**SCREENS**

**Login :**

package Screens;

import FileUtils.CurrentUser;

import FileUtils.PatientFile;

import FileUtils.DoctorFile;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.control.Label;

import javafx.scene.control.PasswordField;

import javafx.scene.control.TextField;

import javafx.stage.Stage;

import javafx.scene.layout.GridPane;

import javafx.scene.image.\*;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import javafx.geometry.Pos;

import javafx.scene.paint.Color;

import javafx.beans.value.ChangeListener;

import javafx.scene.control.RadioButton;

import javafx.scene.control.ToggleGroup;

import javafx.scene.control.\*;

import javafx.beans.value.\*;

import java.lang.String;

import javafx.scene.layout.VBox;

import javafx.scene.layout.HBox;

import javafx.scene.input.MouseEvent;

import javafx.event.EventHandler;

import javafx.scene.control.Alert;

import javafx.scene.control.Alert.AlertType;

public class Login extends Application  {

    @Override

    public void start(Stage primaryStage) throws FileNotFoundException{

        VBox vbox = new VBox(3);

        vbox.setAlignment(Pos.TOP\_CENTER);

        vbox.setSpacing(150);

        vbox.setStyle("-fx-background-color: LIGHTBLUE;");

        HBox hbox = new HBox();

        hbox.setAlignment(Pos.CENTER);

        hbox.setSpacing(10);

        ToggleGroup role = new ToggleGroup();

        RadioButton doctor = new RadioButton("Doctor");

        doctor.setToggleGroup(role);

        RadioButton patient = new RadioButton("Patient");

        patient.setToggleGroup(role);

        role.selectToggle(patient);

        Label heading = new Label("Hospital Management System");

        heading.setStyle("-fx-font: normal bold 20px 'arial' ");

        heading.setAlignment(Pos.CENTER);

        Label register = new Label("Not a user yet? Register");

        register.setUnderline(true);

        TextField user\_field = new TextField();

        user\_field.setPromptText("Username");

        PasswordField pass\_field = new PasswordField();

        pass\_field.setPromptText("Password");

        Button login = new Button("Login");

        FileInputStream inputstream1 = new FileInputStream("Images/user.png");

        Image user\_img = new Image(inputstream1);

        ImageView user\_imv = new ImageView(user\_img);

        user\_imv.setFitHeight(30);

        user\_imv.setFitWidth(20);

        FileInputStream inputstream2 = new FileInputStream("Images/lock.png");

        Image pass\_img = new Image(inputstream2);

        ImageView pass\_imv = new ImageView(pass\_img);

        pass\_imv.setFitHeight(30);

        pass\_imv.setFitWidth(20);

        GridPane gridPane = new GridPane();

        gridPane.setMinSize(300, 300);

        GridPane gp1 = new GridPane();

        gp1.setAlignment(Pos.CENTER);

        gp1.add(doctor, 0, 0);

        gp1.add(patient, 1, 0);

        gp1.setHgap(7);

        gp1.setStyle("-fx-background-color: LIGHTSKYBLUE;");

        gridPane.setVgap(7);

        gridPane.setHgap(7);

        gridPane.setAlignment(Pos.CENTER);

        gridPane.add(gp1, 1, 0);

        gridPane.add(user\_imv, 0, 1);

        gridPane.add(user\_field, 1, 1);

        gridPane.add(pass\_imv, 0, 2);

        gridPane.add(pass\_field, 1, 2);

        gridPane.add(login, 1, 3);

        gridPane.add(register, 1, 4);

        gridPane.setStyle("-fx-background-color: WHITE;");

        hbox.getChildren().add(gridPane);

        vbox.getChildren().add(heading);

        vbox.getChildren().add(hbox);

        login.setStyle("-fx-background-color: lightgreen; -fx-text-fill: white;");

        Scene scene = new Scene(vbox, 700, 600);

        primaryStage.setScene(scene);

        primaryStage.show();

        primaryStage.getIcons().add(new Image("file:Images/icon.png"));

        primaryStage.setTitle("Login");

        primaryStage.setAlwaysOnTop(true);

        register.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                register.setTextFill(Color.BLUEVIOLET);

            } else {

                register.setTextFill(Color.BLACK);

            }

        });

        role.selectedToggleProperty().addListener(new ChangeListener<Toggle>(){

            public void changed(ObservableValue<? extends Toggle> ob, Toggle o, Toggle n){

                String s = "";

                RadioButton rb = (RadioButton)role.getSelectedToggle();

                if (rb != null) {

                    s = rb.getText();

                }

                if(s.equals("Doctor")) {

                    register.setVisible(false);

                }

                else {

                    register.setVisible(true);

                }

            }

        });

        register.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                try{

                    Register register = new Register();

                    register.start(primaryStage);

                } catch (FileNotFoundException e)

                {

                    e.printStackTrace();

                }

            }

        });

        login.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                Alert a = new Alert(AlertType.ERROR);

                a.setHeaderText("Invalid Input");

                a.initOwner(primaryStage);

                if (user\_field.getText().isEmpty()) {

                    a.setContentText("Enter Username/ID");

                    a.showAndWait();

                } else if (pass\_field.getText().isEmpty()) {

                    a.setContentText("Enter Password");

                    a.show();

                } else  {

                    String s = "";

                    RadioButton rb = (RadioButton)role.getSelectedToggle();

                    if (rb != null) {

                        s = rb.getText();

                        boolean authentic;

                        if(s.equals("Doctor")) {

                            authentic = new DoctorFile().IsAuthenticateDoctor(user\_field.getText(),pass\_field.getText());

                            if(authentic){

                                new CurrentUser().UpdateUser(user\_field.getText(), s);

                                try{

                                    DMenu dmenu = new DMenu();

                                    dmenu.start(primaryStage);

                                } catch (FileNotFoundException e)

                                {

                                    e.printStackTrace();

                                }

                            } else {

                                a.setHeaderText("Login Unsuccessful");

                                a.setContentText("Incorrect Username/Password");

                                a.showAndWait();

                            }

                        }

                        else {

                            authentic = new PatientFile().IsAuthenticatePatient(user\_field.getText(),pass\_field.getText());

                            if(authentic){

                                new CurrentUser().UpdateUser(user\_field.getText(), s);

                                try{

                                    Menu menu = new Menu();

                                    menu.start(primaryStage);

                                } catch (FileNotFoundException e)

                                {

                                    e.printStackTrace();

                                }

                            } else {

                                a.setHeaderText("Login Unsuccessful");

                                a.setContentText("Incorrect Username/Password");

                                a.showAndWait();

                            }

                        }

                    } else {

                        a.setContentText("Select Role (Doctor/Patient)");

                        a.showAndWait();

                    }

                }

            }

        });

    }

    public static void main(String[] args) {

        launch(args);

    }

}

**Register :**

package Screens;

import DataModel.Patient;

import FileUtils.PatientFile;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.control.Label;

import javafx.scene.control.PasswordField;

import javafx.scene.control.TextField;

import javafx.stage.Stage;

import javafx.scene.layout.GridPane;

import javafx.scene.image.\*;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import javafx.geometry.Pos;

import javafx.scene.control.RadioButton;

import javafx.scene.control.ToggleGroup;

import javafx.scene.control.ChoiceBox;

import javafx.scene.input.MouseEvent;

import javafx.event.EventHandler;

import javafx.scene.control.Alert;

import javafx.scene.control.Alert.AlertType;

public class Register extends Application  {

    @Override

    public void start(Stage primaryStage) throws FileNotFoundException{

        Label heading = new Label("Register Your Details");

        heading.setStyle("-fx-font: normal bold 20px 'arial' ");

        TextField id = new TextField();

        id.setPromptText("Patient ID");

        TextField name = new TextField();

        name.setPromptText("Name");

        PasswordField password = new PasswordField();

        password.setPromptText("Password");

        PasswordField confirmPassword = new PasswordField();

        confirmPassword.setPromptText("Confirm Password");

        TextField phone = new TextField();

        phone.setPromptText("Phone");

        TextField address = new TextField();

        address.setPromptText("Address");

        Label gen = new Label("Gender: ");

        ToggleGroup gender = new ToggleGroup();

        RadioButton male = new RadioButton("Male");

        male.setToggleGroup(gender);

        RadioButton female = new RadioButton("Female");

        female.setToggleGroup(gender);

        ChoiceBox<String> state = new ChoiceBox<String>();

        state.getItems().addAll("Karnataka", "Maharashtra", "West Bengal", "Uttar Pradesh", "Kerala");

        FileInputStream inputstream1 = new FileInputStream("Images/file.png");

        Image file\_img = new Image(inputstream1);

        ImageView file\_imv = new ImageView(file\_img);

        file\_imv.setFitHeight(200);

        file\_imv.setFitWidth(150);

        Button confirm = new Button("Confirm");

        confirm.setStyle("-fx-background-color: #278fcc; -fx-text-fill: white;");

        GridPane gp = new GridPane();

        gp.setMinSize(700, 600);

        gp.setStyle("-fx-background-color: LIGHTBLUE;");

        GridPane gp1 = new GridPane();

        gp1.setMinSize(300, 600);

        gp1.setAlignment(Pos.CENTER);

        gp1.setHgap(7);

        gp1.setVgap(25);

        gp1.setStyle("-fx-background-color: WHITE;");

        GridPane gp2 = new GridPane();

        gp2.setMinSize(400, 600);

        gp2.setAlignment(Pos.CENTER);

        gp2.setStyle("-fx-background-color: LIGHTBLUE;");

        gp2.add(file\_imv, 0, 0);

        GridPane gp3 = new GridPane();

        gp3.setAlignment(Pos.CENTER);

        gp3.setHgap(7);

        gp3.add(gen, 0, 0);

        gp3.add(male, 1, 0);

        gp3.add(female, 2, 0);

        gp.add(gp1, 0, 0);

        gp.add(gp2, 1, 0);

        gp1.add(heading, 0, 0);

        gp1.add(id, 0, 2);

        gp1.add(name, 0, 3);

        gp1.add(gp3, 0, 4);

        gp1.add(password, 0, 5);

        gp1.add(confirmPassword, 0, 6);

        gp1.add(phone, 0, 7);

        gp1.add(address, 0, 8);

        gp1.add(state, 0, 9);

        gp1.add(confirm, 0, 11);

        Scene scene = new Scene(gp, 700, 600);

        primaryStage.setScene(scene);

        primaryStage.show();

        primaryStage.getIcons().add(new Image("file:Images/icon.png"));

        primaryStage.setTitle("Register");

        primaryStage.setAlwaysOnTop(true);

        gp.requestFocus();

        confirm.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                Alert a = new Alert(AlertType.ERROR);

                a.setHeaderText("Invalid Input");

                a.initOwner(primaryStage);

                //Validate

                if (id.getText().isEmpty()) {

                    a.setContentText("Enter Patient ID");

                    a.showAndWait();

                } else if (name.getText().isEmpty()) {

                    a.setContentText("Enter Name");

                    a.show();

                } else if (gender.getSelectedToggle() == null) {

                    a.setContentText("Select Gender");

                    a.show();

                } else if (password.getText().isEmpty()) {

                    a.setContentText("Enter Password");

                    a.show();

                } else if (confirmPassword.getText().isEmpty() || !confirmPassword.getText().equals(password.getText())) {

                    a.setContentText("Enter Password");

                    a.show();

                } else if (!confirmPassword.getText().equals(password.getText())) {

                    a.setContentText("Enter correct Password");

                    a.show();

                } else if (phone.getText().isEmpty()) {

                    a.setContentText("Enter Phone");

                    a.show();

                } else if (phone.getText().length() != 10 || !phone.getText().chars().allMatch(Character::isDigit)) {

                    a.setContentText("Enter valid Phone");

                    a.show();

                } else if (address.getText().isEmpty()) {

                    a.setContentText("Enter Address");

                    a.show();

                } else if (state.getValue() == null) {

                    a.setContentText("Select State");

                    a.show();

                } else {

                    RadioButton rb = (RadioButton)gender.getSelectedToggle();

                    Patient p = new Patient(id.getText(),name.getText(),phone.getText(),rb.getText(),address.getText(),state.getValue());

                    new PatientFile().addPatient(p,password.getText());

                    try{

                        Login login = new Login();

                        login.start(primaryStage);

                    } catch (FileNotFoundException e)

                    {

                        e.printStackTrace();

                    }

                }

            }

        });

    }

    public static void main(String[] args) {

        launch(args);

    }

}

**Menu :**

**User Details :**

package Screens;

import FileUtils.CurrentUser;

import FileUtils.PatientFile;

import DataModel.Patient;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.control.Label;

import javafx.stage.Stage;

import javafx.scene.layout.GridPane;

import javafx.scene.layout.HBox;

import javafx.scene.layout.VBox;

import javafx.scene.image.\*;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import javafx.geometry.Pos;

import javafx.scene.input.MouseEvent;

import javafx.event.EventHandler;

public class UserDetails extends Application {

    @Override

    public void start(Stage primaryStage) throws FileNotFoundException{

        String sid;

        sid = new CurrentUser().GetUser();

        if (sid.equals("")){

            try{

                Login login = new Login();

                login.start(primaryStage);

            } catch (FileNotFoundException e)

            {

                e.printStackTrace();

            }

        } else {

            Patient p = new PatientFile().GetDetails(sid);

            Label heading = new Label("User Details");

            heading.setStyle("-fx-font: normal bold 20px 'arial' ");

            FileInputStream inputstream = new FileInputStream("Images/user\_profile.png");

            Image user\_img = new Image(inputstream);

            ImageView user\_imv = new ImageView(user\_img);

            user\_imv.setFitHeight(200);

            user\_imv.setFitWidth(200);

            VBox top = new VBox(20);

            top.getChildren().addAll(user\_imv, heading);

            top.setAlignment(Pos.CENTER);

            Label id = dlabel("Patient ID :");

            Label Pid = plabel(p.getId());

            HBox hid = new HBox(5);

            hid.getChildren().addAll(id, Pid);

            Label name = dlabel("Name :");

            Label Pname = plabel(p.getName());

            HBox hname = new HBox(5);

            hname.getChildren().addAll(name, Pname);

            Label gender = dlabel("Gender :");

            Label Pgender = plabel(p.getGender());

            HBox hgender = new HBox(5);

            hgender.getChildren().addAll(gender, Pgender);

            Label phone = dlabel("Phone :");

            Label Pphone = plabel(p.getPhone());

            HBox hphone = new HBox(5);

            hphone.getChildren().addAll(phone, Pphone);

            Label address = dlabel("Address :");

            Label Paddress = plabel(p.getAddress());

            HBox haddress = new HBox(5);

            haddress.getChildren().addAll(address, Paddress);

            Label state = dlabel("State :");

            Label Pstate = plabel(p.getState());

            HBox hstate = new HBox(5);

            hstate.getChildren().addAll(state, Pstate);

            Button ok = new Button("OK");

            ok.setStyle("-fx-background-color: #278fcc; -fx-text-fill: white;");

            VBox element = new VBox(20);

            element.getChildren().addAll(hid, hname, hgender, hphone, haddress, hstate, ok);

            element.setAlignment(Pos.CENTER);

            VBox all = new VBox(30);

            all.getChildren().addAll(top,element);

            all.setAlignment(Pos.CENTER);

            GridPane gp = new GridPane();

            gp.setMinSize(700, 600);

            gp.setStyle("-fx-background-color: LIGHTBLUE;");

            gp.setAlignment(Pos.CENTER);

            gp.add(all,0,0);

            primaryStage.setScene(new Scene(gp,700,600));

            primaryStage.show();

            primaryStage.getIcons().add(new Image("file:Images/icon.png"));

            primaryStage.setTitle("User Details");

            primaryStage.setAlwaysOnTop(true);

            ok.setOnMouseClicked(new EventHandler<MouseEvent>(){

                @Override

                public void handle(MouseEvent arg0) {

                    try{

                        Menu menu = new Menu();

                        menu.start(primaryStage);

                    } catch (FileNotFoundException e)

                    {

                        e.printStackTrace();

                    }

                }

            });

        }

    }

    public Label plabel(String s){

        Label label = new Label(s);

        label.setWrapText(false);

        label.setStyle("-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 1;" +

                "-fx-border-radius: 1;" +

                "-fx-border-color: #5494e3;");

        return label;

    }

    public Label dlabel(String s){

        Label label = new Label(s);

        label.setStyle("-fx-font: normal bold 15px 'arial' ");

        return label;

    }

    public static void main(String[] args) {

        launch(args);

    }

}

**Appointment :**

package Screens;

import FileUtils.CurrentUser;

import FileUtils.PatientFile;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.control.Label;

import javafx.scene.control.TextArea;

import javafx.scene.control.ChoiceBox;

import javafx.scene.control.RadioButton;

import javafx.scene.control.ToggleGroup;

import javafx.stage.Stage;

import javafx.scene.layout.GridPane;

import javafx.scene.layout.HBox;

import javafx.scene.layout.VBox;

import javafx.scene.image.\*;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import javafx.geometry.Pos;

import javafx.scene.input.MouseEvent;

import javafx.event.EventHandler;

import javafx.scene.layout.Priority;

import javafx.scene.layout.Region;

import java.util.ArrayList;

import java.util.List;

import java.util.Arrays;

import javafx.scene.control.Alert;

import javafx.scene.control.Alert.AlertType;

public class Appointment extends Application {

    @Override

    public void start(Stage primaryStage) throws FileNotFoundException{

        String sid;

        String status;

        sid = new CurrentUser().GetUser();

        if (sid.equals("")){

            try{

                Login login = new Login();

                login.start(primaryStage);

            } catch (FileNotFoundException e)

            {

                e.printStackTrace();

            }

        } else {

            status = new PatientFile().GetStatus(sid);

            if (status.equals("")){

                try{

                    Menu menu = new Menu();

                    menu.start(primaryStage);

                } catch (FileNotFoundException e)

                {

                    e.printStackTrace();

                }

            } else if (status.equals("PENDING")){

                Label heading = new Label("Request Pending");

                heading.setStyle("-fx-font: normal bold 30px 'arial' ");

                Button back = new Button("Back to Menu");

                back.setStyle("-fx-background-color: #278fcc; -fx-text-fill: white;");

                VBox elements = new VBox(20);

                elements.setAlignment(Pos.CENTER);

                elements.getChildren().addAll(heading,back);

                primaryStage.setScene(new Scene(elements,700,600));

                primaryStage.show();

                primaryStage.getIcons().add(new Image("file:Images/icon.png"));

                primaryStage.setTitle("Appointment");

                primaryStage.setAlwaysOnTop(true);

                back.setOnMouseClicked(new EventHandler<MouseEvent>(){

                    @Override

                    public void handle(MouseEvent arg0) {

                        try{

                            Menu menu = new Menu();

                            menu.start(primaryStage);

                        } catch (FileNotFoundException e)

                        {

                            e.printStackTrace();

                        }

                    }

                });

            } else if (status.equals("DONE")){

                Label heading = new Label("Feedback");

                heading.setStyle("-fx-font: normal bold 30px 'arial' ");

                String s = new PatientFile().GetFeedback(sid);

                TextArea fb = new TextArea(s);

                fb.setEditable(false);

                Button back = new Button("Back to Menu");

                back.setStyle("-fx-background-color: #278fcc; -fx-text-fill: white;");

                VBox elements = new VBox(20);

                elements.setAlignment(Pos.CENTER);

                elements.setStyle("-fx-background-color: LIGHTBLUE;");

                elements.getChildren().addAll(heading,fb,back);

                primaryStage.setScene(new Scene(elements,700,600));

                primaryStage.show();

                primaryStage.getIcons().add(new Image("file:Images/icon.png"));

                primaryStage.setTitle("Appointment");

                primaryStage.setAlwaysOnTop(true);

                back.setOnMouseClicked(new EventHandler<MouseEvent>(){

                    @Override

                    public void handle(MouseEvent arg0) {

                        new PatientFile().UpdateStatus(sid,"FREE");

                        try{

                            Menu menu = new Menu();

                            menu.start(primaryStage);

                        } catch (FileNotFoundException e)

                        {

                            e.printStackTrace();

                        }

                    }

                });

            } else {

                Label heading = new Label("Appointment Form");

                heading.setStyle("-fx-font: normal bold 20px 'arial' ");

                VBox elements = new VBox(15);

                elements.setAlignment(Pos.TOP\_CENTER);

                HBox e1 = new HBox(2);

                Label symptom = new Label("Symptom :");

                symptom.setStyle("-fx-font: normal bold 15px 'arial' ");

                TextArea symptoms = new TextArea("Mention symptoms in seperate line");

                symptoms.setPrefHeight(100);

                e1.getChildren().addAll(symptom, symptoms);

                HBox e2 = new HBox(2);

                Label specialization = new Label("Specialization :");

                specialization.setStyle("-fx-font: normal bold 15px 'arial' ");

                ChoiceBox<String> specializations = new ChoiceBox<String>();

                specializations.getItems().addAll("General", "ENT", "Cardiologists", "Dermatologists", "Endocrinologists");

                e2.getChildren().addAll(specialization, specializations);

                VBox e3 = new VBox(2);

                Label p1 = new Label("I would like :");

                p1.setStyle("-fx-font: normal bold 15px 'arial' ");

                ToggleGroup choice1 = new ToggleGroup();

                RadioButton op1 = new RadioButton("New patient appointment");

                op1.setToggleGroup(choice1);

                RadioButton op2 = new RadioButton("Routine Checkup");

                op2.setToggleGroup(choice1);

                RadioButton op3 = new RadioButton("Comprehensive Health Exam");

                op3.setToggleGroup(choice1);

                e3.getChildren().addAll(p1, op1, op2, op3);

                VBox e4 = new VBox(2);

                Label p = new Label("Have you previously attended our facility :");

                p.setStyle("-fx-font: normal bold 15px 'arial' ");

                ToggleGroup choice = new ToggleGroup();

                RadioButton yes = new RadioButton("Yes");

                yes.setToggleGroup(choice);

                RadioButton no = new RadioButton("No");

                no.setToggleGroup(choice);

                e4.getChildren().addAll(p, yes, no);

                VBox e5 = new VBox(2);

                Label p2 = new Label("Best time to call you :");

                p2.setStyle("-fx-font: normal bold 15px 'arial' ");

                ToggleGroup choice2 = new ToggleGroup();

                RadioButton morning = new RadioButton("Morning");

                morning.setToggleGroup(choice2);

                RadioButton noon = new RadioButton("Afternoon");

                noon.setToggleGroup(choice2);

                RadioButton eve = new RadioButton("Evening");

                eve.setToggleGroup(choice2);

                e5.getChildren().addAll(p2, morning, noon, eve);

                Button confirm = new Button("Confirm");

                confirm.setStyle("-fx-background-color: #278fcc; -fx-text-fill: white;");

                Button cancel = new Button("Cancel");

                cancel.setStyle("-fx-background-color: #278fcc; -fx-text-fill: white;");

                Region region1 = new Region();

                HBox.setHgrow(region1, Priority.ALWAYS);

                Region region2 = new Region();

                HBox.setHgrow(region2, Priority.ALWAYS);

                HBox btn = new HBox(10);

                btn.getChildren().addAll(region1,confirm,cancel,region2);

                elements.getChildren().addAll(e2,e3,e4,e5);

                FileInputStream inputstream1 = new FileInputStream("Images/appointment.png");

                Image app\_img = new Image(inputstream1);

                ImageView app\_imv = new ImageView(app\_img);

                app\_imv.setFitHeight(250);

                app\_imv.setFitWidth(250);

                GridPane gp1 = new GridPane();

                gp1.add(elements,0,0);

                gp1.add(app\_imv,1,0);

                gp1.setAlignment(Pos.CENTER);

                VBox c = new VBox(15);

                c.getChildren().addAll(gp1,btn);

                c.setAlignment(Pos.CENTER);

                VBox all = new VBox(25);

                all.getChildren().addAll(heading,e1,c);

                all.setAlignment(Pos.TOP\_CENTER);

                GridPane gp = new GridPane();

                gp.setMinSize(700, 600);

                gp.setAlignment(Pos.CENTER);

                gp.setStyle("-fx-background-color: LIGHTBLUE;");

                gp.add(all,0,0);

                primaryStage.setScene(new Scene(gp,700,600));

                primaryStage.show();

                primaryStage.getIcons().add(new Image("file:Images/icon.png"));

                primaryStage.setTitle("Appointment");

                primaryStage.setAlwaysOnTop(true);

                confirm.setOnMouseClicked(new EventHandler<MouseEvent>(){

                    @Override

                    public void handle(MouseEvent arg0) {

                        Alert a = new Alert(AlertType.ERROR);

                        a.setHeaderText("Invalid Input");

                        a.initOwner(primaryStage);

                        if (symptoms.getText().isEmpty()) {

                            a.setContentText("Enter Symptoms");

                            a.showAndWait();

                        } else if (specializations.getValue() == null) {

                            a.setContentText("Select Specialization");

                            a.show();

                        } else if (choice.getSelectedToggle() == null || choice1.getSelectedToggle() == null || choice2.getSelectedToggle() == null) {

                            a.setContentText("Answer all Questions");

                            a.show();

                        } else {

                            //Convert Symptom to list

                            String str[] = symptoms.getText().split("\n");

                            List<String> al = new ArrayList<String>();

                            al = Arrays.asList(str);

                            PatientFile pf = new PatientFile();

                            pf.AddSymptoms(al,sid);

                            pf.AddAppointmentList(sid);

                            pf.UpdateStatus(sid,"PENDING");

                            try{

                                Menu menu = new Menu();

                                menu.start(primaryStage);

                            } catch (FileNotFoundException e)

                            {

                                e.printStackTrace();

                            }

                        }

                    }

                });

                cancel.setOnMouseClicked(new EventHandler<MouseEvent>(){

                    @Override

                    public void handle(MouseEvent arg0) {

                        try{

                            Menu menu = new Menu();

                            menu.start(primaryStage);

                        } catch (FileNotFoundException e)

                        {

                            e.printStackTrace();

                        }

                    }

                });

            }

        }

    }

    public static void main(String[] args) {

        launch(args);

    }

}

**About Us :**

package Screens;

import FileUtils.CurrentUser;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.control.Label;

import javafx.stage.Stage;

import javafx.scene.image.\*;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import javafx.geometry.Pos;

import javafx.beans.value.ChangeListener;

import javafx.scene.paint.Color;

import java.lang.String;

import javafx.scene.layout.VBox;

import javafx.scene.layout.HBox;

import javafx.scene.control.ScrollPane;

import javafx.scene.input.MouseEvent;

import javafx.event.EventHandler;

public class AboutUs extends Application  {

    @Override

    public void start(Stage primaryStage) throws FileNotFoundException{

        //Main Screen

        ScrollPane scrollPane = new ScrollPane();

        //Contents of scrollPane

        VBox vbox = new VBox();

        vbox.setPrefSize(950, 860);

        vbox.setAlignment(Pos.TOP\_CENTER);

        vbox.setSpacing(20);

        vbox.setStyle("-fx-background-color: LIGHTBLUE;");

        scrollPane.setContent(vbox);

        //Contents of vbox

        Label heading = new Label("Question? Look Here.");

        heading.setStyle("-fx-font: normal bold 30px 'arial';"

                        + "-fx-text-fill: #d94364;");

        HBox hbox = new HBox(2);

        hbox.setAlignment(Pos.BASELINE\_LEFT);

        hbox.setSpacing(10);

        //contents of hbox

        VBox vbox1 = new VBox();

        vbox1.setAlignment(Pos.CENTER);

        vbox1.setPrefHeight(215);

        vbox1.setPrefWidth(250);

        vbox1.setSpacing(15);

        vbox1.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 2;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 5;" +

                "-fx-border-color: #5494e3;");

        VBox vbox2 = new VBox();

        vbox2.setAlignment(Pos.CENTER);

        vbox2.setPrefWidth(625);

        vbox2.setSpacing(20);

        vbox2.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 2;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 5;" +

                "-fx-border-color: #5494e3;");

        //contents of vbox1

        Label general = new Label("> General");

        general.setStyle("-fx-font: normal bold 16px 'arial' ");

        Label corona = new Label("Coronavirus");

        corona.setStyle("-fx-font: normal bold 16px 'arial';"

                + "-fx-text-fill : black");

        Label mv = new Label("Mission & Vision");

        mv.setStyle("-fx-font: normal bold 16px 'arial';"

                + "-fx-text-fill : black");

        Label contact = new Label("Contact Us");

        contact.setStyle("-fx-font: normal bold 16px 'arial';"

                + "-fx-text-fill : black");

        Label back = new Label("<- Back to Menu");

        back.setStyle("-fx-font: normal bold 16px 'arial';"

                + "-fx-text-fill : black");

        //contents of vbox2(general)

        VBox vbox3 = new VBox();

        vbox3.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        VBox vbox4 = new VBox();

        vbox4.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        VBox vbox5 = new VBox();

        vbox5.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        VBox vbox6 = new VBox();

        vbox6.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        HBox hbox1 = new HBox(2);

        hbox1.setAlignment(Pos.BASELINE\_LEFT);

        hbox1.setSpacing(10);

        HBox hbox2 = new HBox(2);

        hbox2.setAlignment(Pos.BASELINE\_LEFT);

        hbox2.setSpacing(10);

        HBox hbox3 = new HBox(2);

        hbox3.setAlignment(Pos.BASELINE\_LEFT);

        hbox3.setSpacing(10);

        HBox hbox4 = new HBox(2);

        hbox4.setAlignment(Pos.BASELINE\_LEFT);

        hbox4.setSpacing(10);

        //question naming format-

        //c1q1- category 1, question 1

        //c1a1- category 1, answer 1

        String q1 = "How are diseases prevented?";

        Label c1q1 = new Label(q1);

        c1q1.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String a1 = "\nWashing your hands with soap and water, thoroughly and frequently. \n"

                + "Covering your nose and mouth when you sneeze or cough. \n"

                + "Disinfecting frequently touched surfaces in your home and workplace. \n"

                + "Avoiding contact with sick people or sharing personal items with them.";

        Label c1a1 = new Label(a1);

        c1a1.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        String q2 = "How to overcome any illness?";

        Label c1q2 = new Label(q2);

        c1q2.setStyle("-fx-font: normal bold 16px 'arial';");

        String a2 = "\nMaintain a healthy diet.\r\n"

                + "Get enough sleep\r\n"

                + "Positive thinking\r\n"

                + "Excercise daily\r\n"

                + "Get proper medication\r\n"

                + "Get the support you need\r\n";

        Label c1a2 = new Label(a2);

        c1a2.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        String q3 = "What is the best treatment for chronic disease?";

        Label c1q3 = new Label(q3);

        c1q3.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String a3 = "\nTreatment of chronic illness comes in many forms including surgery, physical\n"

                + "therapy, psychological therapy and radiotherapy. However, one of the most\n"

                + "common treatment forms is the use of medication.";

        Label c1a3 = new Label(a3);

        c1a3.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        String q4 = "How can you help someone with help problems?";

        Label c1q4 = new Label(q4);

        c1q4.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String a4 = "\nAsk what kind of friend the person needs you to be\r\n"

                + "Offer specific help\r\n"

                + "Make plans\r\n"

                + "Visit when possible\r\n"

                + "Send notes and cards\r\n"

                + "Offer to chauffeur or even attend appointments\r\n"

                + "";

        Label c1a4 = new Label(a4);

        c1a4.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        FileInputStream inputstream = new FileInputStream("Images/plus.png");

        Image plus\_img = new Image(inputstream);

        ImageView plus\_imv = new ImageView(plus\_img);

        plus\_imv.setFitHeight(20);

        plus\_imv.setFitWidth(20);

        ImageView plus1\_imv = new ImageView(plus\_img);

        plus1\_imv.setFitHeight(20);

        plus1\_imv.setFitWidth(20);

        ImageView plus2\_imv = new ImageView(plus\_img);

        plus2\_imv.setFitHeight(20);

        plus2\_imv.setFitWidth(20);

        ImageView plus3\_imv = new ImageView(plus\_img);

        plus3\_imv.setFitHeight(20);

        plus3\_imv.setFitWidth(20);

        FileInputStream inputstream1 = new FileInputStream("Images/minus.png");

        Image minus\_img = new Image(inputstream1);

        hbox1.getChildren().add(c1q1);

        hbox1.getChildren().add(plus\_imv);

        hbox2.getChildren().add(c1q2);

        hbox2.getChildren().add(plus1\_imv);

        hbox3.getChildren().add(c1q3);

        hbox3.getChildren().add(plus2\_imv);

        hbox4.getChildren().add(c1q4);

        hbox4.getChildren().add(plus3\_imv);

        vbox3.getChildren().add(hbox1);

        vbox3.getChildren().add(c1a1);

        vbox3.setPrefHeight(80);

        vbox3.setPrefWidth(500);

        c1a1.setVisible(false);

        vbox4.getChildren().add(hbox2);

        vbox4.getChildren().add(c1a2);

        vbox4.setPrefHeight(80);

        vbox4.setPrefWidth(500);

        c1a2.setVisible(false);

        vbox5.getChildren().add(hbox3);

        vbox5.getChildren().add(c1a3);

        vbox5.setPrefHeight(80);

        vbox5.setPrefWidth(500);

        c1a3.setVisible(false);

        vbox6.getChildren().add(hbox4);

        vbox6.getChildren().add(c1a4);

        vbox6.setPrefHeight(80);

        vbox6.setPrefWidth(500);

        c1a4.setVisible(false);

        vbox2.getChildren().add(vbox3);

        vbox2.getChildren().add(vbox4);

        vbox2.getChildren().add(vbox5);

        vbox2.getChildren().add(vbox6);

        //contents of Coronavirus

        VBox vvboxx3 = new VBox();

        vvboxx3.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        vvboxx3.setAlignment(Pos.CENTER);

        HBox hhbox = new HBox();

        hhbox.setAlignment(Pos.CENTER);

        hhbox.setSpacing(10);

        Label hhead = new Label("Coronavirus Updates");

        hhead.setStyle("-fx-font: normal bold 18px 'arial' ");

        hhead.setAlignment(Pos.BASELINE\_CENTER);

        vvboxx3.getChildren().add(hhead);

        vvboxx3.getChildren().add(hhbox);

        VBox cvbox3 = new VBox();

        cvbox3.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #4a5ad9;");

        cvbox3.setAlignment(Pos.BASELINE\_LEFT);

        Label tc = new Label("Total Cases: 46,482,200");

        tc.setStyle("-fx-font: normal bold 12px 'arial';" +

                "-fx-text-fill : #4a5ad9;");

        tc.setAlignment(Pos.CENTER);

        cvbox3.getChildren().add(tc);

        VBox cvbox4 = new VBox();

        cvbox4.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #53d43f;");

        cvbox4.setAlignment(Pos.BASELINE\_CENTER);

        Label tr = new Label("Total Recovered: 33,556,292");

        tr.setStyle("-fx-font: normal bold 12px 'arial';" +

                "-fx-text-fill : #53d43f;");

        tr.setAlignment(Pos.CENTER);

        cvbox4.getChildren().add(tr);

        VBox cvbox5 = new VBox();

        cvbox5.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #d43a2c;");

        cvbox5.setAlignment(Pos.BASELINE\_RIGHT);

        Label td = new Label("Total Deaths: 1,201,759");

        td.setStyle("-fx-font: normal bold 12px 'arial';" +

                "-fx-text-fill : #d43a2c");

        td.setAlignment(Pos.CENTER);

        cvbox5.getChildren().add(td);

        hhbox.getChildren().add(cvbox3);

        hhbox.getChildren().add(cvbox4);

        hhbox.getChildren().add(cvbox5);

        VBox vvbox3 = new VBox();

        vvbox3.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        VBox vvbox4 = new VBox();

        vvbox4.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        VBox vvbox5 = new VBox();

        vvbox5.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        HBox hhbox1 = new HBox(2);

        hhbox1.setAlignment(Pos.BASELINE\_LEFT);

        hhbox1.setSpacing(10);

        HBox hhbox2 = new HBox(2);

        hhbox2.setAlignment(Pos.BASELINE\_LEFT);

        hhbox2.setSpacing(10);

        HBox hhbox3 = new HBox(2);

        hhbox3.setAlignment(Pos.BASELINE\_LEFT);

        hhbox3.setSpacing(10);

        HBox hhbox4 = new HBox(2);

        hhbox4.setAlignment(Pos.BASELINE\_LEFT);

        hhbox4.setSpacing(10);

        String cq1 = "What are the symptoms of the coronavirus?";

        Label c2q1 = new Label(cq1);

        c2q1.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String ca1 = "\nMost common symptoms:\r\n"

                + "\tFever\r\n"

                + "\tdry cough\r\n"

                + "\ttiredness\r\n"

                + "Less common symptoms:\r\n"

                + "\taches and pains\r\n"

                + "\tsore throat\r\n"

                + "\tDiarrhoea\r\n"

                + "\tConjunctivitis\r\n"

                + "\tHeadache\r\n"

                + "\tloss of taste or smell\r\n";

        Label c2a1 = new Label(ca1);

        c2a1.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        String cq2 = "How to prevent catching the coronavirus?";

        Label c2q2 = new Label(cq2);

        c2q2.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String ca2 = "\nClean your hands often. Use soap and water, or an alcohol-based hand rub.\r\n"

                + "Maintain a safe distance from anyone who is coughing or sneezing.\r\n"

                + "Wear a mask when physical distancing is not possible.\r\n"

                + "Dont touch your eyes, nose or mouth.\n"

                + "Cover your nose and mouth with your elbow or a tissue when you cough or sneeze.\r\n"

                + "Stay home if you feel unwell.\r\n";

        Label c2a2 = new Label(ca2);

        c2a2.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        String cq3 = "What are the treatments of the coronavirus?";

        Label c2q3 = new Label(cq3);

        c2q3.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String ca3 = "\nIf you feel sick you should rest, drink plenty of fluid, and eat nutritious food.\n"

                + "Stay in a separate room fromother family members, and use a dedicated \n"

                + "bathroom if possible.\n"

                + "Clean and disinfect frequently touched surfaces.\n"

                + "Everyone should keep a healthy lifestyle at home.\n"

                + "Maintain a healthy diet, sleep, stay active, and make social contact  with loved ones\n"

                + "through the phone or internet. \n"

                + "Children need extra love and attention from adults during difficult times. \n"

                + "Keep to regular routines and schedules as much as possible.\r\n";

        Label c2a3 = new Label(ca3);

        c2a3.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        ImageView cplus\_imv = new ImageView(plus\_img);

        cplus\_imv.setFitHeight(20);

        cplus\_imv.setFitWidth(20);

        ImageView cplus1\_imv = new ImageView(plus\_img);

        cplus1\_imv.setFitHeight(20);

        cplus1\_imv.setFitWidth(20);

        ImageView cplus2\_imv = new ImageView(plus\_img);

        cplus2\_imv.setFitHeight(20);

        cplus2\_imv.setFitWidth(20);

        hhbox1.getChildren().add(c2q1);

        hhbox1.getChildren().add(cplus\_imv);

        hhbox2.getChildren().add(c2q2);

        hhbox2.getChildren().add(cplus1\_imv);

        hhbox3.getChildren().add(c2q3);

        hhbox3.getChildren().add(cplus2\_imv);

        vvbox3.getChildren().add(hhbox1);

        vvbox3.getChildren().add(c2a1);

        vvbox3.setPrefHeight(80);

        vvbox3.setPrefWidth(500);

        c2a1.setVisible(false);

        vvbox4.getChildren().add(hhbox2);

        vvbox4.getChildren().add(c2a2);

        vvbox4.setPrefHeight(80);

        vvbox4.setPrefWidth(500);

        c2a2.setVisible(false);

        vvbox5.getChildren().add(hhbox3);

        vvbox5.getChildren().add(c2a3);

        vvbox5.setPrefHeight(80);

        vvbox5.setPrefWidth(500);

        c2a3.setVisible(false);

        //contents of Mission & Vision

        VBox mvbox3 = new VBox();

        mvbox3.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        VBox mvbox4 = new VBox();

        mvbox4.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        VBox mvbox5 = new VBox();

        mvbox5.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        HBox mhbox1 = new HBox(2);

        mhbox1.setAlignment(Pos.BASELINE\_LEFT);

        mhbox1.setSpacing(10);

        HBox mhbox2 = new HBox(2);

        mhbox2.setAlignment(Pos.BASELINE\_LEFT);

        mhbox2.setSpacing(10);

        HBox mhbox3 = new HBox(2);

        mhbox3.setAlignment(Pos.BASELINE\_LEFT);

        mhbox3.setSpacing(10);

        String mq1 = "Mission";

        Label c3q1 = new Label(mq1);

        c3q1.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String ma1 = "\nWe share our experience serving patients that makes you happy, giving service \n"

                + "that makes you smile, in passionate pursuit of our mission, showing love and caring\n"

                + " in all our actions, to enrich as many lives as we possibly can. ";

        Label c3a1 = new Label(ma1);

        c3a1.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        String mq2 = "Vision";

        Label c3q2 = new Label(mq2);

        c3q2.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String ma2 = "\nTo create and maintain a framework where health care professionals can practice\n"

                + "their skills in the right spirit with empathy, pride and passion.\n"

                + "To be able to adapt to dynamic changes in health care delivery not only\n"

                +"in curing illnesses but also in preserving health. ";

        Label c3a2 = new Label(ma2);

        c3a2.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        String mq3 = "Strength";

        Label c3q3 = new Label(mq3);

        c3q3.setStyle("-fx-font: normal bold 16px 'arial' ;");

        String ma3 = "\nEmployees are recognized as the most valuable asset in the organization.\n"

                + "We are committed to create & maintain a culture that is conducive to learning.";

        Label c3a3 = new Label(ma3);

        c3a3.setStyle("-fx-text-fill: #2f6de0;"

                    + "-fx-font: normal bold 15px 'arial';");

        ImageView mplus\_imv = new ImageView(plus\_img);

        mplus\_imv.setFitHeight(20);

        mplus\_imv.setFitWidth(20);

        ImageView mplus1\_imv = new ImageView(plus\_img);

        mplus1\_imv.setFitHeight(20);

        mplus1\_imv.setFitWidth(20);

        ImageView mplus2\_imv = new ImageView(plus\_img);

        mplus2\_imv.setFitHeight(20);

        mplus2\_imv.setFitWidth(20);

        ImageView mplus3\_imv = new ImageView(plus\_img);

        mplus3\_imv.setFitHeight(20);

        mplus3\_imv.setFitWidth(20);

        mhbox1.getChildren().add(c3q1);

        mhbox1.getChildren().add(mplus\_imv);

        mhbox2.getChildren().add(c3q2);

        mhbox2.getChildren().add(mplus1\_imv);

        mhbox3.getChildren().add(c3q3);

        mhbox3.getChildren().add(mplus2\_imv);

        mvbox3.getChildren().add(mhbox1);

        mvbox3.getChildren().add(c3a1);

        mvbox3.setPrefHeight(80);

        mvbox3.setPrefWidth(500);

        c3a1.setVisible(false);

        mvbox4.getChildren().add(mhbox2);

        mvbox4.getChildren().add(c3a2);

        mvbox4.setPrefHeight(80);

        mvbox4.setPrefWidth(500);

        c3a2.setVisible(false);

        mvbox5.getChildren().add(mhbox3);

        mvbox5.getChildren().add(c3a3);

        mvbox5.setPrefHeight(80);

        mvbox4.setPrefWidth(500);

        c3a3.setVisible(false);

        //Contents of Contact Us

        Label uhead = new Label("Developors");

        uhead.setAlignment(Pos.BASELINE\_CENTER);

        uhead.setStyle("-fx-font: normal bold 15px 'arial';"

                + "-fx-text-fill : black");

        HBox uhbox = new HBox(2);

        uhbox.setAlignment(Pos.CENTER);

        uhbox.setSpacing(10);

        VBox uvbox4 = new VBox();

        uvbox4.setAlignment(Pos.CENTER);

        uvbox4.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        VBox uvbox5 = new VBox();

        uvbox5.setAlignment(Pos.CENTER);

        uvbox5.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 2;" +

                "-fx-border-color: #5494e3;");

        FileInputStream inputstream2 = new FileInputStream("Images/man.png");

        Image man\_img = new Image(inputstream2);

        ImageView man\_imv = new ImageView(man\_img);

        man\_imv.setFitHeight(200);

        man\_imv.setFitWidth(150);

        ImageView man1\_imv = new ImageView(man\_img);

        man1\_imv.setFitHeight(200);

        man1\_imv.setFitWidth(150);

        String name1 = "\nName: Abhinav Rathi\n";

        Label n1 = new Label(name1);

        n1.setAlignment(Pos.CENTER);

        Label e1 = new Label("Email: abhinav.rathi@science.christuniversity.in");

        e1.setAlignment(Pos.CENTER);

        String name2 = "\nName: Vedant Mehrotra\n";

        Label n2 = new Label(name2);

        n2.setAlignment(Pos.CENTER);

        Label e2 = new Label("Email: vedant.mehrotra@science.christuniversity.in");

        e2.setAlignment(Pos.CENTER);

        uvbox4.getChildren().add(man\_imv);

        uvbox4.getChildren().add(n1);

        uvbox4.getChildren().add(e1);

        uvbox5.getChildren().add(man1\_imv);

        uvbox5.getChildren().add(n2);

        uvbox5.getChildren().add(e2);

        uhbox.getChildren().add(uvbox4);

        uhbox.getChildren().add(uvbox5);

        // Adding Contents of vbox

        vbox1.getChildren().addAll(general,corona,mv,contact,back);

        hbox.getChildren().add(vbox1);

        hbox.getChildren().add(vbox2);

        vbox.getChildren().add(heading);

        vbox.getChildren().add(hbox);

        Scene scene = new Scene(scrollPane, 950, 750);

        primaryStage.setScene(scene);

        primaryStage.show();

        primaryStage.getIcons().add(new Image("file:Images/icon.png"));

        primaryStage.setTitle("About Us");

        primaryStage.setAlwaysOnTop(true);

        //event handling starts here

        hbox1.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(plus\_imv.getImage() == plus\_img) {

                    vbox3.setPrefHeight(150);

                    c1a1.setVisible(true);

                    plus\_imv.setImage(minus\_img);

                }

                else {

                    vbox3.setPrefHeight(70);

                    c1a1.setVisible(false);

                    plus\_imv.setImage(plus\_img);

                }

            }

        });

        hbox2.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(plus1\_imv.getImage() == plus\_img) {

                    vbox4.setPrefHeight(185);

                    c1a2.setVisible(true);

                    plus1\_imv.setImage(minus\_img);

                }

                else {

                    vbox4.setPrefHeight(70);

                    c1a2.setVisible(false);

                    plus1\_imv.setImage(plus\_img);

                }

            }

        });

        hbox3.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(plus2\_imv.getImage() == plus\_img) {

                    vbox5.setPrefHeight(135);

                    c1a3.setVisible(true);

                    plus2\_imv.setImage(minus\_img);

                }

                else {

                    vbox5.setPrefHeight(70);

                    c1a3.setVisible(false);

                    plus2\_imv.setImage(plus\_img);

                }

            }

        });

        hbox4.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(plus3\_imv.getImage() == plus\_img) {

                    vbox6.setPrefHeight(185);

                    c1a4.setVisible(true);

                    plus3\_imv.setImage(minus\_img);

                }

                else {

                    vbox6.setPrefHeight(70);

                    c1a4.setVisible(false);

                    plus3\_imv.setImage(plus\_img);

                }

            }

        });

        hhbox1.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(cplus\_imv.getImage() == plus\_img) {

                    vbox.setPrefHeight(980);

                    //vbox2.setPrefHeight(980);

                    vvbox3.setPrefHeight(300);

                    c2a1.setVisible(true);

                    cplus\_imv.setImage(minus\_img);

                }

                else {

                    vbox.setPrefHeight(750);

                    vvbox3.setPrefHeight(70);

                    c2a1.setVisible(false);

                    cplus\_imv.setImage(plus\_img);

                }

            }

        });

        hhbox2.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(cplus1\_imv.getImage() == plus\_img) {

                    vbox.setPrefHeight(980);

                    //vbox2.setPrefHeight(980);

                    vvbox4.setPrefHeight(180);

                    c2a2.setVisible(true);

                    cplus1\_imv.setImage(minus\_img);

                }

                else {

                    vbox.setPrefHeight(750);

                    vvbox4.setPrefHeight(70);

                    c2a2.setVisible(false);

                    cplus1\_imv.setImage(plus\_img);

                }

            }

        });

        hhbox3.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(cplus2\_imv.getImage() == plus\_img) {

                    vbox.setPrefHeight(980);

                    //vbox2.setPrefHeight(980);

                    vvbox5.setPrefHeight(200);

                    c2a3.setVisible(true);

                    cplus2\_imv.setImage(minus\_img);

                }

                else {

                    vbox.setPrefHeight(750);

                    vvbox5.setPrefHeight(70);

                    c2a3.setVisible(false);

                    cplus2\_imv.setImage(plus\_img);

                }

            }

        });

        mhbox1.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(mplus\_imv.getImage() == plus\_img) {

                    mvbox3.setPrefHeight(140);

                    c3a1.setVisible(true);

                    mplus\_imv.setImage(minus\_img);

                }

                else {

                    mvbox3.setPrefHeight(70);

                    c3a1.setVisible(false);

                    mplus\_imv.setImage(plus\_img);

                }

            }

        });

        mhbox2.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(mplus1\_imv.getImage() == plus\_img) {

                    mvbox4.setPrefHeight(145);

                    c3a2.setVisible(true);

                    mplus1\_imv.setImage(minus\_img);

                }

                else {

                    mvbox4.setPrefHeight(70);

                    c3a2.setVisible(false);

                    mplus1\_imv.setImage(plus\_img);

                }

            }

        });

        mhbox3.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                if(mplus2\_imv.getImage() == plus\_img) {

                    mvbox5.setPrefHeight(120);

                    c3a3.setVisible(true);

                    mplus2\_imv.setImage(minus\_img);

                }

                else {

                    mvbox5.setPrefHeight(70);

                    c3a3.setVisible(false);

                    mplus2\_imv.setImage(plus\_img);

                }

            }

        });

        general.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                general.setText("> General");

                corona.setText("Coronavirus");

                mv.setText("Mission & Vision");

                contact.setText("Contact Us");

                vbox2.getChildren().clear();

                vbox2.getChildren().add(vbox3);

                vbox2.getChildren().add(vbox4);

                vbox2.getChildren().add(vbox5);

                vbox2.getChildren().add(vbox6);

            }

        });

        general.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                general.setStyle("-fx-font: normal bold 18px 'arial';"

                        + "-fx-text-fill : #1487d9");

            } else {

                general.setStyle("-fx-font: normal bold 16px 'arial';"

                        + "-fx-text-fill : black");

            }

        });

        corona.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                general.setText("General");

                corona.setText("> Coronavirus");

                mv.setText("Mission & Vision");

                contact.setText("Contact Us");

                vbox2.getChildren().clear();

                vbox2.getChildren().add(vvboxx3);

                vbox2.getChildren().add(vvbox3);

                vbox2.getChildren().add(vvbox4);

                vbox2.getChildren().add(vvbox5);

            }

        });

        corona.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                corona.setStyle("-fx-font: normal bold 18px 'arial';"

                        + "-fx-text-fill : #1487d9");

            } else {

                corona.setStyle("-fx-font: normal bold 16px 'arial';"

                        + "-fx-text-fill : black");

            }

        });

        mv.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                general.setText("General");

                corona.setText("Coronavirus");

                mv.setText("> Mission & Vision");

                contact.setText("Contact Us");

                vbox2.getChildren().clear();

                vbox2.getChildren().add(mvbox3);

                vbox2.getChildren().add(mvbox4);

                vbox2.getChildren().add(mvbox5);

            }

        });

        mv.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                mv.setStyle("-fx-font: normal bold 18px 'arial';"

                        + "-fx-text-fill : #1487d9");

            } else {

                mv.setStyle("-fx-font: normal bold 16px 'arial';"

                        + "-fx-text-fill : black");

            }

        });

        contact.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                general.setText("General");

                corona.setText("Coronavirus");

                mv.setText("Mission & Vision");

                contact.setText("> Contact Us");

                vbox2.getChildren().clear();

                vbox2.getChildren().add(uhead);

                vbox2.getChildren().add(uhbox);

            }

        });

        contact.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                contact.setStyle("-fx-font: normal bold 18px 'arial';"

                        + "-fx-text-fill : #1487d9");

            } else {

                contact.setStyle("-fx-font: normal bold 16px 'arial';"

                        + "-fx-text-fill : black");

            }

        });

        back.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                String role;

                role = new CurrentUser().GetRole();

                if (role.equals("")){

                    try{

                        Login login = new Login();

                        login.start(primaryStage);

                    } catch (FileNotFoundException e)

                    {

                        e.printStackTrace();

                    }

                } else if (role.equals("Doctor")){

                    try{

                        DMenu dmenu = new DMenu();

                        dmenu.start(primaryStage);

                    } catch (FileNotFoundException e)

                    {

                        e.printStackTrace();

                    }

                } else {

                    try{

                        Menu menu = new Menu();

                        menu.start(primaryStage);

                    } catch (FileNotFoundException e)

                    {

                        e.printStackTrace();

                    }

                }

            }

        });

        back.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                back.setStyle("-fx-font: normal bold 18px 'arial';"

                        + "-fx-text-fill : #1487d9");

            } else {

                back.setStyle("-fx-font: normal bold 16px 'arial';"

                        + "-fx-text-fill : black");

            }

        });

    }

    public static void main(String[] args) {

        launch(args);

    }

}

**Doctor Menu :**

package Screens;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Label;

import javafx.stage.Stage;

import javafx.scene.image.\*;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import javafx.geometry.Pos;

import javafx.beans.value.ChangeListener;

import javafx.scene.layout.VBox;

import javafx.scene.layout.HBox;

import javafx.scene.input.MouseEvent;

import javafx.event.EventHandler;

public class DMenu extends Application  {

    @Override

    public void start(Stage primaryStage) throws FileNotFoundException{

        Label heading = new Label("Menu");

        heading.setStyle("-fx-font: normal bold 30px 'arial' ");

        Label user = new Label("User Profile");

        user.setStyle("-fx-text-fill: #67d0eb;" +

                "-fx-font: normal bold 23px 'arial' ");

        ImageView user\_imv = ImgVw("Images/user\_profile.png");

        HBox hbox1 = BHBox();

        hbox1.getChildren().addAll(user\_imv,user);

        Label app = new Label("  Check Up  ");

        app.setStyle("-fx-text-fill: #4ea5d2;" +

                "-fx-font: normal bold 23px 'arial' ");

        ImageView app\_imv = ImgVw("Images/checkup.png");

        HBox hbox2 = BHBox();

        hbox2.getChildren().addAll(app\_imv,app);

        HBox hboxa = new HBox();

        hboxa.setAlignment(Pos.CENTER);

        hboxa.setSpacing(30);

        hboxa.getChildren().addAll(hbox1,hbox2);

        Label qa = new Label("  About Us  ");

        qa.setStyle("-fx-text-fill: #677feb;" +

                "-fx-font: normal bold 23px 'arial' ");

        ImageView qa\_imv = ImgVw("Images/q&a.png");

        HBox hbox3 = BHBox();

        hbox3.getChildren().addAll(qa\_imv,qa);

        Label logout = new Label("   Logout   ");

        logout.setStyle("-fx-text-fill: #677feb;" +

                "-fx-font: normal bold 23px 'arial' ");

        ImageView logout\_imv = ImgVw("Images/logout.png");

        HBox hbox4 = BHBox();

        hbox4.getChildren().addAll(logout\_imv,logout);

        HBox hboxb = new HBox();

        hboxb.setAlignment(Pos.CENTER);

        hboxb.setSpacing(30);

        hboxb.getChildren().addAll(hbox3,hbox4);

        VBox vbox = new VBox(3);

        vbox.setAlignment(Pos.TOP\_CENTER);

        vbox.setSpacing(50);

        vbox.setStyle("-fx-background-color: LIGHTBLUE;");

        vbox.getChildren().addAll(heading,hboxa,hboxb);

        Scene scene = new Scene(vbox, 700, 600);

        primaryStage.setScene(scene);

        primaryStage.show();

        primaryStage.getIcons().add(new Image("file:Images/icon.png"));

        primaryStage.setTitle("Menu");

        primaryStage.setAlwaysOnTop(true);

        hbox1.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                hbox1.setMinSize(200, 200);

            } else {

                hbox1.setMinSize(150, 150);

            }

        });

        hbox2.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                hbox2.setMinSize(250, 200);

            } else {

                hbox2.setMinSize(150, 150);

            }

        });

        hbox3.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                hbox3.setMinSize(250, 200);

            } else {

                hbox3.setMinSize(150, 150);

            }

        });

        hbox4.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

            if (newValue) {

                hbox4.setMinSize(250, 200);

            } else {

                hbox4.setMinSize(150, 150);

            }

        });

        hbox1.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                try{

                    DUserDetails duser = new DUserDetails();

                    duser.start(primaryStage);

                } catch (FileNotFoundException e)

                {

                    e.printStackTrace();

                }

            }

        });

        hbox2.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                try{

                    DAppointment dappointment = new DAppointment();

                    dappointment.start(primaryStage);

                } catch (FileNotFoundException e)

                {

                    e.printStackTrace();

                }

            }

        });

        hbox3.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                try{

                    AboutUs aboutus = new AboutUs();

                    aboutus.start(primaryStage);

                } catch (FileNotFoundException e)

                {

                    e.printStackTrace();

                }

            }

        });

        hbox4.setOnMouseClicked(new EventHandler<MouseEvent>(){

            @Override

            public void handle(MouseEvent arg0) {

                try{

                    Login login = new Login();

                    login.start(primaryStage);

                } catch (FileNotFoundException e)

                {

                    e.printStackTrace();

                }

            }

        });

    }

    private ImageView ImgVw(String s) throws FileNotFoundException{

        FileInputStream inputstream = new FileInputStream(s);

        Image img = new Image(inputstream);

        ImageView imv = new ImageView(img);

        imv.setFitHeight(180);

        imv.setFitWidth(130);

        return imv;

    }

    private HBox BHBox(){

        HBox hbox = new HBox(20);

        hbox.setAlignment(Pos.CENTER);

        hbox.setStyle("-fx-padding: 10;" +

                "-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 2;" +

                "-fx-border-insets: 5;" +

                "-fx-border-radius: 5;" +

                "-fx-border-color: #5494e3;");

        return hbox;

    }

    public static void main(String[] args) {

        launch(args);

    }

}

**Doctor User Details :**

package Screens;

import FileUtils.CurrentUser;

import FileUtils.DoctorFile;

import DataModel.Doctor;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.scene.control.Button;

import javafx.scene.control.Label;

import javafx.stage.Stage;

import javafx.scene.layout.GridPane;

import javafx.scene.layout.HBox;

import javafx.scene.layout.VBox;

import javafx.scene.image.\*;

import java.io.FileInputStream;

import java.io.FileNotFoundException;

import javafx.geometry.Pos;

import javafx.scene.input.MouseEvent;

import javafx.event.EventHandler;

public class DUserDetails extends Application {

    @Override

    public void start(Stage primaryStage) throws FileNotFoundException{

        String did;

        did = new CurrentUser().GetUser();

        if (did.equals("")){

            try{

                Login login = new Login();

                login.start(primaryStage);

            } catch (FileNotFoundException e)

            {

                e.printStackTrace();

            }

        } else {

            Doctor d = new DoctorFile().GetDetails(did);

            Label heading = new Label("Doctor Details");

            heading.setStyle("-fx-font: normal bold 20px 'arial' ");

            FileInputStream inputstream = new FileInputStream("Images/user\_profile.png");

            Image user\_img = new Image(inputstream);

            ImageView user\_imv = new ImageView(user\_img);

            user\_imv.setFitHeight(200);

            user\_imv.setFitWidth(200);

            VBox top = new VBox(20);

            top.getChildren().addAll(user\_imv, heading);

            top.setAlignment(Pos.CENTER);

            Label id = dlabel("Doctor ID :");

            Label Did = plabel(d.getId());

            HBox hid = new HBox(5);

            hid.getChildren().addAll(id, Did);

            Label name = dlabel("Name :");

            Label Dname = plabel(d.getName());

            HBox hname = new HBox(5);

            hname.getChildren().addAll(name, Dname);

            Label phone = dlabel("Phone :");

            Label Dphone = plabel(d.getPhone());

            HBox hphone = new HBox(5);

            hphone.getChildren().addAll(phone, Dphone);

            Label address = dlabel("Address :");

            Label Daddress = plabel(d.getAddress());

            HBox haddress = new HBox(5);

            haddress.getChildren().addAll(address, Daddress);

            Label speciality = dlabel("Speciality :");

            Label Dspeciality = plabel(d.getSpeciality());

            HBox hspeciality = new HBox(5);

            hspeciality.getChildren().addAll(speciality, Dspeciality);

            Button ok = new Button("OK");

            ok.setStyle("-fx-background-color: #278fcc; -fx-text-fill: white;");

            VBox element = new VBox(20);

            element.getChildren().addAll(hid, hname, hphone, haddress, hspeciality, ok);

            element.setAlignment(Pos.CENTER);

            VBox all = new VBox(30);

            all.getChildren().addAll(top,element);

            all.setAlignment(Pos.CENTER);

            GridPane gp = new GridPane();

            gp.setMinSize(700, 600);

            gp.setStyle("-fx-background-color: LIGHTBLUE;");

            gp.setAlignment(Pos.CENTER);

            gp.add(all,0,0);

            primaryStage.setScene(new Scene(gp,700,600));

            primaryStage.show();

            primaryStage.getIcons().add(new Image("file:Images/icon.png"));

            primaryStage.setTitle("User Details");

            primaryStage.setAlwaysOnTop(true);

            ok.setOnMouseClicked(new EventHandler<MouseEvent>(){

                @Override

                public void handle(MouseEvent arg0) {

                    try{

                        DMenu dmenu = new DMenu();

                        dmenu.start(primaryStage);

                    } catch (FileNotFoundException e)

                    {

                        e.printStackTrace();

                    }

                }

            });

        }

    }

    public Label plabel(String s){

        Label label = new Label(s);

        label.setWrapText(false);

        label.setStyle("-fx-background-color: WHITE;" +

                "-fx-border-style: solid inside;" +

                "-fx-border-width: 1;" +

                "-fx-border-insets: 1;" +

                "-fx-border-radius: 1;" +

                "-fx-border-color: #5494e3;");

        return label;

    }

    public Label dlabel(String s){

        Label label = new Label(s);

        label.setStyle("-fx-font: normal bold 15px 'arial' ");

        return label;

    }

    public static void main(String[] args) {

        launch(args);

    }

}

**Doctor Appointment :**

package Screens;

import FileUtils.DoctorFile;

import javafx.application.Application;

import javafx.scene.Scene;

import javafx.stage.Stage;

import javafx.scene.layout.VBox;

import javafx.scene.layout.HBox;

import javafx.scene.control.ListView;

import java.util.ArrayList;

import java.util.List;

import java.util.Arrays;

import javafx.collections.FXCollections;

import javafx.collections.ObservableList;

import javafx.geometry.Pos;

import javafx.scene.control.Label;

import javafx.scene.control.Button;

import javafx.scene.input.MouseEvent;

import javafx.event.EventHandler;

import javafx.scene.paint.Color;

import javafx.beans.value.ChangeListener;

import java.io.FileNotFoundException;

import javafx.scene.image.\*;

import javafx.scene.control.TextArea;

public class DAppointment extends Application{

    @Override

    public void start(Stage primaryStage) throws FileNotFoundException{

        List<String> pList = new DoctorFile().GetAppointmentList();

        if (pList.size() == 0){

            Label heading = new Label("No Appointments");

            heading.setStyle("-fx-font: normal bold 30px 'arial' ");

            Button back = new Button("Back to Menu");

            back.setStyle("-fx-background-color: #278fcc; -fx-text-fill: white;");

            VBox elements = new VBox(20);

            elements.setAlignment(Pos.CENTER);

            elements.getChildren().addAll(heading,back);

            primaryStage.setScene(new Scene(elements,700,600));

            primaryStage.show();

            primaryStage.getIcons().add(new Image("file:Images/icon.png"));

            primaryStage.setTitle("Appointment");

            primaryStage.setAlwaysOnTop(true);

            back.setOnMouseClicked(new EventHandler<MouseEvent>(){

                @Override

                public void handle(MouseEvent arg0) {

                    try{

                        DMenu dmenu = new DMenu();

                        dmenu.start(primaryStage);

                    } catch (FileNotFoundException e)

                    {

                        e.printStackTrace();

                    }

                }

            });

        } else {

            VBox vbox = new VBox();

            vbox.setAlignment(Pos.TOP\_CENTER);

            vbox.setSpacing(50);

            vbox.setStyle("-fx-background-color: LIGHTBLUE;");

            HBox hbox = new HBox();

            hbox.setAlignment(Pos.CENTER);

            hbox.setPrefWidth(400);

            hbox.setPrefWidth(500);

            VBox vbox1 = new VBox();

            vbox1.setPrefSize(300, 400);

            vbox1.setSpacing(20);

            vbox1.setAlignment(Pos.CENTER);

            vbox1.setStyle("-fx-padding: 10;" +

            "-fx-background-color: WHITE;" +

            "-fx-border-style: solid inside;" +

            "-fx-border-width: 2;" +

            "-fx-border-insets: 5;" +

            "-fx-border-radius: 5;" +

            "-fx-border-color: #5494e3;");

            ObservableList<String> patientsList = FXCollections.observableArrayList(pList);

            ListView<String> lv = new ListView<String>(patientsList);

            lv.setMaxSize(200, 180);

            Label head = new Label("Patient List");

            head.setStyle("-fx-font: normal bold 30px 'arial';"

                            + "-fx-text-fill: #d94364;");

            head.setAlignment(Pos.CENTER);

            Button select = new Button("Select");

            select.setStyle("-fx-background-color: lightgreen; -fx-text-fill: white;");

            Label back = new Label("<- Back to Menu");

            back.setStyle("-fx-font: normal bold 16px 'arial';"

                    + "-fx-text-fill : black");

            vbox1.getChildren().add(lv);

            vbox1.getChildren().add(select);

            vbox1.getChildren().add(back);

            hbox.getChildren().add(vbox1);

            vbox.getChildren().add(head);

            vbox.getChildren().add(hbox);

            Scene scene = new Scene(vbox, 700, 600);

            primaryStage.setScene(scene);

            primaryStage.show();

            primaryStage.getIcons().add(new Image("file:Images/icon.png"));

            primaryStage.setTitle("Appointment");

            primaryStage.setAlwaysOnTop(true);

            //Screen2

            HBox hbox1 = new HBox();

            hbox1.setAlignment(Pos.CENTER);

            hbox1.setPrefWidth(400);

            hbox1.setPrefWidth(500);

            VBox vbox2 = new VBox();

            vbox2.setPrefSize(600, 500);

            vbox2.setSpacing(20);

            vbox2.setStyle("-fx-padding: 10;" +

            "-fx-background-color: WHITE;" +

            "-fx-border-style: solid inside;" +

            "-fx-border-width: 2;" +

            "-fx-border-insets: 5;" +

            "-fx-border-radius: 5;" +

            "-fx-border-color: #5494e3;");

            Label head1 = new Label("Appointment");

            head1.setStyle("-fx-font: normal bold 30px 'arial';"

                            + "-fx-text-fill: #d94364;");

            head1.setAlignment(Pos.CENTER);

            Label symptoms = new Label("Symptoms");

            symptoms.setStyle("-fx-font: normal bold 20px 'arial';"

                            + "-fx-text-fill: DARKBLUE;");

            TextArea sym = new TextArea();

            sym.setPrefHeight(150);

            sym.setEditable(false);

            Label feedback = new Label("Doctor's Feedback");

            feedback.setStyle("-fx-font: normal bold 20px 'arial';"

                            + "-fx-text-fill: DARKBLUE;");

            TextArea fb = new TextArea();

            fb.setPrefHeight(150);

            HBox hbox2 = new HBox();

            hbox2.setAlignment(Pos.CENTER);

            hbox2.setSpacing(10);

            Button submit = new Button("Submit");

            submit.setStyle("-fx-background-color: lightgreen; -fx-text-fill: white;");

            Button back1 = new Button("Back");

            back1.setStyle("-fx-background-color: lightgreen; -fx-text-fill: white;");

            hbox2.getChildren().add(back1);

            hbox2.getChildren().add(submit);

            vbox2.getChildren().add(symptoms);

            vbox2.getChildren().add(sym);

            vbox2.getChildren().add(feedback);

            vbox2.getChildren().add(fb);

            vbox2.getChildren().add(hbox2);

            Label cheat = new Label();

            //Event Handling

            back.hoverProperty().addListener((ChangeListener<Boolean>) (observable, oldValue, newValue) -> {

                if (newValue) {

                    back.setTextFill(Color.BLUEVIOLET);

                } else {

                    back.setTextFill(Color.BLACK);

                }

            });

            back.setOnMouseClicked(new EventHandler<MouseEvent>(){

                @Override

                public void handle(MouseEvent arg0) {

                    try{

                        DMenu dm = new DMenu();

                        dm.start(primaryStage);

                    } catch (FileNotFoundException e){

                        e.printStackTrace();

                    }

                }

            });

            select.setOnMouseClicked(new EventHandler<MouseEvent>(){

                @Override

                public void handle(MouseEvent arg0) {

                    if ( lv.getSelectionModel().getSelectedItem() != null){

                        vbox.getChildren().clear();

                        String sid = lv.getSelectionModel().getSelectedItem();

                        cheat.setText(sid);

                        List<String> symptoms = new DoctorFile().GetSymptoms (sid);

                        String symp = String.join("\n", symptoms);

                        sym.setText(symp);

                        hbox1.getChildren().add(vbox2);

                        vbox.getChildren().add(head1);

                        vbox.getChildren().add(hbox1);

                    }

                }

            });

            back1.setOnMouseClicked(new EventHandler<MouseEvent>(){

                @Override

                public void handle(MouseEvent arg0) {

                    try{

                        DAppointment dappointment = new DAppointment();

                        dappointment.start(primaryStage);

                    } catch (FileNotFoundException e)

                    {

                        e.printStackTrace();

                    }

                }

            });

            submit.setOnMouseClicked(new EventHandler<MouseEvent>(){

                @Override

                public void handle(MouseEvent arg0) {

                    if (!fb.getText().isEmpty()){

                        DoctorFile df = new DoctorFile();

                        df.UpdateStatus(cheat.getText(), "DONE", fb.getText());

                        pList.remove(cheat.getText());

                        df.RemoveAppointment(pList);

                        try{

                            DAppointment dappointment = new DAppointment();

                            dappointment.start(primaryStage);

                        } catch (FileNotFoundException e)

                        {

                            e.printStackTrace();

                        }

                    }

                }

            });

        }

    }

    public static void main(String args[]){

        launch(args);

    }

}

**DATA MODELS**

**Patient :**

package DataModel;

public class Patient{

    private String id;

    private String name;

    private String phone;

    private String gender;

    private String address;

    private String state;

    private String status;

    public Patient(String id, String name, String phone, String gender,String address, String state){

        this.id = id;

        this.name = name;

        this.phone = phone;

        this.gender = gender;

        this.address = address;

        this.state = state;

        this.status = "FREE";

    }

    public Patient(String id, String name, String phone, String gender,String address, String state, String status){

        this.id = id;

        this.name = name;

        this.phone = phone;

        this.gender = gender;

        this.address = address;

        this.state = state;

        this.status = status;

    }

    public Patient(Patient p){

        this.id = p.getId();

        this.name = p.getName();

        this.phone = p.getPhone();

        this.gender = p.getGender();

        this.address = p.getAddress();

        this.state = p.getState();

        this.status = p.getStatus();

    }

    public void setStatus(String status){

        this.status = status;

    }

    public String getId(){

        return id;

    }

    public String getName(){

        return name;

    }

    public String getPhone(){

        return phone;

    }

    public String getGender(){

        return gender;

    }

    public String getAddress(){

        return address;

    }

    public String getState(){

        return state;

    }

    public String getStatus(){

        return status;

    }

}

**Doctor :**

package DataModel;

public class Doctor{

    private String id;

    private String name;

    private String phone;

    private String address;

    private String speciality;

    private String status;

    public Doctor(String id, String name, String phone, String address, String speciality){

        this.id = id;

        this.name = name;

        this.phone = phone;

        this.address = address;

        this.speciality = speciality;

        this.status = "FREE";

    }

    public Doctor(String id, String name, String phone, String address, String speciality, String status){

        this.id = id;

        this.name = name;

        this.phone = phone;

        this.address = address;

        this.speciality = speciality;

        this.status = status;

    }

    public void setStatus(String status){

        this.status = status;

    }

    public String getId(){

        return id;

    }

    public String getName(){

        return name;

    }

    public String getPhone(){

        return phone;

    }

    public String getAddress(){

        return address;

    }

    public String getSpeciality(){

        return speciality;

    }

    public String getStatus(){

        return status;

    }

}

**FILE UTILS**

**Patient File :**

package FileUtils;

import DataModel.Patient;

import java.io.\*;

import java.util.\*;

import java.nio.charset.StandardCharsets;

import java.nio.file.\*;

public class PatientFile{

    public void addPatient(Patient p,String password){

        String id = p.getId();

        try{

            File patient = new File(id.concat(".txt"));

            if(patient.createNewFile()) {

                try {

                    FileWriter fw = new FileWriter(patient);

                    BufferedWriter bw = new BufferedWriter(fw);

                    bw.write(p.getName() + "\n" + p.getPhone() + "\n" + p.getGender() + "\n" + p.getAddress() + "\n" + p.getState() + "\n" + password);

                    bw.close();

                    fw.close();

                    UpdateStatus(id,"FREE");

                    System.out.println("Account Created");

                } catch (IOException e) {

                    System.out.println("An error occurred.");

                    patient.delete();

                    e.printStackTrace();

                }

            } else {

                System.out.println("Account already exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

    }

    public boolean IsAuthenticatePatient(String id, String password){

        File user = new File(id.concat(".txt"));

        if(user.exists()){

            try{

                Scanner read = new Scanner(user);

                String pass;

                pass = read.nextLine();

                pass = read.nextLine();

                pass = read.nextLine();

                pass = read.nextLine();

                pass = read.nextLine();

                pass = read.nextLine();

                if(password.equals(pass)){

                   return true;

                }

            } catch (IOException e) {

                System.out.println("An error occurred.");

                e.printStackTrace();

            }

        }

        return false;

    }

    public void UpdateStatus(String id, String s){

        File status = new File(id.concat("\_Status.txt"));

        try {

            FileWriter fw = new FileWriter(status);

            BufferedWriter bw = new BufferedWriter(fw);

            bw.write(s);

            bw.close();

            fw.close();

        } catch (IOException e) {

            System.out.println("An error occurred.");

            status.delete();

            e.printStackTrace();

        }

    }

    public void UpdateStatus(String id, String s, String fb){

        File status = new File(id.concat("\_Status.txt"));

        try {

            FileWriter fw = new FileWriter(status);

            BufferedWriter bw = new BufferedWriter(fw);

            bw.write(s + "\n" + fb);

            bw.close();

            fw.close();

        } catch (IOException e) {

            System.out.println("An error occurred.");

            status.delete();

            e.printStackTrace();

        }

    }

    public String GetStatus(String id){

        String status;

        try{

            File st = new File(id.concat("\_Status.txt"));

            if(st.exists()) {

                Scanner s = new Scanner(st);

                status = s.nextLine();

                s.close();

                return status;

            } else {

                System.out.println("Account does not exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

        return "";

    }

    public String GetFeedback(String id){

        String fb = "";

        String str;

        try{

            File st = new File(id.concat("\_Status.txt"));

            if(st.exists()) {

                Scanner s = new Scanner(st);

                str = s.nextLine();

                while (s.hasNext()){

                    str = s.nextLine();

                    fb = fb.concat(str + "\n");

                }

                s.close();

                return fb;

            } else {

                System.out.println("Account does not exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

        return fb;

    }

    public void AddAppointmentList(String id){

        try {

            File list = new File("AppointmentList.txt");

            if(!list.exists()){

                list.createNewFile();

            }

            try {

                    FileWriter fw = new FileWriter(list,true);

                    BufferedWriter bw = new BufferedWriter(fw);

                    bw.write(id);

                    bw.close();

                    fw.close();

            } catch (IOException e) {

                System.out.println("An error occurred.");

                e.printStackTrace();

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

    }

    public Patient GetDetails(String id){

        String name;

        String phone;

        String gender;

        String address;

        String state;

        String status;

        try{

            File patient = new File(id.concat(".txt"));

            if(patient.exists()) {

                Scanner p = new Scanner(patient);

                name = p.nextLine();

                phone = p.nextLine();

                gender = p.nextLine();

                address = p.nextLine();

                state = p.nextLine();

                p.close();

                status = GetStatus(id);

                return new Patient(id, name, phone, gender, address, state, status);

            } else {

                System.out.println("Account does not exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

        return null;

    }

    public void AddSymptoms(List<String> symptoms,String id){

        ListIterator<String> sitr = symptoms.listIterator();

        try {

            File symp = new File(id.concat("\_symptoms.txt"));

            if(!symp.exists()){

                symp.createNewFile();

            }

            try {

                FileWriter fw = new FileWriter(symp);

                BufferedWriter bw = new BufferedWriter(fw);

                bw.write(sitr.next());

                while(sitr.hasNext()){

                    bw.newLine();

                    bw.write(sitr.next());

                }

                bw.close();

                fw.close();

            } catch (IOException e) {

                System.out.println("An error occurred.");

                e.printStackTrace();

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

    }

    public void ClearSymptoms(String id){

        File symp = new File(id.concat("\_symptoms.txt"));

        if(symp.exists()){

            try {

                PrintWriter pw = new PrintWriter(symp);

                pw.print("");

                pw.close();

            } catch (IOException e) {

                System.out.println("An error occurred.");

                e.printStackTrace();

            }

        }

    }

}

**Doctor File :**

package FileUtils;

import DataModel.Doctor;

import java.io.\*;

import java.util.\*;

import java.nio.charset.StandardCharsets;

import java.nio.file.\*;

public class DoctorFile{

    public boolean IsAuthenticateDoctor(String id, String password){

        File user = new File(id.concat(".txt"));

        if(user.exists()){

            try{

                Scanner read = new Scanner(user);

                String pass;

                pass = read.nextLine();

                pass = read.nextLine();

                pass = read.nextLine();

                pass = read.nextLine();

                pass = read.nextLine();

                if(password.equals(pass)){

                   return true;

                }

            } catch (IOException e) {

                System.out.println("An error occurred.");

                e.printStackTrace();

            }

        }

        return false;

    }

    public void UpdateStatus(String id, String s){

        File status = new File(id.concat("\_Status.txt"));

        try {

            FileWriter fw = new FileWriter(status);

            BufferedWriter bw = new BufferedWriter(fw);

            bw.write(s);

            bw.close();

            fw.close();

        } catch (IOException e) {

            System.out.println("An error occurred.");

            status.delete();

            e.printStackTrace();

        }

    }

    public void UpdateStatus(String id, String s, String fb){

        File status = new File(id.concat("\_Status.txt"));

        try {

            FileWriter fw = new FileWriter(status);

            BufferedWriter bw = new BufferedWriter(fw);

            bw.write(s + "\n" + fb);

            bw.close();

            fw.close();

        } catch (IOException e) {

            System.out.println("An error occurred.");

            status.delete();

            e.printStackTrace();

        }

    }

    public String GetStatus(String id){

        String status;

        try{

            File st = new File(id.concat("\_Status.txt"));

            if(st.exists()) {

                Scanner s = new Scanner(st);

                status = s.nextLine();

                s.close();

                return status;

            } else {

                System.out.println("Account does not exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

        return "";

    }

    public String GetFeedback(String id){

        String pid = "";

        try{

            File st = new File(id.concat("\_Status.txt"));

            if(st.exists()) {

                Scanner s = new Scanner(st);

                pid = s.nextLine();

                pid = s.nextLine();

                s.close();

                return pid;

            } else {

                System.out.println("Account does not exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

        return pid;

    }

    public List<String> GetDoctorList(){

        File list = new File("DoctorList.txt");

        if(list.exists()){

            List<String> lines = Collections.emptyList();

            try

            {

              lines = Files.readAllLines(Paths.get(list.getPath()), StandardCharsets.UTF\_8);

            } catch (IOException e)

            {

              e.printStackTrace();

            }

            return lines;

        }

        return Collections.emptyList();

    }

    public List<String> GetAppointmentList(){

        File list = new File("AppointmentList.txt");

        if(list.exists()){

            List<String> lines = Collections.emptyList();

            try

            {

              lines = Files.readAllLines(Paths.get(list.getPath()), StandardCharsets.UTF\_8);

            } catch (IOException e)

            {

              e.printStackTrace();

            }

            return lines;

        }

        return Collections.emptyList();

    }

    public void RemoveAppointment(List<String> appointments){

        ListIterator<String> sitr = appointments.listIterator();

        File symp = new File("AppointmentList.txt");

        try {

            FileWriter fw = new FileWriter(symp);

            BufferedWriter bw = new BufferedWriter(fw);

            while(sitr.hasNext()){

                bw.newLine();

                bw.write(sitr.next());

            }

            bw.close();

            fw.close();

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

    }

    public List<String> GetSymptoms (String id){

        File list = new File(id.concat("\_symptoms.txt"));

        if(list.exists()){

            List<String> lines = Collections.emptyList();

            try

            {

              lines = Files.readAllLines(Paths.get(list.getPath()), StandardCharsets.UTF\_8);

            } catch (IOException e)

            {

              e.printStackTrace();

            }

            return lines;

        }

        return Collections.emptyList();

    }

    public Doctor GetDetails(String id){

        String name;

        String phone;

        String address;

        String speciality;

        String status;

        try{

            File doctor = new File(id.concat(".txt"));

            if(doctor.exists()) {

                Scanner d = new Scanner(doctor);

                name = d.nextLine();

                phone = d.nextLine();

                address = d.nextLine();

                speciality = d.nextLine();

                d.close();

                //status = GetStatus(id);

                return new Doctor(id, name, phone, address, speciality);

            } else {

                System.out.println("Account does not exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

        return null;

    }

}

**Current User :**

package FileUtils;

import java.io.\*;

import java.util.\*;

public class CurrentUser{

    public void UpdateUser(String id ,String role){

        File user = new File("cuser.txt");

        try {

            FileWriter fw = new FileWriter(user);

            BufferedWriter bw = new BufferedWriter(fw);

            bw.write(id + "\n" + role);

            bw.close();

            fw.close();

        } catch (IOException e) {

            System.out.println("An error occurred.");

            user.delete();

            e.printStackTrace();

        }

    }

    public String GetUser(){

        String id;

        try{

            File user = new File("cuser.txt");

            if(user.exists()) {

                Scanner s = new Scanner(user);

                id = s.nextLine();

                s.close();

                return id;

            } else {

                System.out.println("Account does not exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

        return "";

    }

    public String GetRole(){

        String role;

        try{

            File user = new File("cuser.txt");

            if(user.exists()) {

                Scanner s = new Scanner(user);

                role = s.nextLine();

                role = s.nextLine();

                s.close();

                return role;

            } else {

                System.out.println("Account does not exists.");

            }

        } catch (IOException e) {

            System.out.println("An error occurred.");

            e.printStackTrace();

        }

        return "";

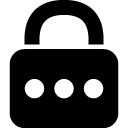
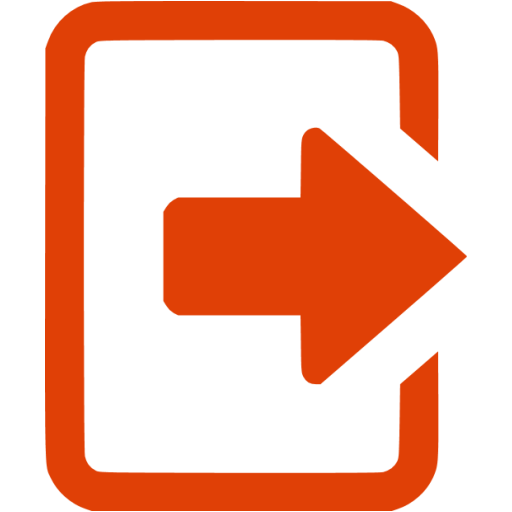
    }

}

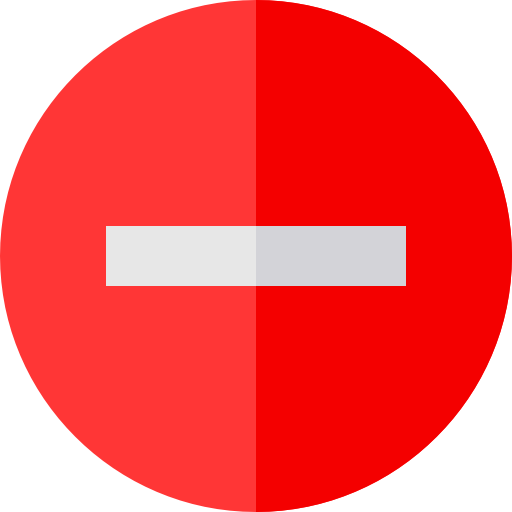
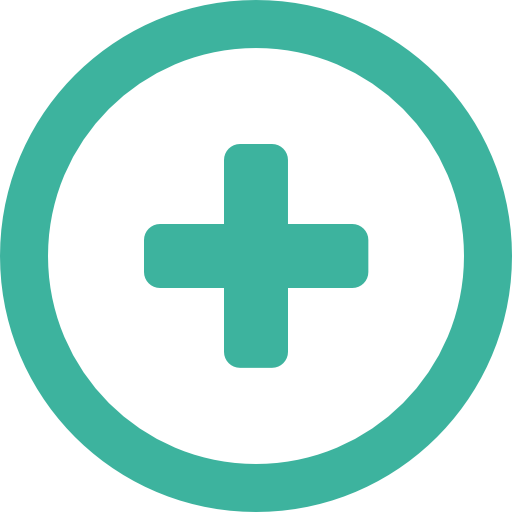
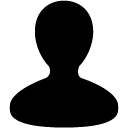
**IMAGES**

**   **

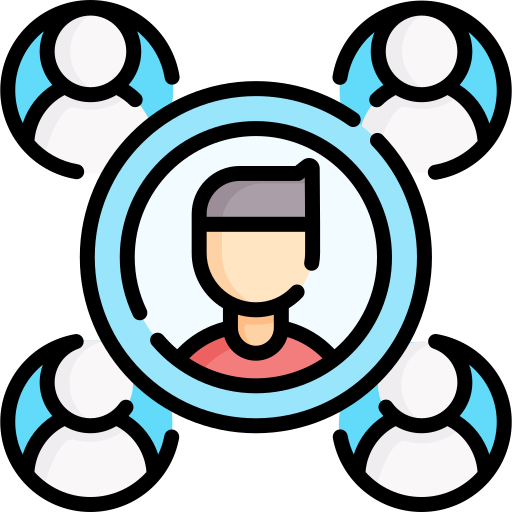
Appointment.png checkup.png file.png icon.png

**   **

icon1.png lock.png logout.png man.png

**   **

minus.png plus.png q&a.png user.png

****

user\_profile.png

**SCREENSHOTS**