

PLANO DE TESTE

NutriVitta
Sistema de cálculo de IMC
(Índice de Massa Corporal)

Aluno: Douglas Maciel
Turma: Full-Stack 1

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
11	CONCLUSÃO

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

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. 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, consequentemente, 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. **primeiroNumero** (altura) deverá ser multiplicado por dois e o resultado será dividido pelo **segundoNumero** (peso). O **resultadoDaDivisao** (resultado) será o IMC.

4 REQUISITOS DE TESTE:

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

1. Assim o software receberá **primeiroNumero** deverá ser multiplicado por ele mesmo e o valor será o resultado esperado.
2. O software receberá **segundoNumero** este dividido pelo resultado da multiplicação do **primeiroNumero** 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 testador 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 **primeiroNumero** seja **1.70** espera-se 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 **segundoNumero** 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 Douglas Maciel.

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	Data finalização
14/07/2022	Multiplicação	14/07/2022
14/07/2022	Divisão	14/07/2022
14/07/2022	Erros	14/07/2022

10 RESULTADOS

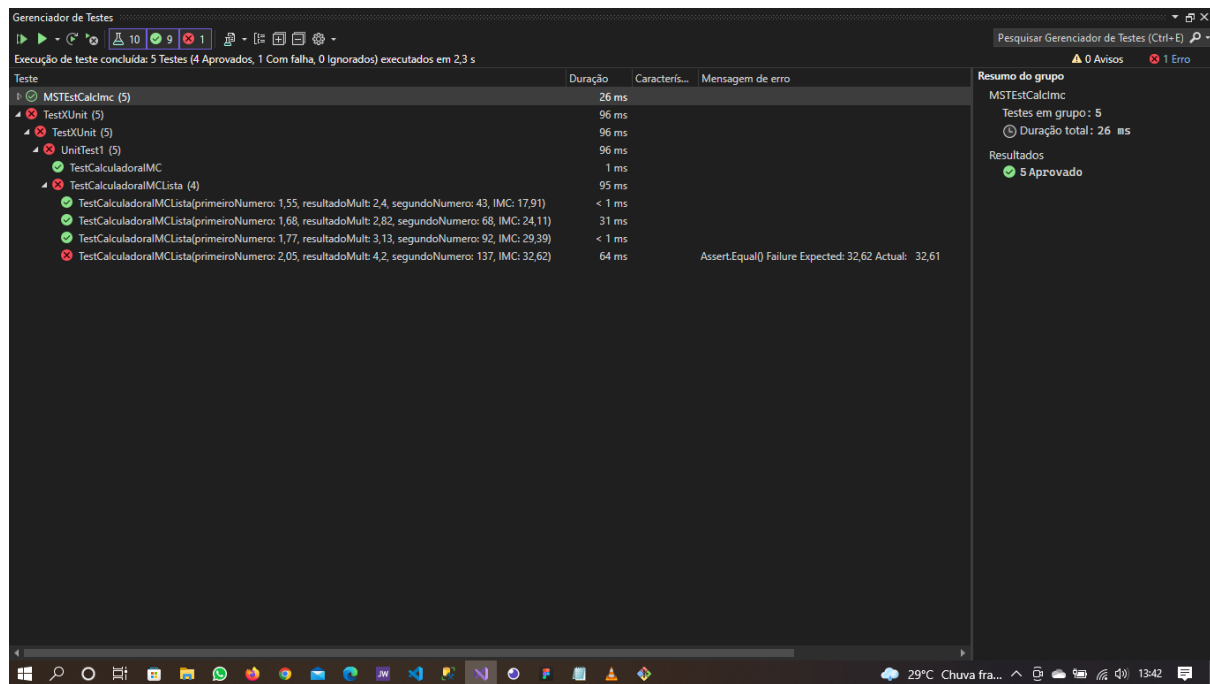
Teste Multiplicação

Data execução	Nome testador	Número inserido	Saída do resultado
14/07/2022	Douglas M	1.91	3.64
14/07/2022	Douglas M	1.68	2.82
14/07/2022	Douglas M	1.55	2.40
14/07/2022	Douglas M	2.05	4.20
14/07/2022	Douglas M	1.77	3.13

Teste Divisão

Data execução	Nome testador	Números inseridos	Saída do resultado
14/07/2022	Douglas M	2.82 -	25.54
14/07/2022	Douglas M	2.82 - 68	24.11
14/07/2022	Douglas M	4.20 - 43	17.91
14/07/2022	Douglas M	4,20 - 137	32.62 error o correto seria 32.61
14/07/2022	Douglas M	3.13 - 92	29.39

Print do teste Multiplicação e Divisão:



Teste de Erros

Data execução	Nome testador	Caracteres inseridos	resultado esperado	Saída do resultado
14/07/2022	Douglas M	1.688 - 68	2.84	2.849344
14/07/2022	Douglas M	1.51 - 41	2.41	2.28
14/07/2022	Douglas M	-2.05 - 137	32.61	32,61 resultado correto
14/07/2022	Douglas M	1*77 - 93	29.39	5929
14/07/2022	Douglas M	-1.77 - 9*3	29.39	3.13

Print do teste de erros:

Pesquisar Gerenciador de Testes (Ctrl+E)

Execução de teste concluída: 10 Testes (6 Aprovados, 4 Com falha, 0 Ignorados) executados em 2,1 s

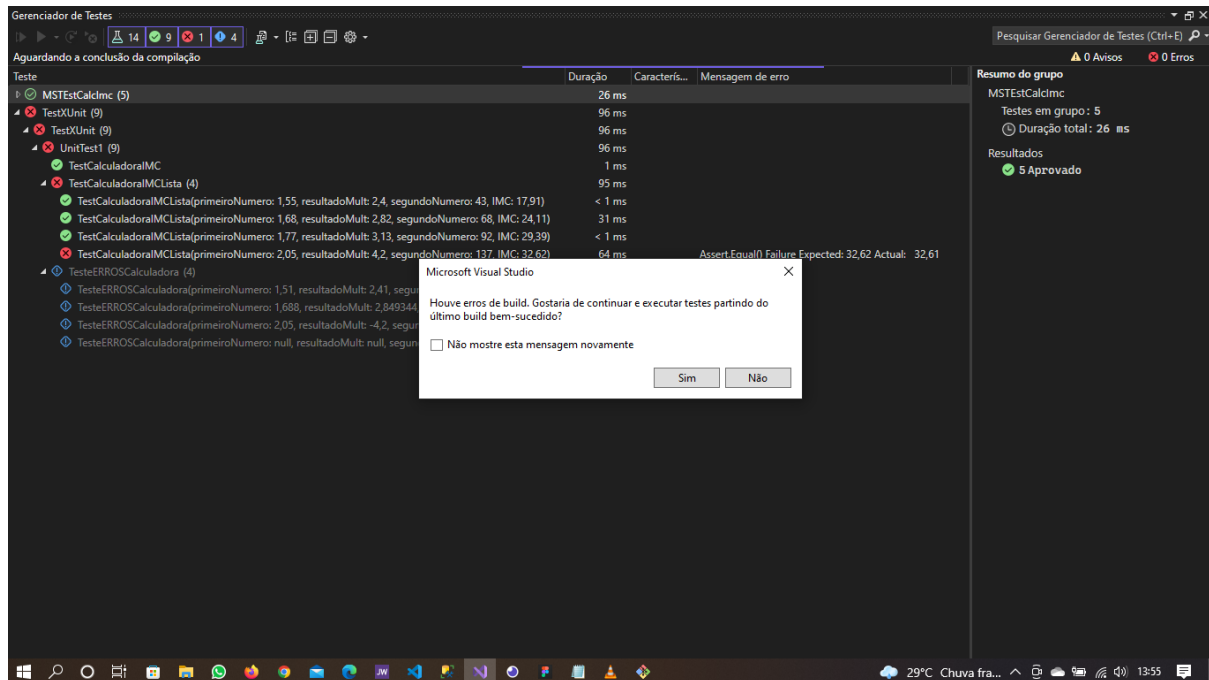
Teste	Duração	Caracteris...	Mensagem de erro
MSTestCaldmc (5)	26 ms		
TestXUnit (10)	70 ms		
TestXUnit (10)	70 ms		
UnitTest1 (10)	70 ms		
TestCalculadoraIMC	< 1 ms		
TestCalculadoraIMCLista (4)	4 ms		
TestCalculadoraIMCLista(primeiroNumero: 1,55, resultadoMult: 2,4, segundoNumero: 43, IMC: 17,91)	< 1 ms		
TestCalculadoraIMCLista(primeiroNumero: 1,68, resultadoMult: 2,82, segundoNumero: 68, IMC: 24,11)	4 ms		
TestCalculadoraIMCLista(primeiroNumero: 1,77, resultadoMult: 3,13, segundoNumero: 92, IMC: 29,39)	< 1 ms		
TestCalculadoraIMCLista(primeiroNumero: 2,05, resultadoMult: 4,2, segundoNumero: 137, IMC: 32,61)	< 1 ms		
TesteERROSCalculadora (5)	66 ms		
TesteERROSCalculadora(primeiroNumero: 1,51, resultadoMult: 2,41, segundoNumero: 41, IMC: 17,91)	< 1 ms		Assert.Equal() Failure Expected: 2,41 Actual: 2,28
TesteERROSCalculadora(primeiroNumero: 1,688, resultadoMult: 2,849344, segundoNumero: 68, IMC: ...)	37 ms		Assert.Equal() Failure Expected: 2,849344 Actual: 2,84
TesteERROSCalculadora(primeiroNumero: -1,77, resultadoMult: 3,1, segundoNumero: 27, IMC: 29,39)	< 1 ms		Assert.Equal() Failure Expected: 3,1 Actual: 3,13
TesteERROSCalculadora(primeiroNumero: -2,05, resultadoMult: 4,2, segundoNumero: 137, IMC: 32,61)	29 ms		
TesteERROSCalculadora(primeiroNumero: 77, resultadoMult: 3,13, segundoNumero: 93, IMC: 29,39)	< 1 ms		Assert.Equal() Failure Expected: 3,13 Actual: 5929

Resumo do grupo
MSTestCaldmc
Testes em grupo: 5
Duração total: 26 ms
Resultados
5 Aprovado

11 Conclusão:

O programa funciona conforme esperado. Consegue fazer cálculos de multiplicação e divisão, e dá o valor correto do IMC.

Porém não aceita caracteres, chegando a não rodar o programa, dando erro de build, conforme a imagem abaixo.



Foi uma surpresa no teste de erro dar um resultado positivo, o qual será necessário fazer análise do porque o teste foi bem sucedido.

Data: 14 de Junho de 2022

Nome testador: Douglas Maciel

Assinatura: _____