Sumário

1 . Introdução.

As atividades de um **QA**, são divididos em **2** modalidades fundamentais para a validação dos requisitos de um software. São eles: **Teste Manual** e **Automatização de Teste**.

Cada um deles é utilizado de forma genérica e particular de acordo com a arquitetura do software.

1 .1 Ferramentas: Testes Manuais.

Essa modalidade consiste em fazer a validação dos requisitos do sistema de forma manual. Usando ferramentas como: '**Postman**'; '**Selenium**'; e qualquer outro software que permita fazer requisições através do micro-serviço '**API**' do sistema.

Os testes são diversificados de acordo com as regras de negocio. Por muitas das vezes mais comum, **por exemplo**: validar um campo **CPF** inserindo caracteres.

Por outras vezes, em uma infra-estrutura **web** ou **móbile**. Os testes são validados manualmente. **Por exemplo**: Se para tal componente, executa sua real função como foi documentada e especificada.

1 .2 Ferramentas: Automatização de Teste.

Essa modalidade consiste em fazer a validação dos requisitos do sistema de forma automatizada, através de scripts ou de uma ferramenta bem robusta para automatização. Sendo assim, a ferramenta sai executando tudo aquilo que foi programado dentro daquele parâmetro para os cenários de teste elaborados.

As ferramentas mais comuns para testes automatizados para **Web são**: '**Cypress'**; '**Opentest'**; '**Apium'**; '**WebDriver'** e etc...

Outras ferramentas para **back-end**, exclusivamente para testes unitários, cobertura de teste e teste de unidade em **Java** são: '**J.Unit'**; '**Mockito**' e etc...

2 . Escrita de cenários.

Em projetos grandes, costumam utilizar tipos de abordagem de teste, cuja finalidade é focada diante o desenvolvimento de software, com intuito de entregar um produto responsivo e de qualidade.

Os tipos são **TDD** (Test-Driven Development)e **BDD** (Behavior-Driven Development).

- **TDD**: Se concentra na escrita de testes unitários para validar o comportamento de unidades individuais de código (métodos, classes, etc...)

- **BDD**: Se concentra na escrita de testes que descrevam o comportamento do sistema a partir da perspectiva do usuário ou do negocio. Conhecido também, como *Cucumber* ou *Gherkin*, geralmente são escritos com linguagem natural. Segue as principais termologias sobre essa metodologia: **GIVEN**; **WHEN**; **THEN**; **AND**; **OR**;

3 . Relatório cenário de teste.

ID	Caso de Teste	Observação
CT01. 00	Validar cadastro de usuário.	PASSED
CT01. 01	Validar campos de login.	EM TESTE
CT02. 00	Validar mensagem de entrada login.	PASSED
CT03. 00	Validar campos de informações do usuário.	ADIADO
CT03. 01	Validar campo layout de pagina.	FALHA
CT03. 02	Validar padrão de cores.	PASSED
CT03. 03	Validar navegação de telas.	PASSED

CT01. 00	Validar cadastro de usuário
Resultado do teste	PASSED
Versão	0.001

Given ao preencher todos os campos de cadastro
And que as informações sejam coerentes com as instruções de casdastro
Then clico no botão de cadastrar
When deverá mostrar uma mensagem de cadastrado com sucesso.

Request	Response

CT01. 01	Validar campos de login.
Resultado do teste	EM TESTE
Versão	0.001

Given que já tenho uma conta cadastrada no sistema
And insiro no campo ‘account’ com uma conta válida
And insiro campo ‘password’ com uma senha válida
Then clico no botão “entrar”
And os campos: ‘account’ e ‘password’ estiverem simultaneamente preenchidos
When o sistema vai liberar acesso e irá navegar entre o painel principal do usuário.

Request	Response

CT02. 00	Validar mensagem de entrada login.
Resultado do teste	PASSED
Versão	0.001

Given
Then
When

Request	Response