WINDOWS PRESENTATION FOUNDATION

Lab Book

Document Revision History

Date	Revision No.	Author	Summary of Changes
21-Oct-09	1.0	Ganesh Desai	Initial Document
21-Oct-09	1.0	Ummeaiman Diwanji	Quality Review, Transfer to new template.
15-July-2011	2.0	Ganesh Desai	Changes made as per integration process
28-Aug-2012	3.0	Abishek Radhakrishnan	Revamp of Lab Book.

Table of Contents

Getting Started

Overview

This lab book is an unguided tour for learning WPF 4.5. It comprises of a use case diagram which will have to be realized in a sequence of exercises. Screen snap shots are provided wherever necessary.

Setup Checklist for WPF

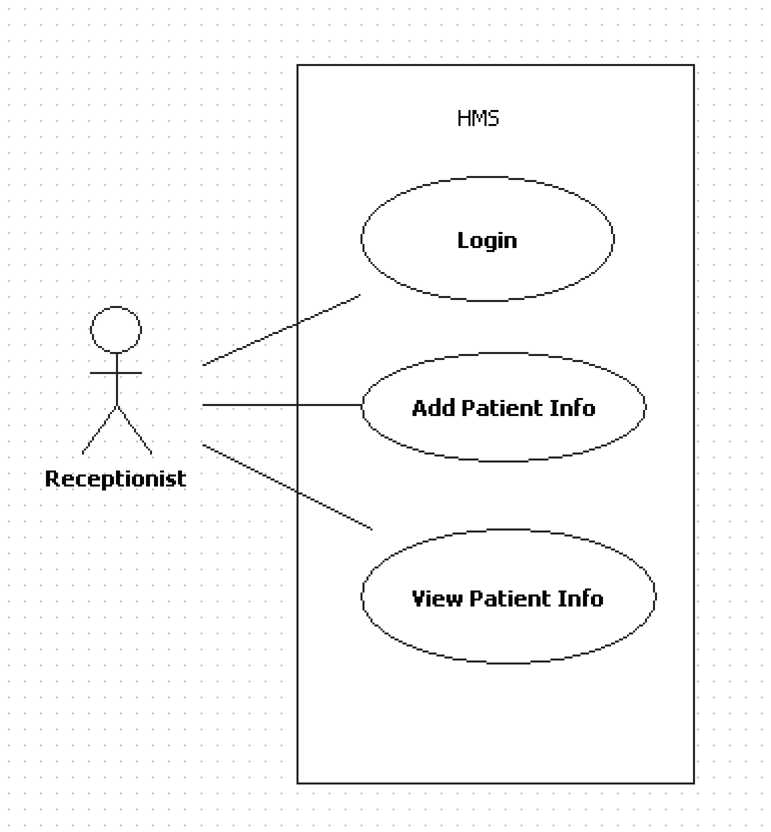
Here is what is expected on your machine in order for the lab to work.

Minimum System Requirements

- *Hardware:* Networked PCs with minimum 1 GB RAM and 60 MB HDD.
- Software: Microsoft Visual C# 2012 Express Edition or Visual Studio.NET 2012.

Problem Statement/ Case Study

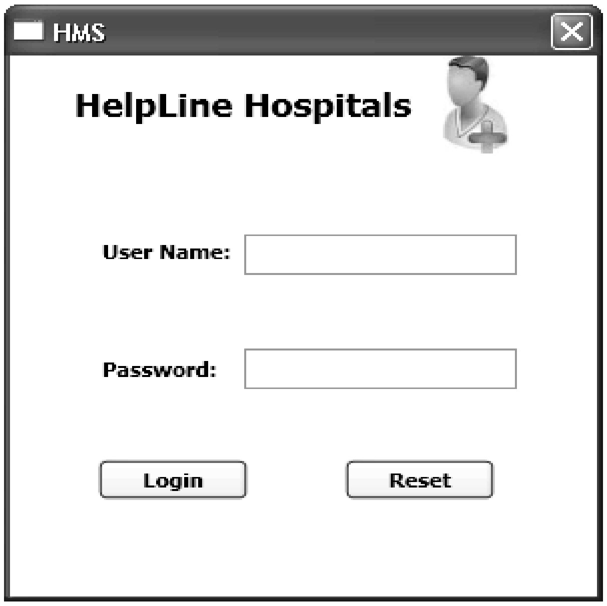
HelpLine Hospitals needs a Hospital Management System (HMS) to manage their daily work flow. You have been given a part of this project to implement. The following use cases have to be realized.



Lab 1.WPF Layouts And Controls

Goals	• Working with a Window. • Using Container controls.
Time	1.5 hours.

<u>1.1:</u> Create a Login form as shown in the sample screen below. The user name and password can be "admin" and "admin". Use a StackPanel,DockPanel and Grid to implement the design.



Note: complete questions 1 to 4 from the "Extended Practice" section.

.

<u>CONCLUSION POINTS:</u>

a. think about scenarios where a WrapPanel can be used.
b. Which container control is the most flexible for a developer to work with? What would be the limitation of the same?

Lab 2.Styles and Resources

Goals	• Working with styles and resources. • Using Gradient colours and shapes.
Time	2 hours.

2.1: Modify the window completed in exercise 1.1 to meet the following requirements:

- the font of the labels must be bold, Arial and size 12.
- the button must have an elliptical shape.
- the login button's opacity property changes from 0.5 to 1 when the mouse focus is on it. Use triggers in XAML to achieve this.
- The window must have a gradient back ground colour of your choice. You must have a minimum of 3 colours.
- <u>All the above requirements must be fulfilled using styles or resources in the App.xaml.</u>

Note: complete questions 5 to 7 from the "Extended Practice" section.

<u>CONCLUSION POINTS:</u>

a. **Where else can styles or resources be applied? What would be the level of reusability in each case?**
b. **Can the above requirements be implemented using C# code? If so, what is the advantage of using XAML ?**

Lab 3.Event Handling and Data Binding

Goals	• Working with Events. • Binding Data and WPF controls.
Time	4 hours.

3.1: The TextBox used for entering the user name must not allow the user to enter any numbers. Implement this functionality using the "PreviewTextInput" event of the TextBox. Is this event a Tunneled event or a Bubbled event ? Discuss.

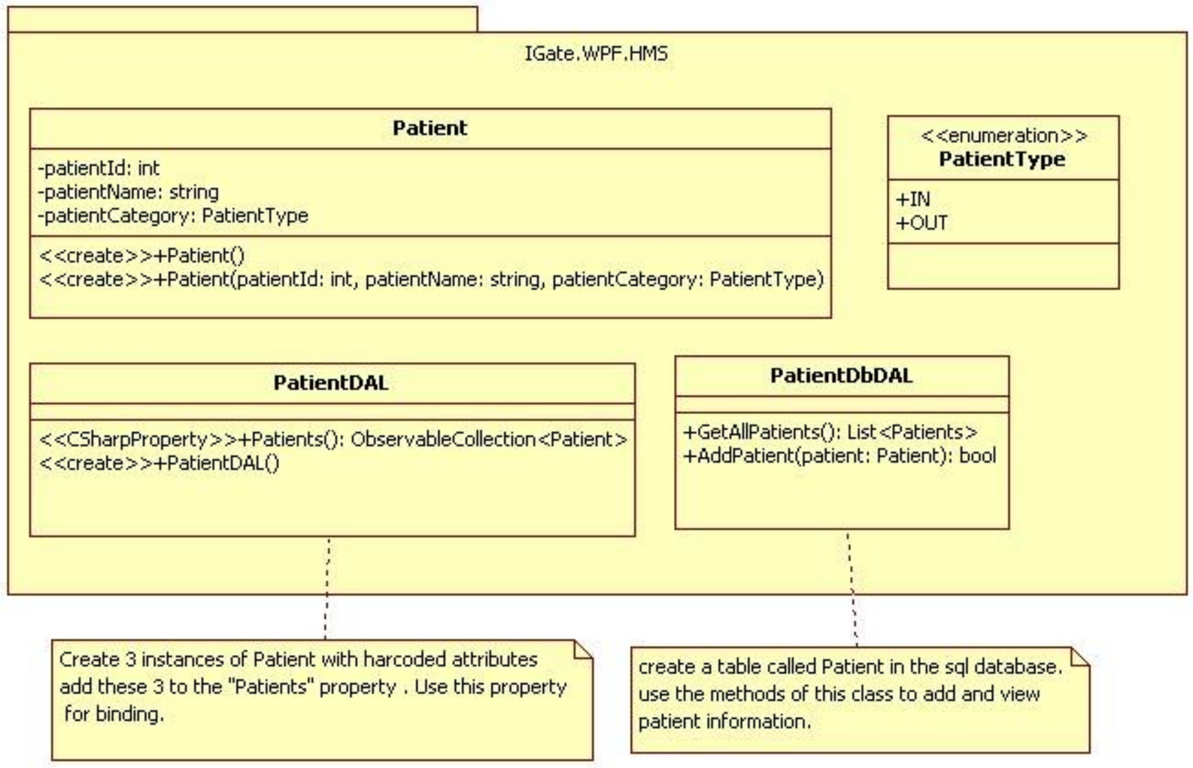
3.2 : Add Functionality to the Login and Reset Button.

enter user name as "admin", password as "admin", click on Login Button	Open a new window as shown in the screen shot
invalid attempt of more than 3	the application exits
clicking on Reset Button.	Clears the fields and brings the focus on the user name TextBox

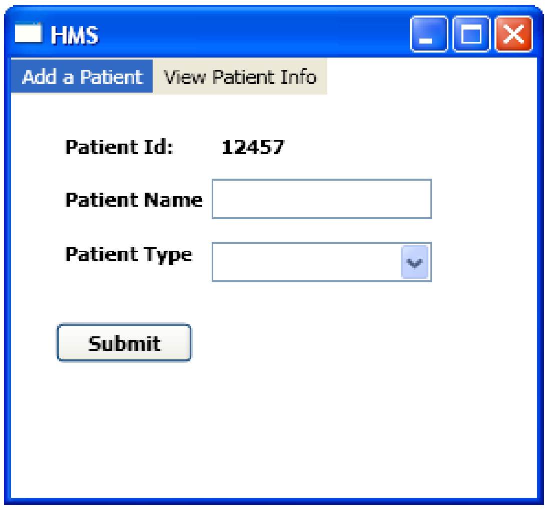
Note : implement the above functionality using bubbled events. [The container of the button controls will handle the click event].

3.3 : Write code to realize the following screens:

Note: You will have to implement this question with two data sources:
a. PatientDbDAL
b. PatientDAL.
Refer to the class diagram for the DAL structure.

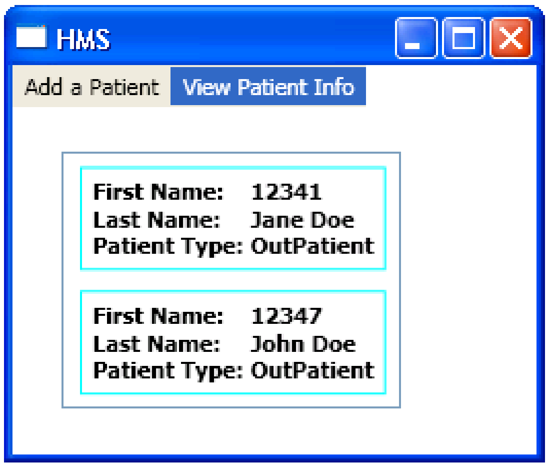

IGate.WPF.HMS
Patient
-patientId: int
-patientName: string
-patientCategory: PatientType
<<create>>+Patient()
<<create>>+Patient(patientId: int, patientName: string, patientCategory: PatientType)
<<enumeration>>
PatientType
+IN
+OUT
PatientDAL
<<CSharpProperty>>+Patients(): ObservableCollection<Patient>
<<create>>+PatientDAL()
PatientDbDAL
+GetAllPatients(): List<Patients>
+AddPatient(patient: Patient): bool
Create 3 instances of Patient with harcoded attributes add these 3 to the "Patients" property . Use this property for binding.
create a table called Patient in the sql database. use the methods of this class to add and view patient information.

The sample screen shots for the window to be displayed after successful login are given in the following page:



Patient Id: 12457
Patient Name
Patient Type

The highlighted menu item exposes the UI to add a patient. Generate the patient Id automatically.



First Name: 12341
Last Name: Jane Doe
Patient Type: OutPatient

First Name: 12347
Last Name: John Doe
Patient Type: OutPatient

The highlighted menu item exposes the UI to view patient details.

CONCLUSION POINTS:

a. **What is the special feature of an Observable Collection?**
b. **What is the specialty of Routed Events in WPF?**
c. **What scenarios can you think for using the various modes in data binding?**

EXTENDED PRACTICE:
TIME: 1 hour.

1. Explore the use of the Expander and Tab Control.
2. What is the use of the “Stretch” property in an Image control?
3. Clip an Image to make it circular or elliptical in shape.
4. A Button’s content needs to have an Image and it’s description in a textblock. Button, being a Content control,can have its content property set only once. How will you achieve the above requirement?
5. Apply a Linear Gradient to a control using C# code.
6. Create 2 pages, Page1.xaml and Page2.xaml. use a hyperlink to navigate from Page1 to Page2.
7. Transform a rectangle using the Render, Translate, Scale and Skew Transforms

CODE DEBUGGING /ENHANCEMENTS
TIME: 1 hour.

Your instructor will provide you with a code which has some errors. Identify the errors, and make the changes necessary to make the code compile and execute