

PLANO DE TESTE

NutriVitta

Sistema de cálculo de IMC (Índice de Massa Corporal)

Aluno: William Rosa

Turma: Full-Stack 4

Modelo

Teste de software de sistema de cálculo de IMC (Índice de Massa Corporal), composto pelas seguintes regras pré-estabelecidas:

- Realização do cálculo do IMC;
- Classificação do IMC de acordo com a tabela da Abeso;

TABELA DE CONTEÚDOS

1 - INTRODUÇÃO

2 - OBJETIVOS

3 - ESCOPO

4 - REQUISITOS DE TESTES

5 - ESTRATÉGIAS DE TESTE

6 - FERRAMENTAS

7 - EQUIPE

8 - SISTEMA

9 - CRONOGRAMA

10 - RESULTADOS DO TESTE

1 - INTRODUÇÃO

Este documento de teste descreve as estratégias, processos, fluxos de trabalho e metodologias adequadas utilizadas para planejar, organizar, executar e gerenciar testes do projeto de software da Calculadora IMC (CIMC) para a empresa NutriVitta.

O Plano de Teste foi criado para comunicar a abordagem de teste aos membros da equipe. Inclui os objetivos, escopo, cronograma, riscos e abordagem. Este documento identifica claramente quais serão os resultados do teste e o que é considerado dentro e fora do escopo.

Este plano tem como finalidade testar o software para a empresa NutriVitta, que consiste em um sistema de Calculadora IMC.

2 - OBJETIVOS

2.1 - Objetivo primário:

Um objetivo primordial dos sistemas de aplicação de teste é assegurar que o sistema atenda aos requisitos completos, incluindo requisitos de qualidade (também conhecidos: requisitos não funcionais) e ajustar métricas para cada requisito de qualidade e satisfazer os cenários de caso de uso e mantenha a qualidade do produto.

Ao final do ciclo de desenvolvimento do projeto, o usuário deve descobrir que o projeto atende ou excede todas as suas expectativas conforme detalhado nos requisitos.

Quaisquer alterações, adições ou exclusões ao documento de requisitos, serão documentadas e testadas no mais alto nível de qualidade permitido dentro do tempo restante do projeto e dentro da capacidade da equipe de teste.

2.2 - Objetivo secundário:

O objetivo secundário dos sistemas de aplicação de teste será: identificar e expor todos os problemas e riscos associados, comunicar todos os problemas conhecidos à equipe do projeto e garantir que todas as questões sejam tratadas antes da liberação.

Como objetivo, isso requer testes cuidadosos e metódicos da aplicação para primeiro garantir que todas as áreas do sistema sejam examinadas e, conseqüentemente, todas as questões (bugs) encontradas sejam tratadas adequadamente.

3 - ESCOPO

Espera-se que o software CIMC consiga fazer cálculos de IMC utilizando a seguinte critério: O IMC é calculado dividindo o peso (em kg) pela altura ao quadrado (em metros).

O teste fará a verificação das funcionalidades da aplicação, certificando-se de que elas apresentam o comportamento esperado e/ou identificando possíveis erros do sistema, através da simulação com dados fictícios para verificar as funcionalidades de cálculo e classificação.

Assim o software receberá dois números. primeiroNum (altura) deverá ser multiplicado por dois e o resultado será dividido pelo segundoNum (peso). O resultadoDiv (resultado) será o IMC.

4 - REQUISITOS DE TESTE

Para a execução do teste será necessário informar dois números que correspondem a altura (primeiroNum) e peso (segundoNum). O teste será dividido em três etapas.

1. Assim o software receberá primeiroNum deverá ser multiplicado por ele mesmo e o valor será o resultado esperado.

2. O software receberá segundoNum este dividido pelo resultado da multiplicação do primeiroNum e apresentar o resultado esperado.

3. Testes de erros, também deverão ser informados caracteres errados (segundo Estratégias de Teste) para simular erros ao usuário.

Durante o teste os resultados deverão ser anotados no item 10 (Resultados do Teste) deste documento.

5 - ESTRATÉGIAS DE TESTE

Abaixo estão algumas etapas para que sejam executados os testes. O tester tem liberdade para usar qualquer número para execução do teste. É necessário que façam as devidas anotações dos números e resultados.

Etapas:

1. Teste de multiplicação: Por exemplo, suponha-se que o primeiroNum seja 1.70 espera que o resultado da multiplicação seja a saída de 2.89. Para esse teste usa-se números 0 a 3 com casa decimais separada por . (ponto)

O resultado deverá sempre ser o valor do número multiplicado por ele mesmo.

2. Teste de divisão: Por exemplo, suponha-se que o segundoNum seja 70 espera-se que o resultado da divisão seja a saída de valor 24,22. Para esse teste usa-se números 0 a 4 com casa decimais separada por . (ponto) O resultado deverá sempre ser resultado da multiplicação (teste 1) dividido pelo número informado.

3. Teste de erros. Para erros deverão ser informados números diferentes dos mencionados acima bem como uso de caracteres especiais, números separados por vírgula e letras. O resultado deverá ser erro no programa. Durante o teste os resultados deverão ser anotados no item 10 (Resultados do Teste) deste documento.

6 – FERRAMENTAS

As ferramentas necessárias para execução deste teste são:

1. OS Windows ou macOS;
2. Microsoft Visual Studio;
3. Linguagem C#;
4. Xunit

7 - EQUIPE

O responsável pela elaboração desse documento, e também pelo desenvolvimento e execução do teste está a cargo de William Rosa.

8 – SISTEMA

O sistema a ser usado para elaboração e execução desse plano de teste é notebook com sistema operacional Windows, usando a ferramenta Microsoft Visual Studio, com a plataforma de desenvolvimento .NET e a linguagem de programação C#, com a ferramenta Xunit.

9 – CRONOGRAMA

Data	Teste	Finalização
01/11/2022	Divisão	01/11/2022
01/11/2022	Multiplicação	01/11/2022
01/11/2022	Erros	01/11/2022

10 – RESULTADOS

Teste de Divisão

Data de execução	Tester	Nº inseridos	Resultados

Teste de Multiplicação

Data de execução	Tester	Nº inseridos	Resultados

Teste de Erros

Data de execução	Tester	Nº inseridos	Resultados