



QUALIDADE DE SOFTWARE

Maria Luciano da Silva

Análise de Qualidade

Rio de Janeiro

2023

1. RESUMO

Este projeto tem como objetivo principal aprimorar a qualidade de software por meio de uma abordagem abrangente e multidisciplinar. Serão investigadas técnicas de teste, automação, boas práticas de desenvolvimento e gerenciamento de processos, visando a otimização da qualidade em todo o ciclo de vida do software. O estudo incorpora a análise de métricas, a avaliação de ferramentas e a aplicação de metodologias ágeis. Além disso, serão exploradas as tendências, como estratégias de teste, critérios de aceitação, casos de testes, repositório github, testes automatizados, integração contínua e testes de performances para aperfeiçoar a qualidade do software em um contexto moderno. Este projeto visa contribuir para o avanço da área de qualidade de software e fornecer insights práticos para profissionais e organizações que buscam entregar produtos de software confiáveis e de alto desempenho.

SUMÁRIO

2.

1. RESUMO	2
2. SUMÁRIO	3
3. INTRODUÇÃO	4
4. O PROJETO	5
4.1 Estratégia de teste	5
4.2 Critérios de aceitação	5
4.2.1 História de usuário 1: [US-0001] – Adicionar item ao carrinho	5
4.2.2 História de usuário 2: [US-0002] – Login na plataforma	6
4.2.3 História de usuário 2: [US-0003] – API de cupons	6
4.3 Casos de testes	6
4.3.1 História de usuário 1:	6
4.3.2 História de usuário 2:	6
4.3.1 História de usuário 3: API de Cupom	7
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

No cenário da tecnologia da informação, a qualidade de software é um fator crítico para o sucesso de produtos e sistemas. A crescente complexidade e a demanda por software confiável, seguro e eficiente exigem abordagens inovadoras para aprimorar a qualidade em todo o ciclo de vida do desenvolvimento de software. Este projeto de conclusão de curso visa abordar esse desafio de maneira abrangente, incorporando uma variedade de disciplinas e técnicas.

Neste trabalho, explorarei um conjunto diversificado de tópicos relacionados à qualidade de software. Irei investigar técnicas de teste, automação, boas práticas de desenvolvimento e gerenciamento de processos, com o objetivo de fornecer um panorama holístico das estratégias disponíveis para otimizar a qualidade do software. Além disso, examinei métricas que podem ser empregadas para avaliar e monitorar a qualidade, bem como a eficácia de ferramentas e metodologias ágeis na busca por esse aprimoramento.

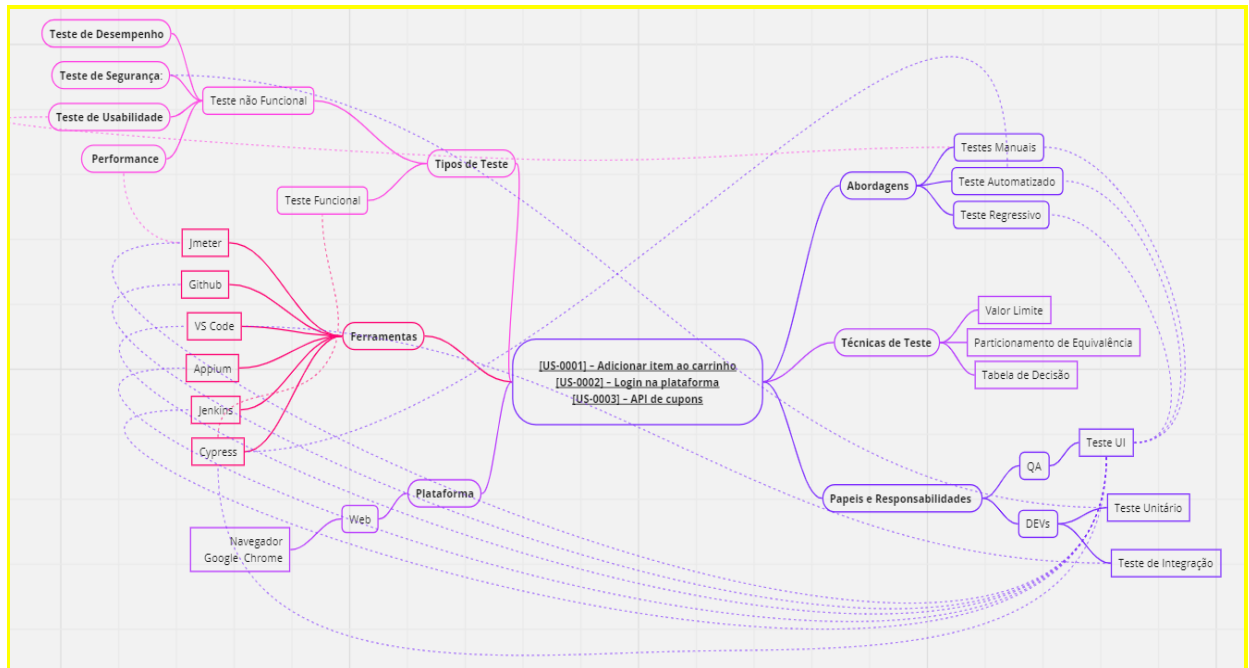
Este projeto almeja contribuir significativamente para a compreensão e aplicação de práticas na área de qualidade de software, beneficiando tanto profissionais quanto organizações que buscam oferecer produtos de software de alta qualidade em um mercado cada vez mais competitivo e exigente.

4. O PROJETO

Para o Trabalho de Conclusão de Curso Qualidade de Software, 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.

4.1 Estratégia de teste



4.2 Critérios de aceitação

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

Critério de aceitação:

Cenário 1: Compra mais de 10 itens de um mesmo produto.

Dado que eu acesse a página de carrinho da EBAC-SHOP;
Quando eu inserir a quantidade “20”, e clicar em Concluir a Compra;
Então deve exibir uma mensagem de alerta “ Você ultrapassou o limite máximo de 10 itens por produto!”

Cenário 2: Compra com valor entre R\$ 200 e R\$ 600

Dado que eu acesse a página de carrinho da EBAC-SHOP
Quando eu inserir a quantidade <quantidade> do produto <produto>
Então deve ser aplicado um desconto de 10%

Exemplos:

quantidade	produto
“5”	“Circe Hooded Ice Fleece - XL, Green”
“3”	“Aether Gym Pant - 36, Brown”

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

Critérios de aceitação:

Cenário 1: Login com email, cpf ou nome de usuário.

Dado que eu acesse a página de login da EBAC-SHOP
Quando eu digitar o login <email, cpf ou nome de usuário>, ea senha <senha@@8733#>

Então deve exibir uma mensagem de boas vindas <“Olá Maria Luciano”>

Exemplos:

login	senha	mensagem
“maria.lusilva@ebac.com.br”	“senha@@8733#”	“Olá Maria Luciano”
“11111111111”	“senha@@8733#”	“Olá Maria Luciano”
“MaLuS2”	“senha@@8733#”	“Olá Maria Luciano”

Critérios de aceitação:

Cenário 1: Login ou senha inválidos

Dado que eu acesse a página de login da EBAC-SHOP
Quando eu digitar o login "aisjifs4845"

E a senha "senha@123"

Então deve exibir uma mensagem de alerta "Algo deu errado verifique suas credenciais e tente novamente!"

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

Critérios de aceitação:

Cenário 1: Listagem de cupons

Dado que eu seja um administrador

E acesse a documentação do serviço da EBAC-SHO;

Quando eu inserir o id "5"

Então deve retornar status "200" com os dados do cupom de id "5"

Cenário 2: Cupom não pode ser cadastrado duas vezes.

Dado que eu acesse a documentação do serviço da EBAC-SHOP

E esteja autenticado como administrador

Quando eu inserir o código do cupom <code>, e o cupom já estiver cadastrado <amount>

Então deve mostrar a seguinte mensagem "Cupom já cadastrado"

Exemplos:

Tipo	Usuário	Senha	Response	
Administrador	"admin_eb"	"@adm123"	500	
Administrador	"admin_ebac"	"@adm123"	"200"	
Cliente	"Malus2"	"@@S2##"	500	

4.3 Casos de testes

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

CT01: Adicionar Produto ao Carrinho Dentro dos Limites de Quantidade e Valor

Descrição: Verificar se é possível adicionar um produto ao carrinho dentro dos limites de quantidade (até 10 unidades do mesmo produto) e valor (não ultrapassar R\$990,00).

Passos:

Acesse a EBAC-SHOP como cliente.

Navegue até a página de produtos.

Escolha um produto que custa R\$100,00.

Adicione 5 unidades do produto ao carrinho.

Resultado Esperado: O produto deve ser adicionado ao carrinho com sucesso sem nenhum erro. O total no carrinho não deve ultrapassar R\$990,00.

Particionamento de Equivalência:

Regras	Entrada	Saída
RN 01	Adicionar 5 produtos ao carrinho	Válido
RN 01	Valor da compra é maior que R\$ 990,00	Invalido

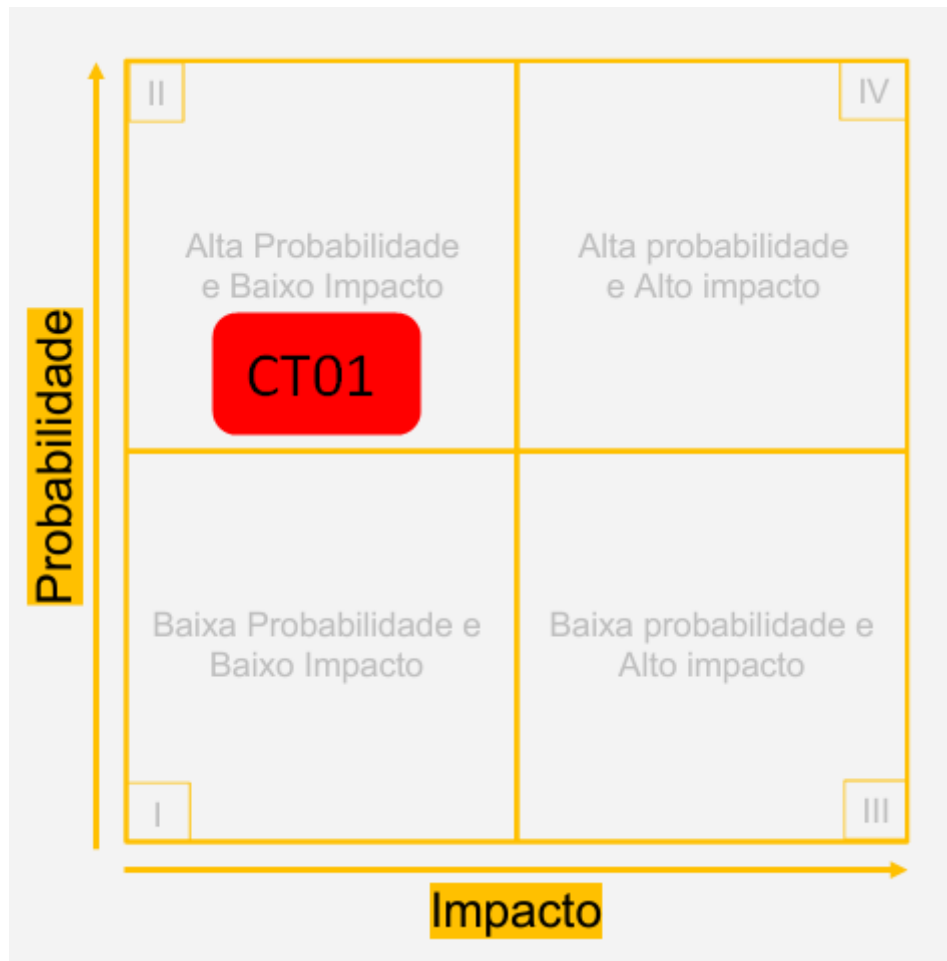
Valor Limite:

Regras	Entrada	Saída
RN 01	Adicionar 9 itens do mesmo produto	Válido
RN 01	Adicionar 10 itens do mesmo produto	Válido
RN 01	Adicionar 15 itens do mesmo produto	Invalido

Tabela de Decisão:

Condições	Regra 1	Regra 2	Regra 3
>10	V	F	F
<10	F	V	F
10	F	F	V
ADICIONAR	-	X	X

Matriz de Risco:



CT02: Tentar Adicionar Mais de 10 Unidades do Mesmo Produto

Descrição: Verificar se o sistema impede o cliente de adicionar mais de 10 unidades do mesmo produto ao carrinho.

Passos:

Acesse a EBAC-SHOP como cliente.

Navegue até a página de produtos.

Escolha um produto que custa R\$40,00.

Tente adicionar 12 unidades do produto ao carrinho.

Resultado Esperado: O sistema deve exibir uma mensagem de erro informando que não é permitido adicionar mais de 10 unidades do mesmo produto ao carrinho.

CT03: Aplicar Desconto ao Carrinho com Valor Entre R\$ 200 e R\$ 600

Descrição: Verificar se o sistema aplica corretamente um desconto de 10% no carrinho quando o valor total estiver entre R\$200 e R\$600.

Passos:

Acesse a EBAC-SHOP como cliente.

Navegue até a página de produtos.

Escolha dois produtos que custam R\$150,00 cada.

Adicione ambos os produtos ao carrinho.

Resultado Esperado: O sistema deve aplicar um desconto de 10% no valor total do carrinho, que deve ser de R\$270,00. O cliente deve ser notificado sobre o desconto aplicado.

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

CT01:Caso de Teste 1: Login bem-sucedido

Pré-condições: O usuário está ativo e possui uma conta na EBAC-SHOP.

Passos:

Abra o navegador da web e acesse a página de login da EBAC-SHOP.

Insira as credenciais corretas do usuário (e-mail, nome de usuário ou CPF) e a senha correta nos campos apropriados.

Clique no botão de login.

Verifique se o sistema permite o login com sucesso.

Após o login bem-sucedido, verifique se o usuário é redirecionado para a página de visualização de pedidos.

Certifique-se de que as informações dos pedidos do usuário sejam exibidas corretamente.

Resultado esperado: O sistema permite o login com sucesso e redireciona o usuário para a página de visualização de pedidos.

CT02: Erro de login com credenciais incorretas

Pré-condições: O usuário está ativo e possui uma conta na EBAC-SHOP.

Passos:

Abra o navegador da web e acesse a página de login da EBAC-SHOP.

Insira credenciais inválidas (por exemplo, um e-mail/nome de usuário/CPF válido, mas senha inválida) nos campos apropriados.

Clique no botão de login.

Verifique se o sistema exibe uma mensagem de erro informando que as credenciais estão incorretas.

Certifique-se de que o sistema não permite o login com credenciais incorretas.

Registre a tentativa de login incorreto para futura referência.

Resultado esperado: O sistema exibe uma mensagem de erro informando que as credenciais estão incorretas e não permite o login. A contagem de tentativas de login incorretas não deve passar de 3.

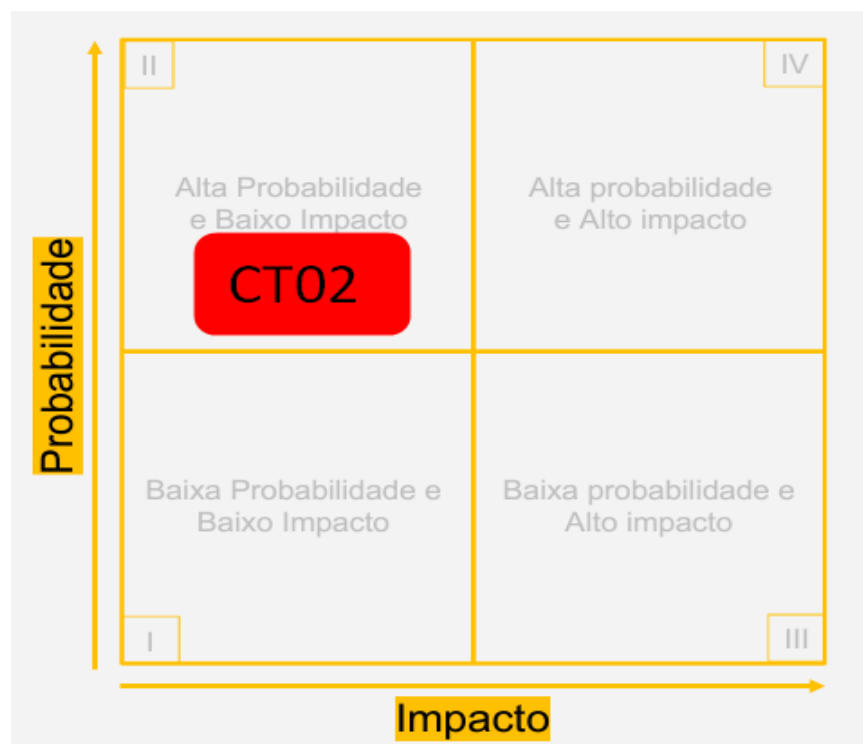
Particionamento de Equivalência

Regras	Entrada	Saída
RN 04	Errar uma vez a senha	Válido
RN 04	Errar mais de 3 vezes a senha	Invalido

Valor Limite:

Regras	Entrada	Saída
RN 04	Errar 2 vezes a senha	Válido
RN 04	Errar 3 vezes a senha	Invalido

Matriz de Risco



CT03: Bloqueio de login após 3 tentativas de senha incorretas

Pré-condições: O usuário está ativo e possui uma conta na EBAC-SHOP.

Passos:

Abra o navegador da web e acesse a página de login da EBAC-SHOP.

Insira credenciais válidas (por exemplo, um e-mail/nome de usuário/CPF válido) nos campos apropriados.

Insira uma senha incorreta.

Clique no botão de login.

Repita os passos 2 e 3 para mais duas tentativas com senha incorreta.

Após a terceira tentativa de senha incorreta, verifique se o sistema bloqueia o login do usuário por 15 minutos.

Certifique-se de que uma mensagem de bloqueio seja exibida, informando o tempo restante para o desbloqueio.

Tente fazer login novamente após o período de bloqueio de 15 minutos e verifique se o sistema permite o acesso.

Resultado esperado: Após a terceira tentativa de senha incorreta, o sistema bloqueia o login do usuário por 15 minutos. Deve ser exibida uma mensagem informando o bloqueio e o tempo restante. O usuário não pode tentar fazer login novamente até que o bloqueio seja levantado após 15 minutos.

1.1.1 História de usuário 3: API de Cupom

CT01: Listar todos os cupons cadastrados

Descrição: objetivo verificar se a API de cupom é capaz de listar todos os cupons cadastrados.

Passos:

Realizar uma requisição GET para a rota "/cupons" da API.

Verificar se a resposta da API apresenta um código de status HTTP 200 (OK).

Analisar o corpo da resposta em formato JSON para assegurar que todos os cupons cadastrados estejam presentes com informações completas.

Certificar-se de que a lista não contenha cupons em excesso ou com faltas.

CT02: Cadastrar um novo cupom com todos os campos obrigatórios

Descrição: Verificar se a API permite o cadastro bem-sucedido de um novo cupom com todos os campos obrigatórios devidamente preenchidos.

Passos:

Criar um novo objeto JSON que representa um cupom válido com todos os campos obrigatórios preenchidos, incluindo código do cupom, valor, tipo de desconto e descrição.

Efetuar uma requisição POST para a rota "/cupons" da API, enviando o objeto JSON recém-criado.

Verificar se a resposta da API contém um código de status HTTP 201 (Created).

Consultar a lista de cupons após o cadastro e confirmar que o novo cupom foi adicionado corretamente.

CT03: Tentar cadastrar um cupom com código repetido

Objetivo: Verificar se a API impede o cadastro de um cupom com um código que já existe no sistema.

Passos:

Identificar um código de cupom que já esteja cadastrado no sistema.

Criar um novo objeto JSON que representa um cupom com o código já existente, preenchendo os outros campos obrigatórios de forma correta.

Realizar uma requisição POST para a rota "/cupons" da API, enviando o objeto JSON recém-criado.

Verificar se a resposta da API contém um código de status HTTP 400 (Bad Request) e uma mensagem de erro informando que o código do cupom já existe no sistema.

1.2 Repositório no Github

Link do repositório: Meu git

<https://github.com/Malusilva/TCC-EBAC>

1.3 Testes automatizados

1.3.1 Automação de UI

Projeto – Teste UI:

Link do repositório: Meu git

<https://github.com/Malusilva/TCC-EBAC>

1.3.2 Automação de API

Projeto – Teste API:

Link do repositório: Meu git

<https://github.com/Malusilva/TCC-EBAC>

1.4 Integração contínua

Jenkinsfile - UI: <https://github.com/Malusilva/TCC-EBAC.git>

Jenkinsfile- API: <https://github.com/Malusilva/TCC-EBAC.git>

1.5 Testes de performance

Projeto – Teste Performance: meu git

Link do repositório: Meu git

<https://github.com/Malusilva/TCC-EBAC>

2. CONCLUSÃO

Coloque sua experiência na realização do trabalho, o que aprendeu, quais lições pode aplicar em sua vida profissional etc.

O presente trabalho foi concebido com o propósito de colocar em prática as técnicas de teste de software adquiridas durante o curso ministrado pela EBAC. Durante a execução deste projeto, me empenhei em aplicar essas técnicas com a finalidade de avaliar a qualidade e eficácia do software em desenvolvimento. Nesse processo, diversas técnicas demonstraram sucesso notável, enquanto outras enfrentaram desafios devido a problemas técnicos. No entanto, é fundamental destacar que todas as atividades propostas ao longo dos módulos foram realizadas com êxito, cumprindo o seu propósito. Em síntese, este projeto não só demonstrou a aplicação eficaz das técnicas de teste de software, mas também sublinhou a importância do aprendizado contínuo e da adaptação a desafios. O conhecimento adquirido e a experiência

obtida contribuirão para o avanço de futuros projetos e para a qualidade do software desenvolvido.

3. REFERÊNCIAS BIBLIOGRÁFICAS

EBAC (Escola Britânica de Artes Criativas). Teste de Software. 2023.
Disponível em: <https://new.lms.ebaonline.com.br/dashboard>.