

**QUALIDADE DE SOFTWARE**

Igor da Silva Galan

Análise de Qualidade

Aparecida do Taboado, MS.

2023

* **RESUMO**

O propósito deste trabalho reside na apresentação de testes pertinentes ao cotidiano de um profissional de Garantia de Qualidade (QA), como aprendidos pelo autor desta pesquisa. Esta exposição contempla uma variedade de testes que representaram tanto uma fonte de grande satisfação quanto de determinação em meu percurso de aprendizado. A constatação da influência da tecnologia na modelagem do mundo contemporâneo corrobora a ideia de que uma mente voltada para o aprimoramento contínuo permanece receptiva para a concepção de estratégias que transponham as barreiras do conhecimento, permitindo um crescimento individual cada vez mais sólido e progressivo.

* **SUMÁRIO**

***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***

* **INTRODUÇÃO**

Este projeto engloba práticas adquiridas durante o curso ministrado pela EBAC, abrangendo áreas como Teste Web, Teste de API, Teste Android, Automação, Gherkin, elaboração de Mapas Mentais, utilização do Jmeter para manipulação de arquivos CSV e implementação do Jenkinsfile para a realização de integração contínua.

* **O PROJETO**

O presente projeto mestrado é dedicado ao módulo 19 e encerra o curso da EBAC, focalizando-se na área de teste de software. Seu objetivo é demonstrar uma série de técnicas e práticas de teste aplicadas para garantir a integridade do aplicativo e, consequentemente, a satisfação do usuário. Este estudo não apenas abarca a realização de testes, mas também apresenta exemplos elucidativos sobre o funcionamento dos temas tratados, tais como:

**Teste Web:** A análise de páginas da web é crucial para identificar problemas que podem prejudicar a experiência dos usuários ou afetar a reputação da empresa, especialmente relacionados à segurança. Uma aplicação online com desempenho rápido, segurança forte, adaptação e design atraente é desejada pelos desenvolvedores e clientes. Nesse sentido, a verificação de sites é vital para identificar questões gerais e detalhes específicos que podem impactar a experiência do usuário.

**Teste de API:** Profissionais na área de desenvolvimento executam testes específicos nas interfaces de programação de aplicativos (APIs) para avaliar se estas atendem às expectativas em termos de funcionalidade, desempenho, confiabilidade e segurança. O principal objetivo desses testes é detectar falhas ou comportamentos inesperados, visando evitar a entrega de um produto de qualidade insatisfatória ou potencialmente perigoso aos usuários. Garantir o funcionamento eficaz das APIs é crucial, pois sua aceitação depende diretamente dessa eficiência.

Porém, é relevante destacar que testar as APIs pode ser mais complexo do que parece inicialmente. Muitas vezes, essas interfaces utilizam protocolos e padrões diferentes das ferramentas usuais. Esses padrões são essenciais para possibilitar a comunicação entre diferentes plataformas, aplicativos e sistemas. Portanto, a avaliação não se restringe apenas à funcionalidade da API, mas inclui também a análise do desempenho, segurança e integração dos componentes, visando estabelecer uma interface confiável e integrada.

**Teste Android:** O Appium é uma ferramenta de automação usada para testar aplicativos móveis nos sistemas iOS, Android e em plataformas móveis híbridas. Sua característica principal é que permite escrever testes uma vez e usá-los em muitos dispositivos diferentes, economizando tempo e recursos. É bastante popular para testar aplicativos em dispositivos reais, não apenas em emuladores.

Ele funciona como um servidor em segundo plano, se comunicando com o dispositivo para enviar comandos de teste. Para isso, usa a API do WebDriver para executar ações no aplicativo, como clicar em botões ou digitar em campos de texto. Também consegue interagir com o sistema operacional do dispositivo, testando funções como abrir e fechar aplicativos.

**Automação:** A automação de testes é fundamental no desenvolvimento de software, usando ferramentas para fazer testes, confirmar se as funções estão corretas e avaliar o desempenho automaticamente. Isso evita fazer repetidamente testes manuais, torna os testes mais eficientes e abrange mais áreas, permitindo testar em vários ambientes de forma rápida e consistente. Apesar disso, não substitui completamente os testes feitos por pessoas, mas os complementa estrategicamente para melhorar a qualidade do software e aproveitar melhor o tempo e os recursos durante o desenvolvimento.

**Gherkin:** O Gherkin é uma linguagem clara usada para explicar como um software deve funcionar. Usa palavras-chave simples como Given, When e Then para criar cenários de teste fáceis de entender, tanto para pessoas técnicas quanto não técnicas. Esses cenários são usados por ferramentas automáticas de teste para verificar o software. Isso ajuda na comunicação entre os membros da equipe e garante que todos entendam o que o software precisa fazer. Assim, promove uma colaboração eficaz entre todos os envolvidos no projeto de software.

**Mapas Mentais:** Um mapa mental é uma representação visual de ideias ou conceitos interligados. Geralmente começa com um tema central e se expande para subtemas relacionados a ele. Essa ferramenta gráfica auxilia na organização do pensamento, tornando mais fácil compreender conceitos complexos, planejar projetos, memorizar informações e estimular a criatividade. Utilizando cores, palavras-chave, imagens e conexões visuais, os mapas mentais mostram as relações entre os elementos, facilitando a assimilação e a visualização das informações.

**Jmeter e arquivo CSV:** O JMeter, uma ferramenta de software de código aberto desenvolvida pela Apache, é usada para conduzir testes de carga, desempenho e estresse em aplicativos, servidores e serviços diversos. Ele possibilita simular múltiplos usuários acessando um aplicativo ou serviço, permitindo avaliar seu comportamento sob diferentes condições de carga.

Por outro lado, um arquivo CSV (Comma-Separated Values) é um formato de arquivo de texto que armazena dados tabulares de forma simples, onde cada linha representa uma entrada na tabela e os valores são separados por vírgulas (ou outro delimitador). Geralmente, é empregado para armazenar grandes conjuntos de dados que podem ser facilmente lidos e processados por diferentes programas, incluindo o JMeter.

Dentro do contexto do JMeter, um arquivo CSV pode fornecer dados de entrada para os testes. Por exemplo, pode conter informações como nomes de usuários, senhas, IDs ou outros parâmetros usados nos testes de carga, simulando diferentes cenários de uso. O JMeter consegue ler esses arquivos CSV e utilizar os dados durante a execução dos testes, permitindo variar os dados de entrada e criar testes mais detalhados e realistas.

**Jenkinsfile para integração contínua:** O Jenkinsfile é um arquivo de definição empregado no Jenkins, uma ferramenta popular para automação de processos de integração contínua e entrega contínua (CI/CD). Este arquivo, redigido em formato de script, tem o propósito de descrever e supervisionar o fluxo de trabalho, conhecido como pipeline, utilizado na integração contínua.

Geralmente inserido no repositório de código do projeto, o Jenkinsfile fornece instruções detalhadas para o Jenkins executar as etapas de construção, testes e implantação do software. Ele especifica os estágios, condições, ações e outras configurações do pipeline de CI/CD, oferecendo uma flexibilidade considerável para adaptar e personalizar o processo de desenvolvimento.

Existem duas maneiras principais de criar um Jenkinsfile:

Pipeline Scriptado: Usa uma linguagem de script baseada em Groovy, sendo altamente flexível, porém, pode demandar mais habilidade para gerenciar em comparação com a abordagem declarativa.

Pipeline Declarativo: Utiliza uma sintaxe estruturada e clara, tornando a leitura e a manutenção do pipeline mais simples, especialmente para pessoas menos familiarizadas com a escrita de scripts.

* **Estratégia de teste**

Uma estratégia destinada ao teste de software é um esquema meticuloso delineador que estipula a condução dos testes ao longo do ciclo de desenvolvimento de um programa computacional. Constitui uma sistemática abordagem que busca assegurar a qualidade intrínseca do software, identificando e remediando eventuais incongruências, com o fito último de assegurar que o produto final esteja em consonância com as expectativas delineadas pelos usuários.



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

**Funcionalidade:** Adicionar itens ao carrinho

Não é permitido inserir mais de 10 itens do mesmo produto ao carrinho

Os valores não podem ultrapassar a R$ 990,00

**Resumo:** Ao entrar na tela do produto no site da EBAC-shop

**Cenário:** Ao adicionar mais de 10 itens ao carrinho

**Quando** o usuário escolher a cor, tamanho e 10 itens

**E** Clicar em adicionar mais produtos do mesmo ao carrinho

**Então** a plataforma deve avisar o usuário com uma mensagem de erro "Apenas 10 itens do mesmo produto são permitos"

**Cenário:** Comprando um produto

**Dado** que o usuário tenha adicionado itens ao carrinho

**E** o preço da compra ultrapasse o valor de R$ 990,00

**Quando** clicar em "Concluir compra"

**Então** um pop-up de erro deve informá-lo "Preço máximo permitido de compra é de R$ 990,00"

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

**Funcionalidade:** Login na plataforma

Apenas usuários ativos podem fazer login na plataforma

Login deve permitir e-mail, nome de usuário ou CPF;

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

**Resumo:** Ao usuário acessar a página de login do site EBAC-shop

**Cenário:** autenticação válida

**Quando** o usuário digitar o email "joao@ebac.com.br"

**E** a senha "JoaobyerMyller123"

**Então** a plataforma deve encaminhá-lo ao perfil do usuário

**Cenário:** Erro de credenciais

**Quando** o usuário digitar o CPF ou senha incorreta

**E** tentar 3 ou mais vezes o login

**Então** o sistema deve negar o login desse ip por 15 minutos

* **História de usuário 3: [US-0003] – API de cupons**

**Funcionalidade:** Como admin da EBAC-SHOP  
 Quero criar um serviço de cupom  
 Para poder listar e cadastrar os cupons

**Resumo:** Ao usuário entrar no postman

**Cenário:** criar e salvar uma API

**Dado** que o usuário crie um novo trabalho no Postman

**E** use a função "GET" no swagger para pegar o url da EBAC-shop

**Quando** adicionar o código gerado no "GET" da EBAC

**E** o token para a autorização

**Então** ao salvar o projeto, deve ser criado um arquivo Json com os códigos da API

**Cenário:** criar teste de API

**Dado** que o usuário acesse a aba "test"

**E** use os parâmetros: method, url e headers

**Quando** denominar a mensagem do teste para "Deve listar o cupom cadastrado"

**E** rodar o teste

**Então** então uma mensagem de "Deve listar o cupom cadastrado" irá aparecer verde como "PASS" em "test results"

* **História de usuário 1:**

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

|  |  |  |
| --- | --- | --- |
| Regras | RN01 | Saída |
| Teste 1 | 10 itens do mesmo produto ou menos no carrinho | Válido |
| Teste 2 | Mais de 10 itens do mesmo produto adicionados no carrinho | Inválido |

RN02: Os valores não podem ultrapassar a R$ 990,00;

|  |  |  |
| --- | --- | --- |
| Regras | RN02 | Saída |
| Teste 1 | Valor do produto igual a R$ 1.500,00 | Inválido |
| Teste 2 | Valor do produto igual a R$ 300,00 | Válido |
| Teste 3 | Valor do produto igual a R$ 10,00 | Válido |
| Teste 4 | Valor do produto igual a R$ 989,99 | Válido |
| Teste 5 | Valor do produto igual a R$ 990,00 | Válido |
| Teste 6 | Valor do produto igual a R$ 990,01 | Inválido |

RN03: Valores entre R$ 200 e R$ 600, ganham cupom de 10%

|  |  |  |
| --- | --- | --- |
| Regras | RN03 | Saída |
| Teste 1 | Cadastro com o valor do produto R$ 199,99 | Inválido |
| Teste 2 | Cadastro com o valor do produto R$ 200,00 | Válido |
| Teste 3 | Cadastro com o valor do produto R$ 200,01 | Válido |
| Teste 4 | Cadastro com o valor do produto R$ 599,99 | Válido |
| Teste 5 | Cadastro com o valor do produto R$ 600,00 | Válido |
| Teste 6 | Cadastro com o valor do produto R$ 600,01 | Inválido |

RN04: Valores acima de R$ 600 ganham cupom de 15%

|  |  |  |
| --- | --- | --- |
| Regras | RN04 | Saída |
| Teste 1 | Cadastro com o valor de R$ 599,99 | Inválido |
| Teste 2 | Cadastro com o valor de R$ 600,00 | Inválido |
| Teste 3 | Cadastro com o valor de R$ 600,01 | Válido |

* **História de usuário 2:**

RN01 - Somente usuários ativos podem fazer login

RN02 - Deve exibir uma mensagem de erro caso o usuário erre o login e senha

RN03 - Login deve permitir e-mail, nome de usuário ou cpf

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Condições | Regra 1 | Regra 2 | Regra 3 | Regra 4 |
| Somente usuários ativos podem fazer login | 1 | 1 | 1 | 0 |
| Deve exibir uma mensagem de erro caso o usuário erre o login e senha | 1 | 1 | 1 | 0 |
| Login deve permitir e-mail, nome de usuário ou cpf | 1 | 0 | 1 | 0 |
| Se o usuário errar por 3 vezes a senha, deve travar por 15 minutos o login | 1 | 0 | 0 | 0 |
| Ação |  |  |  |  |
| Autenticação | 1 | 0 | 0 | 0 |

RN01 - Somente usuários ativos podem fazer login

|  |  |  |
| --- | --- | --- |
| Regras | RN01 | Saída |
| Teste 1 | Tentativa de login de um usuário ativo | Válido |
| Teste 2 | Tentativa de login de um usuário não ativo | Inválido |

RN02 - Deve exibir uma mensagem de erro caso o usuário erre o login e senha

|  |  |  |
| --- | --- | --- |
| Regras | RN02 | Saída |
| Teste 1 | Mensagem de erro "Login ou Senha incorreta, por favor, tente novamente com novas credenciais" aparece ao usuário errar login ou senha | Válido |
| Teste 2 | Mensagem de erro "Login ou Senha incorreta, por favor, tente novamente com novas credenciais" não aparece ao usuário errar login ou senha | Inválido |

RN03 - Login deve permitir e-mail, nome de usuário ou cpf

|  |  |  |
| --- | --- | --- |
| Regras | RN03 | Saída |
| Teste 1 | Login com Email | Válido |
| Teste 2 | Login com nome de usuário | Válido |
| Teste 3 | Login com CPF | Válido |
| Teste 4 | Login com número de celular | Inválido |

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

|  |  |  |
| --- | --- | --- |
| Regras | RN04 | Saída |
| Teste 1 | Tentativa de login de um usuário ativo | Válido |
| Teste 2 | Tentativa de login de um usuário não ativo | Inválido |

* **História de usuário 3: API de Cupom**

RNO1 - Valor mínimo do cupom = R$ 10,00  
 RNO2 - Nome do cupom não pode ser repetido  
 RNO3 - Tipo do desconto: “fixed\_product”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Regras | RN01 | RN02 | RN03 | RN04 |
| Valor mínimo do cupom = R$ 10,00 | 1 | 1 | 0 | 0 |
| Nome do cupom não pode ser repetido | 1 | 1 | 1 | 0 |
| Tipo do desconto: “fixed\_product” | 1 | 0 | 1 | 0 |
| Ações |  |  |  |  |
| cadastrar os cupons | 1 | 0 | 0 | 0 |

* **Repositório no Github**

<https://github.com/igorgalan98/TCC-EBAC>

* **CONCLUSÃO**

O propósito deste projeto não se limita apenas à explanação e distinção de diversos métodos de teste, mas também à sua aplicação prática, visando não apenas à demonstração, mas também ao aprendizado significativo. A oportunidade de realizar este curso e projeto foi um privilégio que me permitiu aprender não somente sobre as técnicas de teste, mas também a identificar fontes relevantes de pesquisa e informação. Esta habilidade é considerada um dos aspectos mais valorosos no papel de um profissional de Garantia de Qualidade (QA), pois a resolução de problemas complexos exige paciência e a capacidade de investigar a fundo, buscando soluções a partir de uma variedade de fontes, tais como pesquisas e referências bibliográficas.

É fundamental estar ciente de que a tecnologia está em constante evolução, o que demanda uma postura de mente aberta para não se apegar a métodos ultrapassados. A mudança, muitas vezes, não é resultado de falhas, mas sim do progresso contínuo do setor. Lidar com a identificação de erros, a escrita de códigos e a apresentação de trabalhos requer diferentes abordagens, e superar esses desafios envolve persistência ao buscar informações, seja através da leitura, pesquisa ou visualização de materiais como vídeos, visando alcançar soluções eficazes.

* **REFERÊNCIAS BIBLIOGRÁFICAS**

Caroli, Paulo. O que faz uma pessoa QA?. caroli.org, 23/05/2023.

Disponível em: <<https://caroli.org/o-que-faz-um-qa/>>. Acesso em: 18/12/2023