|  |
| --- |
| AMF_1  TAP-2016 |

DUM – Documento Único de Manutenção

1. Levantamento dos Requisitos
   * 1. Cenário Atual

|  |
| --- |
| **Cenário Atual – Definição do Problema** |
| Criar um sistema em que apresente a comparação entre variados algoritmos de ordenação. |

1. Detalhamento Funcional

|  |  |  |  |
| --- | --- | --- | --- |
| **ID Req.** | **ID**  **Func.** | **Funcionalidade** | **Descrição – Detalhamento Funcional** |
| 1 | 1.1 | Interface e definições do programa | **O programa foi desenvolvido em C# Windows Forms.**  **O Programa possui uma interface bem simples com um campo *textbox* para o usuário informa a quantidade de números a serem gerados, um *button* que executa a ação de gerar números aleatórios, um *listbox* que mostra os números após serem ordenados, uma *label*  que mostra o tempo demorado para fazer a ordenação e 10 *buttons* , onde cada um executa um método de ordenação diferente.** |
| 2 | 2.1 | Funcionamento do Programa  **Campo “Quantidade de Números”** | * **Ao Ser executado o usuário terá somente o campo “Quantidade de Números” disponível para executar uma informação.** * **Para liberar os outros botões no sistema, o usuário deve inserir no campo somente números inteiros maiores que zero.** * **Além disso, o campo só permite digitar números.** * **Se a tecla enter for pressionada ele chamará o método do botão “Gerar”;** |
| 2 | 2.2 | Funcionamento do Programa  **Botão “Gerar”** | * **O Botão “Gerar” irá gerar números aleatórios que começam em 0 e vai até o número digitado no campo “Quantidade de Números”, podendo se repetir.** |
| 2 | 2.3 | Funcionamento do Programa  **Botões de Ordenação** | * **Cada Botão na tela representa um método de ordenação.** * **Ao ser clicado deve executar o método de ordenação que representa.** * **O resultado da ordenação será exibido no campo “Resultado” junto com o tempo gasto pela ordenação.** * **Será Mostrado No Título do Programa o Método selecionado junto com seu tempo de execução.** |

1. Detalhamento Técnico
   1. Métodos de Ordenação Utilizados

**3.1.1 Insertion Sort**

Esse método de ordenação é um dos mais simples que existe. Ele ordena os elementos um por um comparando-os.

Suas principais características são:

* Bastante eficiente para ordenação com poucos elementos, porém muito ineficiente para listas com muitos elementos.
* Ele realiza a ordenação por inserção adaptativa, o que significa que terá um número de passos reduzidos à medida que tem os números da lista já estarem parcialmente ordenados.
* Sua complexidade é baixa.
* É estável, uma vez que não se altera a ordem relativa dos elementos com chaves iguais.

Análise de Complexidade:

Pior Caso: O(n2)

Melhor Caso: O(n)

Caso Mediano: O(n2)

Complexidade de Espaço: O(1)

Comparação entre Listas com Tamanho Diferentes:







