



ENGENHEIRO DE QUALIDADE DE SOFTWARE

Raphael Pereira Carlos

Análise de Qualidade

Blumenau

2023

1. RESUMO

O projeto de conclusão de curso em Engenharia de Qualidade de Software da EBAC - Escola Britânica de Artes Criativas e Tecnologia aborda a otimização dos processos de desenvolvimento de software com foco na qualidade. Utilizando abordagens de engenharia de software, gerenciamento de projetos e técnicas de teste, o projeto visa aprimorar a eficiência e a confiabilidade das aplicações desenvolvidas.

A pesquisa examina a importância da definição de requisitos claros e do uso de metodologias ágeis, como o Scrum, para garantir uma comunicação eficaz entre as equipes de desenvolvimento. Além disso, o estudo se concentra na implementação de estratégias de automação de testes e na adoção de métricas de qualidade para avaliar o desempenho do software.

O objetivo final é demonstrar como a aplicação de práticas de engenharia de qualidade de software pode resultar em produtos mais robustos, reduzindo custos e tempo de desenvolvimento. O projeto destaca a importância crescente da qualidade de software no mercado atual, onde a satisfação do cliente e a competitividade são fundamentais para o sucesso das empresas de tecnologia.

.

SUMÁRIO

2.

1. RESUMO	2
2. SUMÁRIO	3
3. INTRODUÇÃO	4
4. O PROJETO	5
4.1 Estratégia de teste	6
4.2 Critérios de aceitação	6
4.3 Casos de testes	6
4.4 Repositório no Github	7
4.5 Testes automatizados	7
4.6 Integração contínua	8
4.7 Testes de performance	8
5. CONCLUSÃO	9
6. REFERÊNCIAS BIBLIOGRÁFICAS	9

3. INTRODUÇÃO

A qualidade de software desempenha um papel crucial em nosso mundo cada vez mais digital. À medida que a tecnologia continua a evoluir e se infiltrar em todos os aspectos de nossas vidas, a necessidade de software confiável e de alta qualidade se torna mais evidente do que nunca.

Imagine utilizar um aplicativo ou sistema que frequentemente trava, apresenta erros ou não atende às suas expectativas. Isso não só frustra os usuários, mas também pode causar perdas financeiras e danos à reputação das empresas. É aqui que entra o papel fundamental da Engenharia de Qualidade de Software (QA - Quality Assurance).

A QA é a disciplina que se dedica a garantir que o software seja desenvolvido, testado e entregue com os mais altos padrões de qualidade. Isso envolve a definição de requisitos claros, o uso de metodologias eficazes de desenvolvimento e a implementação de rigorosas práticas de teste. Além disso, a automação de testes e a análise de métricas desempenham um papel vital na busca pela excelência.

Neste contexto, a Engenharia de Qualidade de Software torna-se não apenas uma vantagem competitiva, mas também uma necessidade imperativa para as empresas que desejam prosperar no cenário tecnológico atual. Ela não apenas garante a satisfação do cliente, mas também contribui para a confiabilidade, segurança e eficiência de sistemas que sustentam nossas vidas cotidianas. Portanto, a busca pela qualidade de software é essencial para impulsionar a inovação e o sucesso empresarial no mundo digital de hoje.

A pesquisa "O Futuro da Qualidade de Software", conduzida em colaboração entre a Inmetrics, uma empresa brasileira especializada em soluções digitais e eficiência em TI, a TGT Consult, uma consultoria de gestão focada em estratégia de TI, e o centro de tecnologia da Universidade Estadual de Campinas (Unicamp), será lançada em 9 de maio, com transmissão via LinkedIn Inmetrics.

Este estudo, concluído no final do ano passado, oferece uma visão abrangente sobre a Qualidade de Software no Brasil, identificando tendências do setor e comparando o cenário brasileiro com o mercado internacional. A pesquisa

envolveu um questionário detalhado respondido por empresas participantes, resultando em um conjunto de dados que serviu de base para a elaboração de um White paper analisando o mercado de qualidade de software no Brasil.

A qualidade de software é fundamental para o sucesso das empresas em um mundo digitalizado, impactando a satisfação do cliente, a eficiência dos processos e a redução de custos operacionais. O estudo abordará práticas recomendadas e adotadas pelas empresas, visando inovações em Quality Assurance (QA) e promovendo a evolução do mercado nacional em conformidade com as melhores práticas globais.

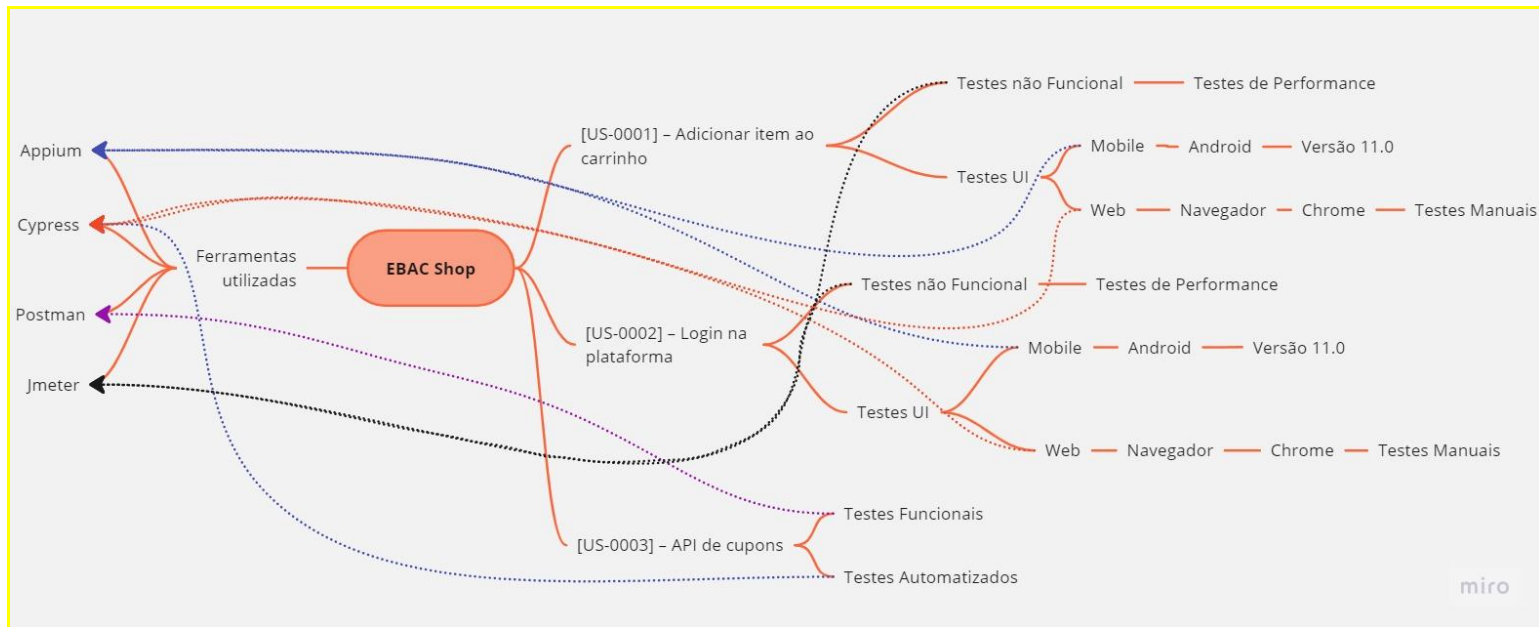
Aderson Bastos, CTO de Qualidade da Inmetrics, destaca a importância dessa pesquisa para acompanhar as últimas atualizações do setor e abordar as demandas crescentes das empresas por produtos de melhor desempenho e entrega mais rápida aos clientes. A pesquisa também busca comparar as práticas brasileiras com as internacionais, fornecendo perspectivas valiosas para o setor de Qualidade de Software no Brasil.

4. 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.

4.1 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.;
- Referência: Módulo 5
- Após fazer sua estratégia de teste, tire um print e cole aqui:



4.2 Critérios de aceitação

- Considere as histórias de usuário:
 - [US-0001] – Adicionar item ao carrinho
 - [US-0002] – Login na plataforma
 - [US-0003] – 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: Inclusão de produtos ao Carrinho

Como cliente da EBAC-SHOP

Quero adicionar produtos no carrinho

Para realizar a compra dos itens

Critério de aceitação: Não deve permitir adicionar produtos sem estoque.

Cenário: Adição de produto em estoque

Dado que eu acesse a página de produtos da EBAC-SHOP

Quando eu selecionar o produto “Augusta Pullover Jacket”

E eu selecionar o tamanho SHOP

E eu selecionar a cor Orange

E eu informar a quantidade igual a 10

E eu clicar no Botão COMPRAR

Então deve exibir uma mensagem de “produto sem estoque”

Critério de aceitação: Deve Incluir o item ao carrinho com sucesso.

Cenário: Não deve permitir adicionar uma quantidade de produto superior a 76

Dado que eu acesse a página de produtos da EBAC-SHOP

Quando eu selecionar o produto “Augusta Pullover Jacket”

E eu selecionar o tamanho SHOP

E eu selecionar a cor Orange

E eu informar a quantidade igual a 100

E eu clicar no Botão COMPRAR

Então deve exibir uma mensagem de “O valor deve ser menor ou igual a 76.”

Critério de aceitação: Deve Incluir o item ao carrinho com sucesso.

Cenário: Mensagem de produto sem estoque

Dado que eu acesse a página de produtos da EBAC-SHOP

Quando eu selecionar o produto “camisa Brasil”

E eu selecionar o tamanho M

E eu clicar no Botão COMPRAR

Então deve exibir uma mensagem de “Desculpe, este produto não está disponível. Escolha uma combinação diferente.”

Critério de aceitação: Não Deve Incluir o item ao carrinho.

Cenário: Compra realizada com sucesso

Dado que eu acesse a página meu carrinho da EBAC-SHOP

Quando eu inserir os dados para pagamento

E eu clicar em concluir

Então deve exibir uma mensagem de “Pagamento aprovado - Compra efetuada com sucesso”

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 (autenticação) na plataforma

Para visualizar meus pedidos

Contexto:

Dado que eu acesse a plataforma EBAC

Cenário: Autenticar acesso com usuário existente

Quando eu inserir o usuário 'paulo@ebac.com.br'

E a senha 'paulo@2023'

Então deve ser direcionado para a tela de checkout

Cenário: Autenticar acesso com usuário inexistente

Quando eu inserir o usuário 'paulo_2023@ebac.com.br'

E a senha 'paulo@2023'

Então deve exibir uma mensagem de alerta 'Erro: Endereço de e-mail desconhecido. Verifique novamente ou tente seu nome de usuário.'

Cenário: Autenticar acesso com usuário existente e senha incorreta

Quando eu inserir o usuário 'paulo@ebac.com.br'

E a senha 'paulo@2022'

Então deve exibir uma mensagem de alerta 'Erro: a senha fornecida para o e-mail paulo@ebac.com.br está incorreta. Perdeu a senha?'

Cenário: Deve bloquear acesso após a 3 tentativa de acesso com senha incorreta

Quando eu inserir o usuário 'paulo@ebac.com.br'

E a senha 'paulo@2022'

Então após a 3 tentativa o usuário deve ser bloqueado.

Esquema do Cenário: Autenticar múltiplos usuários inexistentes

Quando eu inserir o usuário <usuario>

E a senha <senha>

Então deve exibir uma mensagem de alerta <usuario> 'ou senha inválidos'

Exemplos:

usuário	senha	mensagem
"paulo@ebac.com.br"	"paulo@2023"	"Olá Gabriela!"
"pedro@ebac.com.br"	"pedro@2023"	"Olá Maria!"
"joao@ebac.com.br"	"joao@2023"	"Olá Gabriel!"
"gabriel@ebac.com.br"	"gabriel@2023"	"Olá Gabriel!"

4.2.3 Critério de aceitação - [US-0001] – API de cupons

Funcionalidade: Inclusão de produtos ao Carrinho

Como admin da EBAC-SHOP

Quero criar um serviço de cupom

Para poder listar e cadastrar os cupons

Critério de aceitação: Deve apenas mostrar cupons de 10% da loja selecionada

Cenário: Cupom BEMVND010 loja EBAC

Dado que eu acesse a API de cupons

Quando eu efetuar uma pesquisa sobre a loja da EBAC

E procurar por cupons de 10%

Então deve exibir uma mensagem de “Cupom BEMVND010 para primeira compra no site”

Critério de aceitação: Não deve cadastrar cupons com código existente

Cenário: Cupom BEMVND010 loja EBAC

Dado que eu acesse a API de cupons

E eu informar em code um código já cadastrado

Então deve exibir uma mensagem de “O código de cupom já existe”

Critério de aceitação: Deve cadastrar cupons com sucesso

Cenário: Cupom BEMVND010 loja EBAC

Dado que eu acesse a API de cupons

E eu informar os dados para cadastrar o cupom

Então a API deve retornar o status 201 cadastrando um novo cadastro

Critério de aceitação: Deve expor somente cupons válidos

Cenário: Cupons válidos loja EBAC-SHOP

Dado que eu acesse a API de cupons

Quando eu selecionar a loja EBAC-SHOP

E clicar em visualizar cupons

Então deve exibir uma mensagem “Cupons válidos para a data de hoje da loja EBAC-SHOP”

4.2.4 Histórias de usuário - Catálogo de Produtos

Como cliente da EBAC-SHOP

Quero visualizar a lista de produtos disponíveis no catálogo

Para que eu possa ver os itens que posso comprar.

4.2.5 Histórias de usuário - Painel Minha Conta

Como cliente da EBAC-SHOP

Quero acessar o meu Painel "Minha Conta" usando meu e-mail e senha

Para acessar recursos personalizados.

4.2.6 Histórias de usuário - Meus Pedidos

Como cliente da EBAC-SHOP

quero poder visualizar uma lista dos meus pedidos recentes na página "Meus Pedidos"

Para acompanhar o status de cada pedido.

4.2.7 Histórias de usuário - Endereços

Como cliente da EBAC-SHOP

Quero poder editar as informações do meu endereço na página "Meu Endereço"

Para atualizar dados como nome, rua, número, complemento, CEP, cidade e estado

4.2.8 Histórias de usuário - Detalhes da Conta

Como cliente da EBAC-SHOP

Quero poder visualizar as informações da minha conta na página "Detalhes da Conta"

Para atualizar os dados como nome, sobrenome, e-mail, e senha de acesso

4.3 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 ao carrinho;

0 até 10	> 10
0,1,2,...,9, 10	11,12,13,...,98,100
Dados válidos	Dados inválidos

Caso de teste

	Entrada	Saída
Teste 1	inserir mais de 10 itens de um mesmo produto ao carrinho	inválido
Teste 2	inserir menos de 10 itens de um mesmo produto ao carrinho	válido
Teste 3	inserir 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
0,10,00 ,20,00 ,...,100,00,..., 899,99	990,00,1.000,000,...
Dados válidos	Dados inválidos

Caso de teste

	Entrada	Saída
Teste 1	Total do valor de R\$ 1.000,00	inválido
Teste 2	Total do valor de R\$ 80,00	válido
Teste 3	Total do valor de R\$ 899,99	válido

RN03

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

< 199,99	200,00 até 600,00	> 600
0,10,00 ,20,00 ,...,100,00,..., 199,00	200,00,300,000,..., 600,000	700,00,800,00,...
Dados inválidos	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\$ 601,00	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	> 990,00
0,10,00 ,20,00 ,...,100,00,..., 599,00	990,00,1.000,000,...
Dados inválidos	Dados válidos

Caso de teste

	Entrada	Saída
Teste 1	Total do valor de R\$ 1.000,00	válido
Teste 2	Total do valor de R\$ 80,00	inválido
Teste 3	Total do valor de R\$ 599,99	inválido

4.3.2 - Caso de teste: Login na plataforma

RN01

Somente usuários ativos podem fazer login;

Caso de teste

Condições	Regra 1	Regra 2	Regra 3	Regra 4
Usuário Inativo	Não	Sim	Não	Sim
Usuário Bloqueado	Não	Não	Sim	Sim
Ações				
Permitir acesso?	Sim	Não	Não	Não

Casos de testes:

1. Login com usuário **Ativo** e Usuário **desbloqueado**, deve permitir acesso
2. Login com usuário **Inativo** e Usuário **desbloqueado**, não deve permitir acesso
3. Login com usuário **Ativo** e Usuário **bloqueado**, não deve permitir acesso
4. Login com usuário **Inativo** e Usuário bloqueado, não deve permitir acesso

RN02

Deve exibir uma mensagem de erro caso o usuário erre o login e senha;

Caso de teste

Condições	Regra 1	Regra 2	Regra 3	Regra 4
Usuário válido?	Sim	Sim	Não	Não
Senha válida?	Sim	Não	Sim	Não
Ações				
Permitir 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

RN03

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

Caso de teste

Condições	Regra 1	Regra 2
travar acesso por tentativas de erro	< 2	> 3
Ações		
Travar		X
Não travar	X	

RN04

Apenas alunos entre 16 e 80 anos podem ser cadastrados

Caso de teste

Condições	Regra 1	Regra 2	Regra 3
Idade	< 16	16 a 18	> 80
Ações			
Cadastrar		X	
Não cadastrar	X		X

4.3.3 Caso de teste: API de Cupom

RN01

Apenas usuários autenticados podem realizar o cadastro de cupom

API: [/wc/v3/coupons](#)

Caso de teste

Testes	Test 1	Teste 2	Teste 3
Estado Inicial:	Login e Senha corretos	Login correto e Senha incorreta	Login correto e Senha incorreta
Evento:	Integrar	Não integrar	Não integrar
Estado final:	Cadastrar cupom	Não cadastrar cupom	Não cadastrar cupom

RN02

Apenas requisições informando novos nomes podem cadastrar os cupons

API: [/wc/v3/coupons](#)

Caso de teste

Testes	Test 1	Teste 2
Estado Inicial:	Novo nome	Nome repetido
Evento:	Integrar	Não integrar
Estado final:	Cadastrar cupom	Não cadastrar cupom

RN03

Apenas novos nomes de cupons podem ser cadastrados

API: [/wc/v3/coupons](#)

Caso de teste

Testes	Test 1	Teste 2
Estado Inicial:	Novo nome	Nome repetido
Evento:	Integrar	Não integrar
Estado final:	Cadastrar cupom	Não cadastrar cupom

RN04

Deve cadastrar os cupons informando os campos obrigatórios

API: [/wc/v3/coupons](#)

Caso de teste

Testes	Test 1	Teste 2	Teste 3	Teste 4	Teste 5
Estado Inicial:	Informar os campos: Código do cupom, Valor, Tipo do desconto e Descrição	Informar apenas os campos: Valor, Tipo do desconto e Descrição	Informar apenas os campos: Código do cupom,, Tipo do desconto e Descrição	Informar apenas os campos: Código do cupom, Valor e Descrição	Informar apenas os campos: Código do cupom, Valor e Tipo do desconto
Evento:	Integrar	Não integrar	Não integrar	Não integrar	Não integrar
Estado final:	Cadastrar cupom	Não cadastrar cupom	Não cadastrar cupom	Não cadastrar cupom	Não cadastrar cupom

4.4 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/raphaelcarlos/TCC-EBAC-QE>

4.5 Testes automatizados

4.5.1 Automação de UI

Ao considerar ferramentas de automação de testes para nossos projetos, é crucial avaliar cuidadosamente as opções disponíveis, levando em consideração fatores como facilidade de uso, velocidade de execução, manutenção de código e integração contínua. Após uma análise aprofundada, decidimos adotar o Cypress como nossa ferramenta de automação preferida, superando o Selenium e o Playwright, pelas seguintes razões:

Facilidade de Uso e Produtividade Eficiente:

O Cypress se destaca pela sua abordagem amigável e intuitiva para a automação de testes. Sua sintaxe simplificada e estrutura de teste clara permitirão que nossos desenvolvedores escrevam cenários de teste de maneira eficaz e rápida. Isso acelera nosso processo de desenvolvimento de testes e economiza tempo valioso.

Execução Rápida e Eficiente:

A arquitetura de execução direta no navegador do Cypress oferece uma vantagem significativa em termos de velocidade de execução em comparação com o Selenium. Isso nos permitirá realizar testes de regressão e validação de maneira mais rápida, melhorando nossa eficiência geral no ciclo de desenvolvimento.

API Simplificada e Manutenção Acessível:

A API coesa e direta do Cypress reduz a complexidade do nosso código de teste, resultando em testes mais legíveis e fáceis de manter. Ao evitar a verbosidade associada ao Selenium e ao Playwright, conseguiremos criar e atualizar cenários de teste com mais facilidade, reduzindo o esforço de manutenção ao longo do tempo.

Integração Contínua e Suporte Amplo:

O Cypress possui integração sólida com várias ferramentas de integração contínua amplamente utilizadas, como Jenkins e Travis CI. Isso nos permitirá incorporar nossos testes de forma transparente em nosso pipeline de entrega contínua, garantindo que a qualidade seja mantida em cada etapa do processo de desenvolvimento.

Visibilidade de Testes e Depuração Simplificada:

A interface de controle interativo do Cypress oferece visibilidade abrangente sobre a execução dos testes, facilitando a identificação e solução rápida de problemas. Isso reduzirá o tempo necessário para depurar e resolver quaisquer problemas que possam surgir durante a execução dos testes.

Baseados nessas considerações, a escolha do Cypress como nossa ferramenta de automação de testes é uma decisão estratégica que visa aumentar a qualidade de nossos produtos, otimizar nossos processos de teste e melhorar nossa produtividade geral. Estamos confiantes de que o Cypress nos ajudará a atingir nossos objetivos de garantir uma experiência confiável e consistente para nossos usuários.

4.5.2 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! ☺

4.5.3 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

4.6 Integração contínua

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






4.7 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

<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

5. CONCLUSÃO

Meu TCC abordou a otimização dos processos de desenvolvimento de software com foco na qualidade, explorando abordagens de engenharia de software, gerenciamento de projetos e técnicas de teste.

Os principais desafios que enfrentei durante a pesquisa incluíram a configuração de testes mobile, que exigiu mais atenção e configurações complicadas em um computador que deixou a desejar muitas vezes, e a gestão do tempo, já que tive que trabalhar no projeto após o expediente e nos fins de semana.

Para me auxiliar na conclusão realizei muitas pesquisas em cursos anteriores e consultas ao Google.

Aprendi lições valiosas sobre o processo de pesquisa acadêmica, bem como sobre gerenciamento de tempo e recursos.

Lidei com o estresse e a pressão, mesmo em situações pessoais desafiadoras, como uma cirurgia e a descoberta de que seria pai, mantendo o foco e a perseverança

No meu trabalho recebi feedbacks positivos graças ao curso que me ajudaram a padronizar processos de criação de issues de bugs e melhorar o controle de automação.

Os resultados mais significativos do TCC foram a superação das dificuldades na elaboração de testes mobile e o aprimoramento dos testes em Cypress.

As habilidades adquiridas no TCC, como a atenção aos detalhes nos cenários de teste e a seleção cuidadosa dos cenários a serem automatizados, foram aplicadas com sucesso em sua carreira profissional.

6. REFERÊNCIAS BIBLIOGRÁFICAS

Meios digitais:

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 10 setembro de 2023 às 19h10min

Tema: Cypress: passo a passo para começar a usar

<https://testingcompany.com.br/blog/conheca-o-cypress-e-suas-vantagens-para-automacao-de-testes>

Acessado em 01 setembro de 2023 às 21h

Tema: 10 motivos para você usar Cypress na automação de testes web

<https://talkingabouttesting.com/2021/03/18/10-motivos-para-voce-usar-cypress-na-automacao-de-testes-web/>

Acessado em 01 setembro de 2023 às 21h 20m