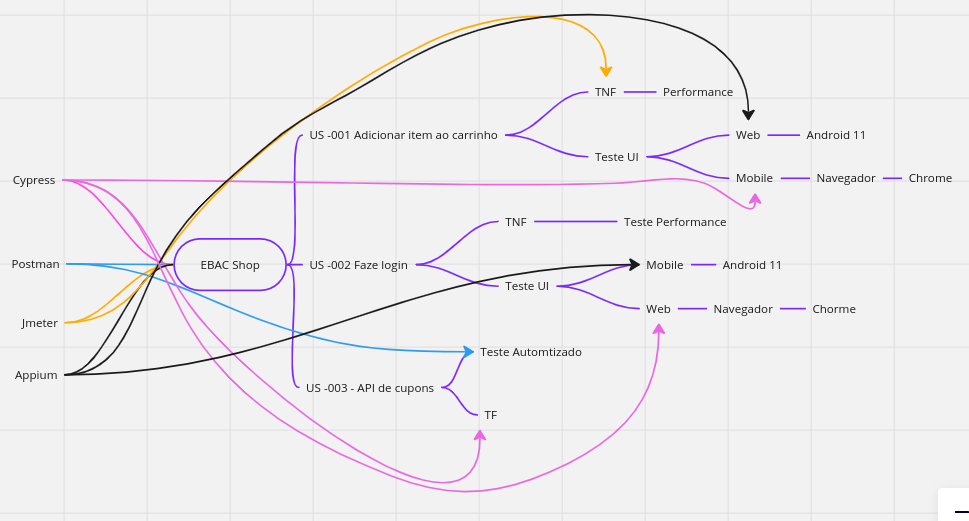
Neste cenário, a Engenharia de Qualidade de Software não é apenas uma vantagem competitividade, mas também uma necessidade para as empresas que querem se destacar no atual panorama tecnológico, não apenas para promover a confiabilidade, segurança e eficácia junto ao cliente, mas também para agregar valor ao negocio do cliente.

# O PROJETO

Para este trabalho de conclusão de curso **Profissão: Engenheiro de Qualidade de software**, você deve utilizar o conhecimento adquirido ao longo do curso para elaborar uma estratégia de testes adequada para validar o e-commerce EBAC Shop (<http://lojaebac.ebaconline.art.br/>). Você deve considerar as histórias de usuário já refinadas como se você estivesse participando de um time ágil. As funcionalidades devem seguir todo o fluxo de trabalho de um *Quality Engineer* (QE), desde o planejamento até a entrega. Siga as etapas dos sub-tópicos para se orientar no trabalho.

## 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:



## Critérios de aceitação

* Considere as histórias de usuário:
  + [US-001] – Adicionar item ao carrinho
  + [US-002] – Login na plataforma
  + [US-003] – API de cupons
* Para cada uma delas crie pelo menos 4 critérios de aceitação usando a linguagem Gherkin;
* Crie histórias de usuário para as funcionalidades:
  + Catálogo de Produtos
  + Painel Minha Conta
  + Meus Pedidos
  + Endereços
  + Detalhes da Conta
* Referência: Módulo 8

4.2.1 Critério de aceitação – [US-0001] – Adicionar item ao carrinho

Funcionalidade: Incluir produtos no carrinho

Criar para a EBAC-SHOP, um serviço para adicionar produtos ao carrinho

Para realizar compra na EBAC-SHOP.

Contexto:

Dado: Usuário acesse o site EBAC-SHOP, página de produto.

Critério de aceitação: Não deve adicionar produto sem estoque.

Cenário: Adicionar produto sem estoque

Quando : Selecionar o produto “Ingrid Running Jacket”

E: Selecionar, tamanho, cor, informe a quantidade desejada igual 1

E: Clicar no botão “Comprar”

Então: Deve exibir uma mensagem de “Produto sem estoque”.ar produto com estoque, com sucesso.

Cenário: Adicionar produto com estoque

Quando: Selecionar o produto “Ingrid Running Jacket”

E: Selecionar, tamanho M, cor Red, informar quantidade 1

E: Clicar no botão COMPRAR

Então: Deve exibir uma mensagem “Produto adicionar ao carrinho!”

Cenário: Não permitir adicionar quantidade de produto maior que 56

Quando: Selecionar o produto “Ingrid Running Jacket”

E: Selecionar , tamanho M, cor Red, informar quantidade 57

E: Clicar no botão COMPRAR

Então: Deve exibir uma mensagem “A quantidade de produto, deve ser menor ou igual a 56”

4.2.2 Critério de aceitação – [US-0002] – Login na plataforma

Funcionalidade: Autenticação de acesso na plataforma EBAC

Como cliente da EBAC-SHOP

Quero fazer o login na plataforma

Para visualizar pedidos

Contexto:

Dado: Que o usuário acesse a plataforma EBAC-SHOP

Cenário: Autenticar acesso com usuário existente

Quando: Inserir o usuário e senha validos

Então: Deve ser direcionado para tela de checkout

Cenário: Autenticar com usuário inexistente

Quando: Inserir um usuário inexistente

E: senha inexistente

Então: Deve exibir uma mensagem de alerta “Usuário ou Senha Incorreta!”

Cenário : Bloquear acesso após 5 tentativa

Quando: Inserir usuário e senha incorretos

E: Clicar no botão Login

Então: Uma mensagem de alerta é apresentado “Usuário bloqueado, pelo número de tentativas!”

Cenário: Autenticar múltiplos usuários inexistente

Quando: Inserir o usuário <usuario>

E: senha <senha>

Então: deve exibir a mensagem <usuario> ‘ou senha inválidos’

Exemplo:

Usuários | Senha |Mensagem

“jose.ebac@ebac.com” | “teste@123 | “Olá Jose“

“maria.ebac@ebac.com”| “teste@123 | “Olá Maria“

“joao.ebac@ebac.com” | “teste@123 | “Olá João “

4.2.3 Critério de aceitação – [US-0002] – API de Cupons

Funcionalidade: Inclusão de cupom

Criar um serviço de cupom

Para pode adicionar cupons, a EBAC-SHOP

Contexto:

Dado: Que acesse a API de cupons

Critério de aceitação: Deve mostrar cupons com 10% de desconto

Cenário: Cupom EBAC10 loja EBAC-SHOP

Quando: Efetuar uma pesquisa, na loja EBAC-Shop

E: pesquisar por cupons 10%

Então: Deve exibir uma mensagem de “Cupom EBAC10 primeira compra”

Critério de aceitação: Não cadastra cupom com código inexistente

Cenário: Cupom EBAC10 EBAC-Shop

Quando: Informar os dados para cadastrar o cupom inexistente

Então: Deve exibir a mensagem de “Cupom Inexistente”

Critério de aceitação:Não deve cadastrar cupons com código existente

Cenário: Cupom EBAC10 EBAC-Shop

E: Informar um código já cadastrado

Então: Deve exibir uma mensagem de “Cupom já existe”--

Critério de aceitação: Deve cadastrar cupons com sucesso

Cenário: Cupom EBAC10 EBAC-Shop

E: Informe os dados para cadastrar o cupom

Então: A API deve retornar o status 201 cadastrando um novo cupom

Critério de aceitação: Deve expor somente cupons válidos

Cenário: Cupons válidos loja EBAC-SHOP

Quando: eu selecionar a loja EBAC-SHOP

E: Clicar em visualizar cupons

Então: Deve exibir uma mensagem “Cupons válidos para hoje da loja EBAC-SHOP”

**4.2.4 Histórias de usuário - Painel Minha Conta**

**Como** cliente da EBAC-SHOP

**Quero** acessar Painel "**Minha Conta**" usando e-mail e senha

**Para** acessar recursos personalizados.

**4.2.5 Histórias de usuário – Catálogo de Produtos**

**Como** cliente da EBAC-Shop

**Quero** visualizar a lista de produtos disponíveis no catálogo

**Para** ver os itens que posso comprar.

**4.2.6 Histórias de usuário - Detalhes da Conta**

**Como** cliente da EBAC-SHOP

**Quero** visualizar informações da conta na página "Detalhes da Conta"

**Para** atualizar nome, sobrenome, e-mail e senha de acesso.

**4.2.7 Histórias de usuário - Meus Pedidos**

**Como** cliente da EBAC-SHOP

**Quero** visualizar uma lista dos meus pedidos "Meus Pedidos”

**Para** obter o status de cada pedido.

**4.2.8 Histórias de usuário - Endereços**

**Como** cliente da EBAC-SHOP

**Quero** editar as informações do meu endereço em "Meu Endereço"

**Para** atualizar nome, rua, número, CEP, cidade e estado.

## Casos de testes

* Crie pelo menos 4 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...”
* Identifique quais os casos de teste serão automatizados, sendo ao menos 1 caminho feliz e 1 caminho alternativo.
* Referência: Módulos 4 e 5

**4.3.1 - Caso de teste: Adicionar item ao carrinho**

**RN01**

Não é permitido inserir mais de 10 itens de um mesmo produto no carrinho

|  |  |
| --- | --- |
| 0 até 10 | >10 |
| 0,1,2,3,8,9,10 | 11,12,99,100 |
| Dados válidos | Dados Inválidos |

**Caso de teste**

|  |  |  |
| --- | --- | --- |
|  | Entrada | Saída |
| Teste 1 | inserir 10 itens de um mesmo produto ao carrinho | Válido |
| Teste 2 | inserir mais de 10 itens de um mesmo produto ao carrinho | Inválido |
| Teste 3 | inserir menos de 10 itens de um mesmo produto ao carrinho | Válido |

**RN02**

Os valores não podem ultrapassar a R$ 990,00

|  |  |
| --- | --- |
| 0 até 899,99 | > 990,00 |
| 0,10, 20 ,….., 90, 100,…., 899,99 | 990,00 1.000,…..,999.99,….. |
| Dados Válidos | Dados Inválido |

**Caso de teste**

|  |  |  |
| --- | --- | --- |
|  | Entrada | Saída |
| Teste 01 | Total do valor de R$ 899,99 | válido |
| Teste 02 | Total do valor de R$ 1.000,00 | inválido |
| Teste 03 | Total do valor de R$ 60,00 | válido |

**NR03**

Valores entre R$ 200 e R$ 600 , ganham cupom de 10%

|  |  |  |
| --- | --- | --- |
| < 199,99 | 200,00 até 600,00 | > 600 |
| 0,10,00 ,20,...,100,…, 199, | 200,00,300,000,..., 600,000 | 601,800,...900,…, |
| Dados Inválido | Dados Válidos | Dados Inválidos |

Caso de teste

|  |  |  |
| --- | --- | --- |
|  | Entrada | Saída |
| Teste 1 | Total do valor de R$ 200,00 | Inválido |
| Teste 2 | Total do valor de R$ 600,50 | Válido |
| Teste 3 | Total do valor de R$ 150,00 | Válido |

RN04

Valores acima de R$ 600 ganham cupom de 15%

|  |  |
| --- | --- |
| 0 até 599,99 | > 99,00 |
| 0,10,20,….,100,…,500,…,599,98 | 990,00,1.000,000,... |
| Dados inválidos | Dados válidos |

Caso de teste

|  |  |  |
| --- | --- | --- |
|  | Entrada | Saída |
| Teste 1 | Total do valor R$ 1.500,00 | Válido |
| Teste 2 | Total do valor R$ 50,00 | Inválido |
| Teste 3 | Total do valor R$599,98 | Válido |

43.2 – Caso de teste: Cadastro na plataforma

RN01

Cadastro de alunos deve estar entre 16 e 80 anos.

|  |  |  |  |
| --- | --- | --- | --- |
| Condições | Regra 1 | Regra 2 | Regra 3 |
| Idade | < 16 | 16 a 18 | > 80 |
| Ação |  |  |  |
| Cadastrar |  | X |  |
| Não | X |  | X |

RN02

Caso erre a senha por 3 vezes, deve bloquear o acesso por 15 minutos.

Caso de teste

|  |  |  |
| --- | --- | --- |
| Condição | Regra 1 | Regra 2 |
| Trava acesso por tentativa | =<2 | >=3 |
| Ação |  |  |
| Bloqueio |  | X |
| Não bloqueado | X |  |

RN03

Exbir mensagem de erro caso o usuário erre login e senha.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Condição | Regra 1 | Regra 2 | Regra 3 | Regra 4 |
| Usuário válido? | Sim | Sim | Não | Não |
| Senha válida? | Sim | Não | Sim | Não |
| Ação |  |  |  |  |
| Acesso | Sim | Não | Não | Não |

Casos de testes:

1. Login com usuário válido e senha válida, deve permitir acesso

2. Login com usuário válido e senha inválida, não deve permitir acesso

3. Login com usuário inválido e senha válida, não deve permitir acesso

4. Login com usuário inválido e senha inválida, não deve permitir acesso

## Repositório no Github

* Crie um repositório no github com o nome TCC-EBAC-QE;

Deixe o repositório publico até a análise dos tutores;

* Neste repositório você deve subir este arquivo e todos os código fontes das automações que criar.
* Referência: Módulo 10
* Link do repositório: <https://github.com/alexgrocha/TCC-EBAC>

## Testes automatizados

* + 1. Automação de UI

Considerando as ferramentas de automação, para nosso projeto é importante ressaltar o cuidado ao selecionar uma ferramenta levando em consideração, velocidade de execução, manutenção de código e integração contínua. Após uma análise aprofundada, decidimos adotar o Cypress como nossa ferramenta de automação, abaixou vou descrever os motivos da utilização:

**API Simples e Manutenção fácil:**

A utilização da API do Cypress diminui a complexidade na utilização, deixa o teste mais simples, e direto, com isso evita muitos termos por exemplo no Selenium ou Playwright.

**Facilidade no uso, codificação:**

O Cyspress foi escolhido pela sua abordagem amigável e simples de utilizar, isso acelera nosso processo de desenvolvimento de testes e economiza tempo valioso.

**Execução Rápida:**

A estrutura nativa do Cypress proporciona uma vantagem substancial em relação à velocidade de execução quando comparada ao Selenium por exemplo, essa característica nos possibilitará conduzir testes de regressão e validação de forma rápida.

**Integração continua:**

O Cypress possui integração com várias ferramentas como Jenkins. Isso permite incorporar nossos testes de forma transparente em nosso pipeline de entrega contínua, garantindo que a qualidade seja mantida.

* + 1. Automação de API
* Crie uma pasta chamada API para os testes de API dos casos de teste que forem automatizados
* Você deve utilizar a ferramenta Supertest para criar seus testes de API
* Não esqueça de validar os contratos! 
  + 1. Automação Mobile
* Considere para os APPs apenas a funcionalidade de Catálogo de Produtos
* Você pode encontrar os APPs em:
  + *Android*: <https://github.com/EBAC-QE/testes-mobile-ebac-shop/tree/main/app/android>
  + *iOS*: <https://github.com/EBAC-QE/testes-mobile-ebac-shop/tree/ios-tests/app/ios>
* Crie uma pasta chamada Mobile para os testes em aplicativos dos casos de teste que forem automatizados
* Utilize ao menos um *Testing Pattern* (à sua escolha) na implementação dos testes.
* Você deve implementar testes para ao menos uma das plataformas Mobile (*Android* ou *iOS*)
* Observações:
  + Considere todas as boas práticas aprendidas até aqui
  + Não esqueça de implementar a geração de relatórios
* Referência: Módulos 11, 12, 14, 16, 17, 22, 23, 24, 29 e 30

## Integração contínua

* Execute os testes automatizados em integração contínua utilizando o Github Actions
* Referência: Módulo 26

## Testes de performance

* Usando o K6, implemente um teste de performance em ao menos 2 casos de testes
* Referência: Módulo 28
* 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

# CONCLUSÃO

Apresento a conclusão do projeto de TCC (trabalho de conclusão de curso). Esse projeto tem objetivo de otimizar os processo de desenvolvimento de software, com foco na excelência da qualidade, explorando abordagens de engenharia de software, gestão de projetos e métodos de teste.

Durante o processo, deparei-me com desafios ao configurar teste em dispositivos móveis, exigindo ajuste complexos.

Para auxiliar na conclusão realizei muitas pesquisas em diferentes documentações e sites.

Os principais resultados no trabalho de conclusão de curso, foram superar as dificuldades no desenvolvimento dos teste, pois foi necessário buscar conhecimento fora do curso, e o aprimoramento nos teste utilizando o Cypress, outros pontos importantes durante o projeto foi o desafio de realizar os teste mobile, que exigiu ainda mais pesquisa e atenção nas configurações, pois o curso fornece poucos recursos para configuração.

Contudo obtive lições valiosas durante todo o projeto de pesquisa e desenvolvimento, do TCC, aprimorei técnica de teste como BDD (Behaviour Driven Testing), casos de teste, conhecimentos em novas tecnologias.

Entendo que o poder da automação ira otimizar o tempo e maximizar os resultados, entregando software com qualidade, buscando atender a expectativa do cliente.

# REFERÊNCIAS BIBLIOGRÁFICAS

Sites:

Tema: Pesquisa inédita sobre o futuro da Qualidade de Software é lançada

<https://gazetadasemana.com.br/noticia/114075/pesquisa-inedita-sobre-o-futuro-da-qualidade-de-software-e-lancada> .acessado em 03 abril de 2024 às 22h10min.

Cypress docs: <https://docs.cypress.io/guides/overview/why-cypress> .acessando em 05 de abril de 2024 21:03 min.

Livros:

Engenharia de Software Uma Abordagem Profissional – Sétima Edição por Roger Pressman