

**ENGENHEIRO DE QUALIDADE DE SOFTWARE**

Hellen Taís de Souza Silva

Análise de Qualidade

Campo Grande

2025

# RESUMO

Garantir a qualidade de um produto certamente não é uma tarefa simples. Exige responsabilidade, usar técnicas adequadas e, principalmente, focar no usuário final. O produto além de funcionar corretamente, deve ser acessível, intuitivo e capaz de suportar múltiplas interações.

Para atingir esse objetivo no projeto da Ebac Shop, foram traçadas algumas estratégias. Inicialmente elaborar um mapa mental como guia com todos os objetivos, responsabilidades, tipos de testes, ferramentas e plataformas. Em seguida, analisaram-se as histórias de usuário e regras de negócio para alcançar o resultado esperado pelo cliente. Também foi elaborado um passo a passo de testes com casos positivos, o famoso “caminho feliz”, e casos alternativos, visando prever possíveis falhas. Foram definidos quais testes seriam automatizados, avaliada a performance das aplicações e, por fim, toda a execução foi documentada.

Todo esse processo demanda bastante tempo e energia, mas é imprescindível para garantir que o objetivo seja alcançado.

# SUMÁRIO

[1. RESUMO 2](#_Toc99483086)

[2. SUMÁRIO 3](#_Toc99483087)

[3. INTRODUÇÃO 4](#_Toc99483088)

[4. O PROJETO 5](#_Toc99483089)

[4.1 Estratégia de teste 6](#_Toc99483090)

[4.2 Critérios de aceitação 6](#_Toc99483091)

[4.3 Casos de testes 6](#_Toc99483092)

[4.4 Repositório no Github 7](#_Toc99483093)

[4.5 Testes automatizados 7](#_Toc99483094)

[4.6 Integração contínua 8](#_Toc99483095)

[4.7 Testes de performance 8](#_Toc99483096)

[5. CONCLUSÃO 9](#_Toc99483097)

[6. REFERÊNCIAS BIBLIOGRÁFICAS 9](#_Toc99483098)

# INTRODUÇÃO

Para que uma aplicação funcione corretamente, é necessário que seja testada de forma contínua. O e-commerce Ebac Shop, nas versões web e mobile, foi utilizado como base para a realização deste projeto. Considerando cuidadosamente as histórias de usuário e as regras de negócio, foram definidas estratégias de teste. Alguns fluxos críticos foram automatizados para evitar retrabalho e aumentar a eficiência, enquanto todo o processo foi documentado e integrado a um fluxo de integração contínua.

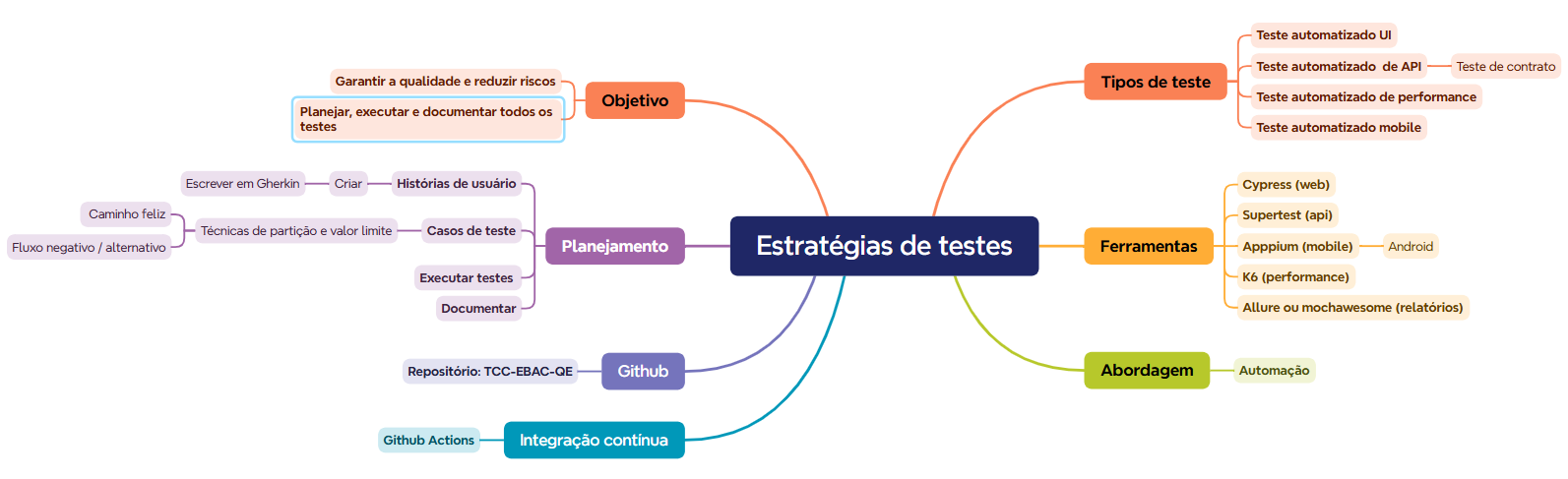
Além de assegurar que o produto testado entregue qualidade, esse projeto também tem como objetivo nos colocar no papel de um QA num time ágil, mostrando como desempenhar bem essa função faz toda a diferença durante todo o processo.

# O PROJETO

Para este trabalho de conclusão de curso **Profissão: Engenheiro de Qualidade de software** está sendo utilizado todo 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/>). A partir das histórias de usuário já refinadas, o projeto é desenvolvido como se estivesse participando de um time ágil. As funcionalidades seguem todo o fluxo de trabalho de um *Quality Engineer* (QE), desde o planejamento até a entrega. A seguir constam todas as etapas com seus respectivos links e imagens.

## Estratégia de teste

* Mapa mental com planejamento, objetivos, técnicas, ferramentas e abordagem.



## Critérios de aceitação

* Três histórias de usuário foram previamente definidas e outras cinco foram criadas. Segue o link: <https://github.com/taistj3/tcc-ebac-qe/tree/main/projetoTCC/documentos/historiasUsuario>
* Para cada história de usuário foram criados pelo menos quatro critérios de aceitação usando Gherkin. Segue o link:

https://github.com/taistj3/tcc-ebac-qe/tree/main/projetoTCC/documentos/cenariosGherkin

## Casos de testes

* Foram criados pelo menos quatro casos de teste para cada história de usuário. Consideramos o “caminho feliz” e o fluxo alternativo como base. Segue o link:

<https://github.com/taistj3/tcc-ebac-qe/tree/main/projetoTCC/documentos/casosDeTeste>

## Repositório no Github

* <https://github.com/taistj3/tcc-ebac-qe.git>

## Testes automatizados

* + 1. Automação de UI
* https://github.com/taistj3/tcc-ebac-qe/tree/main/projetoTCC/automacoes/testeUI/cypress
  + 1. Automação de API
* https://github.com/taistj3/tcc-ebac-qe/tree/main/projetoTCC/automacoes/testeAPI
  + 1. Automação Mobile
* https://github.com/taistj3/tcc-ebac-qe/tree/main/projetoTCC/automacoes/testesMobile

## Integração contínua

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

## Testes de performance

* Usando o K6 foram implementados testes de performance a dois casos de teste: Teste de API Cupons – buscar e cadastrar cupons: https://github.com/taistj3/tcc-ebac-qe/tree/main/projetoTCC/automacoes/testePerformance
* Configurações:

-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

Chegar ao fim desse projeto foi uma experiência muito valiosa. Pude sentir na prática como é estar no papel de um QA, com todas as responsabilidades e desafios envolvidos. Aprendi a ter um olhar mais crítico, pensar além dos detalhes técnicos e considerar sempre o comportamento do usuário para garantir a qualidade.

Cada etapa tem sua importância e não pode ser deixada de lado. Quando seguimos as estratégias certas e aplicamos as técnicas adequadas, conseguimos entregar mais valor para o time e para o cliente. Certamente um grande aprendizado para o início da minha vida profissional.

# REFERÊNCIAS BIBLIOGRÁFICAS

ALLURE FRAMEWORK. Allure Report Documentation. Disponível em: https://allurereport.org/. Acesso em: 10 out. 2025.

BARBOSA, Ernesto; ARAÚJO, Fábio. Engenheiro de Qualidade de Software. EBAC – Escola Britânica de Artes Criativas e Tecnologia. Disponível em: <https://lms.ebaconline.com.br/>. Acesso em: 10 out. 2025.

CYPRESS. Cypress Documentation. Disponível em: https://docs.cypress.io/. Acesso em: 10 out. 2025.

GRAFANA LABS. k6 Documentation. Disponível em: https://grafana.com/docs/k6/latest/. Acesso em: 10 out. 2025.

JEST. Jest Documentation. Disponível em: https://jestjs.io/docs/getting-started. Acesso em: 10 out. 2025.

WEBDRIVERIO. WebdriverIO Documentation. Disponível em: https://webdriver.io/docs/gettingstarted. Acesso em: 10 out. 2025.