

# JAVA Project

## S E M S Result Management App

An app that helps to provide Examination Result in simple Way

Submitted by:

Samundar Singh  
2019272033  
MCA(SS)

## **Abstract**

The main objective of developing this project is to automate student result management. This application will greatly simplify and speed up the result preparation and management process. This is a computerized examinations results management system for tertiary student's examination records. The main aim of the project is to provide the examination result to the student in a simple and accurate way. The application will manage the information about various students enrolled in this course in different years, the marks obtained by the various students in various subjects in different semesters. The manual method of students' academic result processing was found to be tedious, especially when carried out for a large number of students, this makes the entire process time-consuming and error prone. This project is useful for students and institutions for getting the results in simple manner. The system designed is meant to register students as soon as they have paid their departmental registration and only then will they be able to view their results. The system presents a single platform that will be used to manage the processing of all examination records within the institution. admin gives the privileges to the Teacher and student and the guest user to access the result. The student can share or download his/her result.

## **Objectives**

1. The main objective of the project is to provide the examination result to the student in a simple way
2. This project is useful for students and institutions for getting the results in simple manner.
3. The system is intended for the student. And the privileges that are provided to student are to read and execute his/her result by providing user name and password for secure login and in case of new student the registration is available.
4. The CGPA of student is easily calculated here so no need to calculate it manually.
5. The teachers can also easily update the attendance and marks of student in a very simple manner and with less time.

## **Modules:**

### **1.Faculty module:**

- Create the per day Attendance Sheet and print it.
- Create the Marksheet of the Student and Print it.

### **2. Student Module:**

- The Students Can only see there result in a particular semester and there CGPA is automatically calculated.

## **Features:**

### **1.Faculty module:**

- He/she is required to create account for Access.
- View the list of students assigned to them (Automatically show the students list with same semester).
- Add/Update/Display/Delete Marks for the semester's all subjects.
- Generate/Download Report of all Results in PDF Format.
- Generate/Download Attendance in PDF Format.
- Update Profile.

### **2.Student module:**

- He/she is required to create account for Access.
- Can view their Semester Results for All Subjects.
- Generate/Download Report of all Results in PDF Format
- Update Profile

## **App specification:**

- **Front-End :**      **java-Fx**
- **Backend :**       **java**
- **Database :**      **PosgreSQL.**

# Landing Module

## CODES(User Interface):

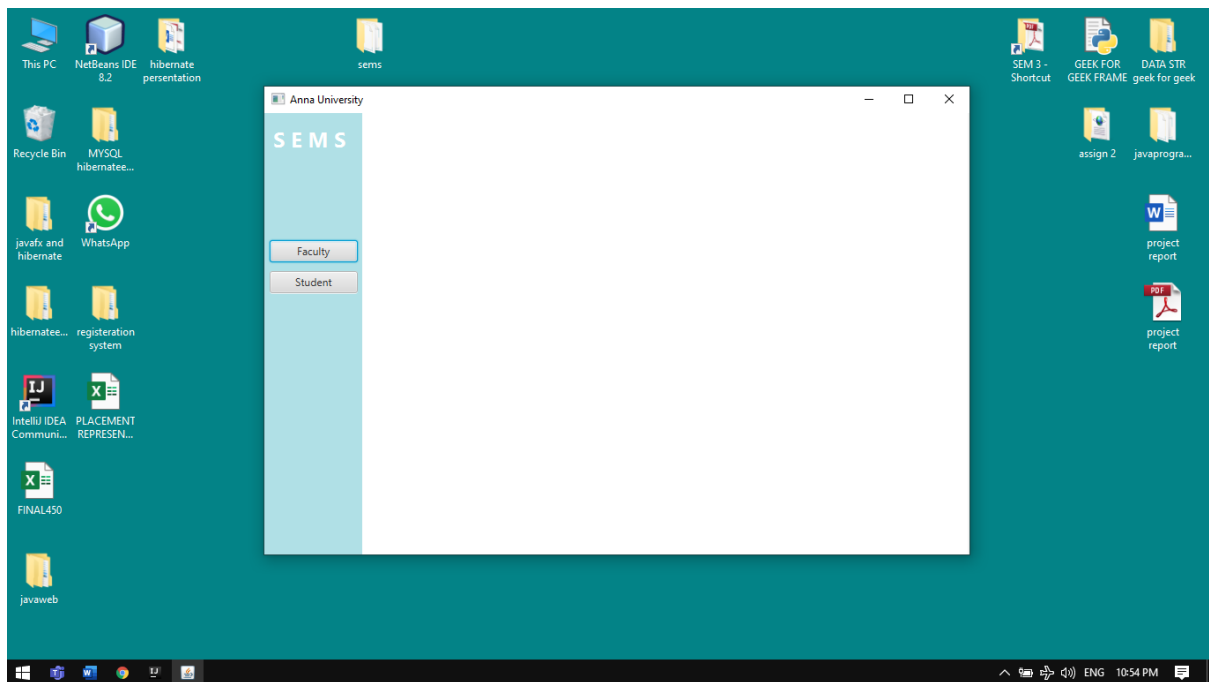
### 1.Home Page:

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.BorderPane?>
<?import javafx.scene.layout.ColumnConstraints?>
<?import javafx.scene.layout.GridPane?>
<?import javafx.scene.layout.RowConstraints?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<GridPane alignment="center" hgap="10" vgap="10"
xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="sample.Controller">
    <columnConstraints>
        <ColumnConstraints />
    </columnConstraints>
    <rowConstraints>
        <RowConstraints />
    </rowConstraints>
    <children>
        <AnchorPane prefHeight="500.0" prefWidth="800.0">
            <children>
                <BorderPane fx:id="dashboardborder" layoutX="-2.0" layoutY="-4.0" prefHeight="508.0" prefWidth="800.0" style="-fx-background-color: white;">
                    <left>
                        <VBox prefHeight="508.0" prefWidth="113.0" style="-fx-background-color: powderblue;" BorderPane.alignment="CENTER">
                            <children>
                                <Button mnemonicParsing="false"
onAction="#facultydashboard" style="-fx-max-width: 100;" text="Faculty"
translateX="8.0" translateY="150.0" />
                                <Button mnemonicParsing="false"
onAction="#studentdashboard" style="-fx-max-width: 100;" text="Student"
translateX="8.0" translateY="160.0" />
                            </children>
                        </VBox>
                    </left>
                </BorderPane>
                <Label layoutX="10.0" layoutY="10.0" prefHeight="44.0"
prefWidth="103.0" text="S E M S" textFill="WHITE">
                    <font>
                        <Font name="System Bold" size="24.0" />
                    </font>
                </Label>
            </children>
        </AnchorPane>
    </children>
</GridPane>
```

## Screenshot:



## Controller Code:

```
package sample;

import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.stage.Stage;

public class Main extends Application {

    @Override
    public void start(Stage primaryStage) throws Exception{
        Parent root =
FXMLLoader.load(getClass().getResource("sample.fxml"));
        primaryStage.setTitle("Anna University");
        primaryStage.setScene(new Scene(root, 800, 500));
        primaryStage.show();
    }

    public static void main(String[] args) {
        launch(args);
    }
}
```

# Faculty Module

## 2.Faculty Signup and login:

### Code:

```
<?xml version="1.0" encoding="UTF-8" ?>

<?import javafx.scene.control.Button ?>
<?import javafx.scene.control.DatePicker ?>
<?import javafx.scene.control.Label ?>
<?import javafx.scene.control.PasswordField ?>
<?import javafx.scene.control.Tab ?>
<?import javafx.scene.control.TabPane ?>
<?import javafx.scene.control.TextField ?>
<?import javafx.scene.image.Image ?>
<?import javafx.scene.image.ImageView ?>
<?import javafx.scene.layout.AnchorPane ?>
<?import javafx.scene.text.Font ?>

<AnchorPane fx:id="faculty_portal" prefHeight="400.0" prefWidth="600.0"
xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="sample.Controller">
    <children>
        <TabPane layoutX="93.0" layoutY="45.0" prefHeight="323.0"
prefWidth="442.0" style="-fx-background-color: powderblue;"
tabClosingPolicy="UNAVAILABLE">
            <tabs>
                <Tab text="login ">
                    <content>
                        <AnchorPane minHeight="0.0" minWidth="0.0"
prefHeight="180.0" prefWidth="200.0">
                            <children>
                                <Label layoutX="149.0" layoutY="87.0"
text="Faculty.ID">
                                    <font>
                                        <Font name="System Bold" size="14.0" />
                                    </font>
                                </Label>
                                <TextField fx:id="login_id" layoutX="222.0"
layoutY="84.0" />
                                <PasswordField fx:id="login_faculty_password"
layoutX="222.0" layoutY="120.0" />
                                <Label layoutX="151.0" layoutY="123.0"
text="Password">
                                    <font>
                                        <Font name="System Bold" size="14.0" />
                                    </font>
                                </Label>
                                <ImageView fitHeight="108.0" fitWidth="113.0"
layoutX="23.0" layoutY="62.0" pickOnBounds="true" preserveRatio="true">
                                    <image>
                                        <Image url="@../images/avatar.png" />
                                    </image>
                                </ImageView>
                                <Button layoutX="258.0" layoutY="160.0"
mnemonicParsing="false" onAction="#faculry_login" prefHeight="25.0"
prefWidth="61.0" style="-fx-background-color: darkgreen;" text="login"
textFill="WHITE" />
                                <Label fx:id="testing_faculty" layoutX="229.0"
layoutY="202.0" prefHeight="35.0" prefWidth="154.0" textFill="#ff0202">
```



```

        <font>
            <Font name="System Bold" size="18.0" />
        </font>
    </Label>
</children>
</AnchorPane>
</content>
</Tab>
<Tab text="Sign up">
    <content>
        <AnchorPane minHeight="0.0" minWidth="0.0"
prefHeight="294.0" prefWidth="456.0">
            <children>
                <Label layoutX="59.0" layoutY="22.0" text="Adhar
">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="59.0" layoutY="52.0" text="Name">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="60.0" layoutY="85.0" text="D.O.B">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="48.0" layoutY="176.0"
text="Address">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="56.0" layoutY="115.0" text="Phone">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <TextField fx:id="faculty_adhar" layoutX="106.0"
layoutY="16.0" />
                <TextField fx:id="faculty_name" layoutX="107.0"
layoutY="49.0" />
                <TextField fx:id="faculty_phone" layoutX="107.0"
layoutY="110.0" />
                <TextField fx:id="faculty_address" layoutX="107.0"
layoutY="171.0" />
                <DatePicker fx:id="faculty_dob" layoutX="106.0"
layoutY="81.0" prefHeight="25.0" prefWidth="150.0" />
                <Button layoutX="139.0" layoutY="246.0"
mnemonicParsing="false" onAction="#faculty_register" prefHeight="25.0"
prefWidth="115.0" style="-fx-background-color: darkgreen;" text="Register"
textFill="WHITE" />
                <ImageView fx:id="faculty_image" fitHeight="149.0"
fitWidth="156.0" layoutX="299.0" layoutY="9.0" pickOnBounds="true"
preserveRatio="true">
                    <image>
                        <Image url="@../images/default%20image.jpg"
/>
                    </image>

```

```

        </ImageView>
        <Button layoutX="300.0" layoutY="165.0"
mnemonicParsing="false" onAction="#faculty_image_chooser" prefHeight="14.0"
prefWidth="107.0" text="Choose img..." />
        <Label layoutX="19.0" layoutY="142.0"
text="Qualification">
            <font>
                <Font size="14.0" />
            </font>
        </Label>
        <TextField fx:id="faculty_qualification"
layoutX="107.0" layoutY="141.0" />
        <Label layoutX="38.0" layoutY="207.0"
text="Password">
            <font>
                <Font size="14.0" />
            </font>
        </Label>
        <TextField fx:id="faculty_password" layoutX="107.0"
layoutY="202.0" />
    </children>
</AnchorPane>
</content>
</Tab>
</tabs>
</TabPane>
<Label layoutX="98.0" layoutY="3.0" text="Faculty Portal">
    <font>
        <Font name="System Bold" size="24.0" />
    </font>
</Label>
</children>
</AnchorPane>

```

## Controller Code:

```

@FXML
private AnchorPane faculty_portal; @FXML
private TextField login_id; @FXML
private PasswordField login_faculty_password; @FXML
private TextField faculty_adhar; @FXML
private TextField faculty_name; @FXML
private TextField faculty_phone; @FXML
private TextField faculty_address; @FXML
private DatePicker faculty_dob; @FXML
private ImageView faculty_image; @FXML
private TextField faculty_qualification; @FXML
private TextField faculty_password;

@FXML
void faculty_register(ActionEvent event) throws Exception {
    String facu_adhar = faculty_adhar.getText();
    String facu_name = faculty_name.getText();
    String facu_phone = faculty_phone.getText();
    String facu_address = faculty_address.getText();
    LocalDate facu_dob = faculty_dob.getValue();
    String facu_quali = faculty_qualification.getText();
    String facu_password = faculty_password.getText();
}

```

```

        try {
            Connection c = null;
            Statement stmt = null;
            Class.forName("org.postgresql.Driver");
            c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/semsproject",
"postgres", "nevada");
            System.out.println("Opened database successfully");
            stmt = c.createStatement();
            PreparedStatement st = c.prepareStatement("insert into
facultysignup values(?,?,?,?,?,?,?)");
            st.setString(1, facu_adhar);
            st.setString(2, facu_name);
            st.setString(3, facu_phone);
            st.setString(4, facu_address);
            st.setDate(5, java.sql.Date.valueOf(facu_dob));
            st.setString(6, facu_quali);
            st.setString(7, facu_password);

            //st.setBinaryStream(8, fis, file.length());
            st.executeUpdate();
            st.close();
            c.close();

            //-----popup ke liye-----
            -----
            Parent root1 =
FXMLLoader.load(getClass().getResource("facultysucessfulpopup.fxml"));
            Stage stage2 = new Stage();
            stage2.setScene(new Scene(root1));
            stage2.show();

        } catch (Exception e) {
            System.out.println("na hoga insert ");
            System.err.println(e.getClass().getName() + ": " + e.getMessage());
            System.exit(0);
        }
    }

@FXML
void faculty_login(ActionEvent event) {
    String log_faculty_id = login_id.getText();
    String log_faculty_pass = login_faculty_password.getText();

    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/semsproject",
"postgres", "nevada");
        stmt = c.createStatement();
        System.out.println("open database sucessfully");
        String sqlStatement = "SELECT * FROM facultysignup WHERE adhar='" +
log_faculty_id + "'";
        ResultSet rs = stmt.executeQuery(sqlStatement);
        String fetch_password = null;

        while (rs.next()) {
            fetch_password = rs.getString("password");

```

```

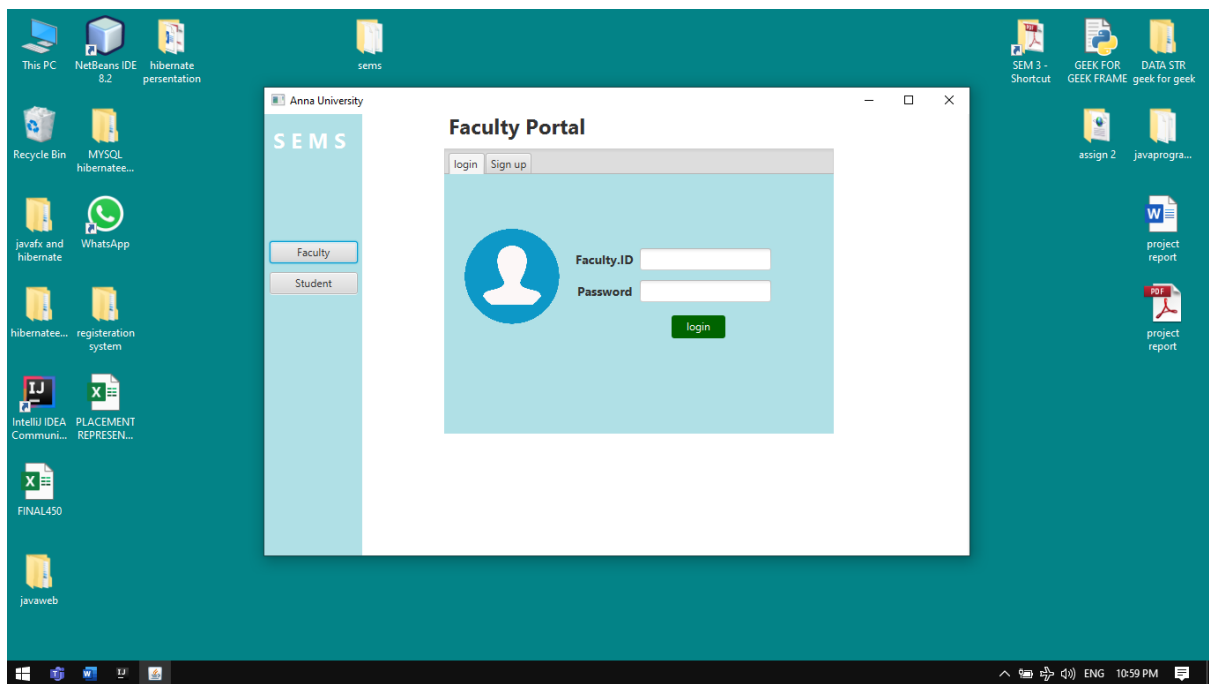
    }
    if (log_faculty_pass.equals(fetch_password)) {
        try {
            Parent root =
FXMLLoader.load(getClass().getResource("facultymarks.fxml"));
            Stage stage = new Stage();
            stage.setScene(new Scene(root));
            stage.show();
        } catch (FileNotFoundException e) {
            System.out.println(" file not found");
        }
    } else {
        System.out.println("galat enter kiya hai ");
        testing_faculty.setText("INVALID USER");
    }
    rs.close();
    stmt.close();
    c.close();
} catch (Exception e) {
    System.err.println(e.getClass().getName() + ": " + e.getMessage());
    System.exit(0);
}
}

@FXML
void faculty_redirect_home(ActionEvent event) throws IOException {
    Parent root =
FXMLLoader.load(getClass().getResource("facultylogin.fxml"));
    Stage stage = new Stage();
    stage.setScene(new Scene(root));
    stage.show();
}
File file1;
FileInputStream fis1;
@FXML
void faculty_image_chooser(ActionEvent event) throws FileNotFoundException
{
    FileChooser filechoose = new FileChooser();
    filechoose.setTitle("Choose Image");
    Stage stage = (Stage) faculty_portal.getScene().getWindow();
    file1 = filechoose.showOpenDialog(stage);
    FileInputStream inputstream = new
FileInputStream(file1.getAbsolutePath());
    Image image = new Image(inputstream);
    if (file1 != null) {
        faculty_image.setImage(image);
    }
    file1 = new File(file1.getAbsolutePath());
    fis1 = new FileInputStream(file1);
}
}

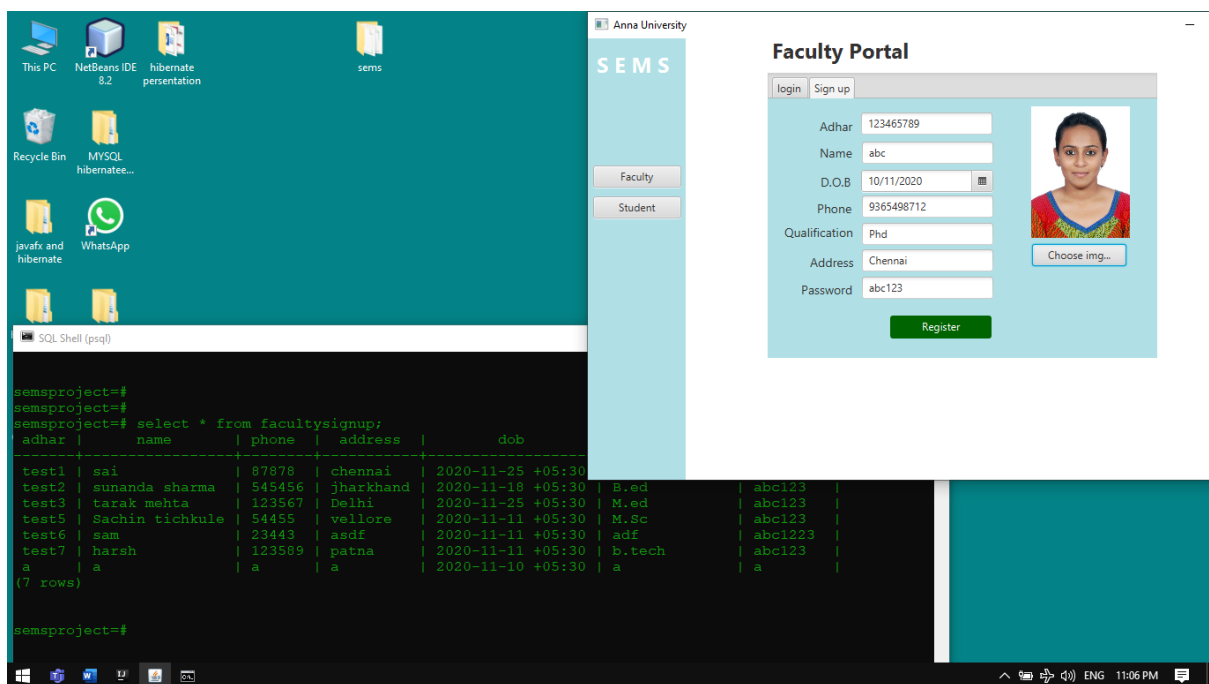
```

Screenshot:

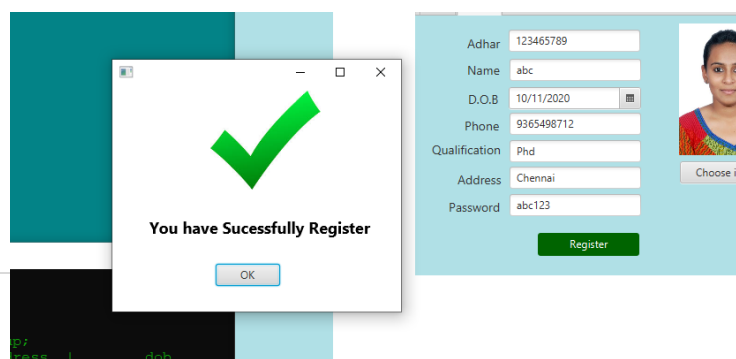
Login page:



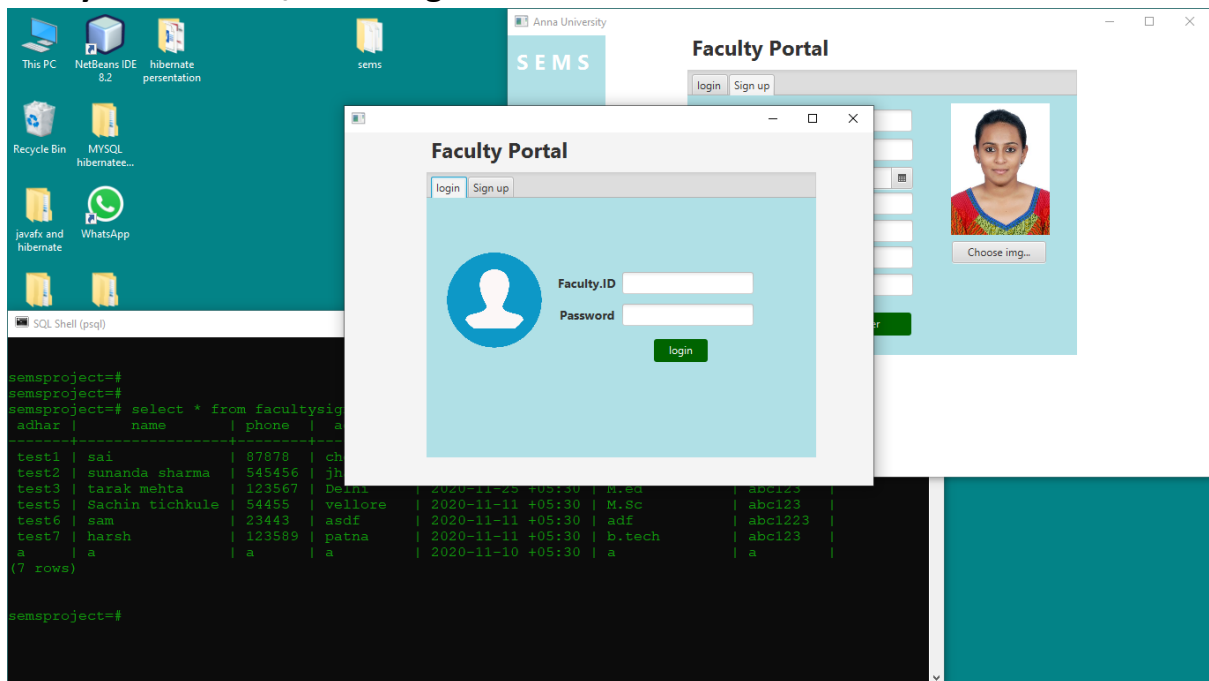
Signup page:



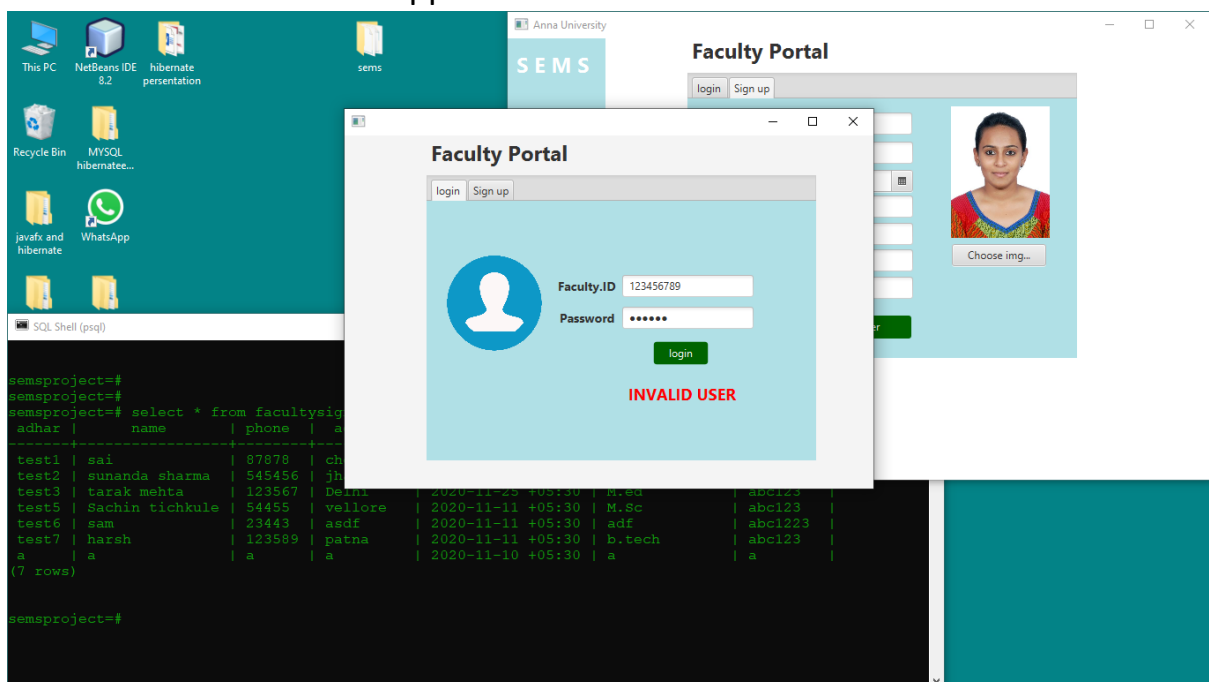
After signup



When Ok is pressed then a cross check window open from where user can verify whether he/she is registered or not.



Input the Aadhar number as Faculty id and password that is registered with that Aadhar number if wrong credential is entered then it will prevent you to access the feature of the app.



If correct credential is provided then it will redirect to faculty portal.

The screenshot shows a desktop environment with a teal background. On the desktop, there are icons for 'This PC', 'NetBeans IDE 8.2', 'hibernate presentation', 'sems', 'Recycle Bin', 'MySQL hibernate...', 'javafx and hibernate', and 'WhatsApp'. A terminal window is open, displaying a SQL query and its results. The web application interface is visible, featuring a sidebar with 'SEMS' and 'Attendance' buttons. The main content area shows a 'Marks' form with dropdowns for Department, Branch, Semester, and Subject, and a table for marks.

**SEMS**

**Attendance**

**Marks**

**Welcome Anonymus**

Deptt.  Branch

Semester  Subject

[Create Marks Sheet](#)

Reg.no	Name	Asses I	Asses II	End sem	Total	Grade
No content in table						

[Print](#) [Update All the Data Of Student](#) [Reset](#)

```
SQL Shell (psql)
a | a | a | a | 2020-
(7 rows)

semsproject=# select * from facultysignup;
      adhar |      name |      phone |      address
-----+-----+-----+-----
test1 | sai | 87878 | chennai
test2 | sunanda sharma | 545456 | jharkhand
test3 | tarak mehta | 123567 | Delhi
test5 | Sachin tichkule | 54455 | vellore
test6 | sam | 23443 | asdf
test7 | harsh | 123589 | patna
a | a | a | a
123465789 | abc | 9365498712 | Chennai
(8 rows)

semsproject=#
```

### 3. The Masks module to upload the result of Student in one click.

Code :

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.ChoiceBox?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="500.0" prefWidth="800.0"
xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="sample.Controller">
    <children>
        <Label layoutX="144.0" layoutY="34.0" text="Welcome" >
            <font>
                <Font name="System Bold" size="24.0" />
            </font>
        </Label>
        <Label layoutX="256.0" layoutY="36.0" text="Anonymus">
            <font>
                <Font size="24.0" />
            </font>
        </Label>
        <VBox prefHeight="528.0" prefWidth="113.0" style="-fx-background-
color: purple;">
            <children>
                <Button mnemonicParsing="false" onAction="#redirect_attendance"
prefHeight="25.0" prefWidth="89.0" text="Attendance">
                    <VBox.margin>
                        <Insets left="10.0" top="100.0" />
                    </VBox.margin>
                    <font>
                        <Font name="System Bold" size="13.0" />
                    </font>
                </Button>
                <Button mnemonicParsing="false" prefHeight="25.0"
prefWidth="89.0" text="Marks">
                    <font>
                        <Font name="System Bold" size="13.0" />
                    </font>
                    <VBox.margin>
                        <Insets left="10.0" top="10.0" />
                    </VBox.margin>
                </Button>
            </children>
        </VBox>
        <Label layoutX="10.0" layoutY="10.0" prefHeight="44.0"
prefWidth="103.0" text="S E M S" textFill="WHITE">
```



```

        <font>
            <Font name="System Bold" size="24.0" />
        </font>
    </Label>
    <Label layoutX="151.0" layoutY="94.0" text="Deptt.">
        <font>
            <Font size="13.0" />
        </font>
    </Label>
    <Label layoutX="494.0" layoutY="95.0" text="Branch">
        <font>
            <Font size="13.0" />
        </font>
    </Label>
    <ChoiceBox fx:id="f_deptt_marks" layoutX="199.0" layoutY="91.0"
prefHeight="25.0" prefWidth="164.0" />
    <ChoiceBox fx:id="f_branch_marks" layoutX="551.0" layoutY="92.0"
prefWidth="150.0" />
    <Label layoutX="136.0" layoutY="134.0" text="Semester">
        <font>
            <Font size="13.0" />
        </font>
    </Label>
    <ChoiceBox fx:id="f_semester_marks" layoutX="200.0" layoutY="131.0"
prefHeight="25.0" prefWidth="164.0" />
    <TableView fx:id="f_marks_table" layoutX="134.0" layoutY="199.0"
prefHeight="246.0" prefWidth="644.0">
        <columns>
            <TableColumn fx:id="marks_regno" prefWidth="88.0" text="Reg.no"
/>
            <TableColumn fx:id="marks_name" prefWidth="210.0" text="Name" />
            <TableColumn fx:id="marks_asses_1" prefWidth="74.0" text="Asses
I" />
            <TableColumn fx:id="marks_asses_2" prefWidth="73.0" text="Asses
II" />
            <TableColumn fx:id="marks_endsem" prefWidth="72.0" text="End
sem" />
            <TableColumn fx:id="marks_total" prefWidth="77.0" text="Total"
/>
            <TableColumn fx:id="marks_grade" prefWidth="47.0" text="Grade"
/>
        </columns>
    </TableView>
    <Label layoutX="491.0" layoutY="134.0" text="Subject">
        <font>
            <Font size="13.0" />
        </font>
    </Label>
    <ChoiceBox fx:id="f_subject_marks" layoutX="551.0" layoutY="131.0"
onMousePressed="#loadsubject" prefWidth="150.0" />
    <Button fx:id="update_marks" layoutX="313.0" layoutY="455.0"
mnemonicParsing="false" onAction="#update_marks" prefHeight="25.0"
prefWidth="339.0" style="-fx-background-color: darkgreen;" text="Update All
the Data Of Student" textFill="WHITE" />

    <Button layoutX="682.0" layoutY="455.0" mnemonicParsing="false"
prefHeight="25.0" prefWidth="97.0" style="-fx-background-color: darkred;"
text="Reset" textFill="#fff8f8" />
    <Button layoutX="134.0" layoutY="168.0" mnemonicParsing="false"
onAction="#create_marks_sheet" prefHeight="25.0" prefWidth="230.0" style="-
fx-background-color: #0095ff;" text="Create Marks Sheet" />

```

```

        <Label fx:id="warning3" layoutX="374.0" layoutY="168.0"
textFill="RED">
        <font>
            <Font name="System Bold" size="18.0" />
        </font>
    </Label>
    <Button layoutX="163.0" layoutY="456.0" mnemonicParsing="false"
onAction="#print_result" prefHeight="25.0" prefWidth="43.0" style="-fx-
background-color: darkblue;" text="Print" textFill="#ffffef" />
    <ImageView fitHeight="28.0" fitWidth="44.0" layoutX="130.0"
layoutY="454.0" pickOnBounds="true" preserveRatio="true">
        <image>
            <Image url="@../images/print%20icon.png" />
        </image>
    </ImageView>
    <Label layoutX="113.0" prefHeight="27.0" prefWidth="686.0" style="-
fx-background-color: black;" text="
Marks " textFill="WHITE">
        <font>
            <Font name="System Bold" size="18.0" />
        </font>
    </Label>
    <Label layoutX="111.0" layoutY="501.0" prefHeight="27.0"
prefWidth="693.0" style="-fx-background-color: black;" textFill="WHITE">
        <font>
            <Font name="System Bold" size="18.0" />
        </font>
    </Label>
</children>
</AnchorPane>

```

## Controller Code:

```

@FXML
private ChoiceBox<String> f_deptt_marks;
@FXML
private ChoiceBox<String> f_branch_marks;
@FXML
private ChoiceBox<String> f_semester_marks;
@FXML
private ChoiceBox<String> f_subject_marks;
@FXML
private TableView<sample.table.marks_table> f_marks_table;
@FXML
private TableColumn<sample.table.marks_table,String> marks_regno;
@FXML
private TableColumn<sample.table.marks_table,String> marks_name;
@FXML
private TableColumn<sample.table.marks_table,String> marks_asses_1;
@FXML
private TableColumn<sample.table.marks_table,String> marks_asses_2;
@FXML
private TableColumn<sample.table.marks_table,String> marks_endsem;
@FXML
private TableColumn<sample.table.marks_table,String> marks_total;
@FXML
private TableColumn<sample.table.marks_table,String> marks_grade;

```

```

private void deptinit(){
    ObservableList<String> li1 = FXCollections.observableArrayList();
    try {
        li1.removeAll(li1);
        li1.add("Information_Technology");
        li1.add("Maths_deptt");
        li1.add("Physics_deptt");
        f_deptt_marks.getItems().addAll(li1);
    }
    catch (NullPointerException e){
        System.out.println("choice box na hoga load");
    }
    branchinit();
}

private void branchinit(){
    ObservableList<String> li2 = FXCollections.observableArrayList();
    try {
        li2.removeAll(li2);
        li2.add("MCA");
        li2.add("Maths");
        li2.add("Physics");
        li2.add("MBA");
        li2.add("Biotechnology");
        li2.add("Electronics");
        li2.add("Civil Engg..");

        f_branch_marks.getItems().addAll(li2);
    }
    catch (NullPointerException e){
        System.out.println("choice box na hoga load");
    }
    semesterinit();
}

private void semesterinit(){
    ObservableList<String> li3 = FXCollections.observableArrayList();
    try {
        li3.removeAll(li3);
        li3.add("Semester_I");
        li3.add("Semester_II");
        li3.add("Semester_III");
        li3.add("Semester_IV");
        li3.add("Semester_V");
        li3.add("Semester_VI");

        f_semester_marks.getItems().addAll(li3);
        Subjectinit(); //isko mouse click wala me daalna hai
    }
    catch (NullPointerException e){
        System.out.println("choice box na hoga load");
    }
}

private void Subjectinit(){
    ObservableList<String> li4 = FXCollections.observableArrayList();
    li4.add("JAVA_programming");
    li4.add("Advance_DataStructure");
}

```

```

        li4.add("Machine_Learning");
        li4.add("Networking");
        li4.add("Social_Relevant_Project");
        li4.add("Ethical_Hacking");
        f_subject_marks.getItems().addAll(li4);
    }

@FXML
public void loadsubject(javafx.scene.input.MouseEvent mouseEvent) {
    try {
        if (f_semester_marks.getValue().equals("Semester_III")) {
            f_subject_marks.getItems().setAll((String) null);
            Subjectinit();
        } else {
            f_subject_marks.getItems().setAll((String) null);
        }
    }
    catch (Exception e){
        System.out.println("hence it is not loaded");
    }
}

@FXML
private Label warning3;

@FXML
void create_marks_sheet(ActionEvent event) {
    String branch=f_branch_marks.getValue();
    String semester=f_semester_marks.getValue();
    String subject=f_subject_marks.getValue();
    String mytable=branch+"_"+semester+"_"+subject;

    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_marks",
"postgres", "nevada");
        System.out.println("Works");
        stmt = c.createStatement();
        String sql = "CREATE TABLE "+mytable+ "(Reg_no varchar(20) primary
key,Name varchar(30)," +
                "Asses_I varchar(3)," +
                "Asses_II varchar(3),end_sem varchar (3)," +
                "total varchar(4),grade varchar(3))";
        System.out.println(sql);
        stmt.executeUpdate(sql);
        stmt.close();
        c.close();
        System.out.println("table created ");
    } catch (Exception e) {
        System.err.println( e.getClass().getName()+" : "+ e.getMessage() );
        warning3.setText("Locked");
    }
    try{
        inittable2();
        System.out.println("hi there");
        loaddata2();
    }
    catch (Exception e){
        System.out.println(e);
    }
}

```

```

    }
}

private void inittable2(){
    initcols2();
}

private void initcols2(){
    // m_regno,m_name,m_asses_I,m_asses_II,m_endsem,m_total,m_grade;
    marks_regno.setCellValueFactory(new PropertyValueFactory<>("m_regno"));
    marks_name.setCellValueFactory(new PropertyValueFactory<>("m_name"));
    marks_asses_1.setCellValueFactory(new
PropertyValueFactory<>("m_asses_I"));
    marks_asses_2.setCellValueFactory(new
PropertyValueFactory<>("m_asses_II"));
    marks_endsem.setCellValueFactory(new
PropertyValueFactory<>("m_endsem"));
    marks_total.setCellValueFactory(new PropertyValueFactory<>("m_total"));
    marks_grade.setCellValueFactory(new PropertyValueFactory<>("m_grade"));

    editablecols2();
}

private void editablecols2(){
    marks_regno.setCellFactory(TextFieldTableCell.forTableColumn());
    marks_regno.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setM_regno(e
.getNewValue());
    });
    marks_name.setCellFactory(TextFieldTableCell.forTableColumn());
    marks_name.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setM_name(e.
getNewValue());
    });
    marks_asses_1.setCellFactory(TextFieldTableCell.forTableColumn());
    marks_asses_1.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setM_asses_I
(e.getNewValue());
    });
    marks_asses_2.setCellFactory(TextFieldTableCell.forTableColumn());
    marks_asses_2.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setM_asses_I
I(e.getNewValue());
    });
    marks_endsem.setCellFactory(TextFieldTableCell.forTableColumn());
    marks_endsem.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setM_endsem(
e.getNewValue());
    });
    marks_total.setCellFactory(TextFieldTableCell.forTableColumn());
    marks_total.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setM_total(e
.getNewValue());
    });
    marks_grade.setCellFactory(TextFieldTableCell.forTableColumn());
    marks_grade.setOnEditCommit(e ->{

```

```

e.getTableView().getItems().get(e.getTablePosition().getRow()).setM_grade(e
.getNewValue());
});
f_marks_table.setEditable(true);
}
ObservableList<String> reg2 = FXCollections.observableArrayList();
ObservableList<String> nam2 = FXCollections.observableArrayList();
ObservableList<sample.table.marks_table> data_table2 =
FXCollections.observableArrayList();

private void loaddata2() {
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/semsproject",
"postgres", "nevada");
        stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery("select * from studentsignup;");
        System.out.println("-----");
        System.out.println("reg_no\t NAME \t\t\n");
        while (rs.next()) {
            String reg_no = rs.getString("reg_no");
            String name = rs.getString("name");
            reg2.add(reg_no);
            nam2.add(name);
            System.out.printf("%s\t%s\t", reg_no, name);
            System.out.println();
        }
        System.out.println("-----");
    };
    rs.close();
    stmt.close();
    c.close();
} catch (Exception e) {
    System.err.println(e.getClass().getName() + ": " + e.getMessage());
}

//-----yahan pe data load karne ka
code -----

System.out.println("ab to chalna hi chhaiye");
for (int i = 0; i < reg2.size(); i++) {

    data_table2.add(new sample.table.marks_table(reg2.get(i),
nam2.get(i), "00", "00", "00", "00", "XY"));
}
f_marks_table.setItems(data_table2);
}

// yahan par marks update hoga

@FXML
void update_marks(ActionEvent event) {
    for (int i = 0; i < data_table2.size(); i++) {
        String t = f_branch_marks.getValue();
        String u = f_semester_marks.getValue();
        String v = f_subject_marks.getValue();
    }
}

```

```

String k=t+"_"+u+"_"+v;

try {
    Connection c = null;
    Statement stmt = null;
    Class.forName("org.postgresql.Driver");
    c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_marks",
"postgres", "nevada");
    System.out.println("Works");
    stmt = c.createStatement();
    String te= ""+data_table2.get(i).getM_regno()+"'"+"," +
        ""+data_table2.get(i).getM_name()+"'"+"," +
        ""+ data_table2.get(i).getM_asses_I()+"'"+"," +
        ""+data_table2.get(i).getM_asses_II()+"'"+"," +
        ""+data_table2.get(i).getM_endsem()+"'"+"," +
        ""+data_table2.get(i).getM_total()+"'"+"," +
        ""+data_table2.get(i).getM_grade()+"'";

    String sql = "insert into " + k + " Values (" + te+")";
    System.out.println(sql);
    stmt.executeUpdate(sql);
    stmt.close();
    c.close();
    System.out.println("table created ");
    warning3.setText("Updated");
} catch (Exception e) {
    System.err.println(e.getClass().getName() + ": " +
e.getMessage());
}

}

}

}

@FXML
void print_result(ActionEvent event) throws FileNotFoundException {
    String t = f_branch_marks.getValue();
    String u = f_semester_marks.getValue();
    String v = f_subject_marks.getValue();
    String k=t+"_"+u+"_"+v;
    String marksfile = "C:/Users/samun/Desktop/sems/Rough doc/marks.pdf";
    PdfDocument pdfDoc = new PdfDocument(new PdfWriter(marksfile));
    Document doc = new Document(pdfDoc, PageSize.A4.rotate());
    doc.add(new Paragraph("-----
"));
    doc.add(new Paragraph("Branch    : "+t));
    doc.add(new Paragraph("Semester  : "+u));
    doc.add(new Paragraph("Subject   : "+v));
    doc.add(new Paragraph("-----
"));

    Table table = new Table(7);
    table.setFontSize(10);
    table.setTextAlignment(TextAlignment.CENTER);
    table.addCell(new Cell().add("Reg_no").setBold());
    table.addCell(new Cell().add("Name").setBold());
    table.addCell(new Cell().add("Asses I").setBold());
    table.addCell(new Cell().add("Asses II").setBold());
    table.addCell(new Cell().add("End Sem").setBold());

```

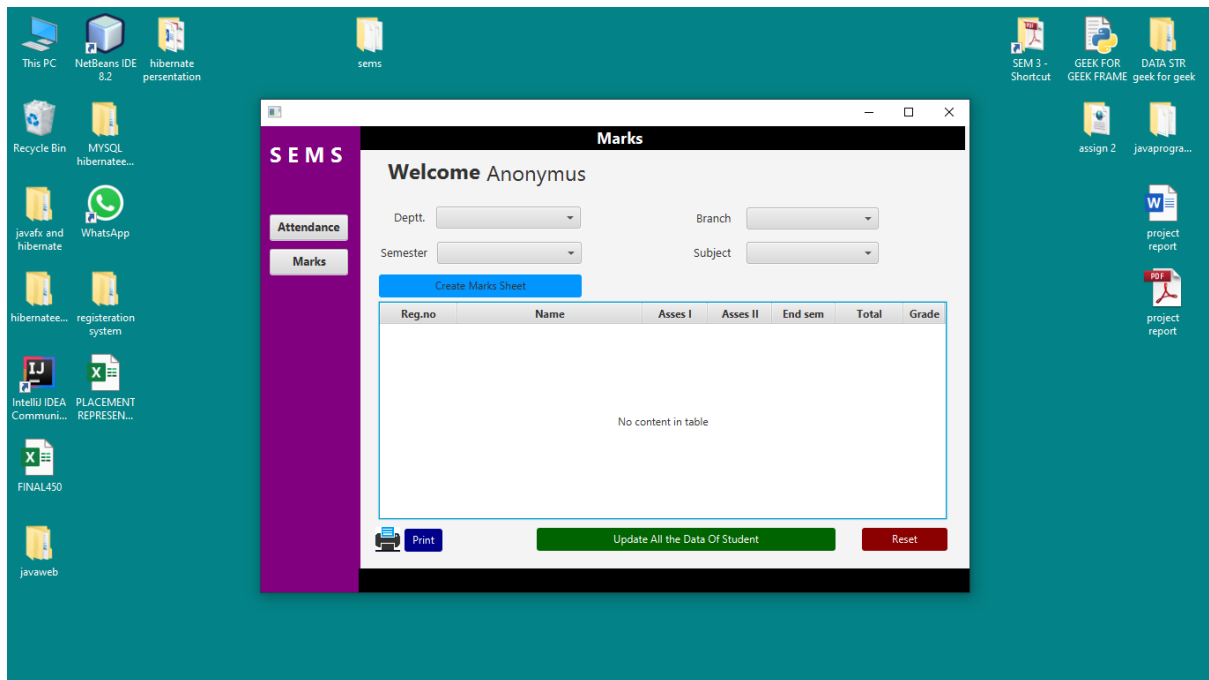




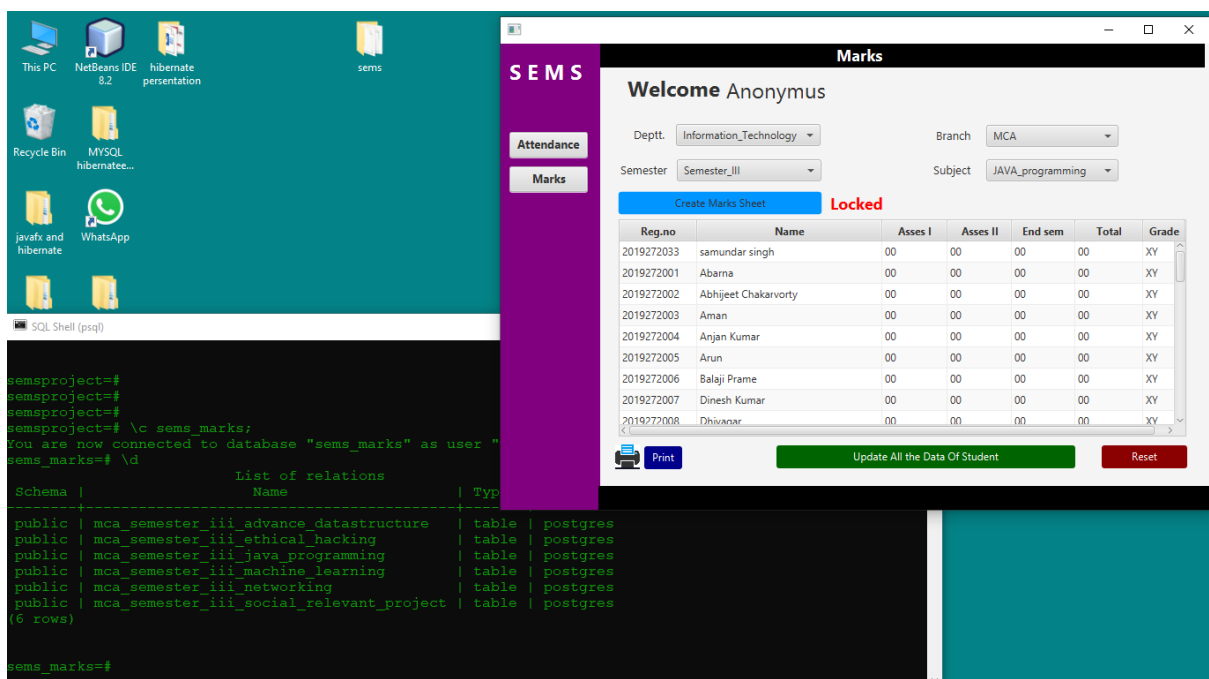
```
FXMLLoader.load(getClass().getResource("facultymarks.fxml"));
Stage stage = new Stage();
stage.setScene(new Scene(root));
stage.show();
}
```

**Screenshot:**

**Module view:**



**If the Marksheet is created then it popup a locked message otherwise it will create a blank table from where Marks of student is entered.**



On clicking the print button, we can see the previous or the marks that has been updated so far on clicking the update all data of student will allow faculty to overwrite the data of the database.

The screenshot displays a web application interface for managing student marks. On the left, a PDF viewer shows a document titled 'marks.pdf' with a table of marks. The table has columns for Reg.no, Name, Asses I, Asses II, End Sem, and Total. The data is as follows:

Reg.no	Name	Asses I	Asses II	End Sem	Total
2019272033	samundar singh	40	45	80	170
2019272001	Abarna	00	00	00	00
2019272002	Abhijeet Chakarvorty	00	00	00	00
2019272003	Aman	00	00	00	00
2019272004	Anjan Kumar	00	00	00	00
2019272005	Arun	00	00	00	00
2019272006	Balaji Prame	00	00	00	00
2019272007	Dinesh Kumar	00	00	00	00
2019272008	Dhivagar	00	00	00	00
2019272009	Hari Vishwesh	00	00	00	00
2019272010	Harithan	00	00	00	00
2019272011	Haritha	00	00	00	00
2019272013	Jayashree	00	00	00	00
2019272015	Karthickeyan	00	00	00	00
2019272016	Kokila	00	00	00	00
2019272017	Krishnaprnya	00	00	00	00
2019272018	Krithiga	00	00	00	00
2019272020	Madhumitha	00	00	00	00

On the right, the 'Marks' section shows a 'Welcome Anonymus' message and a 'Locked' status. Below this is a table with columns for Reg.no, Name, Asses I, Asses II, End sem, and Total. The data is as follows:

Reg.no	Name	Asses I	Asses II	End sem	Total
2019272033	samundar singh	00	00	00	00
2019272001	Abarna	00	00	00	00
2019272002	Abhijeet Chakarvorty	00	00	00	00
2019272003	Aman	00	00	00	00
2019272004	Anjan Kumar	00	00	00	00
2019272005	Arun	00	00	00	00
2019272006	Balaji Prame	00	00	00	00
2019272007	Dinesh Kumar	00	00	00	00
2019272008	Dhivagar	00	00	00	00

A red arrow points to the 'Print' button, which is located at the bottom of the 'Marks' section. The button is labeled 'Print' and has a printer icon next to it. Below the 'Print' button is a green button labeled 'Update All the Data Of Student'.

## 4. The Attendance module to upload the Attendance of Student in one click.(per day Attendance).

### Code:

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.ComboBox?>
<?import javafx.scene.control.DatePicker?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="500.0" prefWidth="805.0"
xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="sample.Controller">
    <children>
        <Label layoutX="134.0" layoutY="30.0" text="Welcome" >
            <font>
                <Font name="System Bold" size="24.0" />
            </font>
        </Label>
        <Label layoutX="246.0" layoutY="32.0" text="Anonymus">
            <font>
                <Font size="24.0" />
            </font>
        </Label>
        <VBox prefHeight="508.0" prefWidth="113.0" style="-fx-background-
color: purple;">
            <children>
                <Button mnemonicParsing="false" prefHeight="25.0"
prefWidth="89.0" text="Attendance">
                    <VBox.margin>
                        <Insets left="10.0" top="100.0" />
                    </VBox.margin>
                    <font>
                        <Font name="System Bold" size="13.0" />
                    </font>
                </Button>
                <Button mnemonicParsing="false" onAction="#redirect_marks"
prefHeight="25.0" prefWidth="89.0" text="Marks">
                    <font>
                        <Font name="System Bold" size="13.0" />
                    </font>
                    <VBox.margin>
                        <Insets left="10.0" top="10.0" />
                    </VBox.margin>
                </Button>
            </children>
        </VBox>
    </children>
</AnchorPane>
```

```

<Label layoutX="10.0" layoutY="10.0" prefHeight="44.0"
prefWidth="103.0" text="S E M S" textFill="WHITE">
    <font>
        <Font name="System Bold" size="24.0" />
    </font>
</Label>
<Label layoutX="483.0" layoutY="95.0" text="Course id">
    <font>
        <Font size="13.0" />
    </font>
</Label>
<Label layoutX="134.0" layoutY="95.0" text="Semester">
    <font>
        <Font size="13.0" />
    </font>
</Label>
<Label layoutX="157.0" layoutY="134.0" text="Date">
    <font>
        <Font size="13.0" />
    </font>
</Label>
<TableView fx:id="attendance_table" layoutX="136.0" layoutY="205.0"
prefHeight="200.0" prefWidth="642.0">
    <columns>
        <TableColumn fx:id="f_table_regno" prefWidth="88.0"
text="Reg.no" />
        <TableColumn fx:id="f_table_name" prefWidth="307.0" text="Name"
/>
        <TableColumn fx:id="f_table_present" minWidth="0.0"
prefWidth="113.0" text="Present" />
        <TableColumn fx:id="f_table_absent" prefWidth="133.0"
text="Absent" />
    </columns>
</TableView>
<Button layoutX="313.0" layoutY="415.0" mnemonicParsing="false"
onAction="#update_attendance" prefHeight="25.0" prefWidth="339.0" style="-
fx-background-color: darkgreen;" text="Update All the Data Of Student"
textFill="WHITE" />
<Button layoutX="682.0" layoutY="415.0" mnemonicParsing="false"
prefHeight="25.0" prefWidth="97.0" style="-fx-background-color: darkred;"
text="Reset" textFill="#fff8f8" />
<Label layoutX="113.0" prefHeight="27.0" prefWidth="678.0" style="-
fx-background-color: black;" text="
Attendance " textFill="WHITE">
    <font>
        <Font name="System Bold" size="18.0" />
    </font>
</Label>
<DatePicker fx:id="attendance_date" layoutX="197.0" layoutY="131.0"
prefHeight="25.0" prefWidth="158.0" />
<ComboBox fx:id="semester_list" layoutX="198.0" layoutY="93.0"
prefWidth="150.0" />
<ComboBox fx:id="course_list" layoutX="553.0" layoutY="92.0"
onMousePressed="#select_course" prefWidth="150.0" />
<Button layoutX="135.0" layoutY="173.0" mnemonicParsing="false"
onAction="#create_attendance_sheet" prefHeight="25.0" prefWidth="230.0"
style="-fx-background-color: #0095ff;" text="Create Attendance Sheet" />
<Button layoutX="546.0" layoutY="173.0" mnemonicParsing="false"
onAction="#fetch_attendance_sheet" prefHeight="25.0" prefWidth="222.0"
style="-fx-background-color: darkgreen;" text="Fetch Attendance Sheet"
textFill="WHITE" />

```

```

        <Label fx:id="warning2" layoutX="378.0" layoutY="173.0"
textFill="#f80202">
            <font>
                <Font name="System Bold" size="18.0" />
            </font>
        </Label>
        <Label layoutX="113.0" layoutY="481.0" prefHeight="27.0"
prefWidth="692.0" style="-fx-background-color: black;" textFill="WHITE">
            <font>
                <Font name="System Bold" size="18.0" />
            </font>
        </Label>
    </children>
</AnchorPane>

```

## Controller code:

```

@FXML
private ComboBox<String> semester_list;
@FXML
private ComboBox<String> course_list;
@FXML
private TableView<sample.table.attendance_table> attendance_table;
@FXML
private DatePicker attendance_date;
@FXML
private TableColumn<sample.table.attendance_table , String> f_table_regno;
@FXML
private TableColumn<sample.table.attendance_table , String> f_table_name;
@FXML
private TableColumn<sample.table.attendance_table , String>
f_table_present;
@FXML
private TableColumn<sample.table.attendance_table , String> f_table_absent;

@FXML
public void select_course(javafx.scene.input.MouseEvent mouseEvent) {

    if (semester_list.getValue().equals("Semester_III")) {
        System.out.println(semester_list.getValue());
        ObservableList<String> li = FXCollections.observableArrayList();
        course_list.getItems().setAll((String) null);
        li.add("JAVA_programming");
        li.add("Advance_DataStructure");
        li.add("Machine_Learning");
        li.add("Networking");
        li.add("Social_Relevant_Project");
        li.add("Ethical_Hacking");
        course_list.getItems().addAll(li);
    }
    else{
        course_list.getItems().setAll((String) null);
    }

}

```

```

@FXML
void create_attendance_sheet(ActionEvent event) {
    String
s=attendance_date.getValue().format(DateTimeFormatter.ofPattern("ddMMyyyy"))
);
    String t=semester_list.getValue();
    String u=course_list.getValue();
    String k=t+"_"+u+"_"+s;
    System.out.println(k);
    //      jab atendance sheet banaya jaye toh Ek table create ho jaye ---
    -----Uska code-----
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_attendan
ce", "postgres", "nevada");
        System.out.println("Works");
        stmt = c.createStatement();
        String sql = "CREATE TABLE "+k+ "(reg_no varchar (20) primary
key,Name varchar(30),Present varchar(2),Absent varchar(2))";
        System.out.println(sql);
        stmt.executeUpdate(sql);
        stmt.close();
        c.close();
        System.out.println("table created ");
    } catch (Exception e) {
        System.err.println( e.getClass().getName()+" : "+ e.getMessage() );
        warning2.setText("Already Exist");
    }
    //-----fetch wala aur autamatic table banane wala
    bhi chhaiye uske liye -----
    try{
        inittable();
        System.out.println("hi there");
        loaddata();
    }
    catch (Exception e){
        System.out.println(e);
    }

    System.out.println("XXXXXXXXXXXXXXXXXXXXrterterrtXXXXXXXXXXXXXXXXXXXXXXXXXXXX");
}

@FXML
private Label warning2;

private void inittable() {
    initcols();
}

private void initcols() {
    // f_atd_regno,f_atd_name,f_atd_present,f_atd_absent
    f_table_regno.setCellValueFactory(new
PropertyFactory<>("f_atd_regno"));
    f_table_name.setCellValueFactory(new
PropertyFactory<>("f_atd_name"));
    f_table_present.setCellValueFactory(new

```

```

PropertyValueFactory<>("f_atd_present"));
    f_table_absent.setCellValueFactory(new
PropertyValueFactory<>("f_atd_absent"));
    editablecols();
}
private void editablecols() {
    f_table_regno.setCellFactory(TextFieldTableCell.forTableColumn());
    f_table_regno.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setF_atd_reg
no(e.getNewValue());
    });

    f_table_name.setCellFactory(TextFieldTableCell.forTableColumn());
    f_table_name.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setF_atd_nam
e(e.getNewValue());
    });

    f_table_present.setCellFactory(TextFieldTableCell.forTableColumn());
    f_table_present.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setF_atd_pre
sent(e.getNewValue());
    });

    f_table_absent.setCellFactory(TextFieldTableCell.forTableColumn());
    f_table_absent.setOnEditCommit(e ->{

e.getTableView().getItems().get(e.getTablePosition().getRow()).setF_atd_abs
ent(e.getNewValue());
    });

    attendance_table.setEditable(true);
}
ObservableList<String> reg = FXCollections.observableArrayList();
ObservableList<String> nam = FXCollections.observableArrayList();
ObservableList<sample.table.attendance_table> data_table =
FXCollections.observableArrayList();
private void loaddata() {
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/semsproject",
"postgres", "nevada");
        System.out.println("Works");
        stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery("select * from studentsignup;");
        System.out.println("-----");
        System.out.println("reg_no\t NAME \t\t\n");
        while (rs.next()) {
            String reg_no = rs.getString("reg_no");
            String name = rs.getString("name");
            reg.add(reg_no);
            nam.add(name);
            System.out.printf("%s\t%s\t", reg_no, name);
            System.out.println();
        }
    }
}

```

```

        System.out.println("-----");
    );
    rs.close();
    stmt.close();
    c.close();
} catch (Exception e) {
    System.err.println(e.getClass().getName() + ": " + e.getMessage());
}
System.out.println("ab to chalna hi chhaiye");
for (int i = 0; i < reg.size(); i++) {

    data_table.add(new sample.table.attendance_table(reg.get(i),
nam.get(i), "", ""));
}
attendance_table.setItems(data_table);
System.out.println(data_table.get(1).getF_atd_name());
}

@FXML
void update_attendance(ActionEvent event) {
    for (int i = 0; i < data_table.size(); i++) {

        String s =
attendance_date.getValue().format(DateTimeFormatter.ofPattern("ddMMyyyy"));
        String t = semester_list.getValue();
        String u = course_list.getValue();
        String k = t + "_" + u + "_" + s;
        try {
            Connection c = null;
            Statement stmt = null;
            Class.forName("org.postgresql.Driver");
            c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_attendan
ce", "postgres", "nevada");
            System.out.println("Works");
            stmt = c.createStatement();
            String te= ""+data_table.get(i).getF_atd_regno()+"'," +
""+data_table.get(i).getF_atd_name()+"'," +
""+ data_table.get(i).getF_atd_present()+"'"+
+", "+
""+data_table.get(i).getF_atd_absent()+"'";

            String sql = "insert into " + k + " Values (" +te+")";
            System.out.println(sql);
            stmt.executeUpdate(sql);
            stmt.close();
            c.close();
            System.out.println("table created ");
            warning2.setText("Updated");
        } catch (Exception e) {
            System.err.println(e.getClass().getName() + ": " +
e.getMessage());
        }

    }

}
}

```



//-----Yahan pe Pdf me badalne wala  
command hai-----

@FXML

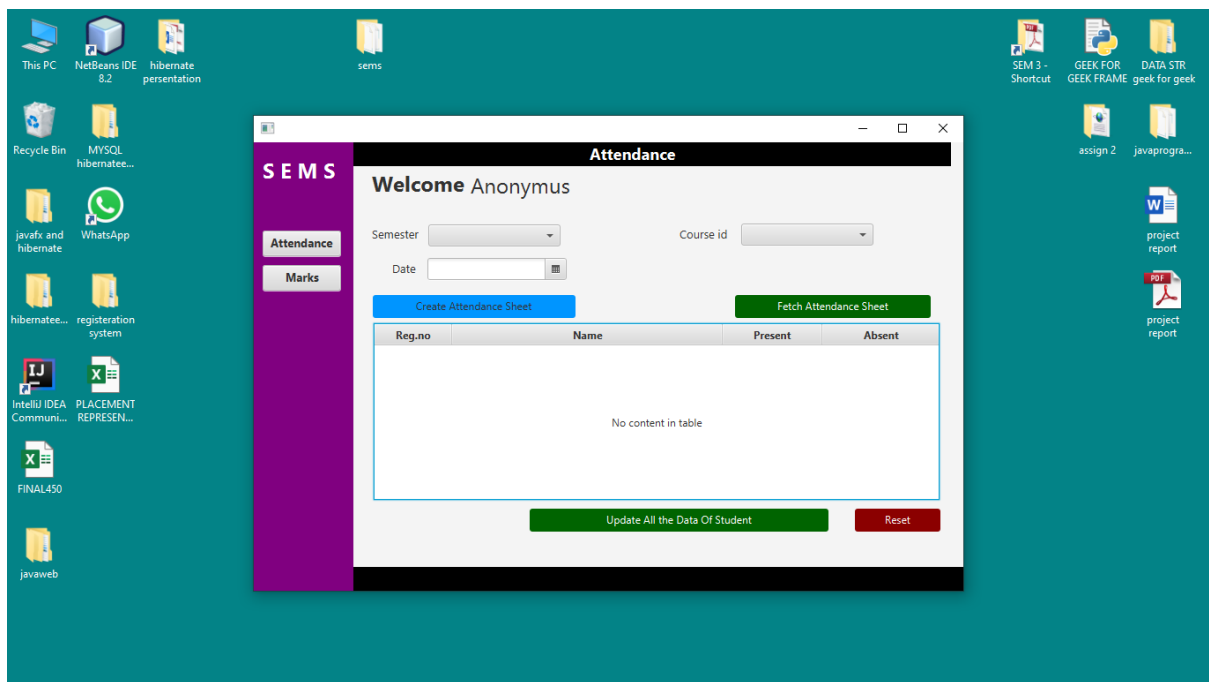
```
void fetch_attendance_sheet(ActionEvent event) throws FileNotFoundException
{
    String
s=attendance_date.getValue().format(DateTimeFormatter.ofPattern("ddMMyyyy"))
);
    String t=semester_list.getValue();
    String u=course_list.getValue();
    String k=t+"_"+u+"_"+s;
    String file = "C:/Users/samun/Desktop/sems/Rough
doc/addingTableToPDF.pdf";
    PdfDocument pdfDoc = new PdfDocument(new PdfWriter(file));
    Document doc = new Document(pdfDoc);

    Table table = new Table(4);
    table.addCell(new Cell().add("Reg_no"));
    table.addCell(new Cell().add("Name"));
    table.addCell(new Cell().add("Present"));
    table.addCell(new Cell().add("Absent"));
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_attendan
ce", "postgres", "nevada");
        stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery( "select * from "+k+";" );
        System.out.println("-----");
        while ( rs.next() ) {
            String pdf_reg=rs.getString("reg_no");
            String pdf_name = rs.getString("name");
            String pdf_present = rs.getString("present");
            String pdf_absent = rs.getString("absent");
            table.addCell(new Cell().add(pdf_reg));
            table.addCell(new Cell().add(pdf_name));
            table.addCell(new Cell().add(pdf_present));
            table.addCell(new Cell().add(pdf_absent));

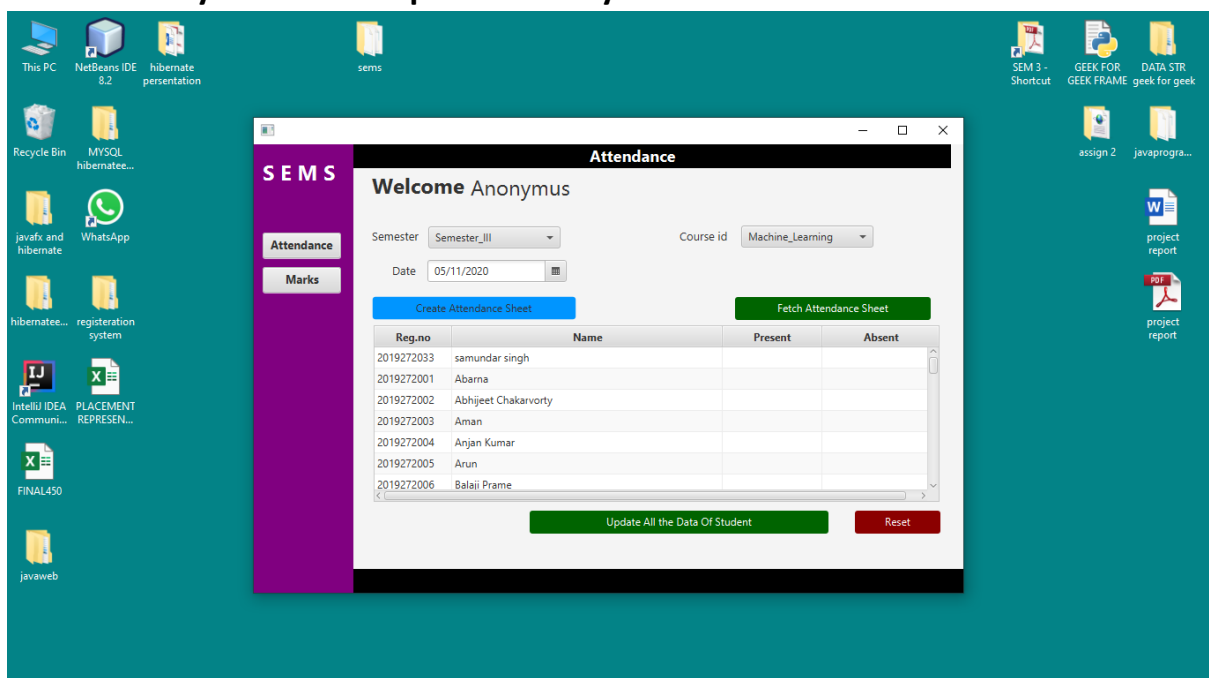
        }
        doc.add(table);
        doc.close();
        rs.close();
        stmt.close();
        c.close();
        File attend_data=new File("C:/Users/samun/Desktop/sems/Rough
doc/addingTableToPDF.pdf");
        try {
            Desktop.getDesktop().open( attend_data);
        } catch (IOException e) {
            e.printStackTrace();
        }
    } catch (Exception e) {
        System.err.println( e.getClass().getName()+" : "+ e.getMessage() );
        System.out.println("kyu nahi chal raha hai");
    }
}
```

**Screenshot:**

**Module view:**



On clicking create attendance sheet it will create a blank attendance sheet from where faculty can create a particular day attendance sheet.



After inserting data on the table click on update data then the attendance is updated for today and a table with that date name is created in database .

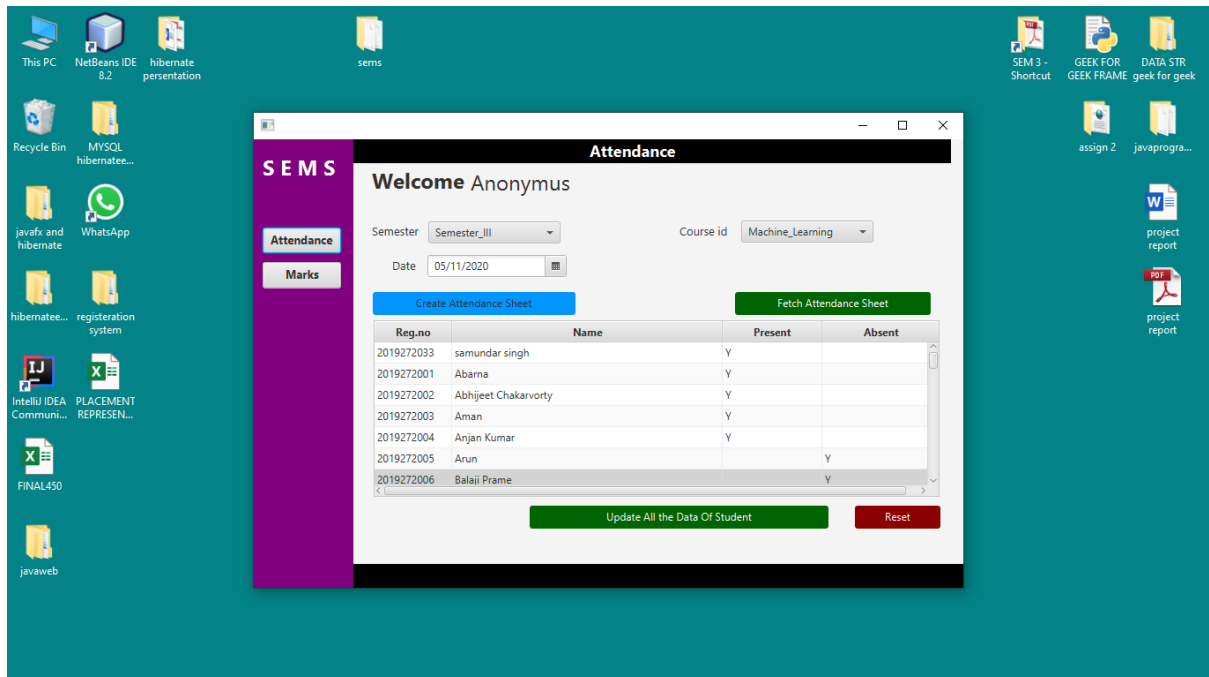
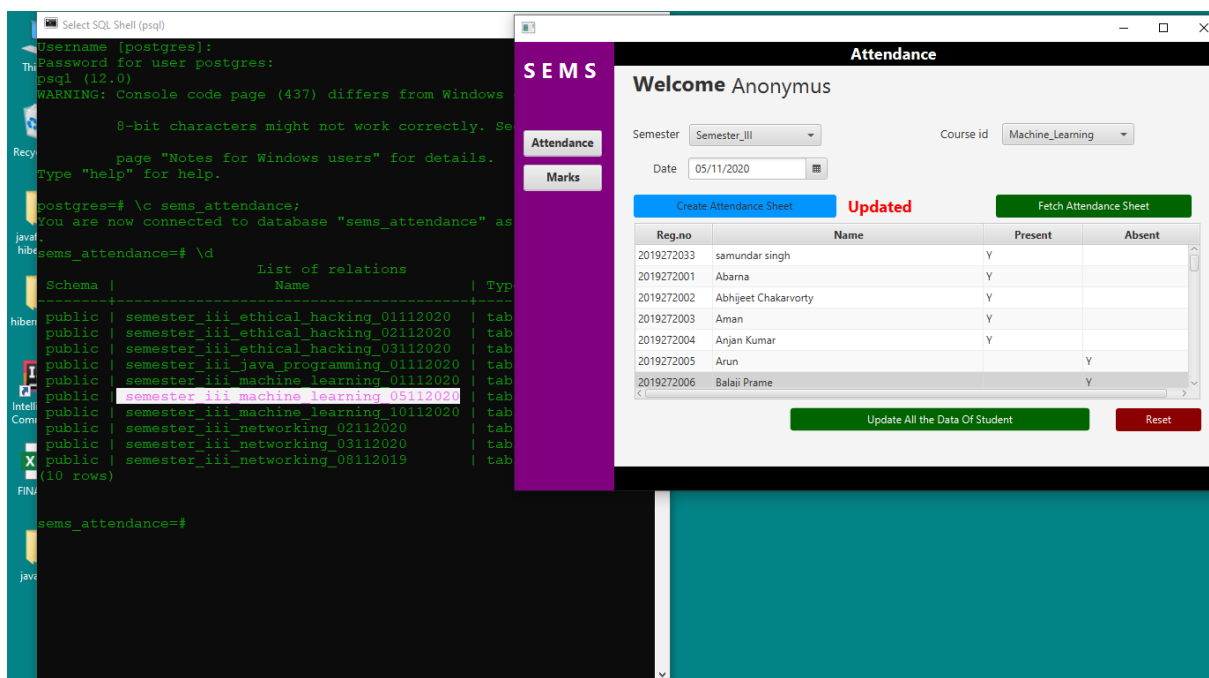


Table with the data has been created.



Data in the table :

The screenshot displays two side-by-side windows. The left window is a SQL Shell (psql) showing the execution of a query to select all records from the 'semester\_iii\_machine\_learning' table. The right window is a web application titled 'Attendance' with a 'Welcome Anonymus' message. It features a sidebar with 'Attendance' and 'Marks' tabs. The main area contains a form to select a semester and course, a date picker, and buttons for 'Create Attendance Sheet', 'Fetch Attendance Sheet', and 'Update All the Data Of Student'. A table below the form shows the current attendance data for the selected semester and course.

Reg.no	Name	Present	Absent
2019272033	samundar singh	Y	
2019272001	Abarna	Y	
2019272002	Abhijeet Chakarvorty	Y	
2019272003	Aman	Y	
2019272004	Anjan Kumar	Y	
2019272005	Arun		Y
2019272006	Balaji Prame		Y
2019272007	Dinesh Kumar		
2019272008	Dhivagar		
2019272009	Hari Vishwesh		
2019272010	Hariharan		
2019272011	Haritha		
2019272013	Jayashree		
2019272015	Karthickeyan		
2019272016	Kokila		
2019272017	Krishnapriya		
2019272018	Krithiga		
2019272020	Madhumitha		
2019272021	Manikandan		
2019272022	Manjula		
2019272023	Mathan Kumar		
2019272024	MD Shabbir Alam		
2019272025	Mukesh Patel		
2019272026	Nallendran		
2019272027	Noorul Ameen		
2019272028	Parkavi		
2019272030	Priyanka		
2019272031	Raja		

We can also download the Attendance by using the fetch attendance Sheet in pdf format.

The screenshot shows a PDF viewer window titled 'addingTableToPDF.pdf - Adobe Reader'. The main content area displays a table with the same attendance data as shown in the previous screenshot. The table has four columns: 'Reg\_no', 'Name', 'Present', and 'Absent'.

Reg_no	Name	Present	Absent
2019272033	samundar singh	Y	
2019272001	Abarna	Y	
2019272002	Abhijeet Chakarvorty	Y	
2019272003	Aman	Y	
2019272004	Anjan Kumar	Y	
2019272005	Arun		Y
2019272006	Balaji Prame		Y
2019272007	Dinesh Kumar		
2019272008	Dhivagar		
2019272009	Hari Vishwesh		
2019272010	Hariharan		
2019272011	Haritha		
2019272013	Jayashree		
2019272015	Karthickeyan		
2019272016	Kokila		
2019272017	Krishnapriya		
2019272018	Krithiga		
2019272020	Madhumitha		
2019272021	Manikandan		
2019272022	Manjula		
2019272023	Mathan Kumar		
2019272024	MD Shabbir Alam		
2019272025	Mukesh Patel		
2019272026	Nallendran		
2019272027	Noorul Ameen		
2019272028	Parkavi		
2019272030	Priyanka		

# Student Module

## 4. Student signup and login:

### Code:

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.DatePicker?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.PasswordField?>
<?import javafx.scene.control.Tab?>
<?import javafx.scene.control.TabPane?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.text.Font?>

<AnchorPane fx:id="student_portal" prefHeight="400.0" prefWidth="600.0"
xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="sample.Controller">
    <children>
        <TabPane layoutX="100.0" layoutY="47.0" prefHeight="316.0"
prefWidth="454.0" style="-fx-background-color: powderblue;"
tabClosingPolicy="UNAVAILABLE">
            <tabs>
                <Tab text="login ">
                    <content>
                        <AnchorPane minHeight="0.0" minWidth="0.0" prefHeight="180.0"
prefWidth="200.0">
                            <children>
                                <Label layoutX="165.0" layoutY="87.0"
text="Reg.no">
                                    <font>
                                        <Font name="System Bold" size="14.0" />
                                    </font>
                                </Label>
                                <TextField fx:id="login_regno" layoutX="222.0"
layoutY="84.0" />
                                <PasswordField fx:id="login_password"
layoutX="222.0" layoutY="120.0" />
                                <Label layoutX="151.0" layoutY="123.0"
text="Password">
                                    <font>
                                        <Font name="System Bold" size="14.0" />
                                    </font>
                                </Label>
                                <ImageView fitHeight="108.0" fitWidth="113.0"
layoutX="23.0" layoutY="62.0" pickOnBounds="true" preserveRatio="true">
                                    <image>
                                        <Image url="@../images/avatar.png" />
                                    </image>
                                </ImageView>
                                <Button layoutX="258.0" layoutY="160.0"
mnemonicParsing="false" onAction="#stud_login" prefHeight="25.0"
prefWidth="61.0" style="-fx-background-color: darkgreen;" text="login"
textFill="WHITE" />
                                    <Label fx:id="testing" layoutX="226.0"
layoutY="197.0" prefHeight="35.0" prefWidth="154.0" textFill="#ff0202">
                                        <font>
```

```

        <Font name="System Bold" size="18.0" />
    </font>
</Label>
</children>
</AnchorPane>
</content>
</Tab>
<Tab text="Sign up">
    <content>
        <AnchorPane minHeight="0.0" minWidth="0.0" prefHeight="287.0"
prefWidth="463.0">
            <children>
                <Label layoutX="37.0" layoutY="22.0" text="Reg.no">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="42.0" layoutY="52.0" text="Name">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="37.0" layoutY="85.0" text="Stream">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="43.0" layoutY="116.0" text="D.O.B">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="31.0" layoutY="176.0"
text="Address">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <Label layoutX="39.0" layoutY="146.0" text="Phone">
                    <font>
                        <Font size="14.0" />
                    </font>
                </Label>
                <TextField fx:id="student_regno" layoutX="89.0"
layoutY="16.0" />
                <TextField fx:id="student_name" layoutX="90.0"
layoutY="49.0" />
                <TextField fx:id="student_stream" layoutX="90.0"
layoutY="82.0" />
                <TextField fx:id="student_phone" layoutX="90.0"
layoutY="142.0" />
                <TextField fx:id="student_address" layoutX="90.0"
layoutY="172.0" />
                <DatePicker fx:id="student_dob" layoutX="89.0"
layoutY="112.0" prefHeight="25.0" prefWidth="150.0" />
                <Button layoutX="122.0" layoutY="241.0"
mnemonicParsing="false" onAction="#student_register" prefHeight="25.0"
prefWidth="115.0" style="-fx-background-color: darkgreen;" text="Register"
textFill="WHITE" />
                <ImageView fx:id="student_image" fitHeight="171.0"
fitWidth="176.0" layoutX="282.0" layoutY="9.0" pickOnBounds="true"

```

```

preserveRatio="true">
    <image>
        <Image url="@../images/default%20image.jpg"
    />
    </image>
</ImageView>
<Button layoutX="282.0" layoutY="185.0"
mnemonicParsing="false" onAction="#image_chooser" prefHeight="25.0"
prefWidth="128.0" text="chose img.." />
<Label layoutX="22.0" layoutY="206.0"
text="Password">
    <font>
        <Font size="14.0" />
    </font>
</Label>
<TextField fx:id="student_password" layoutX="89.0"
layoutY="202.0" />
</children>
</AnchorPane>
</content>
</Tab>
</tabs>
</TabPage>
<Label layoutX="100.0" layoutY="2.0" text="Student Portal">
    <font>
        <Font name="System Bold" size="24.0" />
    </font>
</Label>
</children>
</AnchorPane>

```

## Controller code:

```

@FXML
void redirect_home(ActionEvent event) throws IOException {
    Parent root =
FXMLLoader.load(getClass().getResource("studentlogin.fxml"));
    Stage stage = new Stage();
    stage.setScene(new Scene(root));
    stage.show();
}

```

```

@FXML
private TextField student_regno;
@FXML
private TextField student_name;
@FXML
private TextField student_stream;
@FXML
private TextField student_phone;
@FXML
private TextField student_address;
@FXML
private DatePicker student_dob;
@FXML
private TextField student_password;
@FXML
private Label testing;

```





```

File file;
FileInputStream fis;

@FXML
void image_chooser(ActionEvent event) throws FileNotFoundException {
    FileChooser filechoose = new FileChooser();
    filechoose.setTitle("Choose Image");
    Stage stage = (Stage) student_portal.getScene().getWindow();
    file = filechoose.showOpenDialog(stage);
    FileInputStream inputStream = new
FileInputStream(file.getAbsolutePath());
    Image image = new Image(inputStream);
    if (file != null) {
        student_image.setImage(image);
    }
    File file1 = new File(file.getAbsolutePath());
    fis = new FileInputStream(file1);
}

///----- student login ke liye-----
-----

@FXML
private TextField login_regno;

@FXML
private PasswordField login_password;

@FXML
void stud_login(ActionEvent event) {
    String log_regno = login_regno.getText();
    String log_pass = login_password.getText();

    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/semsproject",
"postgres", "nevada");
        stmt = c.createStatement();
        System.out.println("open database sucessfully");
        // System.out.println(log_regno);
// debugging
        String sqlStatement = "SELECT * FROM studentsignup WHERE reg_no='"
+ log_regno + "'";
        //System.out.println(sqlStatement);
// debugging
        ResultSet rs = stmt.executeQuery(sqlStatement);
        String fetch_password = null;
        String fetch_name = null;

        while (rs.next()) {
            fetch_password = rs.getString("password");
            fetch_name=rs.getString("name");
        }
        System.out.println(log_regno + " " + fetch_password);
        if (log_pass.equals(fetch_password)) {
            try {
                Parent root =

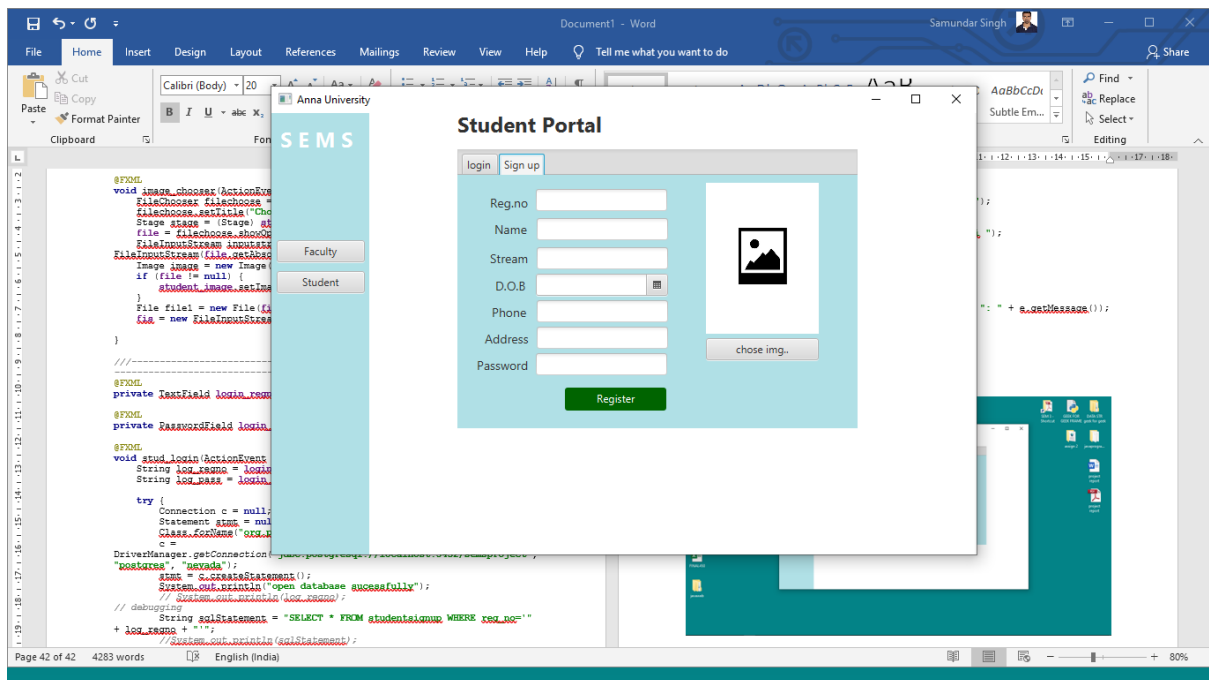
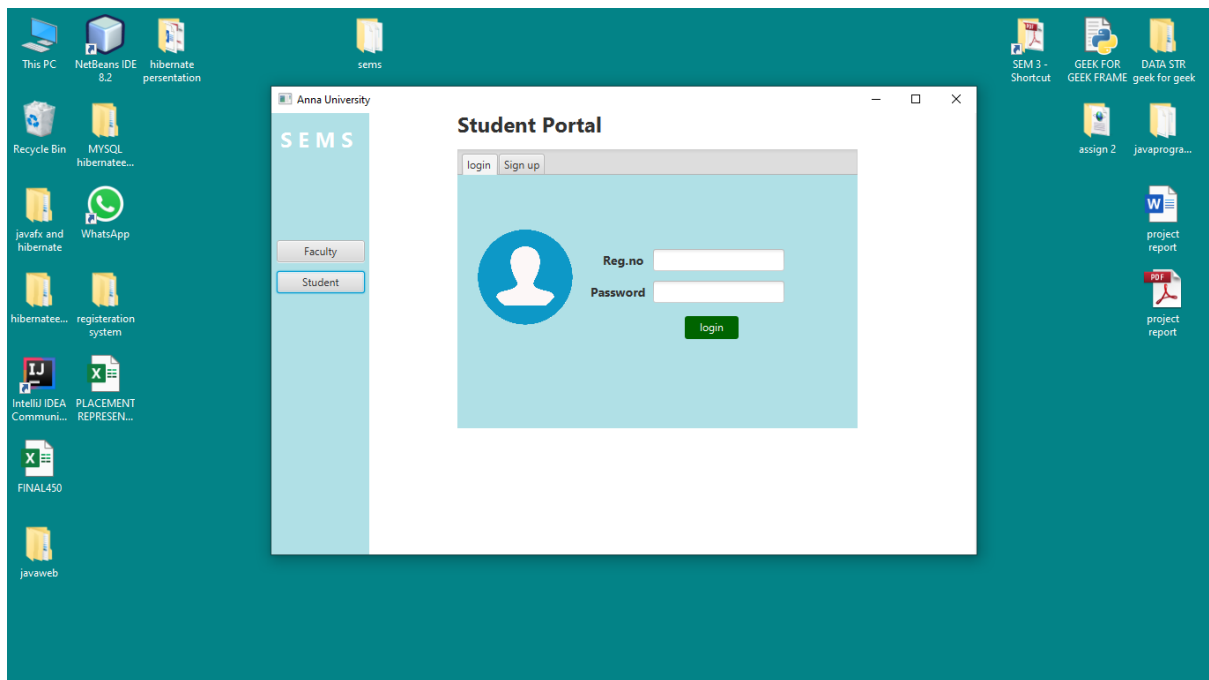
```

```

FXMLLoader.load(getClass().getResource("studentmarksview.fxml"));
    Stage stage = new Stage();
    stage.setScene(new Scene(root));
    stage.show();
} catch (FileNotFoundException e) {
    System.out.println("file not found");
}
} else {
    System.out.println("galat enter kiya hai ");
    testing.setText("INVALID USER");
}
rs.close();
stmt.close();
c.close();
} catch (Exception e) {
    System.err.println(e.getClass().getName() + ": " + e.getMessage());
    System.exit(0);
}
}

```

## Screenshot:



The login and signup page functions is same as mentioned in faculty section.

## 4. Student Marks page:

### Code:

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.geometry.Insets?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.ChoiceBox?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.TableColumn?>
<?import javafx.scene.control.TableView?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>
<?import javafx.scene.text.Text?>

<AnchorPane prefHeight="500.0" prefWidth="800.0"
xmlns="http://javafx.com/javafx/11.0.1" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="sample.Controller">
    <children>
        <AnchorPane fx:id="marsk_portal" prefHeight="500.0"
prefWidth="800.0">
            <children>
                <TableView fx:id="s_marks_table2" layoutX="134.0"
layoutY="163.0" prefHeight="256.0" prefWidth="642.0">
                    <columns>
                        <TableColumn fx:id="m_serial" prefWidth="39.0"
text="S.no" />
                        <TableColumn fx:id="smarks_name" prefWidth="259.0"
text="Subject Name" />
                        <TableColumn fx:id="smarks_asses_1" prefWidth="74.0"
text="Asses I" />
                        <TableColumn fx:id="smarks_asses_2" prefWidth="73.0"
text="Asses II" />
                        <TableColumn fx:id="smarks_endsem" prefWidth="72.0"
text="End sem" />
                        <TableColumn fx:id="smarks_total" prefWidth="77.0"
text="Total" />
                        <TableColumn fx:id="smarks_grade" prefWidth="47.0"
text="Grade" />
                    </columns>
                </TableView>
                <Label layoutX="144.0" layoutY="18.0" text="Welcome " >
                    <font>
                        <Font name="System Bold" size="24.0" />
                    </font>
                </Label>
                <Label fx:id="s_name_falg" layoutX="256.0" layoutY="20.0"
text="Anonymus">
                    <font>
                        <Font size="24.0" />
                    </font>
                </Label>
                <VBox prefHeight="508.0" prefWidth="113.0" style="-fx-
background-color: teal;">
                    <children>
```

```

        <Button mnemonicParsing="false" prefHeight="25.0"
prefWidth="89.0" text="Marks">
            <VBox.margin>
                <Insets left="10.0" top="100.0" />
            </VBox.margin>
            <font>
                <Font name="System Bold" size="13.0" />
            </font>
        </Button>
    </children>
</VBox>
<Label layoutX="10.0" layoutY="10.0" prefHeight="44.0"
prefWidth="103.0" text="S E M S" textFill="WHITE">
    <font>
        <Font name="System Bold" size="24.0" />
    </font>
</Label>
<Label layoutX="144.0" layoutY="97.0" text="Stream">
    <font>
        <Font size="13.0" />
    </font>
</Label>
<Label layoutX="491.0" layoutY="97.0" text="Semester">
    <font>
        <Font size="13.0" />
    </font>
</Label>
<ChoiceBox fx:id="s_marks_semester" layoutX="555.0"
layoutY="94.0" prefWidth="150.0" />
<Button layoutX="677.0" layoutY="431.0" mnemonicParsing="false"
prefHeight="35.0" prefWidth="97.0" style="-fx-background-color: darkred;"
text="Close" textFill="#fff8f8">
    <font>
        <Font size="14.0" />
    </font></Button>
<Label layoutX="139.0" layoutY="437.0" text="C.G.P.A :">
    <font>
        <Font name="System Bold" size="18.0" />
    </font>
</Label>
<Text fx:id="cgpa_cal" layoutX="219.0" layoutY="457.0"
strokeType="OUTSIDE" strokeWidth="0.0" text="0.00">
    <font>
        <Font name="System Bold" size="18.0" />
    </font>
</Text>
<Label layoutX="143.0" layoutY="69.0" text="Reg.no">
    <font>
        <Font size="13.0" />
    </font>
</Label>
<TextField fx:id="s_marks_reg" layoutX="202.0" layoutY="66.0"
onMousePressed="#s_marks_fetchreg" promptText="Enter reg.no.." />
<TextField fx:id="s_marks_stream" layoutX="202.0"
layoutY="94.0" />
<Button layoutX="132.0" layoutY="132.0" mnemonicParsing="false"
onAction="#Show_result" prefHeight="25.0" prefWidth="642.0" style="-fx-
background-color: darkgreen;" text="Click here to Show the Result.."
textFill="WHITE" />
<Button layoutX="559.0" layoutY="430.0" mnemonicParsing="false"
onAction="#print_marks" prefHeight="35.0" prefWidth="97.0" style="-fx-

```

```

background-color: darkblue;" text="      Print" textFill="#ffffef">
    <font>
        <Font name="System Bold" size="14.0" />
    </font>
</Button>
<ImageView fitHeight="28.0" fitWidth="44.0" layoutX="566.0"
layoutY="433.0" pickOnBounds="true" preserveRatio="true">
    <image>
        <Image url="@../images/print%20icon.png" />
    </image>
</ImageView>
</children>
</AnchorPane>
</children>
</AnchorPane>

```

## Controller Code:

```

@FXML
private Label s_name_falg;
@FXML
private ChoiceBox<String> s_marks_semester;
@FXML
private Text cgpa_cal;
@FXML
private TextField s_marks_reg;
@FXML
private TextField s_marks_stream;
@FXML
private TableView<sample.table.result> s_marks_table2;
@FXML
private TableColumn<sample.table.result,String> m_serial;
@FXML
private TableColumn<sample.table.result,String> smarks_name;
@FXML
private TableColumn<sample.table.result,String> smarks_asses_1;
@FXML
private TableColumn<sample.table.result,String> smarks_asses_2;
@FXML
private TableColumn<sample.table.result,String> smarks_endsem;
@FXML
private TableColumn<sample.table.result,String> smarks_total;
@FXML
private TableColumn<sample.table.result,String> smarks_grade;

@FXML
public void s_marks_fetchreg(javafx.scene.input.MouseEvent mouseEvent) {
    String reg = s_marks_reg.getText();
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
        DriverManager.getConnection("jdbc:postgresql://localhost:5432/semsproject",
"postgres", "nevada");
        stmt = c.createStatement();
        System.out.println("open database sucessfully");
    }
}

```

```

        String sqlStatement = "SELECT * FROM studentsignup WHERE reg_no='"+
+reg + "'";
        ResultSet rs = stmt.executeQuery(sqlStatement);

        String fetch_name = null;
        String fetch_stream = null;

        while (rs.next()) {
            fetch_name = rs.getString("name");
            fetch_stream = rs.getString("stream");
        }
        s_name_falg.setText(fetch_name);
        s_marks_stream.setText(fetch_stream);
        initilizable3();

    } catch (SQLException | ClassNotFoundException e) {
        System.out.println(" file not found");
    }
}

private void initilizable3(){
    ObservableList<String> li5 = FXCollections.observableArrayList();
    s_marks_semester.getItems().setAll((String) null);
    li5.add("Semester_I");
    li5.add("Semester_II");
    li5.add("Semester_III");
    li5.add("Semester_IV");
    li5.add("Semester_V");
    li5.add("Semester_VI");
    s_marks_semester.getItems().addAll(li5);
}

ObservableList<String> temp1 = FXCollections.observableArrayList();
ObservableList<String> temp2 = FXCollections.observableArrayList();
ObservableList<String> temp3 = FXCollections.observableArrayList();
ObservableList<String> temp4 = FXCollections.observableArrayList();
ObservableList<String> temp5 = FXCollections.observableArrayList();
ObservableList<String> temp6 = FXCollections.observableArrayList();
@FXML
void Show_result(ActionEvent event) {
    String stmt1="select * from mca_semester_iii_advance_datastructure
where reg_no='"+s_marks_reg.getText()+"'";
    String stmt2="select * from mca_semester_iii_ethical_hacking where
reg_no='"+s_marks_reg.getText()+"'";
    String stmt3="select * from mca_semester_iii_java_programming where
reg_no='"+s_marks_reg.getText()+"'";
    String stmt4="select * from mca_semester_iii_machine_learning where
reg_no='"+s_marks_reg.getText()+"'";
    String stmt5="select * from mca_semester_iii_networking where
reg_no='"+s_marks_reg.getText()+"'";
    String stmt6="select * from mca_semester_iii_social_relevant_project
where reg_no='"+s_marks_reg.getText()+"'";

    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
        DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_marks",
"postgres", "nevada");
        stmt = c.createStatement();

```



```

        ResultSet rs = stmt.executeQuery(stmt1);
        while ( rs.next() ){
            String f_regno=rs.getString("reg_no");
            String f_assel=rs.getString("asses_i");
            String f_asse2=rs.getString("asses_ii");
            String f_endsem=rs.getString("end_sem");
            String f_total=rs.getString("total");
            String f_grade=rs.getString("grade");
            temp1.addAll("01", "Advance_datastructure", f_assel,
f_asse2, f_endsem, f_total, f_grade);
            System.out.println();
        }
    }
    catch (Exception e){

    }
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_marks",
"postgres", "nevada");
        stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery(stmt2);
        while ( rs.next() ){
            String f_regno=rs.getString("reg_no");
            String f_assel=rs.getString("asses_i");
            String f_asse2=rs.getString("asses_ii");
            String f_endsem=rs.getString("end_sem");
            String f_total=rs.getString("total");
            String f_grade=rs.getString("grade");
            temp2.addAll("02", "Ethical_Hacking", f_assel,
f_asse2, f_endsem, f_total, f_grade);
        }
    }
    catch (Exception e){

    }
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_marks",
"postgres", "nevada");
        stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery(stmt3);
        while ( rs.next() ){
            String f_regno=rs.getString("reg_no");
            String f_assel=rs.getString("asses_i");
            String f_asse2=rs.getString("asses_ii");
            String f_endsem=rs.getString("end_sem");
            String f_total=rs.getString("total");
            String f_grade=rs.getString("grade");
            temp3.addAll("03", "JAVA_Programming", f_assel,
f_asse2, f_endsem, f_total, f_grade);
        }
    }
    catch (Exception e){

```

```

    }
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_marks",
"postgres", "nevada");
        stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery(stmt4);
        while ( rs.next() ){
            String f_regno=rs.getString("reg_no");
            String f_asse1=rs.getString("asses_i");
            String f_asse2=rs.getString("asses_ii");
            String f_endsem=rs.getString("end_sem");
            String f_total=rs.getString("total");
            String f_grade=rs.getString("grade");
            temp4.addAll("04", "Machine_Learning", f_asse1,
f_asse2, f_endsem, f_total, f_grade);
        }
    }
    catch (Exception e){

    }
    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_marks",
"postgres", "nevada");
        stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery(stmt5);
        while ( rs.next() ){
            String f_regno=rs.getString("reg_no");
            String f_asse1=rs.getString("asses_i");
            String f_asse2=rs.getString("asses_ii");
            String f_endsem=rs.getString("end_sem");
            String f_total=rs.getString("total");
            String f_grade=rs.getString("grade");
            temp5.addAll("05", "Networking", f_asse1,
f_asse2, f_endsem, f_total, f_grade);
        }
    }
    catch (Exception e){

    }

    try {
        Connection c = null;
        Statement stmt = null;
        Class.forName("org.postgresql.Driver");
        c =
DriverManager.getConnection("jdbc:postgresql://localhost:5432/sems_marks",
"postgres", "nevada");
        stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery(stmt6);
        while ( rs.next() ){
            String f_regno=rs.getString("reg_no");
            String f_asse1=rs.getString("asses_i");
            String f_asse2=rs.getString("asses_ii");

```

```

        String f_endsem=rs.getString("end_sem");
        String f_total=rs.getString("total");
        String f_grade=rs.getString("grade");
        temp6.addAll("06","Social_relevant_project",f_asse1,
f_asse2,f_endsem,f_total,f_grade);
    }

    initcols3();
    loaddata3();
    loaddata4();
    loaddata5();
    loaddata6();
    loaddata7();
    loaddata8();
    cgpa();

}
catch (Exception e){
    System.out.println(e);
}

}

private void initcols3(){
    m_serial.setCellValueFactory(new PropertyValueFactory<>("sno"));
    smarks_name.setCellValueFactory(new
PropertyValueFactory<>("sub_name"));
    smarks asses_1.setCellValueFactory(new
PropertyValueFactory<>("asses1"));
    smarks asses_2.setCellValueFactory(new
PropertyValueFactory<>("asses2"));
    smarks_endsem.setCellValueFactory(new
PropertyValueFactory<>("endsem"));
    smarks_total.setCellValueFactory(new PropertyValueFactory<>("total"));
    smarks_grade.setCellValueFactory(new PropertyValueFactory<>("grade"));

}
ObservableList<sample.table.result> data_table3 =
FXCollections.observableArrayList();
private void loaddata3(){
    data_table3.add(new
sample.table.result(temp1.get(0),temp1.get(1),temp1.get(2),temp1.get(3),tem
p1.get(4),temp1.get(5),temp1.get(6)));
    s_marks_table2.setItems(data_table3);
}

private void loaddata4(){
    System.out.println("9");
    data_table3.removeAll();
    data_table3.add(new
sample.table.result(temp2.get(0),temp2.get(1),temp2.get(2),temp2.get(3),tem
p2.get(4),temp2.get(5),temp2.get(6)));
    s_marks_table2.setItems(data_table3);
}

private void loaddata5(){
    data_table3.removeAll();
    data_table3.add(new
sample.table.result(temp3.get(0),temp3.get(1),temp3.get(2),temp3.get(3),tem
p3.get(4),temp3.get(5),temp3.get(6)));
    s_marks_table2.setItems(data_table3);
}

```

```

private void loaddata6() {
    data_table3.removeAll();
    data_table3.add(new
sample.table.result(temp4.get(0), temp4.get(1), temp4.get(2), temp4.get(3), tem
p4.get(4), temp4.get(5), temp4.get(6)));
    s_marks_table2.setItems(data_table3);
}
private void loaddata7() {
    data_table3.removeAll();
    data_table3.add(new
sample.table.result(temp5.get(0), temp5.get(1), temp5.get(2), temp5.get(3), tem
p5.get(4), temp5.get(5), temp5.get(6)));
    s_marks_table2.setItems(data_table3);
}
private void loaddata8() {
    data_table3.removeAll();
    data_table3.add(new
sample.table.result(temp6.get(0), temp6.get(1), temp6.get(2), temp6.get(3), tem
p6.get(4), temp6.get(5), temp6.get(6)));
    s_marks_table2.setItems(data_table3);
}

private int fetchgrade(String j){
    if (j.equals("O")){
        return 10;
    }
    else if (j.equals("A+")){
        return 9;
    }
    else if (j.equals("A")){
        return 8;
    }
    else if(j.equals("B+")){
        return 7;
    }
    else if (j.equals("B")){
        return 6;
    }
    else{
        return 0;
    }
}

private void cgpa() {
    int a1=fetchgrade(temp1.get(6));
    int a2=fetchgrade(temp2.get(6));
    int a3=fetchgrade(temp3.get(6));
    int a4=fetchgrade(temp4.get(6));
    int a5=fetchgrade(temp5.get(6));
    int a6=fetchgrade(temp6.get(6));

    int total=(a1*3+a2*3+a3*3+a4*3+a5*3+a6*3)/18;
    cgpa_cal.setText(String.valueOf(total));
}

@FXML
void print_marks(ActionEvent event) throws FileNotFoundException {
    String t =s_marks_reg.getText();

```

```

        String u = s_marks_stream.getText();
        String v = s_marks_semester.getValue();
        String w=s_name_falg.getText();
        String marksfile = "C:/Users/samun/Desktop/sems/Rough
doc/marksheet.pdf";
        PdfDocument pdfDoc = new PdfDocument(new PdfWriter(marksfile));
        Document doc = new Document(pdfDoc,PageSize.A4.rotate());

        doc.add(new Paragraph("-----
"));
        doc.add(new Paragraph("Reg.No  :
"+t).setBold().setFontColor(Color.BLUE));
        doc.add(new Paragraph("Name   : "+w).setBold());
        doc.add(new Paragraph("Stream : "+u).setBold());
        doc.add(new Paragraph("Semester : "+v).setBold());
        doc.add(new Paragraph("-----
"));
        doc.add(new Paragraph(""));
        doc.add(new Paragraph(""));
        doc.add(new Paragraph(""));
        doc.add(new Paragraph("Marks:
").setBold().setTextAlignment(TextAlignment.CENTER).setFontSize(18));
        doc.add(new Paragraph(""));
        Table table = new Table(7);
        table.setFontSize(10);
        table.setTextAlignment(TextAlignment.CENTER);
        table.addCell(new Cell().add("s.no").setBold());
        table.addCell(new Cell().add("Subjects").setBold());
        table.addCell(new Cell().add("Asses I").setBold());
        table.addCell(new Cell().add("Asses II").setBold());
        table.addCell(new Cell().add("End Sem").setBold());
        table.addCell(new Cell().add("Total").setBold());
        table.addCell(new Cell().add("Grades").setBold());

        table.addCell(new Cell().add(temp1.get(0)));
        table.addCell(new Cell().add(temp1.get(1)));
        table.addCell(new Cell().add(temp1.get(2)));
        table.addCell(new Cell().add(temp1.get(3)));
        table.addCell(new Cell().add(temp1.get(4)));
        table.addCell(new Cell().add(temp1.get(5)));
        table.addCell(new Cell().add(temp1.get(6)));

        table.addCell(new Cell().add(temp2.get(0)));
        table.addCell(new Cell().add(temp2.get(1)));
        table.addCell(new Cell().add(temp2.get(2)));
        table.addCell(new Cell().add(temp2.get(3)));
        table.addCell(new Cell().add(temp2.get(4)));
        table.addCell(new Cell().add(temp2.get(5)));
        table.addCell(new Cell().add(temp2.get(6)));

        table.addCell(new Cell().add(temp3.get(0)));
        table.addCell(new Cell().add(temp3.get(1)));
        table.addCell(new Cell().add(temp3.get(2)));
        table.addCell(new Cell().add(temp3.get(3)));
        table.addCell(new Cell().add(temp3.get(4)));
        table.addCell(new Cell().add(temp3.get(5)));
        table.addCell(new Cell().add(temp3.get(6)));

        table.addCell(new Cell().add(temp4.get(0)));
        table.addCell(new Cell().add(temp4.get(1)));
        table.addCell(new Cell().add(temp4.get(2)));

```

```

table.addCell(new Cell().add(temp4.get(3)));
table.addCell(new Cell().add(temp4.get(4)));
table.addCell(new Cell().add(temp4.get(5)));
table.addCell(new Cell().add(temp4.get(6)));

table.addCell(new Cell().add(temp5.get(0)));
table.addCell(new Cell().add(temp5.get(1)));
table.addCell(new Cell().add(temp5.get(2)));
table.addCell(new Cell().add(temp5.get(3)));
table.addCell(new Cell().add(temp5.get(4)));
table.addCell(new Cell().add(temp5.get(5)));
table.addCell(new Cell().add(temp5.get(6)));

table.addCell(new Cell().add(temp6.get(0)));
table.addCell(new Cell().add(temp6.get(1)));
table.addCell(new Cell().add(temp6.get(2)));
table.addCell(new Cell().add(temp6.get(3)));
table.addCell(new Cell().add(temp6.get(4)));
table.addCell(new Cell().add(temp6.get(5)));
table.addCell(new Cell().add(temp6.get(6)));

doc.add(table);
doc.add(new Paragraph("-----
"));
doc.add(new Paragraph("CGPA :
"+cgpa_cal.getText()).setBold().setFontColor(Color.RED));
doc.add(new Paragraph("-----
"));

doc.close();

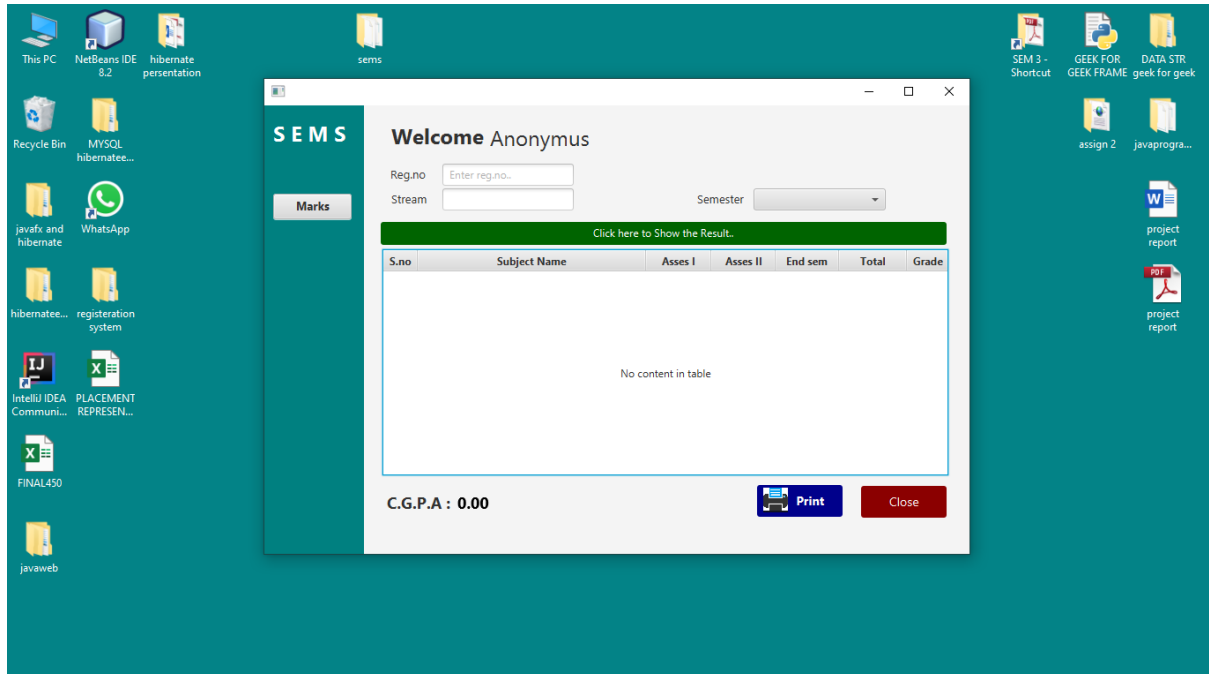
File marks_data=new File("C:/Users/samun/Desktop/sems/Rough
doc/marksheet.pdf");
try {
    Desktop.getDesktop().open(marks_data);
} catch (IOException e) {
    e.printStackTrace();
}

}

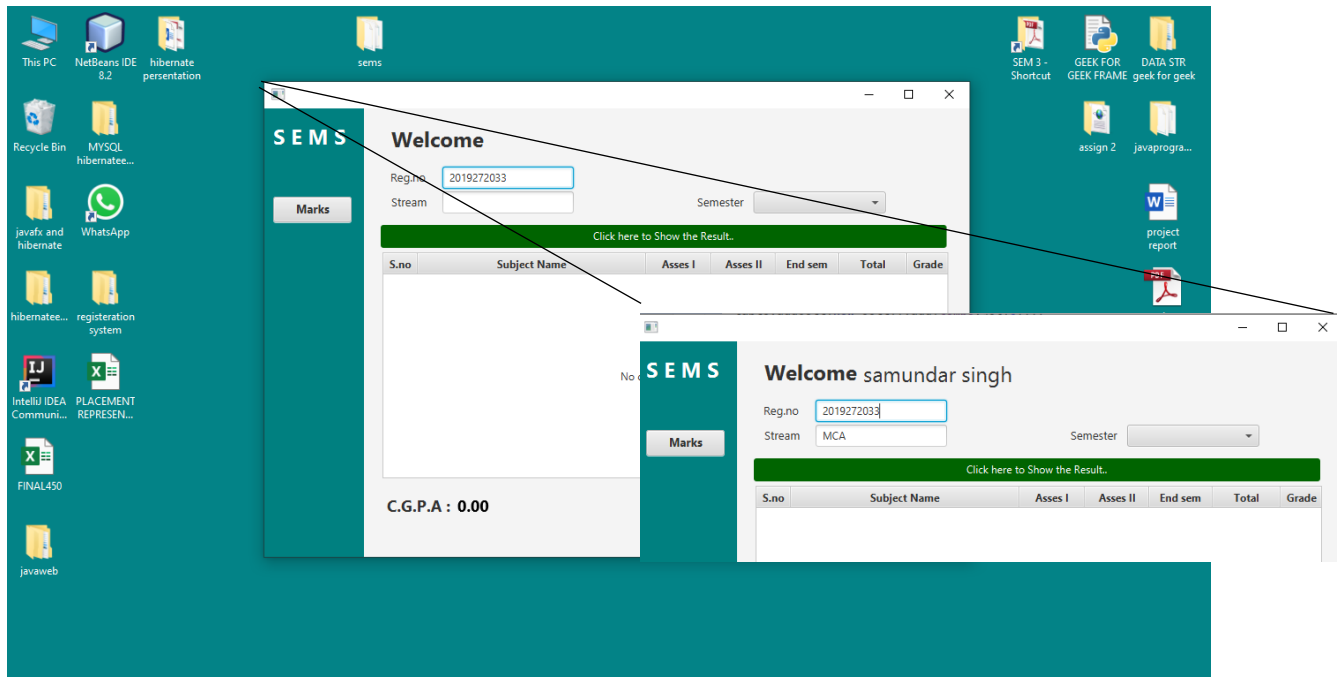
```

# Screenshot

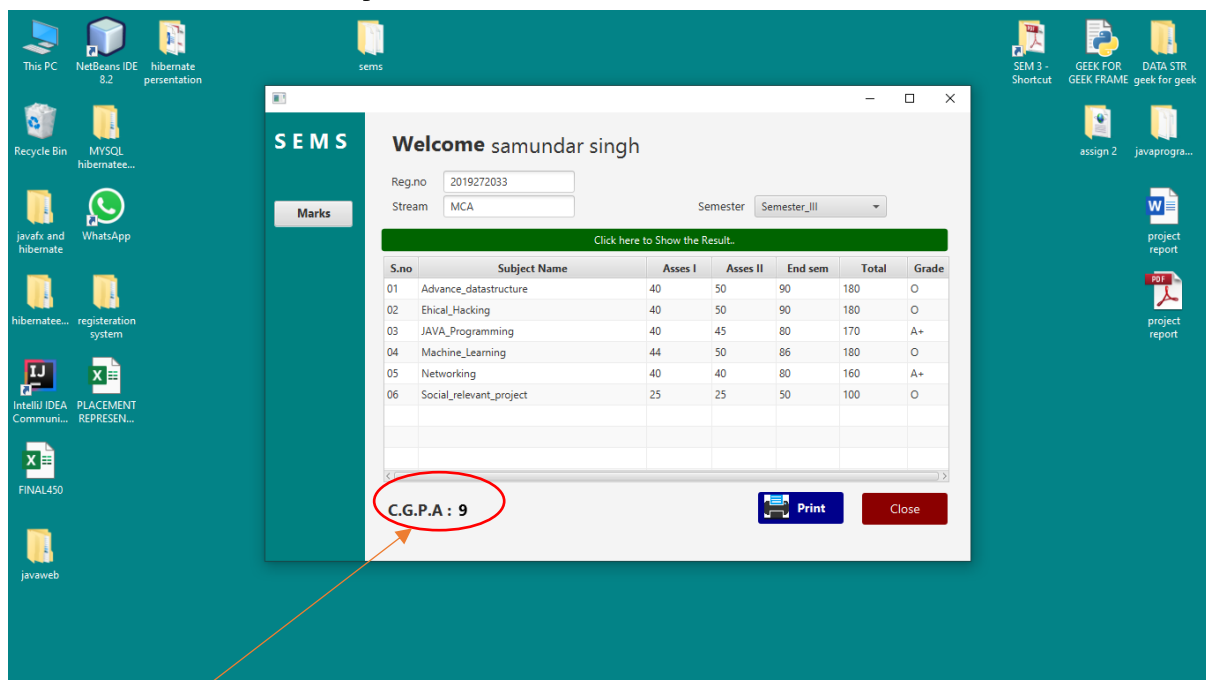
## Module View:



By Entering the reg.no it will automatically fetch the stream and name of student.



Select the semester and click to show the result button to fetch your Marks:



The CGPA is automatically Calculated.

We can also download the Marksheet.by clicking the print button

