**Pharmacy Stock Management System**

***Programming Language* :** JAVA

***Concepts Involved*** : OOP , GUI , FileHandling (Serialization) , Scene Builder

***Team Members*** : [Huzaifa Basharat-------sp22-bcs-126 ]

[Umair Arshad----sp22-bcs- 108]

We are developing a stock management app for a pharmacy. It includes a **LOGIN page** which check for valid credentials of the user. Then we have a **HomePage** which have different menu buttons for different tasks like RegisterUser button for adding new user, AddMedicine button for adding new medicine to the stock and ViewMedicine button for viewing current stock etc.. we have used file handling to store data on hard drive through serialization. We have used 2 files, one for storing data of the user and other one for storing data of the medicines. We have created scenes by using **SceneBuilder** and then adding their FXML files to our project folder.

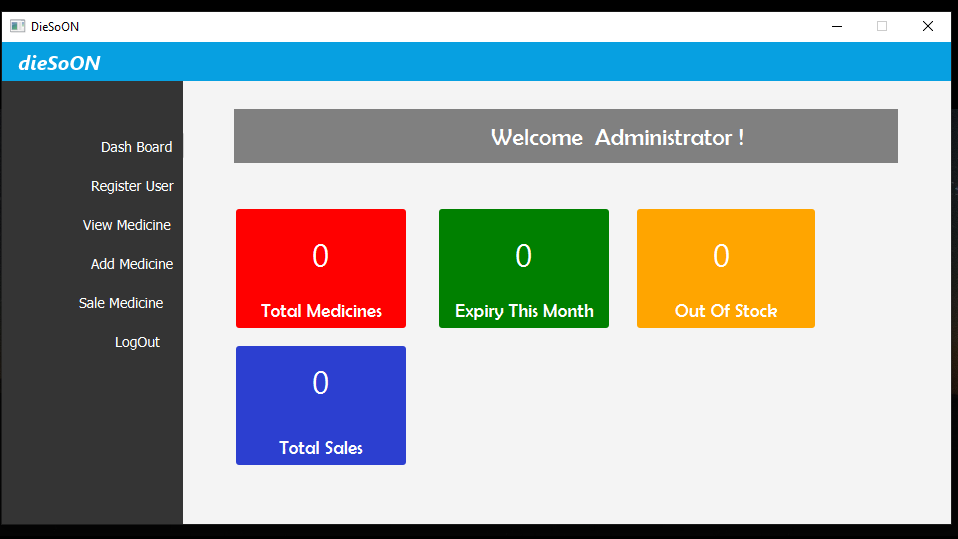
After signing in Home page appears , on that scene we have used the logic of hiding and unhiding panes on the clicking of different buttons instead of creating new scenes for every button action. This decreases the complexity of out code by making only 2 FXML files instead of large number of FXML files for every scene.

**LOGIN SCENE**

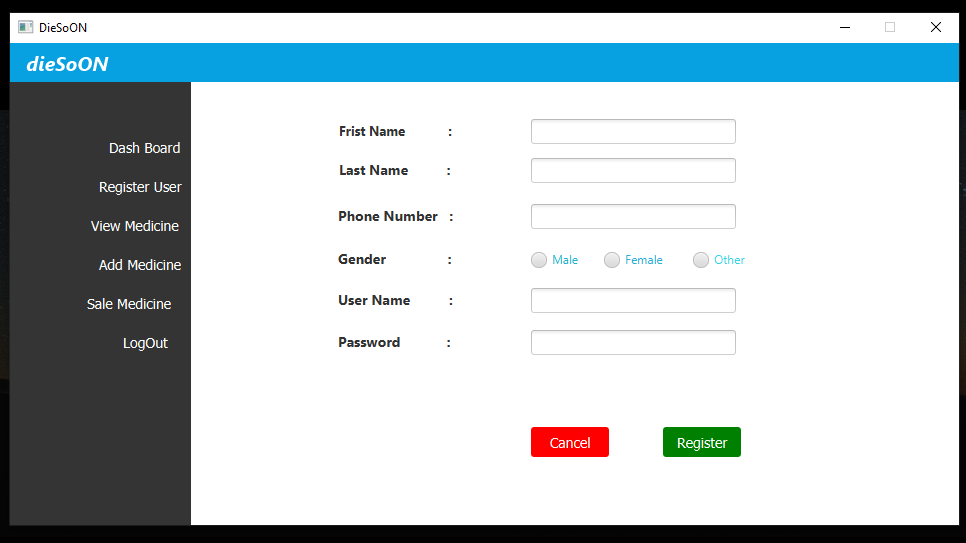
Graphical user interface, application, website

Description automatically generated

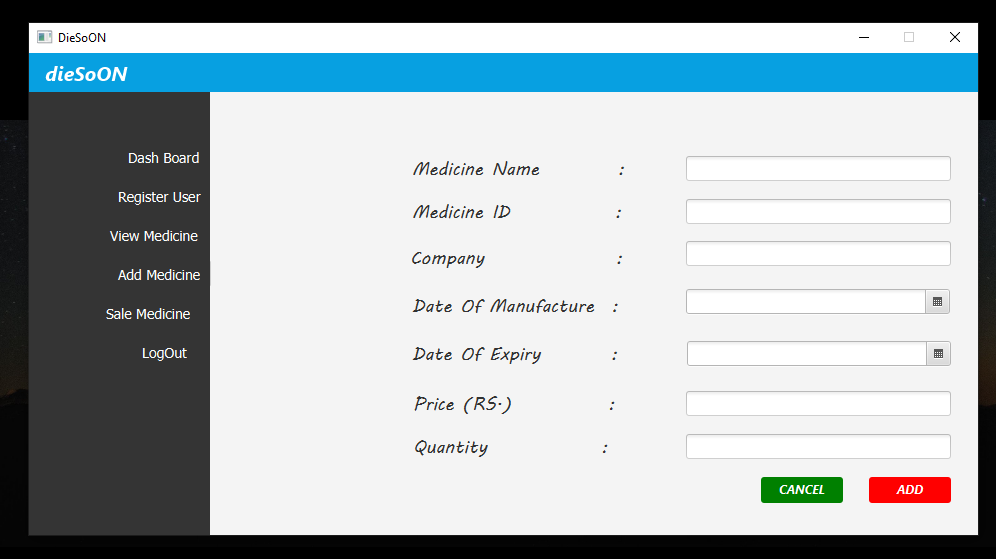
**Home Scene**

****

**Register User Pane**



**Add Medicine Pane**

****

**CODE**

**Driver Class :**

**package** com.example.diesoon;  
**import** javafx.application.Application;  
**import** javafx.fxml.FXML;  
**import** javafx.fxml.FXMLLoader;  
**import** javafx.scene.Scene;  
**import** javafx.stage.Stage;  
**import** java.io.IOException;  
**import** java.util.ArrayList;  
**import** java.util.Objects;  
**public class** Driver **extends** Application {  
 **private static** Stage *mainStage*;  
 **public static** Stage getMainStage() {  
 **return** *mainStage*;  
 }  
 @Override  
 **public void** start(Stage stage) **throws** IOException {  
 *mainStage* = stage;  
 FXMLLoader fxmlLoader = **new** FXMLLoader(Driver.**class**.getResource("/com/example/diesoon/LoginIn.fxml"));  
 Scene scene = **new** Scene(fxmlLoader.load());  
 *mainStage*.setScene(scene);  
 *mainStage*.setTitle("DieSoON");  
 *mainStage*.setResizable(**false**);  
 *mainStage*.show();  
 }  
  
 **public static void** main(String[] args) **throws** IOException, ClassNotFoundException {  
  
 *launch*();  
 }  
}

**File Class :**

**package** com.example.diesoon;  
**import** java.io.\*;  
**import** java.util.ArrayList;**public class** fileWork **implements** Serializable {  
   
 // Method For Reading Uses Data From File  
 **public static** ArrayList<User> userFileRead(String fileName) **throws** IOException, ClassNotFoundException {  
  
 ArrayList<User> users = **new** ArrayList<>();  
 FileInputStream fis = **new** FileInputStream(fileName);  
 ObjectInputStream ois = **new** ObjectInputStream(fis);  
  
 **while** (fis.available() > 0 ) {  
 users.add((User) ois.readObject());  
 }  
 ois.close();  
 fis.close();  
 **return** users;  
  
 }  
 // Method For Writing Objects To File  
 **public static** <T>**void** UserFileWrite(T u,String filePath) {  
  
 File f = **new** File(filePath);  
 FileOutputStream fos;  
  
 **try** {  
  
 **if**( f.length() == 0){  
 fos = **new** FileOutputStream( f);  
 ObjectOutputStream oos = **new** ObjectOutputStream(fos);  
 oos.writeObject(u);  
 oos.close();  
 System.*out*.println("User Registered");  
 }  
 **else** {  
  
 fos = **new** FileOutputStream( f,**true**);  
 MyObjectOutputStream oos = **new** MyObjectOutputStream(fos);  
 oos.writeObject(u);  
 System.*out*.println("User Registered1111");  
 oos.close();  
 }  
 fos.close();  
 }  
 **catch** (IOException e){  
 **throw new** RuntimeException(e);  
 }  
 }  
 **public static** ArrayList<Medicine> medicineFileRead(String fileName) **throws** IOException, ClassNotFoundException {  
  
 File f = **new** File(fileName);  
 ArrayList<Medicine> medicines = **new** ArrayList<Medicine>();  
  
 FileInputStream fis = **new** FileInputStream(f);  
// System.out.println("HERE");  
 ObjectInputStream ois = **null**;  
  
  
  
 **while** (f.length() >0 && fis.available() > 0 ) {  
 ois = **new** ObjectInputStream(fis);  
 medicines.add((Medicine) ois.readObject());  
 }  
 **assert** ois != **null**;  
 ois. close();  
  
 fis.close();  
 **return** medicines;  
  
 }  
  
}