JAVA Mini Project

On

CGPA CALCULATOR

LOVELY PROFESSIONAL UNIVERSITY

Regg. No	Name	Roll No	Section
12100514	N. Vamsi	RK21RXB65	K21RX
12109224	B. Vignesh Reddy	RK21RXB66	K21RX
12111646	Shaik Abdul Salam	RK21RXA31	K21RX

Submitted to:

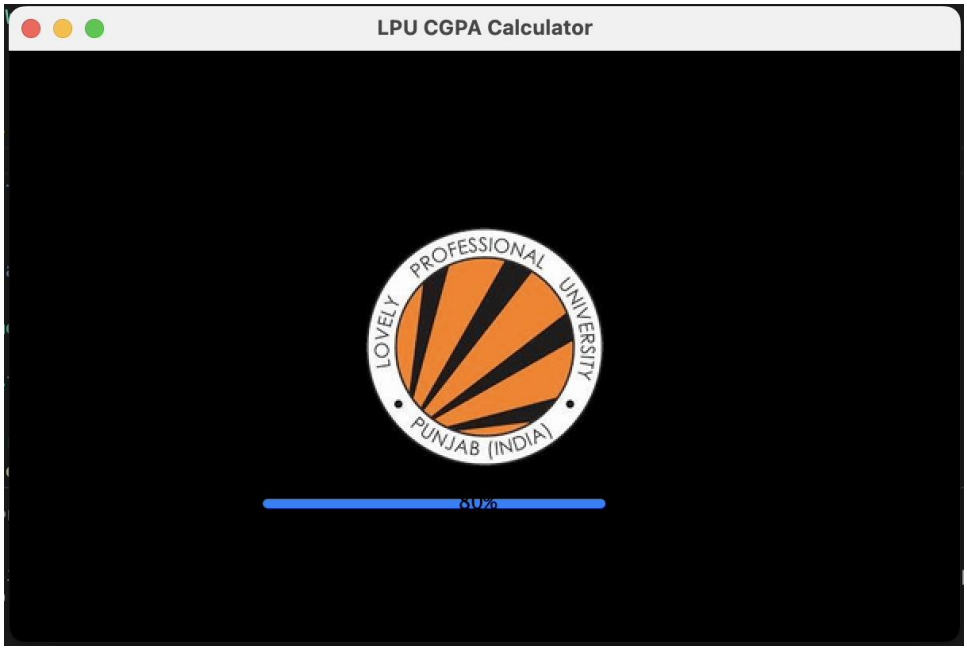
Dr. Jeevan Bala, Asst. Professor,

School of Computer Science and Engineering

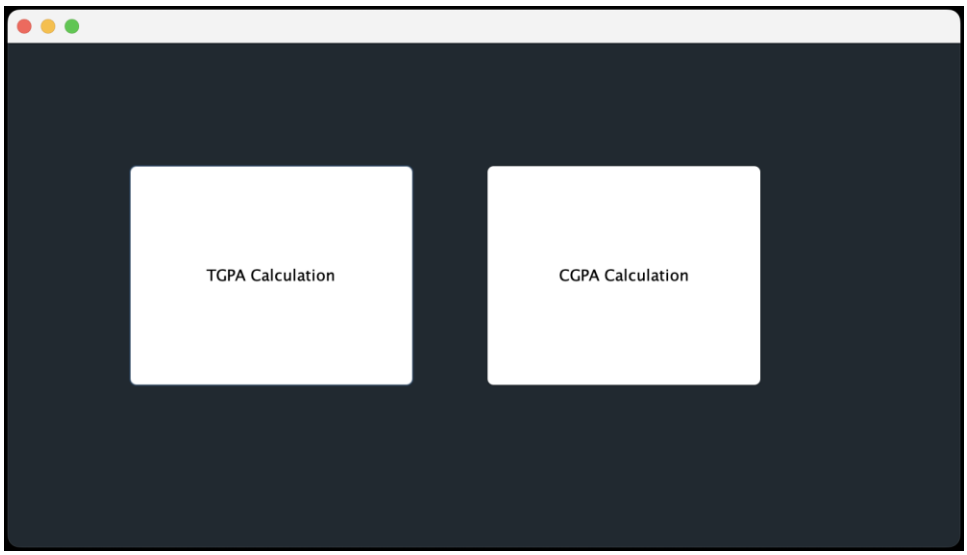
UID: 26699

Introduction:

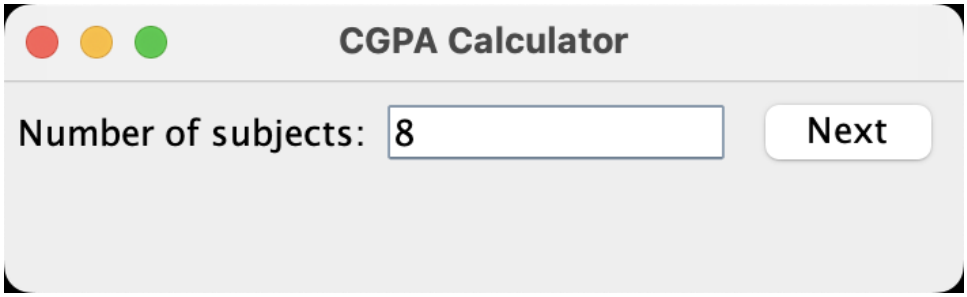
We created an LPU CGPA and TGPA Calculator using Java. For the GUI of the application, we used Java Swing.

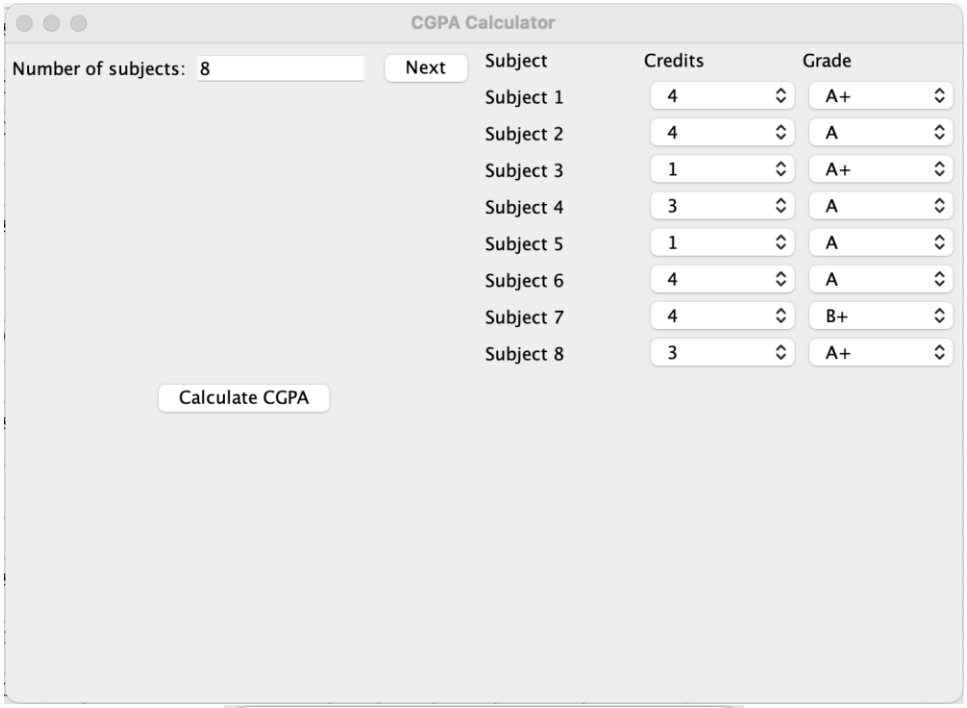


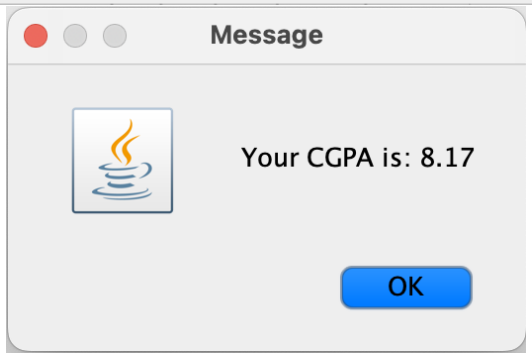
In this Project, we have added two different options for the user:



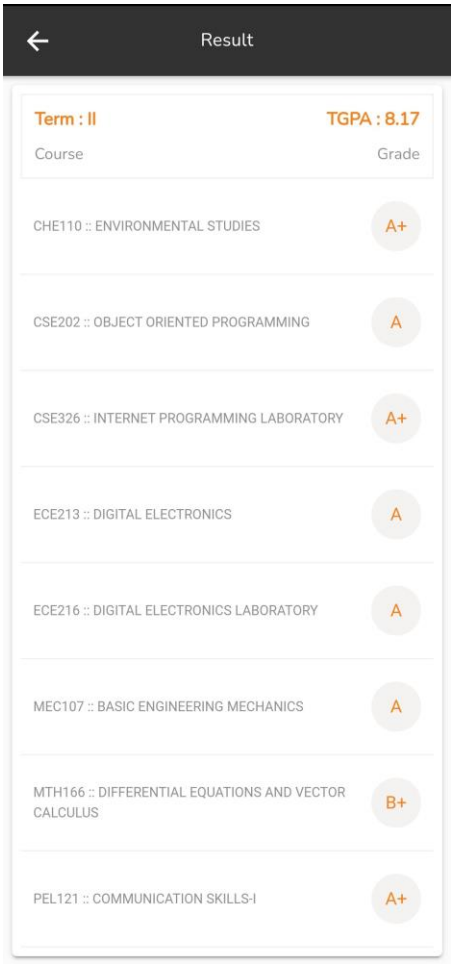
1) Term GPA Calculation (TGPA): The user needs to provide the total number of subjects in the current Semester along with the subject credits and grades obtained. The selection is done by using the COMBOBOXES in JavaSwing and the term CGPA is calculated accordingly.



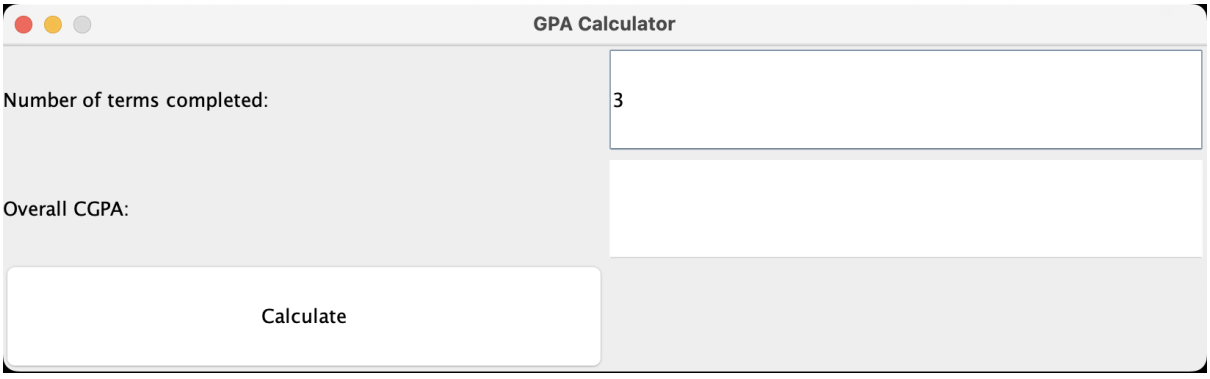


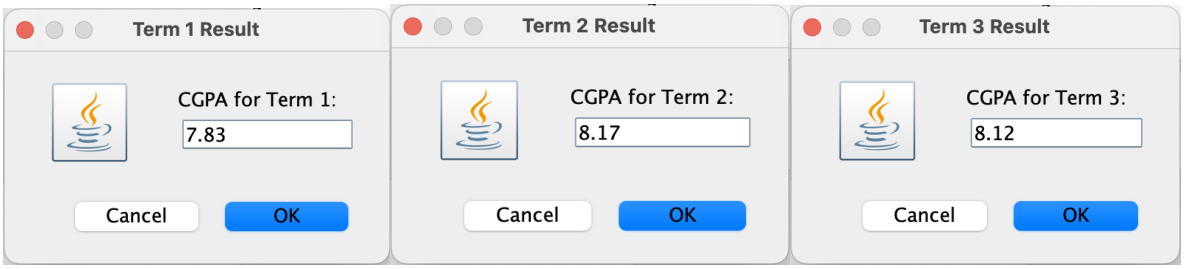


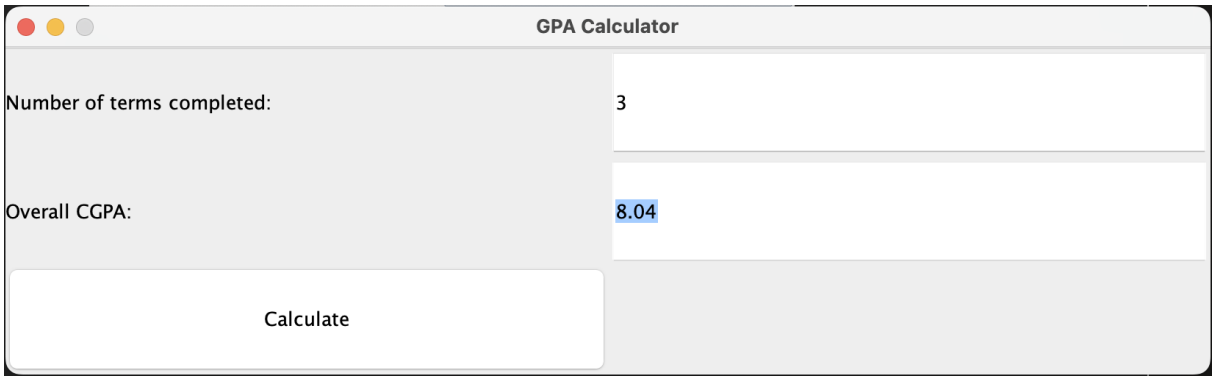
Reference from LPU Touch:



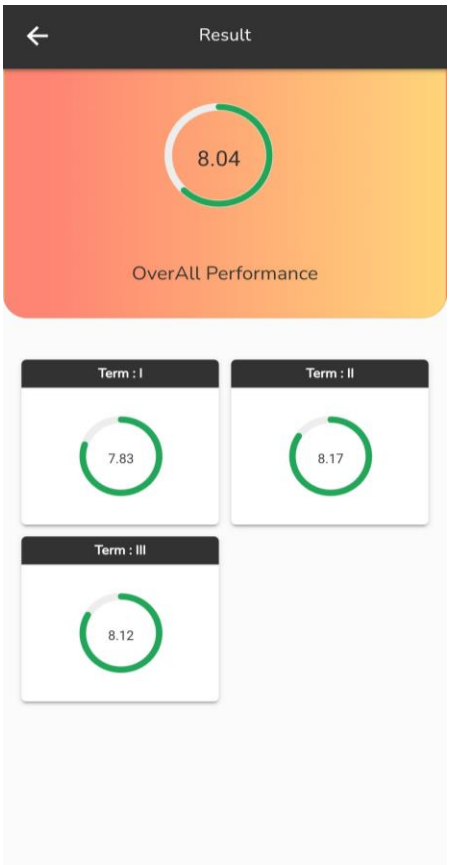
2) Cumulative GPA Calculation (CGPA): In this the user can calculate his/her overall CGPA by entering the past semesters TGPA obtained.







Reference from LPU Touch:



Code:

The program has 4 files named as:

1. WelcomeScreen.java
2. SelectionPage.java
3. TGPACalculator.java
4. CGPACalculator.java

And an Image, which is available in GitHub Link provided at the bottom of the report.

1. WelcomeScreen.java

```
import java.awt.Color;
import javax.swing.ImageIcon;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JProgressBar;


public class WelcomeScreen {


    private static int value;
    public static void main(String[] args){
        JFrame frame = new JFrame();
        JLabel label = new JLabel();
        JProgressBar bar = new JProgressBar();
        bar.setValue(value);
        bar.setBounds(160, 280, 270, 9);
        bar.setStringPainted(true);



Thread thread = new Thread(() -> {
    for (int i = 0; i <= 100; i += 20) {
```

```
        try {
            Thread.sleep(600);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
        bar.setValue(i);
        if(bar.getValue()==100){

            frame.dispose();
            SelectionPage mywindow = new SelectionPage();

        }
    }
});
thread.start();



        ImageIcon image1 = new ImageIcon("lpu_logo3.png");
        label.setIcon(image1);
        label.setHorizontalAlignment(JLabel.CENTER);
        label.setVerticalAlignment(JLabel.CENTER);
        label.add(bar);
        frame.setSize(600, 400);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setTitle("LPU CGPA Calculator");
        frame.setLocationRelativeTo(null);
        frame.getContentPane().setBackground(new Color(0,0,0));
        frame.add(label);
        frame.setVisible(true);
    }
```

```
}
```

2. SelectionPage.java

```
import java.awt.Color;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;



public class SelectionPage implements ActionListener {

   JButton button = new JButton();
   JButton button2 = new JButton();



SelectionPage(){

      JLabel label = new JLabel();

      button.setBounds(100,100,243,190);
      button.setText("TGPA Calculation");
      label.add(button);

      button2.setBounds(400,100,235,190);
      button2.setText("CGPA Calculation");
      label.add(button2);
```

```
        JFrame frame = new JFrame();
        frame.setSize(800,450);
        frame.setLocationRelativeTo(null);
        frame.getContentPane().setBackground(new Color(33,41,48));
        frame.add(label);
        frame.setVisible(true);
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                frame.setVisible(false);
                TGPACalculator w2 = new TGPACalculator();
            }
        });

        button2.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                frame.setVisible(false);
                CGPACalculator W1 = new CGPACalculator();
            }
        });
}
@Override
public void actionPerformed(ActionEvent e) {}
}
```

3. TGPACalculator.java

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
```

```
public class TGPACalculator extends JFrame implements ActionListener {
    private JLabel numSubjectsLabel;
    private JTextField numSubjectsField;
    private JButton nextButton;
    private JLabel[] subjectLabels;
    private JComboBox<Integer>[] creditBoxes;
    private JComboBox<String>[] gradeBoxes;
    private JButton calculateButton;

    public TGPACalculator() {
        setTitle("CGPA Calculator");
        numSubjectsLabel = new JLabel("Number of subjects:");
        numSubjectsField = new JTextField(10);
        nextButton = new JButton("Next");
        nextButton.addActionListener(this);
        JPanel numSubjectsPanel = new JPanel();
        numSubjectsPanel.add(numSubjectsLabel);
        numSubjectsPanel.add(numSubjectsField);
        numSubjectsPanel.add(nextButton);
        setLayout(new GridLayout(2, 1));
        add(numSubjectsPanel);
        pack();
        setLocationRelativeTo(null);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setVisible(true);
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == nextButton) {
```

```
    int numSubjects = Integer.parseInt(numSubjectsField.getText());
    subjectLabels = new JLabel[numSubjects];
    creditBoxes = new JComboBox[numSubjects];
    gradeBoxes = new JComboBox[numSubjects];
    JPanel subjectsPanel = new JPanel();
    subjectsPanel.setLayout(new GridLayout(numSubjects + 1, 3));
    subjectsPanel.add(new JLabel("Subject"));
    subjectsPanel.add(new JLabel("Credits"));
    subjectsPanel.add(new JLabel("Grade"));
    for (int i = 0; i < numSubjects; i++) {
        subjectLabels[i] = new JLabel("Subject " + (i+1));
        subjectsPanel.add(subjectLabels[i]);
        creditBoxes[i] = new JComboBox<>();
        for (int j = 1; j <= 5; j++) {
            creditBoxes[i].addItem(j);
        }
        subjectsPanel.add(creditBoxes[i]);
        gradeBoxes[i] = new JComboBox<>(new String[]{"O", "A+", "A", "B+", "B", "C","E"});
        subjectsPanel.add(gradeBoxes[i]);
    }
    calculateButton = new JButton("Calculate CGPA");
    calculateButton.addActionListener(this);
    JPanel buttonPanel = new JPanel();
    buttonPanel.add(calculateButton);
    setLayout(new GridLayout(2, 1));
    add(subjectsPanel);
    add(buttonPanel);
    pack();
    setLocationRelativeTo(null);
}
else if (e.getSource() == calculateButton) {
```

```
double totalCredits = 0;
double totalGradePoints = 0;
for (int i = 0; i < subjectLabels.length; i++) {
    int credits = (int) creditBoxes[i].getSelectedItem();
    totalCredits += credits;
    String grade = (String) gradeBoxes[i].getSelectedItem();
    Double gradePoint;
    switch (grade) {
        case "O":
            gradePoint = 10.0;
            break;
        case "A+":
            gradePoint = 9.0;
            break;
        case "A":
            gradePoint = 8.0;
            break;
        case "B+":
            gradePoint = 7.0;
            break;
        case "B":
            gradePoint = 6.0;
            break;
        case "C":
            gradePoint = 5.0;
            break;
        case "E":
            gradePoint = 0.0;
        default:
            gradePoint = 0.0;
            break;
```

```
            }
            totalGradePoints += credits * gradePoint;
        }
        Double cgpa = totalGradePoints / totalCredits;
        JOptionPane.showMessageDialog(this, "Your CGPA is: " + String.format("%.2f", cgpa));
    }
  }
  public static void main(String[] args) {
    new TGPACalculator();
  }
}
```

4. CGPACalculator.java

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class CGPACalculator implements ActionListener {
  JFrame frame;
  JPanel panel;
  JLabel noOfTermsLabel, overallCGPALabel;
  JTextField noOfTermsField, overallCGPAField;
  JButton calculateButton;

  public CGPACalculator() {
    frame = new JFrame("GPA Calculator");
    frame.setSize(820, 250);
    frame.setResizable(false);
    frame.setLocationRelativeTo(null);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
        frame.getContentPane().setBackground(new Color(33,41,48));
        panel = new JPanel();
        panel.setLayout(new GridLayout(3, 2));
        noOfTermsLabel = new JLabel("Number of terms completed: ");
        panel.add(noOfTermsLabel);
        noOfTermsField = new JTextField();
        panel.add(noOfTermsField);
        overallCGPALabel = new JLabel("Overall CGPA: ");
        panel.add(overallCGPALabel);
        overallCGPAField = new JTextField();
        overallCGPAField.setEditable(false);
        panel.add(overallCGPAField);
        calculateButton = new JButton("Calculate");
        calculateButton.addActionListener(this);
        panel.add(calculateButton);
        frame.add(panel);
        frame.setVisible(true);
    }
    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == calculateButton) {
            int noOfTerms = Integer.parseInt(noOfTermsField.getText());
            double overallCGPA = 0.0;
            for (int i = 1; i <= noOfTerms; i++) {
                JTextField termCGPAField = new JTextField();
                Object[] fields = {"CGPA for Term " + i + ":", termCGPAField};
                int result = JOptionPane.showConfirmDialog(null, fields, "Term " + i + " Result",
JOptionPane.OK_CANCEL_OPTION);
                if (result == JOptionPane.OK_OPTION) {
                    double termCGPA = Double.parseDouble(termCGPAField.getText());
                    overallCGPA += termCGPA;
                } else {
```

```
                return;
            }
        }
        overallCGPA /= noOfTerms;
        overallCGPAField.setText(String.format("%.2f", overallCGPA));
    }
  }
  public static void main(String[] args) {
    new CGPACalculator();
  }
}
```

GitHub Link: -

https://github.com/vamsi-120304/java-mini-project