**Trabalho N1 – 2º bimestre – VERSÃO 1.0**

* Estrutura de dados
* Programação Orientada a Objeto

Cadastro e player de Mídias.

* Trabalho deverá ser feito em dupla.
* Desenvolver um sistema (Windows Forms ou WPF). O objetivo será analisar a capacidade do aluno de criar e manipular classes e objetos utilizando os conceitos de orientação a objetos, além de estruturas de dados.
* O Trabalho desenvolvido irá substituir a N2 do 2º bimestre.
* Desenvolver no Visual Studio 2015
* O trabalho deverá ser apresentado em, no máximo, 10 minutos. Neste tempo está incluso a configuração da máquina.
* A nota de quem não apresentar será zero.
* Será apresentado na semana de provas N2.
* A ordem de apresentação será dada por sorteio.
* Toda estrutura de dados utilizada, como por exemplo listas e ordenação, devem ser criadas pelo e aluno, que não deve utilizar a classe LIST do C#, nem nenhuma outra classe pronta para este fim.
* Os dados deverão ser salvos (persistidos) em disco em arquivo texto ou XML / JSON. As classes DataTable e DataSet podem ser utilizadas para persistência dos dados em XML.
* A manipulação dos dados em memória NA TELA DO PLAYER deve ser feita utilizando-se estruturas de dados criadas pelo aluno, como listas, pilhas, filas, árvores, etc.
* Toda ordenação também deve ter sido implementada pelo aluno.
* Resolução máxima da tela: 1024X768px

A avaliação levará em conta as partes técnica, visual e a organização.

**Classes e Interfaces que deverão ser criadas:**

Interface ICatalogo

Propriedades:

Int AnoLancamento – ano que a mídia foi lançada

Interface ILocal

Propriedades:

String arquivoMidia – local e nome do arquivo da mídia

Métodos:

Bool ValidaCaminho() valida se o arquivo de mídia realmente existe no local informado.

Classe Midia : Object // Esta classe deve ser abstrata e deverá ter os atributos:

Propriedades:

Int Id;

String descricao;

String arquivoDeDados - local e nome do arquivo de dados (onde as mídias serão armazenadas)

Métodos:

Inclui() // inclui um novo elemento no arquivo. Não permita Id´s repetidos independente do tipo de mídia

Alterar() // altera o emento no arquivo

Excluir() // exclui o elemento do arquivo

Midia Consultar(int id) // preenche os atributos da classe com o id do elemento informado.

Classe Musica : Midia, ILocal,

Atributos:

FormatoEnum formato (enumador que pode ser: MP3, WAV ou WMA)

Double Duração

Int Volume – volume da música, para ser utilizado no media player

Classe AlbumMusical : Midia, ICatalogo

Atributos:

String Artista

Lista de músicas: Lista com todas as músicas que fazem parte do álbum.

Classe Video : Midia, ILocal, ICatalogo

Atributos:

FormatoEnum formato (enumador que pode ser: AVI, WMV, MKV, MP4, MPEG, OUTROS)

IdiomaEnum idioma (portugues, inglês, espanhol, outros)

Boolean PossuiLegenda

Classe Foto: Midia, ILocal, ICatalogo

Atributos:

String Local (local onde a foto foi tirada)

Double MegaPixels

Int TempoEmSegundosParaExibir (tempo em segundos que a foto será exibida no player)

Telas:

* Criar uma tela para efetuar cada um dos cadastros e persistir os dados salvos em um arquivo
* **Não utilize banco de dados SQL ou de qualquer outro tipo**
* Criar uma forma de associar as músicas e os álbuns.
* Criar uma tela onde o usuário poderá reproduzir os vários tipos de mídias cadastradas. Nesta tela, o usuário poderá escolher as mídias que deseja tocar no player. Deve-se listar as mídias e permitir que o usuário monte um playlist para reproduzir as mídias.

Ao exibir as mídias para que o usuário as selecione, deve ser possível lista-las das seguintes maneiras:

* Ordem Alfabética (não importa a mídia, todas em ordem alfabética)
* Por tipo de mídia (álbum / música / vídeo / foto) e uma vez escolhido o tipo, exibir em ordem alfabética.
* Por ano (exibir uma forma da pessoa selecionar o ano e então devem ser exibidas as mídias que constam nesse ano)

Nesta parte, se necessário, pode-se utilizar qualquer componente visual para exibição das mídias, como por exemplo as classes ListBox , TreeView, GridView, etc. É Importante salientar que apesar dos elementos serem exibidos em componentes visuais, como o ListBox por exemplo, a ordenação e o tratamento destes dados deve ser realizado nas estruturas de dados criadas pelo aluno.

O objetivo destes “filtros” é permitir que o usuário monte uma sequência de mídias que ele deseja tocar. As mídias podem estar misturadas. Sendo assim, no player pode tocar uma música e logo após tocar um vídeo e depois outra música novamente, seguido de uma foto e assim por diante.

Obs: Quando o usuário seleciona um álbum, é como se ele tivesse selecionado individualmente todas as músicas que o compõe.

As mídias deverão ser tocadas exibindo-se o seguinte padrão: por lista, por pilha e por fila. Segue a seguir a explicação de cada um:

* Tocar no formato de lista: o usuário monta a lista e as mídias devem ser reproduzidas em:
  + Em ordem aleatória ou,
  + Ordenadas pela descrição.
* Tocar no formato de pilha: as mídias devem ser reproduzidas obedecendo a ordem LIFO (armazenar em uma pilha)
* Tocar no formato de fila: as mídias devem ser reproduzidas obedecendo ao formato de FIFO. (armazenar em uma fila)

Ao reproduzir uma mídia, exiba em algum local da tela TODAS as informações desta mídia. As mídias podem ser reproduzidas nos componentes media player / picturebox / webBrowser.

Entrega:

**Todo código fonte deve ser salvo em uma pasta na rede. O conteúdo do trabalho não poderá ultrapassar 100 mb.**

**Criar uma pasta com o SEU RA completo.**

**O local será determinado no dia da apresentação.**

**Avaliação:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tópico** | **Nota máxima** | **Nota** |  |
| **Cadastro de mídias** | 1,0 |  | Ok |
| **Tela principal do player** | 1,0 |  | Ok |
| **Tocar por Fila** | 1,0 |  |  |
| **Tocar por Pilha** | 1,0 |  |  |
| **Tocar por Lista** | 1,0 |  |  |
| **Ordenação (aleatória, por descrição)** | 1,0 |  | Ok |
| **Tocar todos os tipos de mídia em um mesmo playlist** | 1,0 |  |  |
| **Classes e estruturas de dados / Orientação a objetos** | 2,5 |  | Ok |
| **Tela sobre com os nomes dos integrantes** | 0,5 |  | Ok |