Plano de Teste

Função de login - ApiChapter

Adriano de Jesus Reis da Silva

Programador Back-End Senai

Gestor: Matheus de Holanda

Brasília 06/02/2023

Índice

ÍNDICE

1. INTRODUÇÃO....................................................................................................... 2
   1. PROPÓSITO DO SISTEMA............................................................................... 3
   2. REFERÊNCIAS/DOCUMENTAÇÃO DO PROJETO............................................. 4
2. ABORDAGEM DE TESTES...................................................................................... 5
   1. Categorização dos requisitos em funcionais e não funcionais...................... 5
   2. Detalhamento da abordagem de teste.......................................................... 6
   3. Ferramentas................................................................................................... 6
3. Ambiente de teste............................................................................................... 7
   1. Definições do ambiente de teste................................................................... 7
4. Programação dos testes...................................................................................... 7
   1. Objetivos e prioridades.................................................................................. 7
5. Casos de teste...................................................................................................... 7

1.INTRODUÇÃO

* 1. Propósito do Sistema

A solução é desenvolver uma interface (API) para disponibilizar dados que serão apresentados e utilizados pela equipe de mobile.

A solução terá as funções de ler, alterar, buscar e deletar dados.

A solução terá a função de cadastrar usuário e realizar login.

* 1. Referências /Documentação do projeto

A tabela abaixo identifica a documentação utilizada para a elaboração deste plano de teste:

|  |  |  |
| --- | --- | --- |
| Documento | Fará parte do projeto? | Obeservações |
| Especificação de Requisitos | Sim | Define funcionalidades do sistema |
| Especificação de casos de uso | Sim | Descreve as funcionalidades do sistema |

2.ABORDAGEM DE TESTES

2.1 Categorização de requisitos em funcionais e não funcionais

|  |  |
| --- | --- |
| Requisito funcional | Requisito não funcional |
| RF001 – O sistema deve realizar cadastro de usuários. | NF001 – O sistema deve ser desenvolvido em linguagem de programação C#. |
| RF002 – O sistema deve realizar login de usuários. | NF002 – O sistema deve ser compatível com os sistemas operacionais Windows e Linux. |

2.2 Detalhamento da abordagem de teste

|  |  |
| --- | --- |
| Tipo de teste: | Funcional |
| Subtipo de teste: | Requisitos |
| Objetivo do teste: | Testar a funcionalidade de cadastro de usuários. |
| Requisitos que motivaram o teste: | RF001 - O sistema deve realizar o cadastro de usuários. |

|  |  |
| --- | --- |
| Tipo de teste: | Funcional |
| Subtipo de teste: | Requisitos |
| Objetivo do teste: | Testar a funcionalidade de login de usuários cadastrados |
| Requisitos que motivaram o teste: | RF002 – O sistema deve efetuar login somente de usuários cadastrados. |

2.3 Ferramentas

As ferramentas que serão empregadas neste projeto de testes:

|  |  |  |
| --- | --- | --- |
| Ferramenta | Fabricante | Versão |
| MsTest | Microsoft | Mstest |
| XUnit | Microsoft | XUnit |
| Moq | Daniel Cazzulino | 4.18.4 |
|  |  |  |
|  |  |  |

3.AMBIENTE DE TESTE

3.1 Definições do Ambiente de Teste

* Teste unitário desenvolvidos por desenvolvedores com um volume pequeno de dados.
* O sistema deverá ser desenvolvido em linguagem C#.

4.PROGRAMAÇÃO DOS TESTES

4.1 Objetivos e Prioridades

As prioridades do sistema são:

* O sistema deve verificar se o usuário está cadastrado
* O sistema deve verificar o token de acesso para validar login

5. CASOS DE TESTE

|  |  |  |  |
| --- | --- | --- | --- |
| Caso de uso | ID | Passos | Resultado esperado |
| RF001 | 1 | Criar usuário fake usando a ferramenta Moq e testar validação. | Verificar se o usuário é valido |
| RF002 | 1 | Criar token fake usando a ferramenta Moq para verificar o token de acesso para usuário. | Confirma se o token gerado é compatível com o token fake. |