



QUALIDADE DE SOFTWARE

Bruna Daniela Kalschne

Análise de Qualidade

Três de Maio - RS

2024

SUMÁRIO

RESUMO	3
INTRODUÇÃO	4
O PROJETO	5
1.1 Estratégia de teste	5
1.2 Critérios de aceitação	6
1.2.1 História de usuário 1: [US-0001] – Adicionar item ao carrinho	7
1.2.2 História de usuário 2: [US-0002] – Login na plataforma	7
1.2.3 História de usuário 2: [US-0003] – API de cupons	8
1.3 Casos de testes	8
1.3.1 História de usuário 1:	9
1.3.2 História de usuário 2:	10
1.3.3 História de usuário 3: API de Cupom	11
1.4 Repositório no Github	13
1.5 Repositório no Github	13
1.5.1 Automação de UI	13
1.5.2 Automação de API	13
1.6 Integração contínua	14
1.7 Testes de performance	14
CONCLUSÃO	16
REFERÊNCIAS BIBLIOGRÁFICAS	17

RESUMO

Este projeto tem como objetivo aplicar todos os conhecimentos adquiridos ao longo do curso de Qualidade de Software da EBAC. O trabalho final tem como proposta, simular todo o fluxo de trabalho de um Analista de Qualidade dentro de um time ágil, envolvendo desde a etapa de planejamento, até a entrega final.

INTRODUÇÃO

A missão do analista de qualidade no cenário de trabalho atual, é garantir o máximo de qualidade dos produtos e serviços disponibilizados pelas empresas de software, fazendo uma análise crítica, sempre buscando por vulnerabilidades e pontos a melhorar.

A proposta principal deste trabalho final, é simular todo o fluxo de trabalho de um Analista de Qualidade dentro de um time ágil, envolvendo desde a etapa de planejamento, até a entrega final. Para iniciar a etapa de planejamento, foram tomadas como base as histórias de usuários e suas regras de negócios, e a partir delas foram desenvolvidas as estratégias de testes, critérios de aceitação e casos de testes.

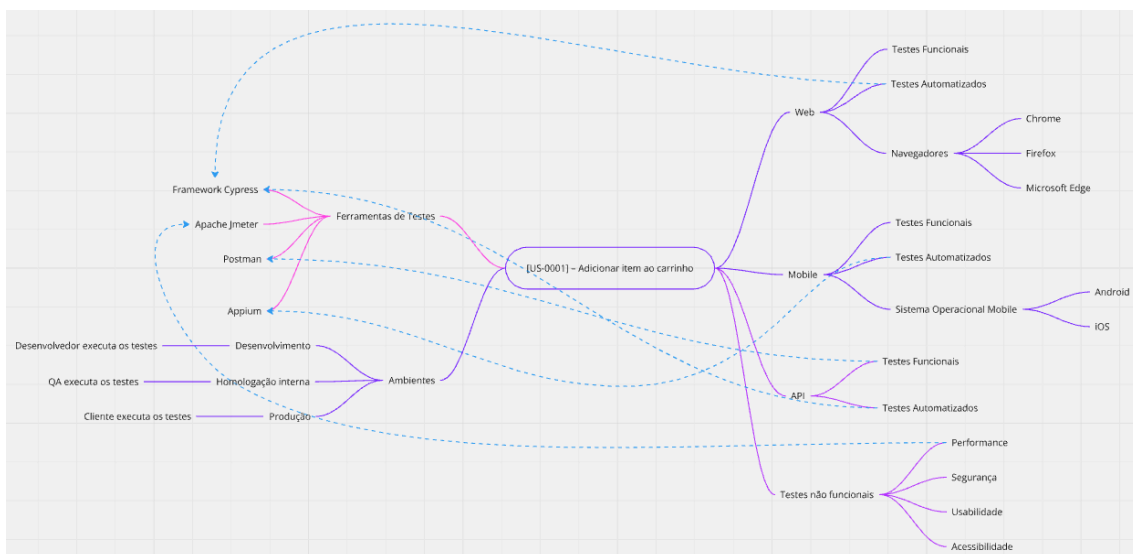
Para desenvolver a parte prática, foi realizada a implementação de testes automatizados da interface Web, e API, utilizando o framework Cypress. Buscando aplicar as boas práticas do DevOps, também foi implementada a integração contínua destes mesmos testes, em uma pipeline no Jenkins. E buscando assegurar a performance do sistema utilizado como base para desenvolver este projeto, também foram implementados testes de performance, utilizando a ferramenta Apache Jmeter.

O PROJETO

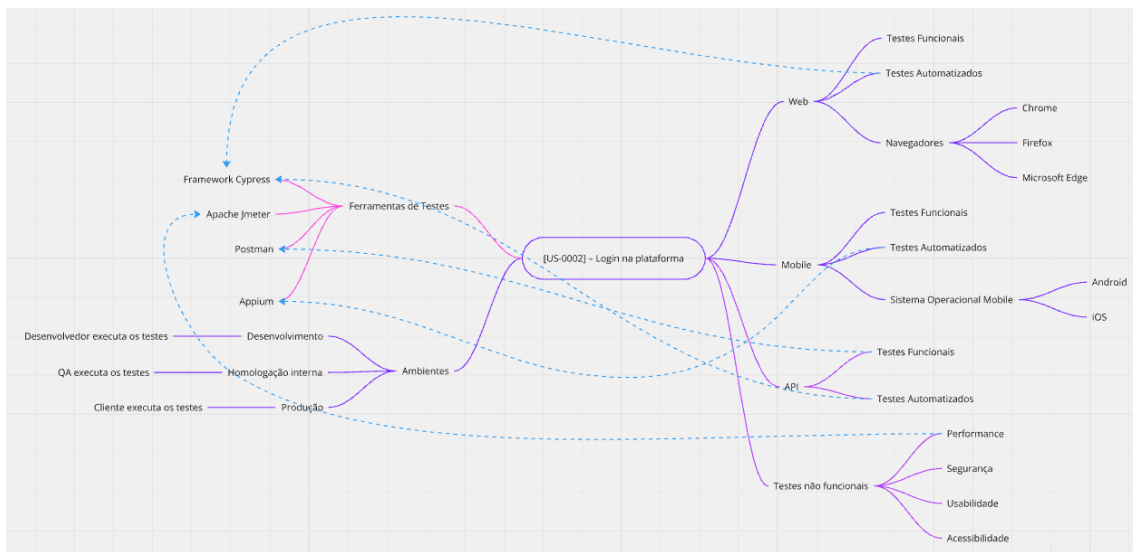
Para o Trabalho de Conclusão de Curso Qualidade de Software, foram consideradas as histórias de usuário já refinadas, e simulado todo o fluxo de trabalho de um Analista de Qualidade dentro de um time ágil, envolvendo desde a etapa de planejamento, até a entrega final. Todo o trabalho se baseia na validação do e-commerce EBAC Shop, que também foi utilizado durante as aulas do curso.

1.1 Estratégia de teste

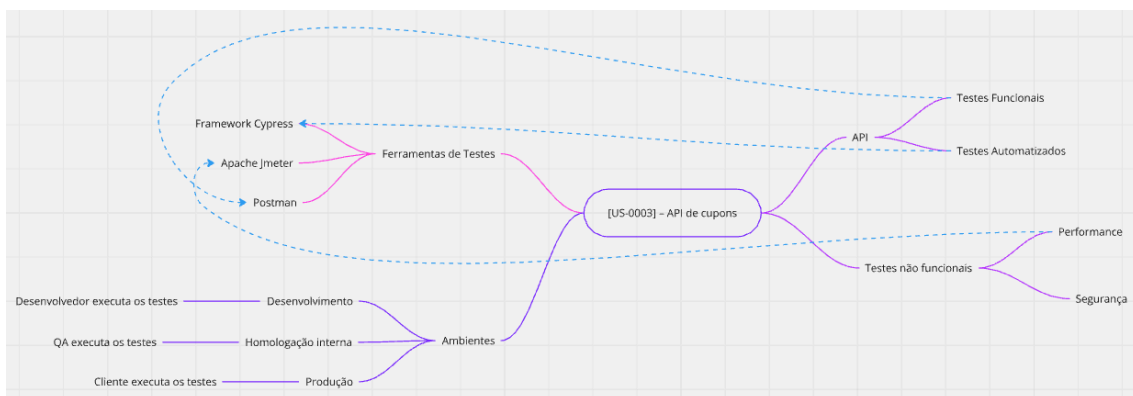
- Faça uma estratégia de testes em um mapa mental, seguindo algumas diretrizes como objetivos, papéis e responsabilidades, fases de testes, padrões, tipos de testes, técnicas de testes, ambientes, ferramentas, abordagem (manual ou automatizado), framework ou ferramenta usados, plataformas (web, api, mobile), etc.;
- US001 - Adicionar item ao carrinho



- US002 - Login na Plataforma



- US003 - API de Cupons



1.2 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

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

Critérios de aceitação:

Cenário 1: Limitar quantidade de itens para compra

```
Dado que eu adicione um valor maior que 10 na quantidade do produto
Quando eu clicar no botão Comprar
Então deve exibir uma mensagem de alerta: "Não é permitido inserir mais de 10 itens de um mesmo produto para compra"
```

Cenário 2: Validar valor da compra para aplicação do cupom de desconto

```
Dado que eu adicione produtos ao carrinho de compras
Quando o valor total da compra for <valor>
Então deverá aplicar automaticamente um cupom de <cupom>
```

Exemplos:

valor	cupom
entre R\$ 200,00 e R\$ 600,00	10%
maior que R\$ 600,00	15%

Cenário 3: Validar compra sem a aplicação de cupom de desconto

```
Dado que eu adicione produtos ao carrinho de compras
Quando o valor total da compra for menor que R$ 200,00
Então não deverá aplicar nenhum cupom de desconto
```

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

Critérios de aceitação:

Cenário 1: Validação da situação do usuário ao realizar o login

```
Dado que meu usuário esteja <status> no sistema
Quando eu informar os dados para fazer o login na plataforma
Então <resultado> fazer o login
```

Exemplos:

status	resultado	
ativo	será possível	
inativo	não será possível	

Cenário 2: Validação das informações ao realizar o login

```
Dado que meu usuário esteja ativo no sistema
Quando eu preencher <situacao> as informações de login
Então será apresentada a mensagem <mensagem>
```

Exemplos:

situacao	mensagem	
corretamente	"Login realizado com sucesso"	
incorretamente	"Usuário ou senha inválidos, tente novamente"	

Cenário 3: Validar tentativas sequenciais de login inválido

```
Dado que eu informe uma senha inválida no login
Quando eu tentar fazer o login pela terceira vez em sequência
Então o processo de login será bloqueado pelo tempo de 15 minutos
```

1.2.3 História de usuário 2: [US-0003] – API de cupons

Critérios de aceitação:

Cenário 1: Validação do nome do cupom de desconto

```
Dado que eu informe os dados para cadastrar um novo cupom de desconto
Quando existir um cupom de desconto com o mesmo nome já cadastrado
Então deve exibir uma mensagem de alerta, informando que não é possível repetir o nome do cupom
```

Cenário 2: Validação dos campos obrigatórios

```
Dado que eu informe os dados para cadastrar um novo cupom de desconto
Quando eu deixar de preencher algum dos campos obrigatórios
Então deve exibir uma mensagem de alerta, informando que é necessário preencher todos os campos obrigatórios
```

1.3 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

1.3.1 História de usuário 1:

Regra de Negócio:

RN01 - Não é permitido inserir mais de 10 itens de um mesmo produto ao carrinho

< 10	> 10
Dados válidos	Dados inválidos

Casos de Testes:

- **CT01:** Inserir 8 itens do mesmo produto ao carrinho, deve permitir.
- **CT02:** Inserir 12 itens do mesmo produto ao carrinho, não deve permitir.

Regra de Negócio:

RN02 - Os valores não podem ultrapassar R\$ 990,00

	Valor limite 990,00	
Dados válidos	Dados válidos	Dados inválidos
989,99	990,00	990,01

Casos de Testes:

- **CT01:** Realizar uma compra no valor de 989,99, deve permitir a compra.
- **CT02:** Realizar uma compra no valor de 990,00, deve permitir a compra.
- **CT03:** Realizar uma compra no valor de 990,01, não deve permitir a compra.

Regra de Negócio:

RN04 - Valores acima de R\$ 600 ganham cupom de 15%

Condições	Regra 1	Regra 2
Valor da compra	<= 600,00	> 600,00
Ações		
Aplicar cupom de 15%		x
Não aplicar cupom de 15%	x	

Casos de Testes:

- **CT01:** Fazer uma compra com valor de 599,99, não deve aplicar o cupom de 15%
- **CT02:** Fazer uma compra com valor de 600,00 não deve aplicar o cupom de 15%
- **CT03:** Fazer uma compra com valor de 600,01 deve aplicar o cupom de 15%

1.3.2 História de usuário 2:**Regra de Negócio:****RN01 - Somente usuários ativos podem fazer login**

Condições	Regra 1	Regra 2
Usuário ativo?	Sim	Não
Usuário inativo?	Não	Sim
Ações		
Realizar login?	Sim	Não

Casos de Testes:

- **CT01:** Usuários que estejam ativos no sistema, podem realizar o login.
- **CT02:** Usuários que estejam inativos no sistema, não podem realizar o login.

Regra de Negócio:**RN02 - Deve exibir uma mensagem de erro caso o usuário erre o login e senha**

Condições	Regra 1	Regra 2	Regra 3	Regra 4
Login inválido?	Sim	Sim	Não	Não
Senha inválida?	Sim	Não	Sim	Não
Ações				
Exibir mensagem de erro?	Sim	Sim	Sim	Não

Casos de Testes:

- **CT01:** Login com usuário e senha inválidos, deve exibir mensagem de erro.
- **CT02:** Login com usuário inválido e senha válida, deve exibir mensagem de erro.
- **CT03:** Login com usuário válido e senha inválida, deve exibir mensagem de erro.
- **CT04:** Login com usuário e senha válidos, não deve exibir mensagem de erro.

Regra de Negócio:

RN04 - Se o usuário errar por 3 vezes a senha, deve travar por 15 minutos o login

< 3	= 3
Não bloqueia o acesso	Bloqueia o acesso

Casos de Testes:

- **CT01:** Ao errar a senha por 2 vezes, não deve bloquear o acesso do usuário no login.
- **CT02:** Ao errar a senha por 3 vezes, deve bloquear o acesso do usuário no login durante 15 minutos.

1.3.3 História de usuário 3: API de Cupom

Regra de Negócio:

RN01 - Deve listar todos os cupons cadastrado ou listar buscando por ID do cupom

Casos de Testes:

- **CT01:** Ao fazer a busca sem informar um ID, deve retornar todos os cupons cadastrados.
- **CT02:** Ao fazer a busca informando um ID, deve retornar apenas o cupom correspondente ao ID informado.

Regra de Negócio:

RN02 - Deve cadastrar os cupons com os campos obrigatórios abaixo:

- **Código do cupom**
- **Valor**
- **Tipo do desconto**
- **Descrição**

Condições	R1	R2	R3	R4	R5	R6
Preencheu Código do cupom?	Não	Não	Sim	Sim	Sim	Sim
Preencheu Valor?	Não	Sim	Não	Sim	Sim	Sim
Preencheu Tipo do desconto?	Não	Sim	Sim	Não	Sim	Sim
Preencheu Descrição	Não	Sim	Sim	Sim	Não	Sim
Ações						
Realizar o cadastro?	Não	Não	Não	Não	Não	Sim

Casos de Testes:

- **CT01:** Cadastro sem preencher os campos Código do cupom, Valor, Tipo do desconto e Descrição, não deve realizar o cadastro.
- **CT02:** Cadastro sem preencher o campo Código do cupom, não deve realizar o cadastro.
- **CT03:** Cadastro sem preencher o campo Valor, não deve realizar o cadastro.
- **CT04:** Cadastro sem preencher o campo Tipo do desconto, não deve realizar o cadastro.
- **CT05:** Cadastro sem preencher o campo Descrição, não deve realizar o cadastro.
- **CT06:** Cadastro preenchendo os campos Código do cupom, Valor, Tipo do desconto e Descrição, deve realizar o cadastro.

Regra de Negócio:

RN03 - Nome do cupom não pode ser repetido

Casos de Testes:

- **CT01:** Ao realizar um cadastro informando um nome de cupom que já está cadastrado no sistema, não deve realizar o cadastro.
- **CT02:** Ao realizar um cadastro informando um nome de cupom que não está cadastrado no sistema, deve realizar o cadastro.

1.4 Repositório no Github

- Crie um repositório no github com o nome TCC-EBAC;
- Deixe o repositório público 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/brukalschne/TCC-EBAC>

1.5 Repositório no Github

1.5.1 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.

1.5.2 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

1.6 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

1.7 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

<input type="checkbox"/>	Nome de usuário	Nome	E-mail	Função
<input type="checkbox"/>	 user1_ebac	—	user1_ebac@ebac.com	Assinante
<input type="checkbox"/>	 user2_ebac	—	user2_ebac@ebac.com	Assinante
<input type="checkbox"/>	 user3_ebac	—	user3_ebac@ebac.com	Assinante
<input type="checkbox"/>	 user4_ebac	—	user4_ebac@ebac.com	Assinante
<input type="checkbox"/>	 user5_ebac	—	user5_ebac@ebac.com	Assinante

- 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:

HTTP Request

Name: -31

Comments: Detected the start of a redirect chain

Basic Advanced

Web Server

Protocol (http): Server Name or IP:

HTTP Request

POST Path: /minha-conta/

☐ Redirect Automatically
 ☒ Follow Redirects
 ☒ Use KeepAlive
 ☐ Use multipart/form-data
 ☐ Browser-compatible headers

Parameters Body Data Files Upload

Send Parameters With the Request:

Name:	Value
username	\$(usuario)
password	\$(senha)
wocommerce-login-nonce	
_wp_http_referer	/minha-conta/
login	Login

CONCLUSÃO

O desenvolvimento deste projeto ampliou meus conhecimentos teóricos e práticos na área de qualidade de software. Eu já atuava nesta área, como profissional, mas aprendi muita coisa nova durante o curso, o que agregou ainda mais para a minha carreira. A técnica de testes BDD, usando a linguagem gherkin, automação de testes, implementação de processos de integração contínua, e aplicação de testes de performance, são alguns dos exemplos de conteúdos que aprendi neste curso e contribuíram muito para o meu trabalho.

REFERÊNCIAS BIBLIOGRÁFICAS

Curso Teste de Software da EBAC: <https://lms.ebaconline.com.br/dashboard>