MS.NET

Mini Project

Music-JukeBox-System (MJS)

Document Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Revision No.** | **Author** | **Summary of Changes** |
| 13-07-2017 | 1.0 | Pankaj Patil | Initial Draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[Introduction 4](#_Toc304553070)

[Setup Checklist 4](#_Toc304553071)

[Instructions 4](#_Toc304553072)

[Problem Statement 5](#_Toc304553073)

[Objective 5](#_Toc304553074)

[Abstract of the project 5](#_Toc304553075)

[Functional components of the project 5](#_Toc304553076)

[Technology used: 8](#_Toc304553077)

[Implementation 9](#_Toc304553078)

[Summary of the functionality to be built: 9](#_Toc304553079)

[Guidelines on the functionality to be built: 9](#_Toc304553080)

Introduction

This document outlines a mini project for the .NET Line of Technology (LOT). The project is to develop Music-JukeBox-System (MJS). This document contains the requirements, work flow of the system and gives guidelines on how to build the functionality gradually in each of the course modules of the .NET LOT.

## Setup Checklist

Minimum System Requirements

* Intel Pentium 4 and above Windows 2007, 2008 and 2010
* Memory 4 GB
* Internet Explorer 8.0 or higher
* SQL Server 2012 client and access to SQL Server 2012 server
* Visual Studio 2015/2017

## Instructions

* The code modules in the mini project should follow all the coding standards.
* Create a directory by your name in drive **<drive>**. In this directory, create a subdirectory **MiniProject**. Store your Project here.
* You can refer to your course material.
* You may also look up the help provided in the MSDN
* Since this project work will span over couple of months, you will need to take care of maintaining the code

Problem Statement

## Objective

Development of **Music Jukebox System** (MJS)

**Abstract of the project**

The main aim of developing this **Music Jukebox System** it’s a collection of MP3 songs of different languages in one place where users can search, based on the year also play and listen the songs in our website at free of cost only is to provide user friendly tool for music web sites. This is one type of online music Player. Most of the websites now a day’s Music products through online but download music files with free of cost makes problem with piracy so this is the best place to stop music piracy over the internet. Here the customer should login through Music store website and play & Lesion selected music files like songs, and private albums by using papal. Then selected music files can be downloaded directly to the local system of the customer. The main advantage is it is user friendly, provide us 24 hours customer service, and decrease the manual efforts and Time.

**Music Jukebox System** allows to store, process, retrieve and analyze information with respective to the administrative and inventory management within a Music Store.

Music Jukebox System will have following features:

* Collection of MP3 songs of different languages in one place.
* Music Jukebox User System - User Registration, Managing User information
* Music Search – based on Singer, Year, Movie, Actor, Actress, Language

Users of the System will be: Music Jukebox System Administrator

Role of Music Jukebox System Administrator:

* Login to System
* Upload, Remove the Music
* Add and Remove Singer
* Add and Remove Customer
* Update Album details
* Delete the Album details
* Search Customer, Singer
* View Album

Role of Music Jukebox System User:

* Login to System
* Upload, Remove the Music in Playlist
* Update Playlist details
* Delete the Playlist
* Search Music, Singer, Album
* View Album
* Upload songs
* Download songs

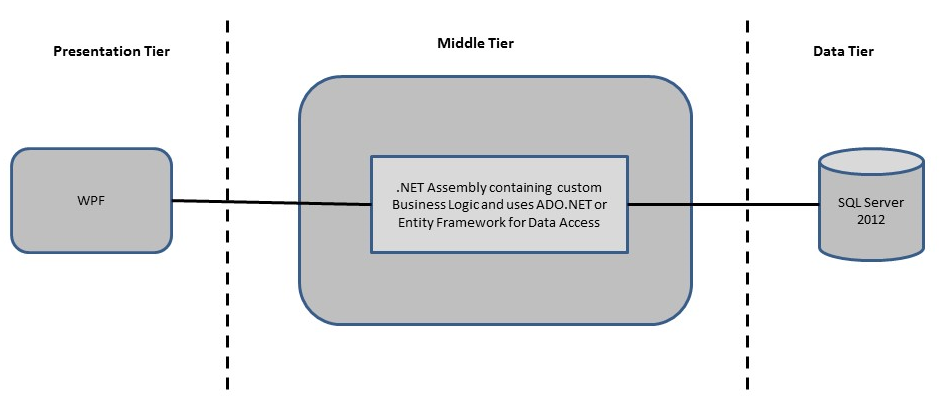
## Functional components of the project

**Application Architecture:**

Distributed web applications traditionally to be designed and built across three logical tiers:

* Database Access Layer (DAL)
* Business Logic Layer (BLL)
* Presentation Layer

The DAL refers to the database itself, the stored procedures, and the component that provides an interface to the database. The BLL refers to the component that encapsulates all the business logic of the application. And, the Presentation layer refers to the web application pages.

****

**Design guidelines**

* All the exceptions/errors to be captured and user friendly message to be displayed on the CommonError page.
* Data access layer of 3-tier use ADO.NET data access using SQL stored procedures - All the database interaction would be performed using Data Access Component. Most common methods in Data Access Component would be –

1. Create Connection to the Database
2. Create Command Object
3. Set Command Type to Stored Procedure
4. Create and Populate Parameters
5. Execute the Command
6. Close the Connection

## Technology used:

* + - *Presentation Layer* 
      1. *WPF 4.5*
    - *Business Layer*
      1. *Business Logic Components and Services :-* 
         1. C# 5.0 / 6.0
    - *Database Layer*
      1. *Databases:-*
         1. SQL Server 2012 / +

Implementation

## Summary of the functionality to be built:

The participants need to develop the Music JukeBox System by building the functionality incrementally in each of the course modules of .NET LOT.

|  |  |
| --- | --- |
|  | |
| **Sr. No** | | **Course** | **Duration** | **Functionality to be built** |
| **(in PDs)** |
| 1 | | MS SQL Server 2012 / + | 4 | Creating relevant database tables and stored procedures |
| 2 | | NET Framework 4.5 + C# 5.0 / 6.0 | 7.5 | Developing Business components (C# classes) |
| 3 | | ADO.NET 4.5 | 3 | Integrating the DAL ,BLL and PL |
| 4 | | LINQ to Entity Framework | 2 | Creating data model and data context and using LINQ to entities |
| 5 | | WPF 4.5 | 2 | Incorporating advanced UI functionality with WPF 4.5 |
| 6 | | Mini Project Presentation | 1 | The Mini Project Presentation day |

# Note: Saturday half day will be devoted for mini project

## Guidelines on the functionality to be built:

The functionality and components to be built in each of the course modules of .NET LOT is as follows:

1. **Course: SQL Server 2012 / +**

This section describes some of the basic steps involved in designing and creation of the database for the application.

Create Data Model - identify the different tables and fields that we will need, which would later be used for building the rest of the application.

Database Schema - Taking these objects, we can easily identify our main tables in the database.

* 1. Create the following database tables with following fields: [make your assumptions in case you require few more fields]
     + 1. **Custome**r :CustomerID, CustName, Address, DOB, City, Password MobileNo
       2. **Employee** : EmployeeID, EmpName, Address, DOB, City, Password, MobileNo
       3. **MusicOrder** : OrdederID, OrderDate, DeleveryDate, OrdPaymentID, EmployeeID, CustomerID
       4. **Payment** : PaymentID, PayAccName, PayOrderID, PaymentDate, Amount PaymentMethod
       5. **MusicOrderDtetails** : MusicOrderID, AlbumID, UnitPrice, Qty, Discount, PaymentID, TotAmount
       6. **Album** : AlbumID, AlbumName, Category, No\_Of\_Songs, Release Date, Company, Price, Language
       7. **Songs**: SongID, SongName, Singer, Movie, ComposedBy, Lyrics, Year, AlbumID, Language

1. **Course: WPF 4.5**
   1. Develop the prototypes for following functionalities:
      1. **Login / Sign In**: Login screen would display asking user to enter ‘User Id’ & ‘Password’. If the supplied user credentials are valid the HomePage would be displayed, else appropriate Login error message would be displayed
      2. **Home Page**: On successful user authentication (validation of userid/password provided by the user in login screen) the homepage would be displayed. The Homepage would contain below sections/contents:

Header section: The header section would be common across all the pages and would mainly have –

* Sign In Link – On click it would take to Login page
* Registration Link- On click it would take to Registration page
* Search: Search for information

Main content section: It would display generic Welcome message giving overview of the site. Should have menu for Employee, Customer

* 1. For Customer:
     + - 1. User will get option to add, update the personal information
         2. User will get option to Search

Album By ReleaseYear.

Album By Singer..

Songs by Singer.

Songs by Composer

Songs by Genre

* + - * 1. User will able to Create Personal Album
  1. For Employee
     1. User will option to add, update the personal information
     2. View the details of all Albums
     3. View the details of all Active Users
     4. View the details of all Orders Placed
     5. Update details of all Albums
     6. Update the details of all Orders Placed
     7. Delete the Albums
     8. Delete the Customer
     9. Delete the Orders
     10. Search for Album, Order and Customer

1. **Course: C# 5.0 / 6.0 and ADO.NET 4.5**
   1. Develop business components (C# classes) for the following functionality:
      1. Customer Class :- This class will Contain methods which will allow to manage all of the Customer
      2. Employee Class :- This class will Contain methods which will allow to manage all of the Employee
      3. MusicOrder Class :- This class will Contain methods which will allow to manage all Order Placed by Customer
      4. MusicOrderDetails Class :- :- This class will Contain methods which will allow to manage all Order Placed by Customer
      5. Album Class :- This class will Contain methods which will allow to manage Album
      6. Songs Class :- This class will Contain methods which will allow to manage Songs.

You need to create Layered Architecture which comprises of Presentation Layer ( WPF), Business Logic Layer (C# Classes) and DAL Layer (Using ADO.Net 4.5 / Linq and Entity Framework)

DAL Layer of ADO.NET 4.5 will include all the required code snippets for CRUD Operations.

All the CRUD operations should use SQL Server Stored Procedures (For Insert, Update, Delete and Search).

The connectionString should be stored in the configuration file only.

**OR**

1. **Course : LINQ and EF**
   1. Use the database first approach and create the Entity data model consist of the following entities :
      1. Customer Class :- This class will hold the basic details of the Customer
      2. Employee Class :- This class will hold the details of Employee
      3. MusicOrder Class :- This class will hold the basic details of Order Placed by Customer
      4. MusicOrderDetails Class :- This class will hold the details of Order Placed by Customer
      5. Album Class :- This class will hold the details of Album
      6. Songs Class :- This class will hold the details of Songs.
   2. Use Data context to perform CRUD operations and Implement the required logic using LINQ (like retrieving data, sorting data, searching data etc).
2. **Course: ADO.NET 4.5 and WPF 4.5**
   1. Integrate all screens (WPF pages) with business components (C# classes) to complete the entire functionality Project Evaluation Guidelines

The project it is to be evaluated based on the following five parameters:

1. Proper Database Structure and UI designing as per the specifications –(15 Marks)
   1. Proper Database Design and Stored Procedure
   2. Visual look and feel of the UI
2. Project Completion – (20 Marks)
   1. Timely Completion of the project
   2. Integration of all component of the system
3. Defect free execution – (30 Marks)
   1. Error free execution of individual modules and the whole system
   2. Validation
   3. Functionality as per the specified requirements
4. Compliance of standard and guidelines – (15 Marks)
   1. Appropriate comments entries
   2. Adherence to naming conventions for classes, functions, variables and files
   3. Simplicity of user interface and screen layouts
   4. Maintainability of codes (for example, no one function should be more than 100 lines)
5. Group Presentation and Query handling – (20 Marks)
   1. Participants (Group of 3 to 4) to present the project with UML Diagrams(use case diagram and one of the sequence or activity diagram) and PPT