

Testes Automatizados

Fundação Centro Universitário Estadual da Zona Oeste – UEZO
Graduação Ciências da Computação

Testes Automatizados
Calculadora Desenvolvida na Linguagem Java

Discente
Ana Claudia Gomes Souza

Rio de Janeiro
2018

Sumário

1 – Definição Teste Automatizado	4
2 – Apresentação da Ferramenta Junit.....	6
3 – Plano de Teste.....	8
4 –Bibliografia.....	12

1 – Definição Teste Automatizado

Testes automatizados são técnicas que utilizamos para avaliar o comportamento de um sistema / aplicação. Desta forma, podemos dectar erros de execução e consequentemente trabalharmos na correção dos erros e em melhorias do produto.

Existem várias ferramentas para implementação dos testes automatizados para cada nível de teste. Veremos a seguir os principais tipos de testes e suas respectivas ferramentas.

Testes unitários

Pertencem ao primeiro nível de teste sendo responsável por avaliar o software como unidades independentes do produto. Suas principais ferramentas de acordo com a linguagem de programação são:

Linguagem	Ferramentas
Java	Junit
.NET	Visual Studio Unit Testing Framework
JavaScript	Unit.js
PHP	PHPUnit
Python	PyUnit(Unittest)

Teste de integração

Representam o segundo nível de teste sendo responsável por verificar se os componentes montados se comportam como uma unidade coesa. Suas principais ferramentas implementadas em APIs e banco de dados são:

Aplicação	Ferramenta
Banco de dados	DB Unit
APIs	Postman

Testes de aceitação

Último nível de teste antes do software ser direcionado para produção. São testes de caixa preta que executam a jornada completa do usuário com o software.

Ferramentas para teste de aceitação aplicado na interface

Ambiente	Ferramenta
Web	Selenium
Android	Espresso
IOS	Xcode Unit Tests
Windows	Visual Studio Coded Ui Test

Ferramentas para testes de aceitação

Linguagem	Ferramenta
Java	Cucumber
.NET	SpecFlow
JavaScript	Jasmine
PHP	PHPSpec
Python	Behave

Além destes testes, temos os testes não funcionais que garantem que o produto cumprirá o propósito para o qual foi desenvolvido.

Ambiente	Ferramenta
Apache	Jmeter

2 – Apresentação da Ferramenta Junit

Junit

É um framework open-souce desenvolvido por Erich Gamma e Kent Beck para testes automatizados na linguagem Java.

Com esta ferramenta podemos verificar se cada método de uma classe funciona de forma esperada indicando possíveis erros ou falhas. Podendo ser executado como baterias de testes como para extensão.

O Junit é aplicado nos testes utomatizados de unidade que são responsáveis por testar o menor dos componentes de um sistema de maneira isolada. Cada uma das unidades define um conjunto de estímulos (chamada de metodos), e de dados de entrada e saída associados a cada estimulo.

Algumas vantagens de usar o Junit:

- 1- Permite a criação de código de teste enquanto possibilita um aumento na quantidade do sistema sendo desenvolvido e testado;
- 2- Não é necessário escrever o próprio framework;
- 3- Uma vez escrito os testes são executados rapidamente sem que, para isso seja interrompido o processo de desenvolvimento;
- 4- Junit verifica os resultados dos testes e fornece uma resposta imediata;
- 5- Pode-se criar uma hierarquia de testes que permitirá testar apenas uma parte do sistema ou todo ele.

Para realizarmos os testes é preciso declarar uma dos seguintes métodos: @Test, @After, @Before, @AfterClass, @BeforeClass ou @Ignore

@Test

Informa ao Junit que o método public void ao qual está conectado pode ser executado como caso de teste.

@After

É um método de teste que será executado depois do método @Test

@Before

Vários teste precisam de objetos semelhantes criados antes de poderem ser executados. O método @Before faz com que o mesmo seja executado antes de cada método@Test

@BeforeClass

É executado uma vez antes de qualquer um dos métodos de teste na classe

@AfterClass

É executado depois de todos os testes terem terminado. Podendo ser usado para executar atividades de limpeza

@Ignore

É usado quando queremos ignorar um teste fazendo com o que o mesmo não seja executado

3 – Plano de Teste

Para cada caso de teste foi implementado o teste de unidade verificando a funcionalidade de todas as operações matemáticas que a calculadora irá operar.

A ferramenta utilizada nos testes de unidade foi o Junit porque o software foi desenvolvido na linguagem de programação java.

O método @Before foi utilizado para verificar se as variáveis foram inicializadas com zero

O método @Test foi utilizado para verificar se o valor retornador pela função era um valor correto

O método @After foi utilizado após o método @Test para verificar se após a execução das operações matemáticas as variáveis possuíam valor zero

Caso de Teste	CT 001 – Operação soma
Pré-condições	Ter inicializado a calculadora
Procedimento	1) O ator digita o primeiro valor 2) O ator seleciona a opção soma 3) O ator digita o segundo valor 4) O ator seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor da soma de dois números
Dados de entrada	Dois números do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1º iteração

Caso de Teste	CT 002 – Operação subtrair
Pré-condições	Ter inicializado a calculadora
Procedimento	1) O ator digita o primeiro valor 2) O ator seleciona a opção subtrair 3) O ator digita o segundo valor 4) O ator seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor da subtração de dois números
Dados de entrada	Dois numeros do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1º iteração

Caso de Teste	CT 003 – Operação multiplicação
Pré-condições	Ter inicializado a calculadora
Procedimento	1) O ator digita o primeiro valor 2) O ator seleciona a opção multiplicação

	3) O ator digita o segundo valor 4) O ator seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor da multiplicação de dois números
Dados de entrada	Dois números do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1º iteração

Caso de Teste	CT 004 – Operação divisão
Pré-condições	Ter incialiado a calculadora
Procedimento	1) O ator digita o primeiro valor 2) O ator seleciona a opção divisão 3) O ator digita o segundo valor 4) O ator seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor da soma de dois números
Dados de entrada	Dois números do tipo double
Critérios especiais	Caso o numerador for valor zero será impresso uma mensagem informando que este valor não é divisível e solicita que o ator digite um novo valor para numerador
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1º iteração

Caso de Teste	CT 005 – Operação valor máximo entre dois números
Pré-condições	Ter incialiado a calculadora
Procedimento	1) O ator digita o primeiro valor 2) O ator seleciona a opção valor máximo 3) O ator digita o segundo valor 4) O ator digita seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor do maior valor entre dois números
Dados de entrada	Dois números do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1º iteração

Caso de Teste	CT 006 – Operação valor mínimo entre dois números
---------------	---

Pré-condições	Ter incialiado a calculadora
Procedimento	<ol style="list-style-type: none"> 1) O ator digita o primeiro valor 2) O ator seleciona a opção menor valor 3) O ator digita o segundo valor 4) O ator digita seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor do menor número entre dois números
Dados de entrada	Dois números do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1° iteração

Caso de Teste	CT 007 – Operação Logaritmo
Pré-condições	Ter incialiado a calculadora
Procedimento	<ol style="list-style-type: none"> 1) O ator digita o valor 2) Seleciona a opção logaritmo 3) Seleciona a opção de igual para obter o resultado da operação
Resultado esperado	Impressão do valor do logaritmo
Dados de entrada	Um valor do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1° iteração

Caso de Teste	CT 008 – Operação raiz quadrada
Pré-condições	Ter incialiado a calculadora
Procedimento	<ol style="list-style-type: none"> 1) O ator digita o valor 2) Seleciona a opção raiz quadrada 3) Seleciona o sinal de igual para obter o resultado da operação
Resultado esperado	Impressão do valor da raiz quadrada
Dados de entrada	Um valor do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1° iteração

Caso de Teste	CT 009 – Operação potenciação
Pré-condições	Ter incialiado a calculadora
Procedimento	<ol style="list-style-type: none"> 1) O ator digita o primeiro valor 2) O ator seleciona a opção potencia 3) O ator digita o segundo valor 4) O ator digita seleciona a opção igual para obter o resultado da operação

Resultado esperado	Impressão do valor da potência de dois números
Dados de entrada	Dois números do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1° iteração

Caso de Teste	CT 010 – Operação seno
Pré-condições	Ter incialiado a calculadora
Procedimento	<ol style="list-style-type: none"> 1) O ator um valor 2) Seleciona a opção seno 3) Seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor do seno
Dados de entrada	Um número do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1° iteração

Caso de Teste	CT 011 – Operação cosseno
Pré-condições	Ter incialiado a calculadora
Procedimento	<ol style="list-style-type: none"> 1) O ator digita um valor 2) Seleciona a opção cosseno 3) Seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor da soma de dois numeros
Dados de entrada	Um número do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1° iteração

Caso de Teste	CT 012 – Operação cosseno trigonométrico
Pré-condições	Ter incialiado a calculadora
Procedimento	<ol style="list-style-type: none"> 1) O ator digita um valor 2) Seleciona a opção cosseno 3) Seleciona a opção igual para obter o resultado da operação
Resultado esperado	Impressão do valor da soma de dois numeros
Dados de entrada	Um número do tipo double
Critérios especiais	Não se aplica
Ambiente	Windows, netbeans
Implementação	Manual
Iteração	1° iteração

4 –Bibliografia

- ✓ Conceitos sobre os métodos @Test, @After, @Before, @AfterClass , @BeforeClass e @Ignore
https://www.tutorialspoint.com/junit/junit_using_assertion.htm
- ✓ Conceitos sobre testes automatizados
<https://www.knowledge21.com.br/blog/qualidade-do-software-testes-automatizados/>
- ✓ Conteitos sobre a ferramenta Junit
<https://pt.wikipedia.org/wiki/JUnit>