MS.NET

Parallel Project

Hospital Management System

(HMS)

Capgemini Public

Document Revision History

Date	Revision No.	Author	Summary of Changes

Table of Contents

©2016 Capgemini. All rights reserved.

The information contained in this document is proprietary and confidential. For Capgemini only. | **3** / 16

Capgemini Public

INTRODUCTION

This document outlines a parallel project for the .NET Line of Technology (LOT). The project is to develop Hospital Management System. 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 2017

INSTRUCTIONS

- The code modules in the parallel project should follow all the coding standards.
- Create a directory by your name in drive **<drive>**. In this directory, create a subdirectory **ParallelProject**. 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

Capgemini Internal

PROBLEM STATEMENT

OBJECTIVE

Development of Hospital Management System (HMS)

Abstract of the project

Hospital Management System (HMS) is an application, which helps to manage Patient (Both inpatient and Outpatient) and their Treatment details. Patient details like Patient ID, Name, Age, Phone No etc. And Treatment details like – Doctor Name, Room No, Date of Admission, and discharge details etc.

HMS aims to manage Patient Lab reports like adding report information – Test type, Test Date, Price & Result. Along with add, system should display all Report summaries (View) and must support search facility. The Application should also generate Bills for the patients like Bill id, Room amount, Total days, Theatre amount (if applicable), medicine bill, Total Amount, etc...

- **Phase 1 :** The system will first develop using C# only – where Patient Info, Lab Report & Billing data will be store as a Collection (List & Dictionary). For user interaction, system will use Console Application.

- **Phase 2 :** Later on data will be store in MS SQL Server database; system will use ADO.NET or LINQ and Entity Framework for the same. User Interface will be designed using WPF

- **Phase 3** : HMS will become web based application, following MVC design pattern. Here the application will be develop in ASP.NET MVC.

Macro level Operations/offerings:

Capgemini Internal

1. Adding Patient Information

2. Modify Patient Info

3. Add Patient Treatment Information

4. View Patient History

5. Search Patient Info

6. Generate Lab Report

7. Generate Medical Bills

MODULE LIST and MODULE DETAILS

CREATE PATIENT DATA

Following info need to capture

- Patient ID (Must be Unique)

- Name

- Gender

- Age

- Address

- Phone No

- Weight

CREATE PATIENT APPOINTMENT DATA (INPATIENT / OUTPATIENT)

Following info need to capture

- Appointment ID (Must be Unique)

- Patient ID

- Doctor Name (Based on Disease)

- Room No

- Date of Visit (For Inpatient)

- Admission Date (For Outpatient)

- Discharge Date (For Outpatient)

- Remarks

SEARCH PATIENT

Admin / Hospital Staff should be able to search a Patient details by Patient ID, Date of Visit, Doctor Name, Date of Admission or Discharge. They may use any one, two or all the applicable parameters to search a patient.

MODIFY PATIENT

Search (Patient Id) Patient and modify the information. System should show existing data/info of the patient and should support modify the eligible details.

©2016 Capgemini. All rights reserved.

The information contained in this document is proprietary and confidential. For Capgemini only. | **7** / 16

Capgemini Internal

FILM SUMMARY (VIEW)

System should show (display) patient / patient list in a tabular format (one row for each patient, and columns for patient info). It is not required to show all the patient data. Only important info like – Name, Date of Visit, Doctor name, etc… can be displayed in table.

LAB REPORT

Patients can take various medical test as per doctor's prescription. After completion, they will get lab report which contains information like – Report ID, Name, Test Date, Doctor, Test Type & Remarks.

BILL GENERATION

After completion of treatment, system should generate Bill which contains information like – Bill ID, Room Amount (Inpatient), Operation Theatre amount (Inpatient), Medical Bill & Lab Bill.

Constrains

- Proper validation is required (especially Patient Id, Age, Amount, etc…) System must check uniqueness on film Patient ID, Report ID, Bill ID.

- System must show appropriate massage on all activity (whether activity is successful or failure)

- User must have proper menu to select the activity (create, modify, search, view, remove) that user want to perform.

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.

Capgemini Internal
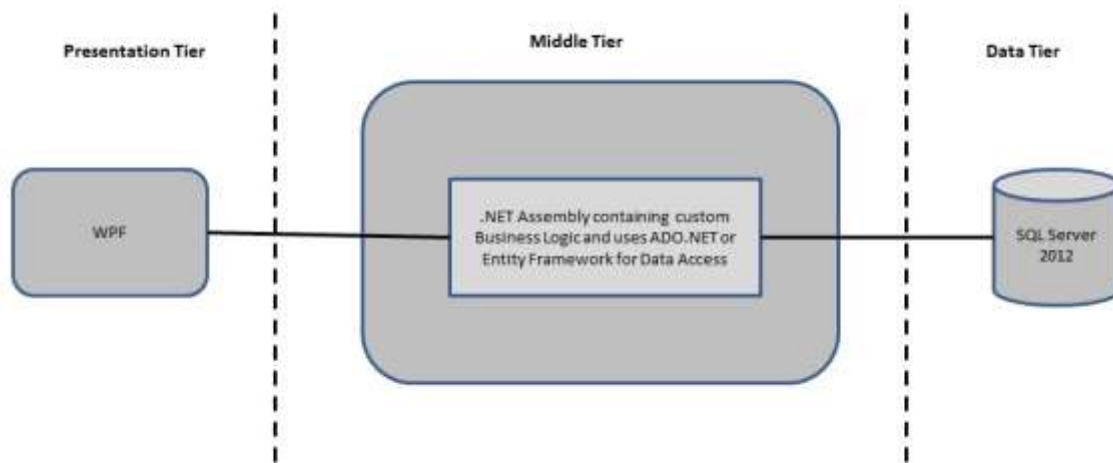
Design guidelines

- 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 Entity Framework data access using SQL stored procedures - All the database interaction would be performed using Data Access Component.

TECHNOLOGY USED:

> *Presentation Layer*

   1. *Console Application, WPF, ASP.NET MVC 5*

> *Business Layer*

   1. *Business Logic Components and Services :-*
      a. *C# 5.0*

> *Database Layer*

   1. *Databases:-*
      a. *SQL Server 2012*

Capgemini Internal

IMPLEMENTATION

SUMMARY OF THE FUNCTIONALITY TO BE BUILT:

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

Sr. No	Course	Duration (in PDs)	Functionality to be built
1	MS SQL Server 2012	4	Creating relevant database tables and stored procedures
2	NET Framework 4.6 + C# 7.0 + Introduction to WPF	10	Developing Business components (C# classes)
3	ADO.NET with LINQ and Entity Framework	4	Creating data model and data context and using LINQ to entities
4	ASP.NET MVC 5	4	Incorporating advanced UI functionality with ASP.NET MVC 5
5	Parallel Project Presentation	1	The Parallel Project Presentation day

Note: Saturday half day will be devoted for Parallel project

Capgemini Internal

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.

   a. Create the following database tables with following fields: [make your assumptions in case you require few more fields]

Table Name: Patient

Field Name	Constraints	Data Type
Patient id	Primary Key	Text
Name		Text
Age		Number
Weight		Number
Gender		Text
Address		Text
Phone no		Number
Disease		Text
Doctor id	Foreign Key	Text

Capgemini Internal

Table Name: Doctor

Field Name	Constraints	Data Type
Doctor id	Primary Key	Text
Doctor name		Text
Dept		Text

Table Name: Lab

Field Name	Constraints	Data Type
Lab Id	Primary Key	Text
Pid	Foreign Key	Text
Doctor id		Text
Test Date		Date
Test Type		Text
Patient Type		Text

Table Name: Inpatient

Field Name	Constraints	Data Type
Pid	Primary Key	Text
Room No	Foreign Key	Text
Doctor id	Foreign Key	Text
Admission Date		Date
Discharge Date		Date
Lab No	Foreign Key	Text
Amount Per Day		Number

Table Name: Outpatient		
Field Name	Constraints	Data Type
Pid	Primary Key	Text
Treatment Date		Date
Doctor id	Foreign Key	Text
Lab No	Foreign Key	Text

Table Name: Room Data

Field Name	Constraints	Data Type
Room No	Primary Key	Text
Treatment Date		Date
Doctor id	Foreign Key	Text
Lab No	Foreign Key	Text

Table Name: Bill Data

Field Name	Constraints	Data Type
Bill No	Primary Key	Text
Pid	Foreign Key	Text
Patient Type		Text
Doctor id	Foreign Key	Text
Doctor Fees		Number
Room Charge		Number
Operation Charges		Number
Medicine Fees		Number
Total Days		Number
Lab Fees		Number
Total Amount		Number

Capgemini Internal

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)

   a. Proper Database Design and Stored Procedure

   b. Visual look and feel of the UI

2) Project Completion – (20 Marks)

   a. Timely Completion of the project

   b. Integration of all component of the system

3) Defect free execution – (30 Marks)

   a. Error free execution of individual modules and the whole system

   b. Validation

   c. Functionality as per the specified requirements

4) Compliance of standard and guidelines – (15 Marks)

   a. Appropriate comments entries

   b. Adherence to naming conventions for classes, functions, variables and files

   c. Simplicity of user interface and screen layouts

   d. Maintainability of codes (for example, no one function should be more than 100 lines)

5) Group Presentation and Query handling – (20 Marks)

   a. 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

Capgemini Internal