A. Introduction

Overview

The University Grade Encoding System is a software application built using Java's graphical user interface (GUI). This system simplifies the process of recording and managing student grades, targeting university-level administration and faculty. By automating the calculation of grades and their respective classifications, it reduces human errors and speeds up the grading process.

Key Objectives:

The system is designed with the following goals:

- **Ease of Use:** Provide a user-friendly interface for professors and administrative staff to input, calculate, and store grades.
- **Efficiency**: To streamline the process of entering and managing large volumes of student records.
- **Consistency**: To maintain consistent grading standards based on pre-set thresholds for grades like "Excellent" or "Passed."

Primary Functions:

The system performs several core tasks:

- **Input Grades:** Allow users to enter grades for different grading periods (Prelim, Midterm, Final).
- **Calculate Grades:** The system automatically calculates the average of the grades entered and determines a grade rating (e.g., Excellent, Satisfactory, Passed).
- **Display Grades:** The user can view a table listing all students, their calculated averages, and their grade ratings.
- **Delete Grades:** The user can delete grade in the list on the table.

B. System Overview

Key Features of the System:

The University Grade Encoding System is designed to streamline the process of recording, calculating, and managing student grades. Its core functionalities include:

Grade Input:

Users can input student information, including the student's name and grades for specific grading periods (e.g., Prelim, Midterm, Final). This data entry is critical as it serves as the foundation for the grade calculation process.

Grade Calculation:

Once the grades for the Prelim, Midterm, and Final exams are entered, the system automatically calculates the student's average grade. The formula used is:

$$\text{Average} = \frac{\text{Prelim Grade} + \text{Midterm Grade} + \text{Final Grade}}{3}$$

Based on the calculated average, the system assigns a grade rating using predefined thresholds (e.g., >= 90 = "Excellent", 80-89 = "Satisfactory", etc.). This ensures consistency in grading standards and reduces manual errors.

Result Display:

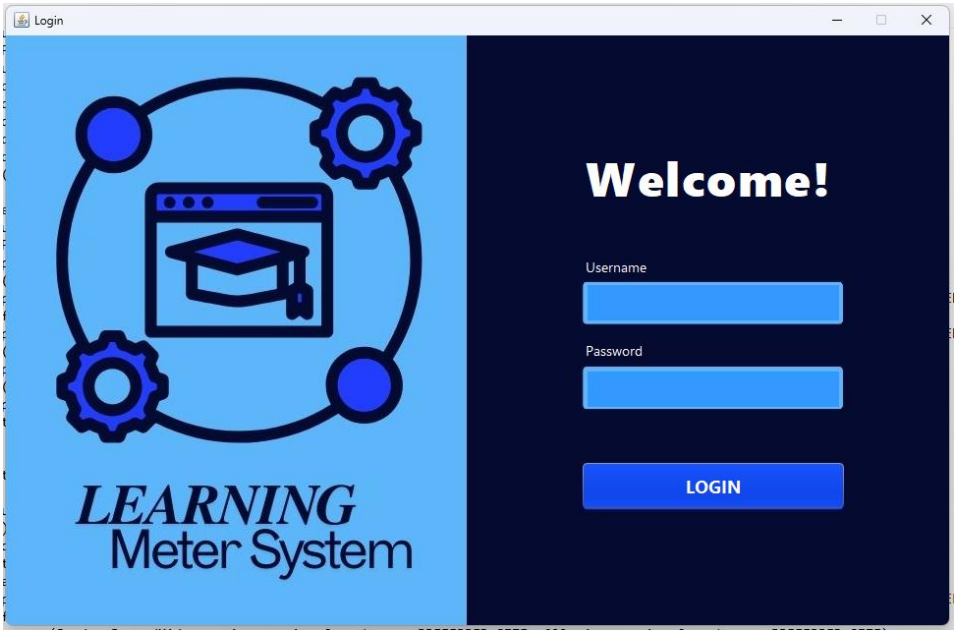
After calculating the grades, the system displays the student's name, their average grade, and the assigned rating. This output can be viewed in a tabular format, where all students' records are listed together. This allows users to review, update, or manage multiple student records simultaneously.

Together, these components ensure a smooth and efficient process for managing student grades, from input to calculation and final display.

User Interface:

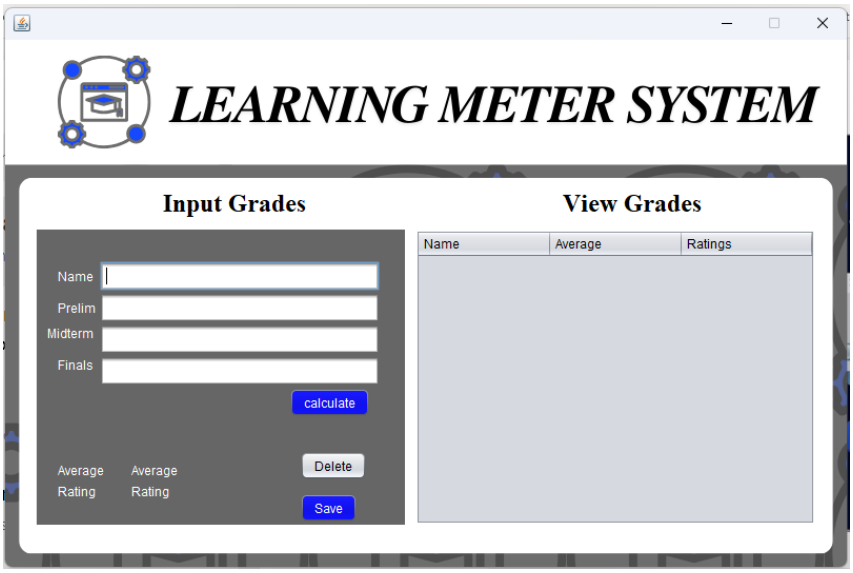
The system is designed to be intuitive and easy to use for university faculty, such as professors or administrative staff. The user interface (UI) is divided into several key screens to guide users through the various tasks:

- **Login Screen**



Users begin by logging into the system using a username and password. Only authorized personnel (such as professors or staff) have access to the system, ensuring security and privacy of student records.

- **Home Screen:**



After a successful login, the user is directed to the Home Screen, where they are presented with several options to navigate the system:

1. **Input Grades:** Users can input grades for new or existing students.
2. **View Grades:** Allows the user to access all saved student records, displaying them in a list format with relevant information such as student names, averages, and ratings.

Delete Grades:

Allows the user to delete the selected grade in the table.

Typical Tasks Users Can Perform:

- **Add New Student Records:**

  This function allows professors or admin staff to input a new student's name and their grades for Prelim, Midterm, and Final exams.

  Step-by-Step Process:

  Access the Input Form: From the main menu, the user selects the "Input Grades" section.

  Enter Student Information: A form appears with fields for:

  A. **Student Name**: The name of the student.
  B. **Prelim Grade**: The grade for the prelim period.
  C. **Midterm Grade**: The grade for the midterm period.
  D. **Final Grade**: The grade for the final period.

- **Submit the Grades**: After inputting all necessary information, the user clicks the "calculate" button. The system processes the input, calculating the average grade as well as the performance rating.

- **Result Display**: The system immediately displays the calculated average and assigned rating (e.g., Excellent, Good, Passed). The user is prompted to save the record the grade to view in table.

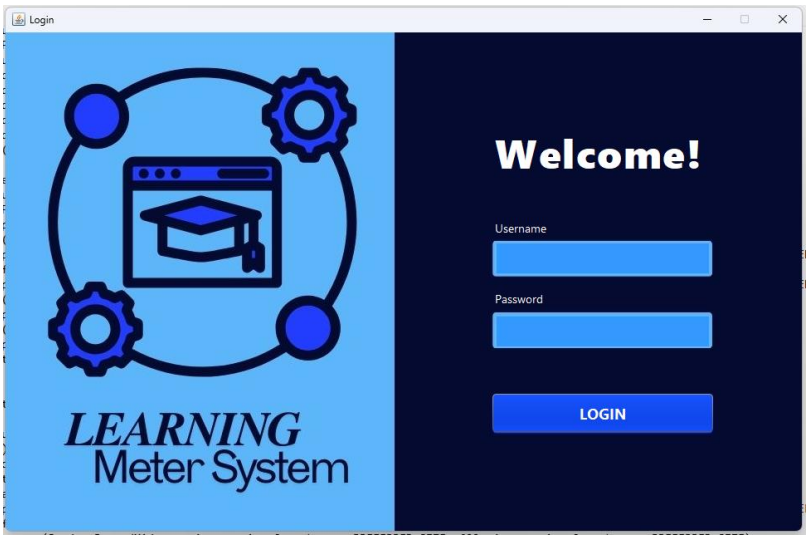
- **Viewing Student Grades**: This feature enables users to view a list of all students, along with their grades and performance ratings. The table format makes it easy to navigate and find specific records.

  Step-by-Step Process:

  1. **Open the Grade Table**: Users check the "View Grades" section from the home.
  2. **Review the Data**: The table shows all students' names, along with their calculated average and rating.
  3. Delete the data: The grade can be deleted by highlighting the row of the grade to delete thru clicking the delete button.

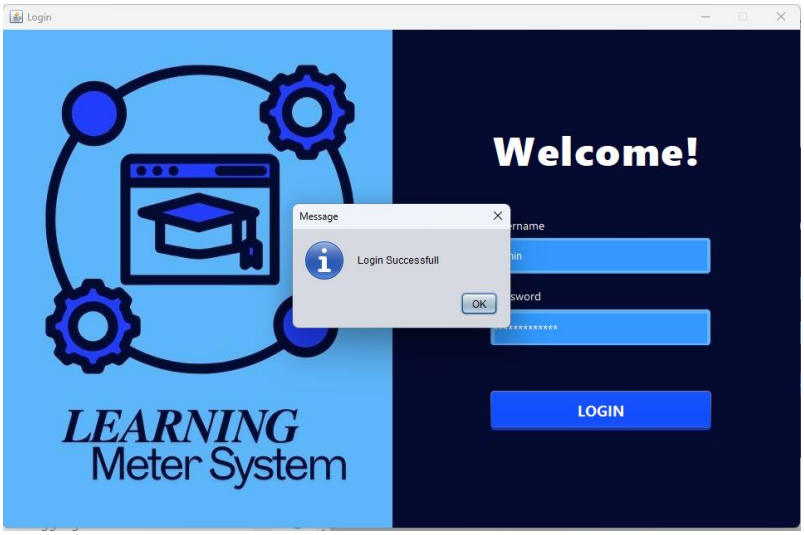
C. User Manual

Login Procedure:



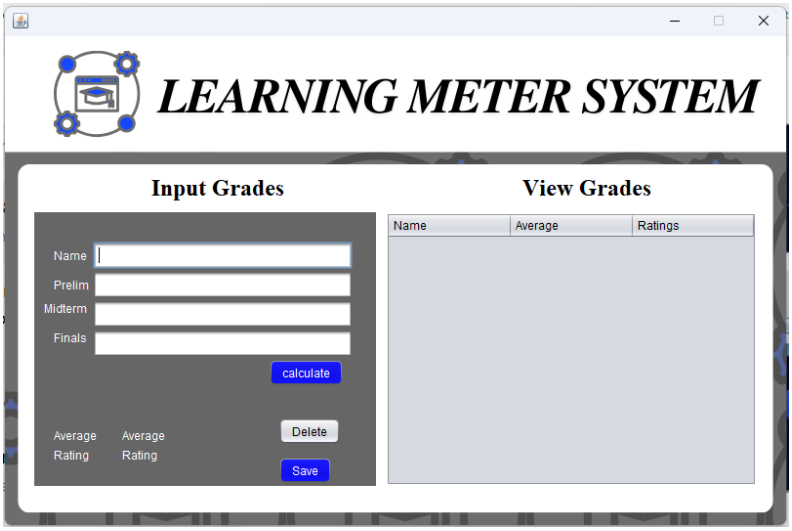
Step 1: Open the application. The login screen appears.

Step 2: Enter the username (Admin) and password (Administrator).



Step 3: Click "Login". If the credentials are correct, the system will display the Home Screen. If incorrect, an error message will appear.

Using the Home Screen:



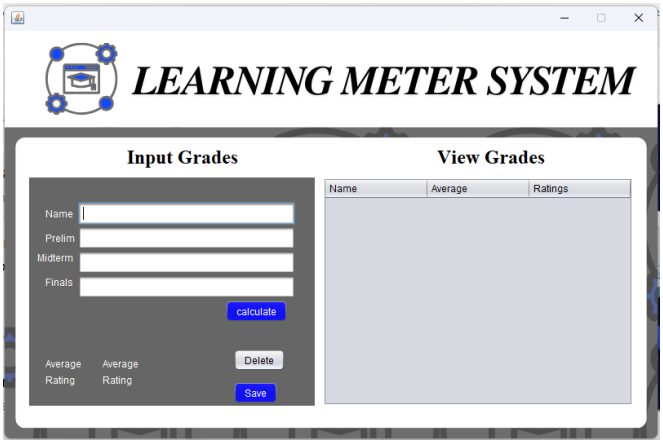
Step 1: Upon successful login, you are presented with three options:

1. Input Grades
2. View Grades

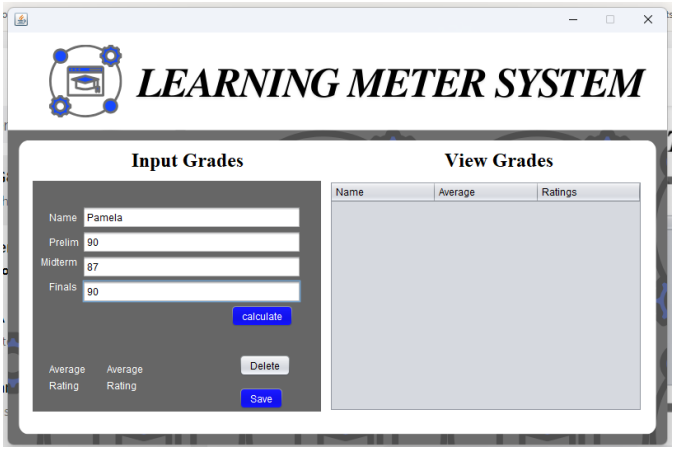
Step 2: Click the button corresponding to the task you wish to perform:

1. **Input Grades**: Opens the grade input form.
2. **View Grades**: Displays a table of saved student records.

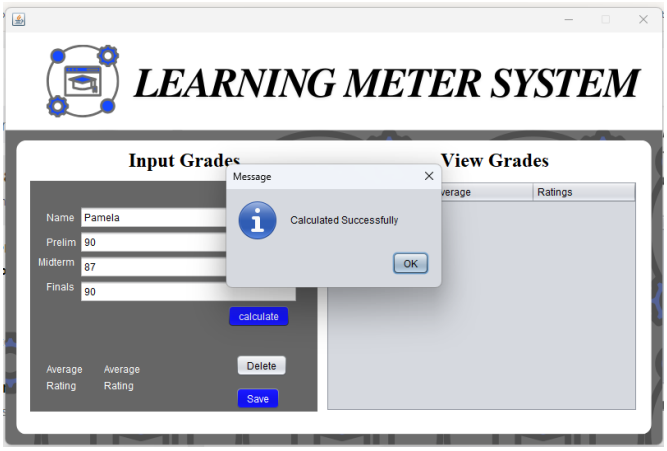
Inputting Grades:



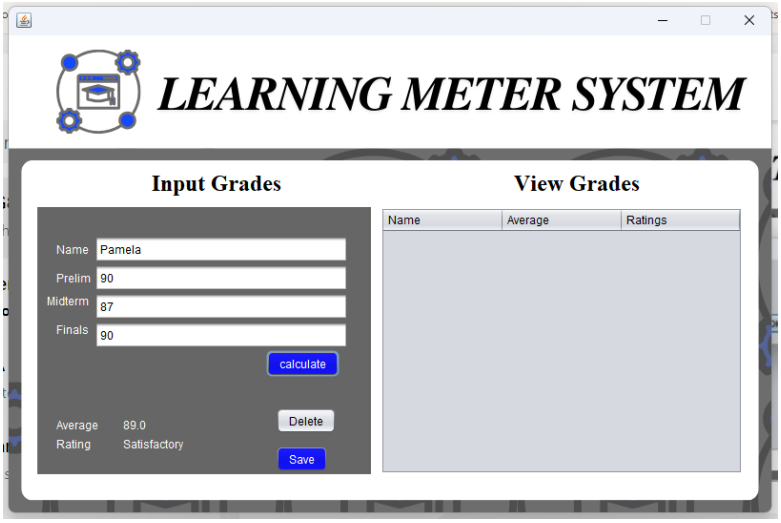
1. Enter the student's name.
2. Input grades for Prelim, Midterm, and Final.



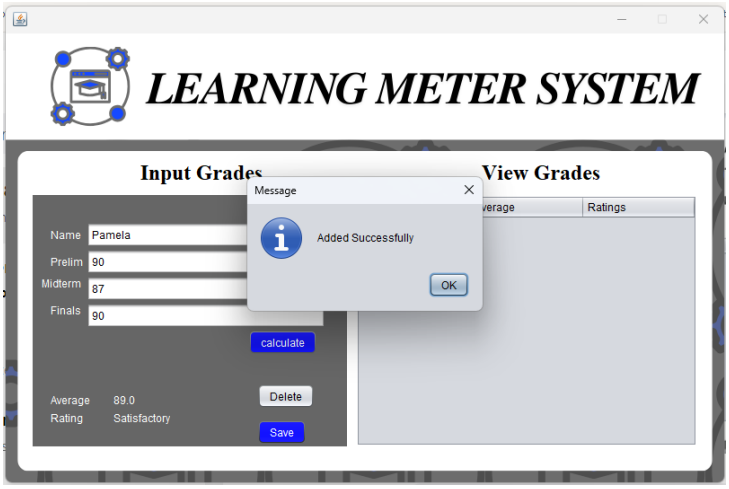
3. Click "Calculate" to calculate the average grade.



4. The message of calculation notice will pop-up

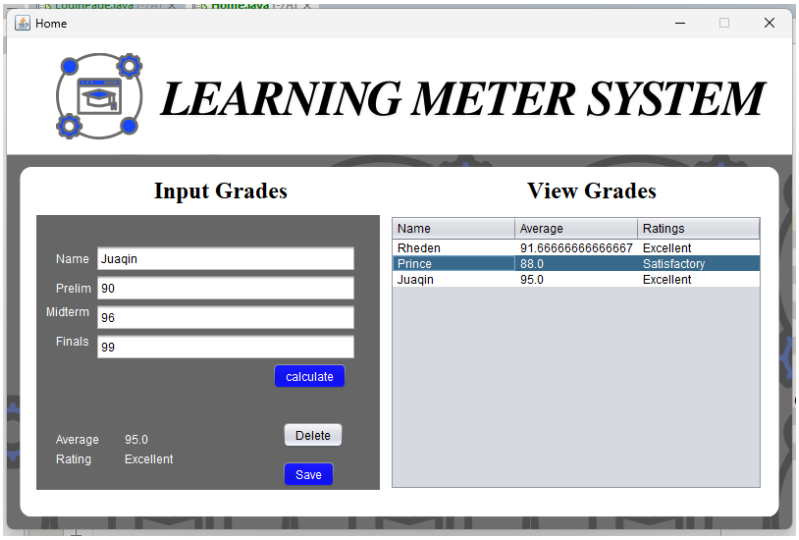


5. To save the grades, click the save button.



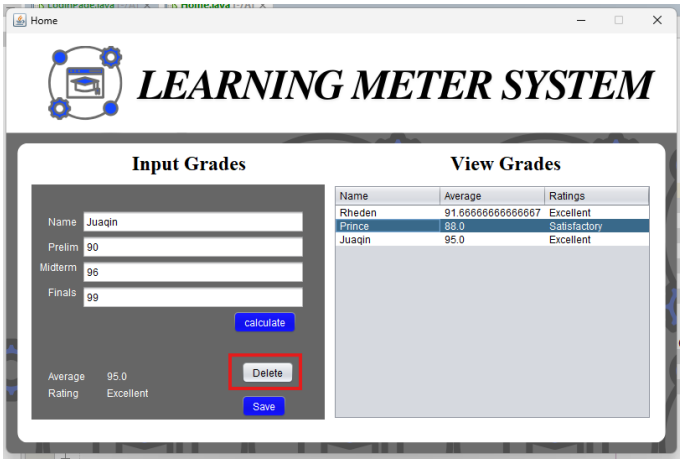
6. A message will appear after savings successfully.

Viewing Grades:



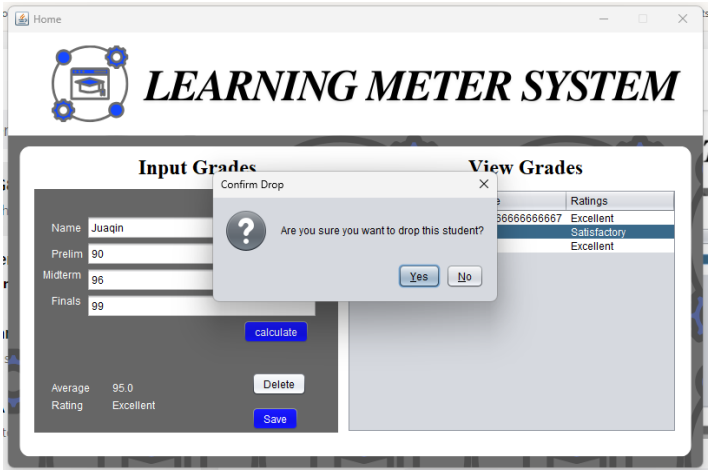
1. The system will display a table of student names, averages, and ratings.

Delete Row:

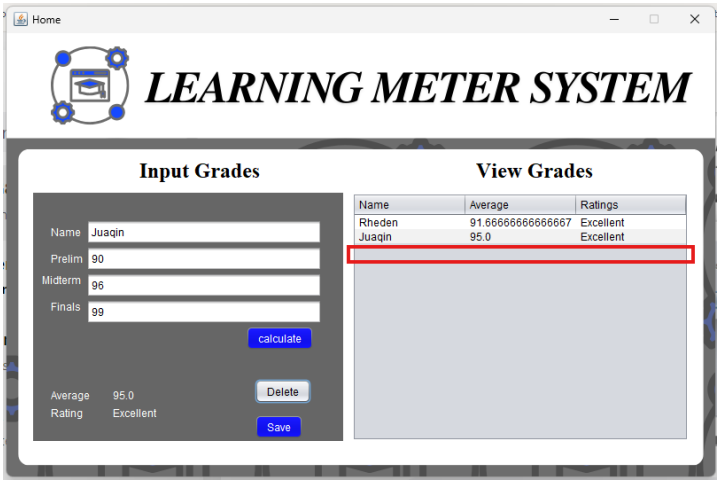


1. Select the row of the grade.

2. Click the delete button to delete.



3. A confirmation message will appear, click yes to proceed.



4. The row selected is now deleted.

D. Technical Documentation

System Architecture:

- **Login Module**: Handles user authentication.
- **Grade Management Module**: Manages grade input, calculation, and saving.
- **View Module**: Displays stored student records.

Pseudocode Overview:

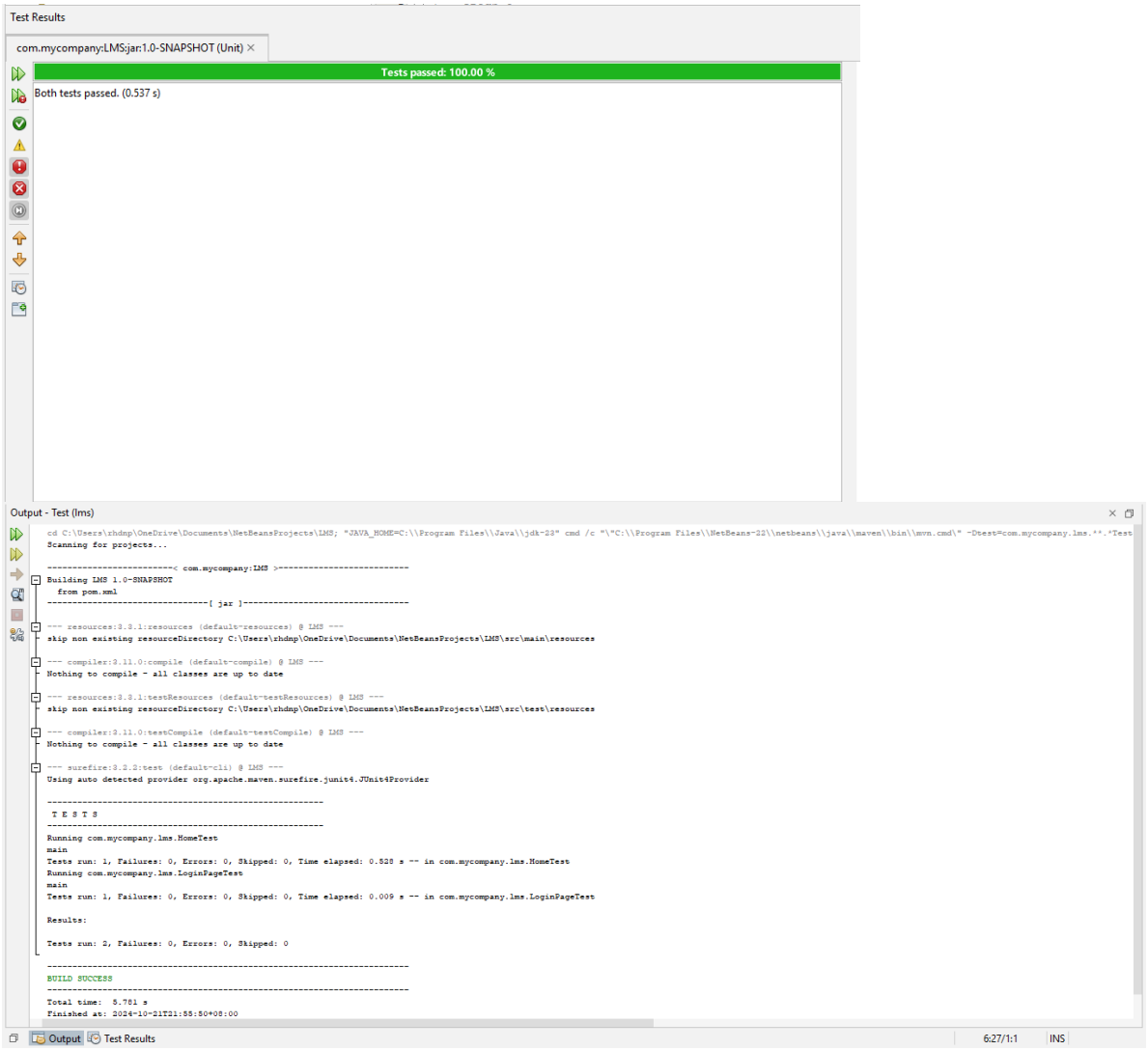
- **Login**: Prompts for credentials and verifies them.
- **Grade Input**: Collects student name and grades, calculates the average, assigns a rating, and allows saving.

- **Grade Display**: Shows stored student records in a table.

Algorithms:

- **Grade Calculation**: Adds Prelim, Midterm, and Final grades, divides by 3 for an average, and assigns ratings based on pre-set ranges.
- **Record Management**: When grades are finalized, the student's information is stored in a table.

E. Testing and Validation



With the integrated functionalities of the IDE NetBeans, we tested our project seamlessly. The results shows that the line of codes was written correctly as it passed after some debugs and attempts to clean the code. The program has passed the test as it shown in the figure above.

F. Known Issues and Limitations

Bugs:

- The decimal count in average part of the grade isn't well controlled leading to horrific and long decimals counts.

Limitations:

- Only one user role is supported (Administrator).
- The system is not scalable for handling thousands of records efficiently without optimization.

Recommendations:

- Add multiple user roles (admin, professor, etc.).
- Implement grade import functionality from files for easier data entry.

G. Appendices

Pseudocode:

Full pseudocode is provided for login, grade input, calculation, and viewing operations.

Sample Input and Output:

Example 1 (High-Performance "**Excellent**" Case):

- **Input**:
    - Student: Eminem
    - Prelim Grade: 95
    - Midterm Grade: 92
    - Final Grade: 96
- **Output**:
    - Average: (95 + 92 + 96) / 3 = 94.33
    - Rating: **Excellent**

Example 2 (Low "**Satisfactory**" Case)**:**

- **Input:**
    - Student: Michael Joseph Jackson
    - Prelim Grade: 80
    - Midterm Grade: 82
    - Final Grade: 81
- **Output:**
    - Average: (80 + 82 + 81) / 3 = 81
    - Rating: **Satisfactory**

Example 3 (Borderline "**Good**" Case):

- **Input:**
    - Student: Shawn Corey Carter
    - Prelim Grade: 70
    - Midterm Grade: 71
    - Final Grade: 69
- **Output:**
    - Average: (70 + 71 + 69) / 3 = 70
    - Rating: **Good**

Example 4 (Low "**Passed**" Case)**:**

- **Input:**
    - Student: Beyoncé Giselle Knowles-Carter
    - Prelim Grade: 60
    - Midterm Grade: 61
    - Final Grade: 59
- **Output:**
    - Average: (60 + 61 + 59) / 3 = 60
    - Rating: **Passed**

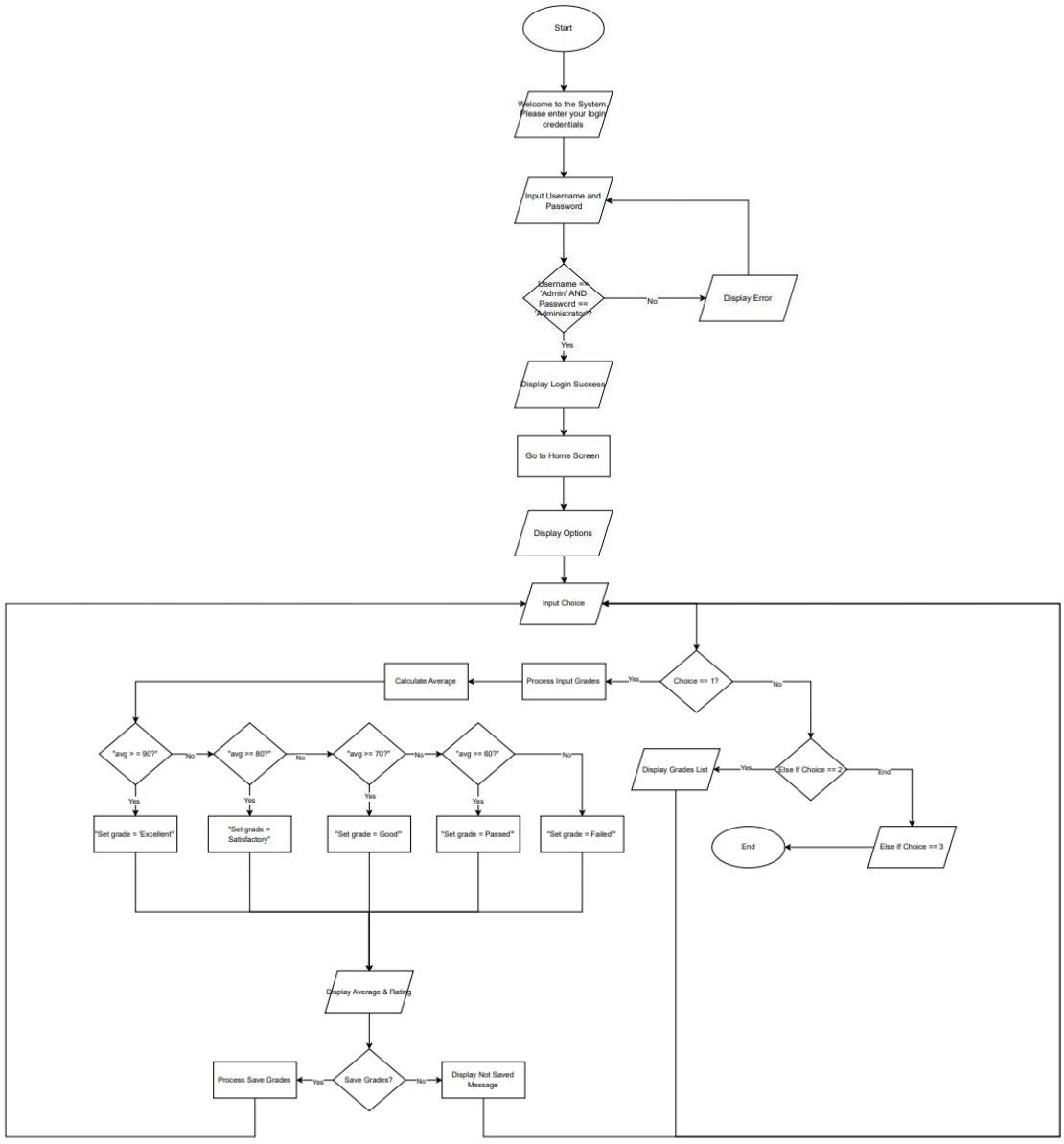
Example 5 (Failing Case):

- **Input**:
    - Student: P. Diddy

- Prelim Grade: 45
- Midterm Grade: 50
- Final Grade: 48

- **Output**:
  - Average: (45 + 50 + 48) / 3 = 47.67
  - Rating: **Failed**

Flowchart:



INPUT (Code):

```
/*  Group 8
    October 21, 2024
    2CpE-A
*/

package com.mycompany.lms;
import javax.swing.JOptionPane;

public class LoginPage extends javax.swing.JFrame {

    /**
     * Creates new form LoginFrame
     */
    public LoginPage() {
        initComponents();
    }

    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        Header = new javax.swing.JLabel();
        user_Label = new javax.swing.JLabel();
        pass_Label = new javax.swing.JLabel();
        Username = new javax.swing.JTextField();
        Password = new javax.swing.JPasswordField();
        login = new javax.swing.JButton();
        Background = new javax.swing.JLabel();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setTitle("Login");
        setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
        setResizable(false);
        getContentPane().setLayout(new
org.netbeans.lib.awtextra.AbsoluteLayout());

        Header.setBackground(new java.awt.Color(0, 0, 153));
        Header.setFont(new java.awt.Font("Segoe UI Black", 1, 48)); // NOI18N
        Header.setForeground(new java.awt.Color(255, 255, 255));
        Header.setText("Welcome!");
        getContentPane().add(Header, new
org.netbeans.lib.awtextra.AbsoluteConstraints(590, 110, 290, 70));
```

```
        user_Label.setFont(new java.awt.Font("Segoe UI Emoji", 0, 14)); // NOI18N
        user_Label.setForeground(new java.awt.Color(255, 255, 255));
        user_Label.setText("Username");
        getContentPane().add(user_Label, new
org.netbeans.lib.awtextra.AbsoluteConstraints(590, 230, -1, -1));

        pass_Label.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        pass_Label.setForeground(new java.awt.Color(255, 255, 255));
        pass_Label.setText("Password");
        getContentPane().add(pass_Label, new
org.netbeans.lib.awtextra.AbsoluteConstraints(590, 310, -1, -1));

        Username.setBackground(new java.awt.Color(51, 153, 255));
        Username.setForeground(new java.awt.Color(255, 255, 255));
        Username.setBorder(null);
        getContentPane().add(Username, new
org.netbeans.lib.awtextra.AbsoluteConstraints(590, 252, 260, 40));

        Password.setBackground(new java.awt.Color(51, 153, 255));
        Password.setFont(new java.awt.Font("Segoe UI", 0, 14)); // NOI18N
        Password.setForeground(new java.awt.Color(255, 255, 255));
        Password.setBorder(null);
        getContentPane().add(Password, new
org.netbeans.lib.awtextra.AbsoluteConstraints(590, 340, 260, 40));

        login.setBackground(new java.awt.Color(0, 51, 204));
        login.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N
        login.setForeground(new java.awt.Color(255, 255, 255));
        login.setText("LOGIN");
        login.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                loginActionPerformed(evt);
            }
        });
        getContentPane().add(login, new
org.netbeans.lib.awtextra.AbsoluteConstraints(585, 433, 270, 50));

        Background.setBackground(new java.awt.Color(102, 153, 255));
        Background.setForeground(new java.awt.Color(255, 255, 255));
        Background.setIcon(new
javax.swing.ImageIcon("C:\\Users\\rhdnp\\Downloads\\login.jpg")); //
NOI18N
        getContentPane().add(Background, new
org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, -1, -1));

        pack();
```

```
        setLocationRelativeTo(null);
    }// </editor-fold>//GEN-END:initComponents

    private void loginActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_loginActionPerformed
        // TODO add your handling code here:
        //default values of login
        String admin = "Admin";
        String pssword = "Administrator";

        // variable for login entry
        String user = Username.getText();
        String pass = Password.getText();

        if(user.equals(admin) && pass.equals(pssword)){
            JOptionPane.showMessageDialog(null, "Login Successfull");
            Home home = new Home();
            setVisible(false);
            home.setVisible(true);}
        else{
            JOptionPane.showMessageDialog(null, "Incorrect Credentials", "Error",
JOptionPane.ERROR_MESSAGE);
        }

    }//GEN-LAST:event_loginActionPerformed

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting
code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with the
default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
```

```
java.util.logging.Logger.getLogger(LoginPage.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(LoginPage.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(LoginPage.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(LoginPage.class.getName()).log(java.util.l
ogging.Level.SEVERE, null, ex);
        }
        //</editor-fold>
        //</editor-fold>
        //</editor-fold>
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new LoginPage().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JLabel Background;
    private javax.swing.JLabel Header;
    private javax.swing.JPasswordField Password;
    private javax.swing.JTextField Username;
    private javax.swing.JButton login;
    private javax.swing.JLabel pass_Label;
    private javax.swing.JLabel user_Label;
    // End of variables declaration//GEN-END:variables
}
/*  Group 8
    October 21, 2024
    2CpE-A
*/

package com.mycompany.lms;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
```

```
public class Home extends javax.swing.JFrame {

    public Home() {
        initComponents();
    }
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        jPanel1 = new javax.swing.JPanel();
        jPanel5 = new javax.swing.JPanel();
        jLabel12 = new javax.swing.JLabel();
        jLabel13 = new javax.swing.JLabel();
        Name = new javax.swing.JTextField();
        M_grades = new javax.swing.JTextField();
        F_grades = new javax.swing.JTextField();
        jLabel8 = new javax.swing.JLabel();
        lblaverage1 = new javax.swing.JLabel();
        jLabel9 = new javax.swing.JLabel();
        lblrating = new javax.swing.JLabel();
        calc2 = new javax.swing.JButton();
        jLabel14 = new javax.swing.JLabel();
        P_grades = new javax.swing.JTextField();
        jLabel15 = new javax.swing.JLabel();
        Save = new javax.swing.JButton();
        Delete = new javax.swing.JButton();
        jLabel11 = new javax.swing.JLabel();
        jLabel16 = new javax.swing.JLabel();
        jScrollPane1 = new javax.swing.JScrollPane();
        jTable1 = new javax.swing.JTable();
        jLabel1 = new javax.swing.JLabel();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setTitle("Home");
        setResizable(false);
        getContentPane().setLayout(new
org.netbeans.lib.awtextra.AbsoluteLayout());

        jPanel1.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

        jPanel5.setBackground(new java.awt.Color(102, 102, 102));
        jPanel5.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

        jLabel12.setForeground(new java.awt.Color(255, 255, 255));
```

```
        jLabel12.setText("Midterm");
        jPanel5.add(jLabel12, new
org.netbeans.lib.awtextra.AbsoluteConstraints(10, 90, -1, -1));

        jLabel13.setForeground(new java.awt.Color(255, 255, 255));
        jLabel13.setText("Finals");
        jPanel5.add(jLabel13, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 120, -1, -1));

        Name.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                NameActionPerformed(evt);
            }
        });
        jPanel5.add(Name, new
org.netbeans.lib.awtextra.AbsoluteConstraints(60, 30, 266, -1));
        jPanel5.add(M_grades, new
org.netbeans.lib.awtextra.AbsoluteConstraints(60, 90, 266, -1));

        F_grades.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                F_gradesActionPerformed(evt);
            }
        });
        jPanel5.add(F_grades, new
org.netbeans.lib.awtextra.AbsoluteConstraints(60, 120, 266, -1));

        jLabel8.setForeground(new java.awt.Color(255, 255, 255));
        jLabel8.setText("Average");
        jPanel5.add(jLabel8, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 220, -1, -1));

        lblaverage1.setForeground(new java.awt.Color(255, 255, 255));
        lblaverage1.setText("Average");
        jPanel5.add(lblaverage1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(90, 220, -1, -1));

        jLabel9.setForeground(new java.awt.Color(255, 255, 255));
        jLabel9.setText("Rating");
        jPanel5.add(jLabel9, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 240, -1, -1));

        lblrating.setForeground(new java.awt.Color(255, 255, 255));
        lblrating.setText("Rating");
        jPanel5.add(lblrating, new
org.netbeans.lib.awtextra.AbsoluteConstraints(90, 240, -1, -1));
```

```
        calc2.setBackground(new java.awt.Color(0, 0, 204));
        calc2.setForeground(new java.awt.Color(255, 255, 255));
        calc2.setText("calculate");
        calc2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                calc2ActionPerformed(evt);
            }
        });
        jPanel5.add(calc2, new
org.netbeans.lib.awtextra.AbsoluteConstraints(240, 150, -1, -1));

        jLabel14.setForeground(new java.awt.Color(255, 255, 255));
        jLabel14.setText("Prelim");
        jPanel5.add(jLabel14, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 60, -1, 28));

        P_grades.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                P_gradesActionPerformed(evt);
            }
        });
        jPanel5.add(P_grades, new
org.netbeans.lib.awtextra.AbsoluteConstraints(60, 60, 266, -1));

        jLabel15.setForeground(new java.awt.Color(255, 255, 255));
        jLabel15.setText("Name");
        jPanel5.add(jLabel15, new
org.netbeans.lib.awtextra.AbsoluteConstraints(20, 30, -1, 28));

        Save.setBackground(new java.awt.Color(0, 0, 204));
        Save.setForeground(new java.awt.Color(255, 255, 255));
        Save.setText("Save");
        Save.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                SaveActionPerformed(evt);
            }
        });
        jPanel5.add(Save, new
org.netbeans.lib.awtextra.AbsoluteConstraints(250, 250, -1, -1));

        Delete.setText("Delete");
        Delete.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                DeleteActionPerformed(evt);
            }
        });
```

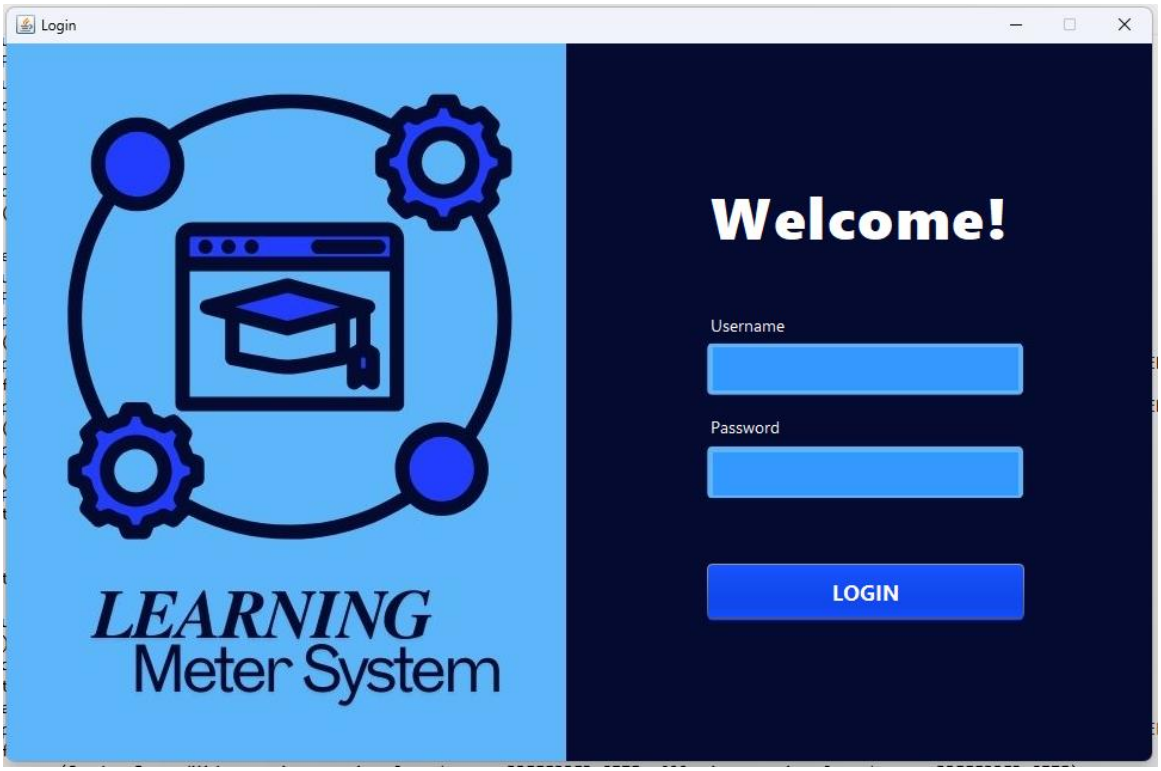
```
        jPanel5.add(Delete, new
org.netbeans.lib.awtextra.AbsoluteConstraints(250, 210, -1, -1));

        jPanel1.add(jPanel5, new
org.netbeans.lib.awtextra.AbsoluteConstraints(30, 180, 350, 280));

        jLabel11.setFont(new java.awt.Font("Serif", 1, 24)); // NOI18N
        jLabel11.setText("View Grades");
        jPanel1.add(jLabel11, new
org.netbeans.lib.awtextra.AbsoluteConstraints(530, 140, -1, 28));

        jLabel16.setFont(new java.awt.Font("Serif", 1, 24)); // NOI18N
        jLabel16.setText("Input Grades");
        jPanel1.add(jLabel16, new
org.netbeans.lib.awtextra.AbsoluteConstraints(150, 140, -1, 28));

        jTable1.setModel(new javax.swing.table.DefaultTableModel(
            new Object [][] {

            },
            new String [] {
                "Name", "Average", "Ratings"
            }
        ) {
            Class[] types = new Class [] {
                java.lang.String.class, java.lang.String.class, java.lang.String.class
            };

            public Class getColumnClass(int columnIndex) {
                return types [columnIndex];
            }
        });
        jScrollPane1.setViewportView(jTable1);

        jPanel1.add(jScrollPane1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(390, 180, 380, 280));

        jLabel1.setFont(new java.awt.Font("Showcard Gothic", 1, 24)); // NOI18N
        jLabel1.setIcon(new
javax.swing.ImageIcon("C:\\Users\\rhdnp\\Downloads\\Home.png")); //
NOI18N
        jPanel1.add(jLabel1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, -1, -1));

        getContentPane().add(jPanel1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(0, 0, 800, 500));
```

```
        pack();
        setLocationRelativeTo(null);
    }// </editor-fold>//GEN-END:initComponents

    private void F_gradesActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_F_gradesActionPerformed

    }//GEN-LAST:event_F_gradesActionPerformed

    private void NameActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_NameActionPerformed

    }//GEN-LAST:event_NameActionPerformed

    private void calc2ActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_calc2ActionPerformed

        double prelims, midterm, finals, sum, avg;
        String grade;
        prelims = Integer.parseInt(P_grades.getText());
        midterm = Integer.parseInt(M_grades.getText());
        finals = Integer.parseInt(F_grades.getText());

        sum = prelims + midterm + finals;
        avg = sum/3;




        if(avg >= 90){
            grade = "Excellent";
        } else if(avg >= 80){
            grade = "Satisfactory";
        } else if (avg >= 70){
            grade = "Good";
        } else if (avg >= 60){
            grade = "Passed";
        } else{
            grade = "Failed";
        }

        JOptionPane.showMessageDialog(null, "Calculated Successfully");
        lblaverage1.setText(String.valueOf(avg));
        lblrating.setText(grade);
    }//GEN-LAST:event_calc2ActionPerformed
```

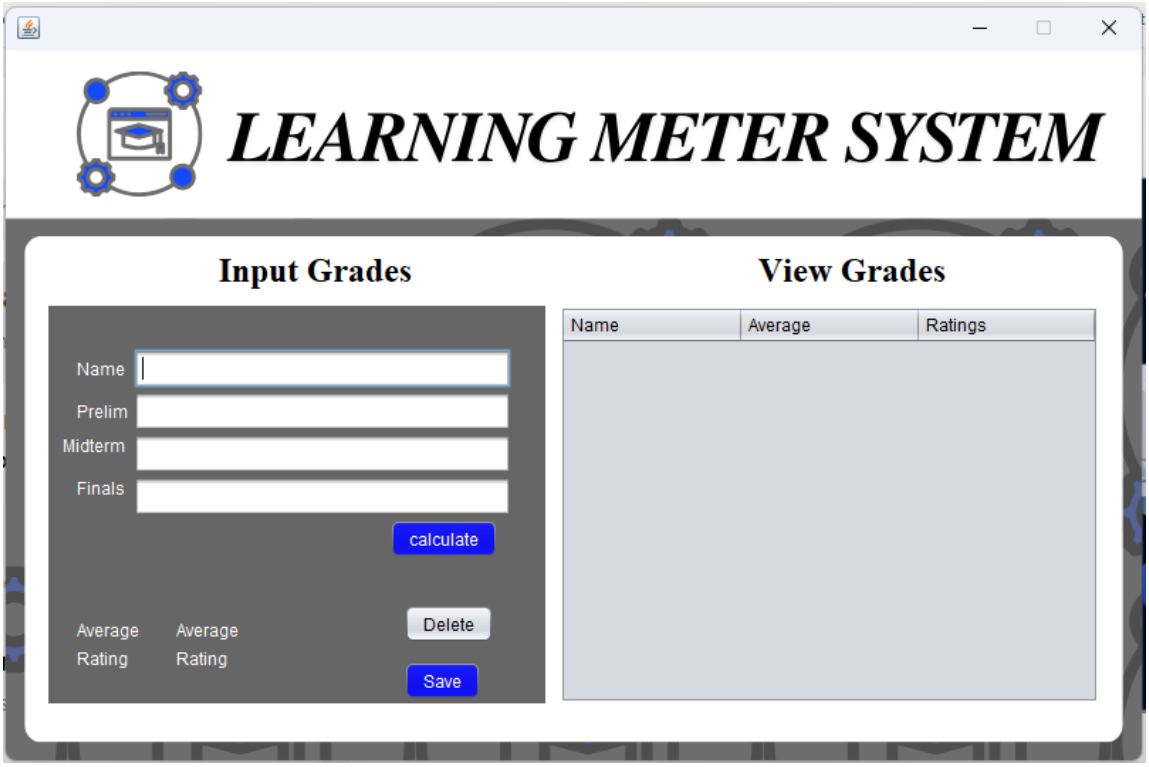
```
    private void P_gradesActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_P_gradesActionPerformed

    }//GEN-LAST:event_P_gradesActionPerformed

    private void SaveActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_SaveActionPerformed
        JOptionPane.showMessageDialog(null, "Added Successfully");
        DefaultTableModel table = (DefaultTableModel)jTable1.getModel();
        table.addRow(new Object[]{Name.getText(),
lblaverage1.getText(),lblrating.getText()});
    }//GEN-LAST:event_SaveActionPerformed

    private void DeleteActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_DeleteActionPerformed
            DefaultTableModel model = (DefaultTableModel)
jTable1.getModel();
    int selectedRow = jTable1.getSelectedRow();

    // Check if a row is selected
    if (selectedRow != -1) {
        // Confirm deletion
        int confirmation = JOptionPane.showConfirmDialog(this, "Are you sure
you want to drop this student?", "Confirm Drop",
JOptionPane.YES_NO_OPTION);

        if (confirmation == JOptionPane.YES_OPTION) {
            // Remove the selected row from the table
            model.removeRow(selectedRow);
        }
    } else {
        JOptionPane.showMessageDialog(this, "Please select a student to drop.",
"No Selection", JOptionPane.WARNING_MESSAGE);
    }

    }//GEN-LAST:event_DeleteActionPerformed

    public static void main(String args[]) {
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
```

```
        } catch (ClassNotFoundException | InstantiationException |
IllegalAccessException | javax.swing.UnsupportedLookAndFeelException ex)
{

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.loggi
ng.Level.SEVERE, null, ex);
        }
        //</editor-fold>
        //</editor-fold>

        //</editor-fold>
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(() -> {
            new Home().setVisible(true);
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton Delete;
    private javax.swing.JTextField F_grades;
    private javax.swing.JTextField M_grades;
    private javax.swing.JTextField Name;
    private javax.swing.JTextField P_grades;
    private javax.swing.JButton Save;
    private javax.swing.JButton calc2;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel11;
    private javax.swing.JLabel jLabel12;
    private javax.swing.JLabel jLabel13;
    private javax.swing.JLabel jLabel14;
    private javax.swing.JLabel jLabel15;
    private javax.swing.JLabel jLabel16;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JPanel jPanel5;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JTable jTable1;
    private javax.swing.JLabel lblaverage1;
    private javax.swing.JLabel lblrating;
    // End of variables declaration//GEN-END:variables
}
```

OUTPUT (GUI):



LOGIN PAGE



HOME PAGE

H. Members

Leader

Padilla, Rheden N.

Members

Punongbayan, Pamela

Rivera, Ivy Jean

Alipio, Christian Jericho

Cruz, Jazzen