ENVISION: EMPIRICAL DATA VISUALISATION

APP FOR VIT

A PROJECT REPORT

Submitted by

Anushka Khare – 19BCE10273

Aditi Nigam – 19BCE10300

Laveena Dudani– 19BCE10378

Divya Mathew – 19BCE10456

in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

VIT BHOPAL UNIVERSITY

KOTHRIKALAN, SEHORE

MADHYA PRADESH – 466114

APRIL 2021

VIT BHOPAL UNIVERSITY, KOTHRIKALAN,

SEHORE

MADHYA PRADESH – 466114

BONAFIDE CERTIFICATE

Certified that this project report titled "**ENVISION: EMPIRICAL DATA VISUALISATION APP FOR VIT"** is the bonafide work of "**ANUSHKA KHARE-19BCE10273, ADITI NIGAM-19BCE10300, LAVEENA DUDANI-19BCE10378, DIVYA MATHEW-19BCE10456"** who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here does not form part of any other project / research work on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

PROGRAM CHAIR	**PROJECT GUIDE**
M. Ashwin ,	Dr.Rakesh R,
Dean-SCSE,	Assistant Professor,
School of Computer Science and Engineering	School of CSE,
VIT BHOPAL UNIVERSITY	VIT BHOPAL UNIVERSITY

The Project Exhibition I Examination is held on 4th May 2021.

ACKNOWLEDGEMENT

First and foremost, I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to M. Ashwin, Head of the Department, School of Computer Science, for much of his valuable support and encouragement in carrying out this work.

I would like to thank my internal guide Dr.Rakesh R, for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Computer Science, who extended directly or indirectly all support.

Last, but not the least, I am deeply indebted to my parents who have been the greatest support while,I worked day and night for the project to make it a success.

LIST OF ABBREVIATIONS

S.NO	ABBREVIATIONS	FULL FORM
1	VIT	Vellore Institute Of Technology
2	SCSE	School Of Computer Science Engineering
3	CGPA	Cumulative Grade Points Average
4	GPA	Grade Points Average
5	GUI	Graphical User Interface
6	SEEE	School of Electrical and Electronics Engineering
7	SMEC	School of Mechanical Engineering
8	SAS	School of
9	SASL	School of Advanced Science and Language

LIST OF FIGURES

ABSTRACT

ENVISION aims to provide information to the users regarding different aspects of VIT Bhopal using data visualization. Our app emphasizes the fact that graphical representations of data are more effective for analysis than long textual files. It also helps students get information about their curriculum, credits earned, cgpa and gpa procured by a particular student and teachers of different department. We have tried to achieve transparency by providing a docket of curriculum for each and every branch accessible to all. They could see above information in form of bar charts or pie charts for better analysis.

TABLE OF CONTENT

INTRODUCTION

1.1 INTRODUCTION

Data Visualization is the graphical and explicit representation of information and data. With the use of visual elements like charts, graphs, and maps, data visualization makes attainable and convenient to understand the patterns and motif in data.

1.2 MOTIVATION FOR THE WORK

During course registration, discerning faculty and their respective departments is strenuous. Also, while calculating credits, cgpa or gpa, students have to access unsanctioned applications or online calculators which sometimes proves to be tedious. Currently, students also don't have access to scrutinize other branches' curriculum and freedom to have an overlook over the different curriculums or courses offered by the university. So, to minimize the problems of students, we decided to prepare an app ENVISION: Empirical Data Visualization App for VIT.

1.3 OBJECTIVE OF THE WORK

This project focuses on the points:

- To understand data visualization and its fundamentals.
- Its working and applications in different areas.
- Implementation of ENVISION app through data visualization.

LITERATURE REVIEW

2.1 INTRODUCTION

In the mid – 1980s, the field of visualization separated from computer graphics, to discern graphics added from scientific and engineering data.

In the early 1990s, a foster partition took place to discern scientific data from intellectual "information visualization".

Data visualization usual techniques are Line Charts, Bubble Charts, Bar Charts, Scatterplots, Pie Charts, Images and Maps.

To understand data by diagrams and maps, visualization was used in China as early as 1137.

Visualization techniques has been used widely in all the fields. To examine information and data, visualization help in expressing ideas in architecture. With the coming of computer simulation, visualization apposite has been foster notation.

Information visualization concentrates on data sets underlying Two – Dimensional (2D) or Three – Dimensional (3D) substance. For visualization, there are a lot of long – familiar methods for data sets, such as, x-y plots, histograms, and line – plots.

Data visualization covers geographical information systems, graphical user interface, user interface, digital images, graphs, multidimensional tables, virtual reality and three – dimensional advantages.

2.2 RESEARCH WORK DONE BY PEOPLE

- French mathematician and philosopher Rene Descartes developed a two-dimensional coordinate system for displaying values, consisting of horizontal axis for one variable and a vertical for another, primarily as a graphical means of performing mathematical operations.

- Scotsman William Playfair pioneered many of the graphs that are commonly used today. He was the first person to use a line moving up and down as it progressed from left to right to show how values change through time.

- John Tukey (1977) gave a whole new statistical approach called *exploratory data analysis.*

- Edward Tufte (1983) published his groundbreaking book *The Visual Display of Quantitative Information,* in which he pointed out the effective ways of displaying data visually.

- William Cleveland extended and refined data visualization techniques for statisticians.

PROJECT PROCEDURE

3.1 ENVISION APP

Our app aims to provide information to the users regarding different aspects of VIT Bhopal using data visualization. Our app emphasizes the fact that graphical representations of data are more effective for analysis than long textual files. It provides students with information regarding curriculum, credits earned, and teachers of different subjects, no. of students in each branch, through an interactive GUI and by simply clicking on a few buttons.

3.2 A BRIEF OF PROPOSED APPROACH

- **VISUAL ENCODING –** Translates data into visual form by applying unique graphical attributes like shape, color, size, area, volume, etc. to both quantitative and qualitative data.
- **TYPOGRAPHY –** Text can be used to label different chart elements like chart titles, data labels, axis labels and legend.
- **ICONOGRAPHY –** Used to represent different types of data in a chart and improve a chart's overall usability.
- **LEGENDS AND ANNOTATIONS –** Used to describe a chart's information by highlighting data points, data outliers, and any noteworthy content.
- **DASHBOARD DESIGN –** A dashboard's purpose should be reflected in its layout, style and interaction patterns. Its design should suit how it will be used, whether it's a tool for making a presentation or deeply exploring data.

3.3 METHODOLOGY

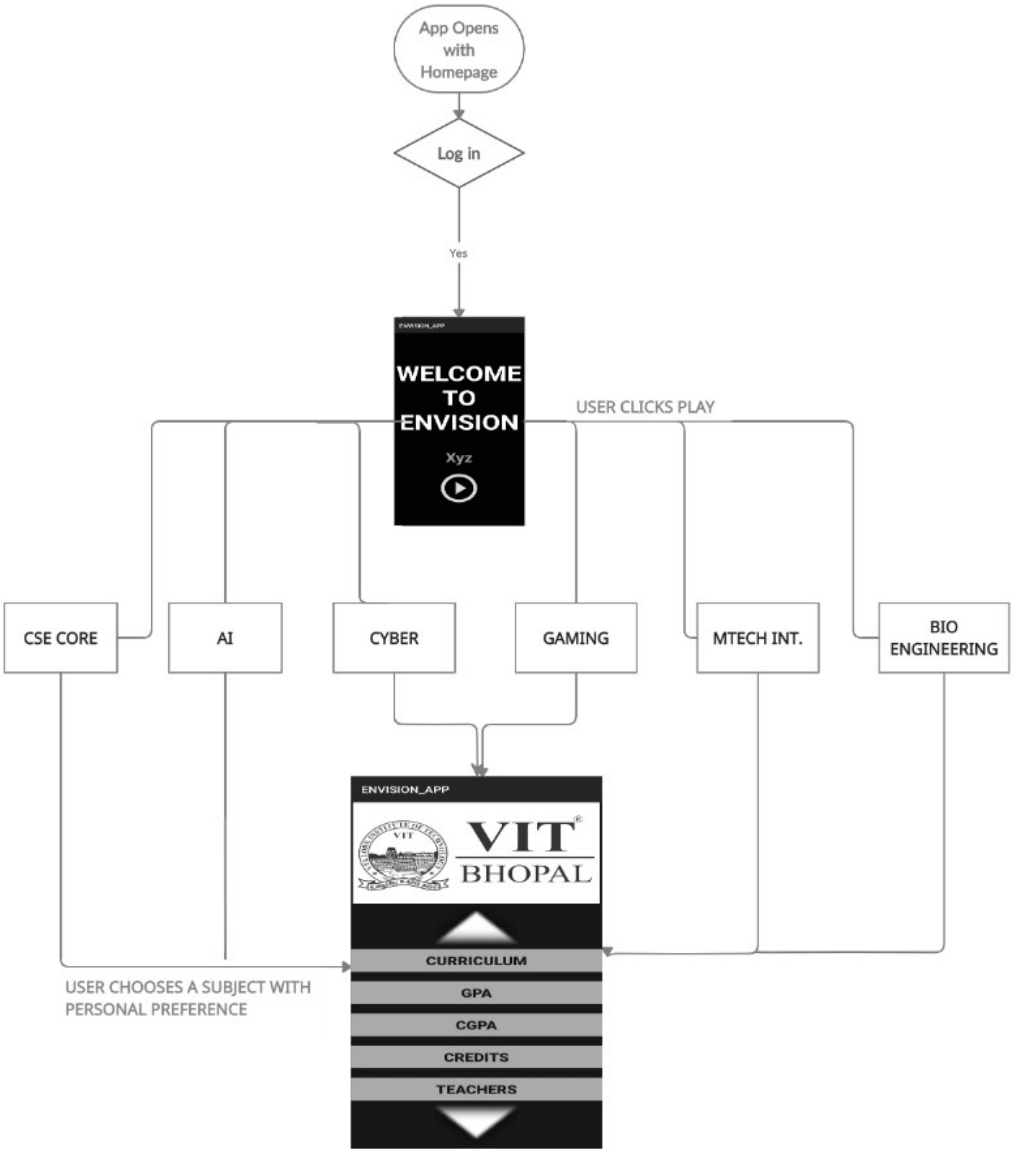
- **LINE CHART –** The simplest technique, a line plot is used to plot the relationship or dependence of one variable on another. To plot the relationship between the two variables, we can simply create and call the line dataset (plot) function.

- **BAR CHART –** Bar charts are used for comparing the quantities of different categories or groups. Values of a category represented with the help of vertical or horizontal bars, with the length or height of each bar representing the value.

- **PIE CHART –** It is a circular statistical graph which decides slices to illustrate numerical proportion. Here the arc length of each slide is proportional to the quantity it represents.

- **RADAR CHART –** A radar chart is a graphical method of displaying multivariate data in the form of a two-dimensional chart of three or more quantitative variables represented on axes starting from the same point.

3.4 FLOW DIAGRAM



3.5 HARDWARE REQUIREMENTS

- Windows 10 64 bit
- Intel Core i7 processors
- Minimum RAM = 8GB
- Maximum RAM = 16GB
- Intel UHD Graphics
- 128GB SSD

3.6 SOFTWARE REQUIREMENTS

- **ANDROID STUDIO:**

Provides the fastest tools for building apps on every type of Android device.

- VISUAL LAYOUT EDITOR: Used to create complex layouts with Constraint Layout by adding constraints from each view to other views and guidelines.
- APK ANALYZER: Inspect the manifest file resources, and DEX files.
- FAST EMULATOR: Used to simulate different configurations and features.

- **CHARTING LIBRARY (MP ANDROID CHART):**

A powerful Android chart view/graph view library, supporting line – bar – pie – radar – bubble – and candlestick charts as well as scaling, panning and animations.

- **DEPENDENCIES:**

The dependencies can be located on your machine or in a remote repository, and any transitive dependencies they declare are automatically included as well.

```
repositories {
maven { url "https://jitpack.io }}
dependencies {
implementation 'com.github.PhilJay:MPAndroidChart:v2.2.4'}
```

3.7 ADVANTAGES OF DATA VISUALIZATION

1. BETTER ANALYSIS

- Data visualization helps business stakeholders analyze reports regarding sales, marketing strategies and product interest.
- Based on the analysis, they can focus on the areas that require attention to increase profits, which in turn makes the business more productive.

2. QUICK ACTION

- The human brain grasps visuals more easily than table reports.
- Data visualizations allow decision makers to be notified quickly of new data insights and take necessary actions for business growth.

3. IDENTIFYING PATTERNS

- Large amounts of complicated data can provide many opportunities for insights when we visualize them.
- Visualization allows business users to recognize relationships between the data, providing greater meaning to it.

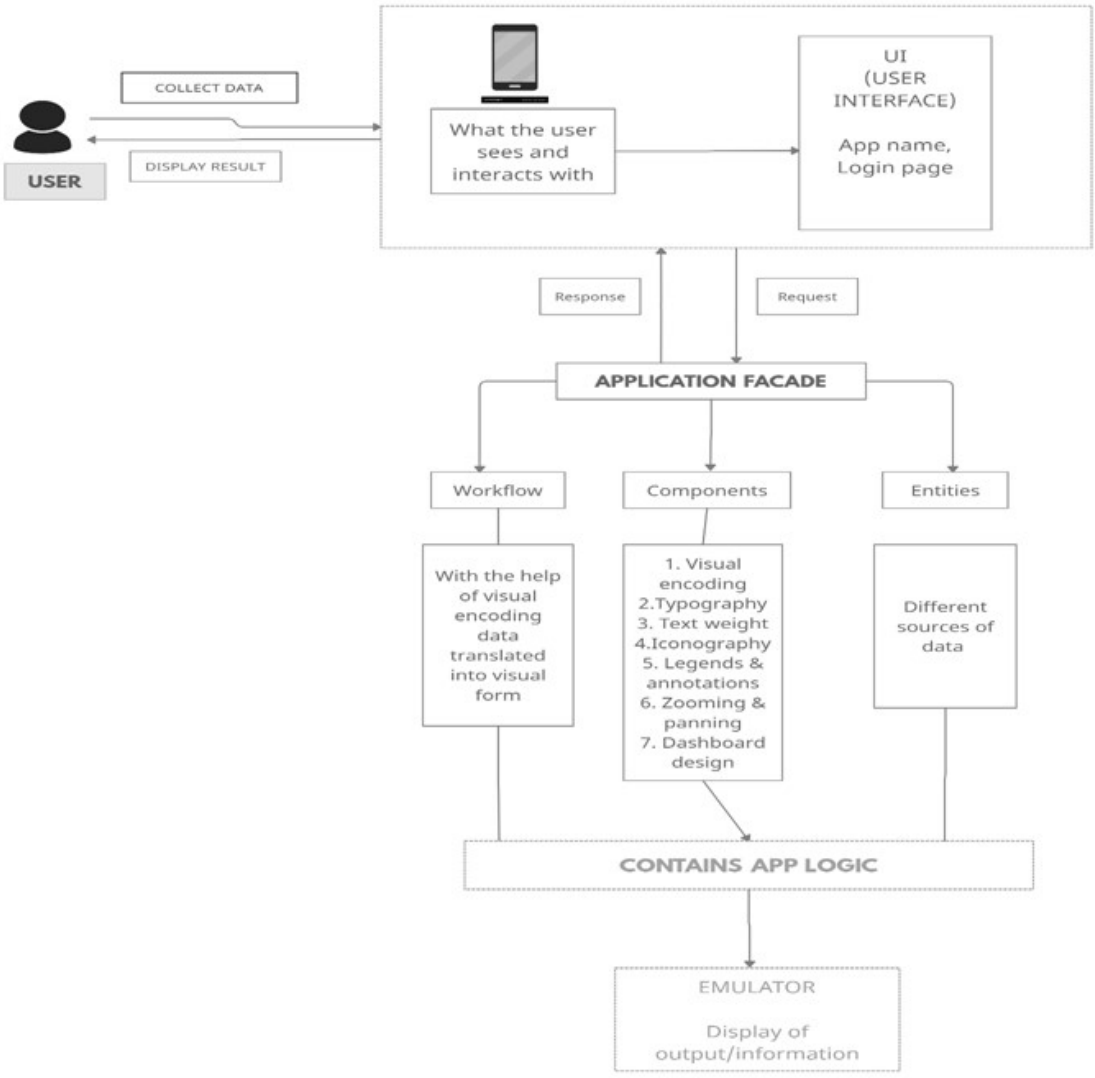
4. FINDING ERRORS

- Visualizing your data helps quickly identify any errors in the data.
- If the data tends to suggest the wrong actions, visualizations help identify erroneous data sooner so that it can be removed from analysis.

5. UNDERSTANDING THE STORY

- Storytelling is the purpose of your dashboard.
- By designing your visuals in a meaningful way, you help the target audience grasp the story in a single glance.

ENVISION APP'S ARCHITECTURE DIAGRAM



ENVISION APP ADVANTAGES

- **FAST CONSULTANCY**

Through the ENVISION app students can access details and information regarding curriculum, credits earned, teachers of different subjects, number of students in each branch very accurately and spritely.

- **PERSONAL SUPPORT**

Students can get personal assistance by knowing their curriculum and calculating there cgpa and gpa.

- **BETTER UNDERSTANDING**

Students will face no vacillation and uncertainty as they will be able to keep a record of their scored gpa in each semester along with the credits owned.

- **SAFE AND SECURE**

Our app is 100 per cent safe and secure as it does not store any information like student's name or scored gpa or credits owned. Without the burning question of uncertainty in their mind, students can use this app safely without second thoughts.

WORK DONE

4.1 CURRICULUM

4.1.1 Bar Chart

```
public class MainActivity29 extends AppCompatActivity
{
BarChart barchart;
ArrayList<BarEntry> barEntryArrayList;
ArrayList<String> labels;
ArrayList<Teachers> teachersArrayList = new ArrayList<>();
Button b;
@Override
protected void onCreate(Bundle savedInstanceState)
{
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main29);
b = (Button)findViewById(R.id.button14);
b.setOnClickListener(new View.OnClickListener()
{
@Override
public void onClick(View view)
{
Intent intent=new Intent(MainActivity29.this , MainActivity30.class);
startActivity(intent);
}
});
barchart = findViewById(R.id.barChart);
barEntryArrayList = new ArrayList<>();
labels = new ArrayList<>();
fillsubjects();
```

```
for (int i =0; i<teachersArrayList.size();i++) {
String teacher = teachersArrayList.get(i).getTeacher();
int number = teachersArrayList.get(i).getNumber();
barEntryArrayList.add(new BarEntry(i,number));
labels.add(teacher);
}
BarDataSet barDataSet = new BarDataSet(barEntryArrayList, "NUMBER OF
SUBJECTS");
barDataSet.setColors(ColorTemplate.COLORFUL_COLORS);
barDataSet.setValueTextSize(20f);
Description description = new Description();
description.setText("Teachers");
barchart.setDescription(description);
BarData barData = new BarData(barDataSet);
barchart.setData(barData);
// set axis value formater
XAxis xAxis = barchart.getXAxis();
xAxis.setValueFormatter(new IndexAxisValueFormatter(labels));
//set position of labels
 xAxis.setPosition(XAxis.XAxisPosition.TOP);
xAxis.setDrawGridLines(false);
xAxis.setDrawAxisLine(false);
xAxis.setGranularity(1f);
xAxis.setLabelCount(labels.size());
xAxis.setLabelRotationAngle(270);
barchart.animateY(2000);
barchart.invalidate();
}
private void fillsubjects()
{
teachersArrayList.clear();
teachersArrayList.add(new Teachers("CSE",23));
teachersArrayList.add(new Teachers("DSN",5));
```

```
teachersArrayList.add(new Teachers("ECE",5));
teachersArrayList.add(new Teachers("NAS",4));
teachersArrayList.add(new Teachers("CHY",2));
teachersArrayList.add(new Teachers("ENG",2));
teachersArrayList.add(new Teachers("MAT",7));
teachersArrayList.add(new Teachers("PHY",3));
teachersArrayList.add(new Teachers("PLA",2));
teachersArrayList.add(new Teachers("STS",2));
teachersArrayList.add(new Teachers("OTHERS",3));
}}
```

4.1.2 Pie Chart

```
public class MainActivity31 extends AppCompatActivity
{
private PieChart pieChart;
Button b;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main31);
b = (Button)findViewById(R.id.button14);
b.setOnClickListener(new View.OnClickListener() {
@Override
 public void onClick(View view) {
 Intent intent=new Intent(MainActivity31.this , MainActivity30.class);
startActivity(intent);
}});
pieChart = findViewById(R.id.pieChart);
setupPieChart();
loadPieChartData();
}
private void setupPieChart() {
```

```
pieChart.setDrawHoleEnabled(true);
pieChart.setUsePercentValues(false);
pieChart.setEntryLabelTextSize(16);
pieChart.setEntryLabelColor(Color.BLACK);
pieChart.setCenterText("SUBJECTS");
pieChart.setCenterTextSize(24);
pieChart.getDescription().setEnabled(false);
Legend l = pieChart.getLegend();
l.setVerticalAlignment(Legend.LegendVerticalAlignment.TOP);
l.setHorizontalAlignment(Legend.LegendHorizontalAlignment.RIGHT);
l.setOrientation(Legend.LegendOrientation.VERTICAL);
l.setDrawInside(false);
l.setEnabled(true);
}
private void loadPieChartData() {
ArrayList<PieEntry> entries = new ArrayList<>();
entries.add(new PieEntry(23, "CSE"));
entries.add(new PieEntry(5,"DSN"));
entries.add(new PieEntry(5,"ECE"));
entries.add(new PieEntry(4,"NAS"));
entries.add(new PieEntry(2,"CHY"));
entries.add(new PieEntry(2,"ENG"));
entries.add(new PieEntry(7,"MAT"));
entries.add(new PieEntry(3,"PHY"));
entries.add(new PieEntry(2,"PLA"));
entries.add(new PieEntry(2,"STS"));
entries.add(new PieEntry(3,"OTHERS"));
ArrayList<Integer> colors = new ArrayList<>();
for (int color: ColorTemplate.COLORFUL_COLORS) {
colors.add(color);
}
for ( int color: ColorTemplate.VORDIPLOM_COLORS) {
colors.add(color);
```

```
}
PieDataSet dataSet = new PieDataSet(entries,"SUBJECTS");
dataSet.setColors(colors);
PieData data = new PieData(dataSet);
data.setDrawValues(true);
data.setValueFormatter(new PercentFormatter(pieChart));
data.setValueTextSize(14f);
data.setValueTextColor(Color.BLACK);
pieChart.setData(data);
pieChart.invalidate();
}}
```

4.2 GPA

```
public class MainActivity extends AppCompatActivity
{
TextView tv;
Button b;
EditText t1,t15,t2,t25,t3,t35,t4,t45,t5,t55,t6,t65,t7,t75,t8,t85,t9,t95;
@Override
protected void onCreate(Bundle savedInstanceState)
{
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
t1=(EditText)findViewById(R.id.editText1);
t15=(EditText)findViewById(R.id.editText15);
t2=(EditText)findViewById(R.id.editText2);
t25=(EditText)findViewById(R.id.editText25);
…
 b=(Button)findViewById(R.id.button);
b.setOnClickListener(new View.OnClickListener()
{
@Override
public void onClick(View view)
{
int N1=0;
```

```
if(t15.getText().toString().equals("S"))        {       N1=10;          }
if(t15.getText().toString().equals("A"))        {       N1=9;           }
if(t15.getText().toString().equals("B"))        {       N1=8;           }
if(t15.getText().toString().equals("C"))        {       N1=7;           }
if(t15.getText().toString().equals("D"))        {       N1=6;           }
if(t15.getText().toString().equals("E"))        {       N1=5;           }
if(t15.getText().toString().equals("F"))        {       N1=0;           }
if(t15.getText().toString().equals("N"))        {       N1=0;           }
int num1 =N1*Integer.parseInt(t1.getText().toString());
…
float y = Integer.parseInt(t9.getText().toString())+
Integer.parseInt(t8.getText().toString())+ Integer.parseInt(t7.getText().toString())+
nteger.parseInt(t6.getText().toString())+ Integer.parseInt(t5.getText().toString())+
Integer.parseInt(t4.getText().toString())+ Integer.parseInt(t3.getText().toString())+
Integer.parseInt(t2.getText().toString())+ Integer.parseInt(t1.getText().toString());
float z=(num1+num2+num3+num4+num5+num6+num7+num8+num9);
float x =z/y;
tv.setText(Float.toString(x));
Toast.makeText(MainActivity.this,"Result="+x, LENGTH_SHORT).show();
Log.d("string","Result="+x);
}
});
}
}
```

4.3 CGPA

```
public class MainActivity extends AppCompatActivity
{
TextView tv;
Button b;EditText t,t1,t2,t3,t4,t5,t6,t7,t8;
 @Override protected void onCreate(Bundle savedInstanceState)
        super.onCreate(savedInstanceState);setContentView(R.layout.activity_main);t
=(EditText)findViewById(R.id.editText);t1=(EditText)findViewById(R.id.editText1)
;t2=(EditText)findViewById(R.id.editText2);t3=(EditText)findViewById(R.id.editTe
```

```
xt3)t4=(EditText)findViewById(R.id.editText4);t5=(EditText)findViewById(R.id.edi
tText5);t6=(EditText)findViewById(R.id.editText6)t7=(EditText)findViewById(R.id.
editText7);t8=(EditText)findViewById(R.id.editText8);tv=(TextView)findViewById(
R.id.textView);b=(Button)findViewById(R.id.button);b.setOnClickListener(new
View.OnClickListener(){@Overridepublic void onClick(View view) {Float num =
(Float.parseFloat(t1.getText().toString()) +Float.parseFloat(t2.getText().toString())
+Float.parseFloat(t3.getText().toString()) +Float.parseFloat(t4.getText().toString())
+Float.parseFloat(t5.getText().toString()) +Float.parseFloat(t6.getText().toString())
+Float.parseFloat(t7.getText().toString()) +Float.parseFloat(t8.getText().toString())) /
Integer.parseInt(t.getText().toString())
tv.setText(Float.toString(num));Toast.makeText(MainActivity.this,"Result="+num,
LENGTH_SHORT).show();Log.d("string","Result="+ num)}});}}
```

4.4 CREDITS CALCULATION

```
public class MainActivity extends AppCompatActivity
 {
TextView tv,tv1,tv2,t11,t12;
…
CheckBox c,c1,c2,c3;
…
Button button,b1,b2,b3;
...
@Override
protected void onCreate(Bundle savedInstanceState)
{
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
c=(CheckBox)findViewById(R.id.checkBox);
c1=(CheckBox)findViewById(R.id.checkBox1);
…
tv=(TextView)findViewById(R.id.textView);
tv1=(TextView)findViewById(R.id.textView1);
tv2=(TextView)findViewById(R.id.textView2);
t11=(TextView)findViewById(R.id.textView11);
```

```
t12=(TextView)findViewById(R.id.textView12);
…
b1=(Button)findViewById(R.id.button11);
…
b1.setOnClickListener(new View.OnClickListener()
{
@Override
public void onClick(View view)
{
String s1=" ";
int b1=0;
if(c.isChecked())
{
t11.setText(c.getText().toString());
s1=s1+" "+c.getText().toString();
b1=b1+4;
}
if(c21.isChecked())
{
t11.setText(c21.getText().toString());
s1=s1+" "+c21.getText().toString();
b1=b1+4;
}
if(c39.isChecked())
{
t11.setText(c39.getText().toString());
s1=s1+" "+c39.getText().toString();
b1=b1+4;
}
if(c42.isChecked())
{
t11.setText(c42.getText().toString());
s1=s1+" "+c42.getText().toString();
b1=b1+2;
}
if(c43.isChecked())
{
t11.setText(c43.getText().toString());
s1=s1+" "+c43.getText().toString();
b1=b1+2;
```

```
}
if(c45.isChecked())
{
t11.setText(c45.getText().toString());
s1=s1+" "+c45.getText().toString();
b1=b1+2;
}
if(c46.isChecked())
{
t11.setText(c46.getText().toString());
s1=s1+" "+c46.getText().toString();
b1=b1+4;
}
if(c53.isChecked())
{
t11.setText(c53.getText().toString());
s1=s1+" "+c53.getText().toString();
b1=b1+4;
}
t11.setText(s1);
t12.setText(Integer.toString(b1));
}
});
…
tv.setText(s);
tv1.setText(Integer.toString(b));
int x=180-b;
tv2.setText(Integer.toString(x));
}
}
```

4.5 TEACHERS' SLOT

4.5.1 Bar Graph

```
public class MainActivity15 extends AppCompatActivity
{
BarChart barchart;
ArrayList<BarEntry> barEntryArrayList;
ArrayList<String> labels;
ArrayList<Teachers> teachersArrayList = new ArrayList<>();
Button b1,b2,b3,b4,b5,b6;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main15);
b1 = (Button)findViewById(R.id.button8);
b2 = (Button)findViewById(R.id.button9);
b3 = (Button)findViewById(R.id.button10);
b4 = (Button)findViewById(R.id.button11);
b5 = (Button)findViewById(R.id.button12);
b6 = (Button)findViewById(R.id.button13);
b1.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
Intent intent=new Intent(MainActivity15.this , MainActivity16.class);
startActivity(intent);
}
});
b2.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
Intent intent=new Intent(MainActivity15.this , MainActivity17.class);
```

```
startActivity(intent);
}
});
b3.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
Intent intent=new Intent(MainActivity15.this , MainActivity18.class);
startActivity(intent);
}
});
b4.setOnClickListener(new View.OnClickListener() {
 @Override
  public void onClick(View view) {
  Intent intent=new Intent(MainActivity15.this , MainActivity19.class);
  startActivity(intent);
  }
  });
   b5.setOnClickListener(new View.OnClickListener() {
   @Override
   public void onClick(View view) {
   Intent intent=new Intent(MainActivity15.this , MainActivity20.class);
   startActivity(intent);
  }
  });
  b6.setOnClickListener(new View.OnClickListener() {
  @Override
  public void onClick(View view) {
  Intent intent=new Intent(MainActivity15.this , MainActivity21.class);
  startActivity(intent);
  }
  });
      barchart = findViewById(R.id.barChart);
      barEntryArrayList = new ArrayList<>();
```

```
    labels = new ArrayList<>();
    fillteachers();
    for (int i =0; i<teachersArrayList.size();i++) {
    String teacher = teachersArrayList.get(i).getTeacher();
    int number = teachersArrayList.get(i).getNumber();
    barEntryArrayList.add(new BarEntry(i,number));
    labels.add(teacher);
    }
    BarDataSet barDataSet = new BarDataSet(barEntryArrayList, "NUMBER OF
TEACHERS");
    barDataSet.setColors(ColorTemplate.COLORFUL_COLORS);
    barDataSet.setValueTextSize(20f);
    Description description = new Description();
    description.setText("Teachers");
    barchart.setDescription(description);
    BarData barData = new BarData(barDataSet);
    barchart.setData(barData);
    // set axis value formater
    XAxis xAxis = barchart.getXAxis();
    xAxis.setValueFormatter(new IndexAxisValueFormatter(labels));
    //set position of labels
    xAxis.setPosition(XAxis.XAxisPosition.TOP);
    xAxis.setDrawGridLines(false);
    xAxis.setDrawAxisLine(false);
    xAxis.setGranularity(1f);
    xAxis.setLabelCount(labels.size());
    xAxis.setLabelRotationAngle(270);
    barchart.animateY(2000);
    barchart.invalidate();
    }
    private void fillteachers() {
    teachersArrayList.clear();
    teachersArrayList.add(new Teachers("SCSE",38));
```

```
        teachersArrayList.add(new Teachers("SAS",20));
        teachersArrayList.add(new Teachers("SASL",23));
        teachersArrayList.add(new Teachers("SEEE",15));
        teachersArrayList.add(new Teachers("SMEC",9));
        teachersArrayList.add(new Teachers("OTHERS",6));
}}
```

4.5.2 Pie Chart

```
public class MainActivity22 extends AppCompatActivity {
private PieChart pieChart;
Button b1,b2,b3,b4,b5,b6;
@Override
protected void onCreate(Bundle savedInstanceState) {
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main22);
b1 = (Button)findViewById(R.id.button8);
b2 = (Button)findViewById(R.id.button9);
b3 = (Button)findViewById(R.id.button10);
b4 = (Button)findViewById(R.id.button11);
b5 = (Button)findViewById(R.id.button12);
b6 = (Button)findViewById(R.id.button13);
b1.setOnClickListener(new View.OnClickListener() {
@Override
 public void onClick(View view) {
Intent intent=new Intent(MainActivity22.this , MainActivity16.class);
startActivity(intent);
}
});
b2.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
Intent intent=new Intent(MainActivity22.this , MainActivity17.class);
```

```
startActivity(intent);
 }
});
b3.setOnClickListener(new View.OnClickListener() {
 @Override
public void onClick(View view) {
Intent intent=new Intent(MainActivity22.this , MainActivity18.class);
startActivity(intent);
}
});
b4.setOnClickListener(new View.OnClickListener() {
 @Override
 public void onClick(View view) {
 Intent intent=new Intent(MainActivity22.this , MainActivity19.class);
startActivity(intent);
}
});
b5.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
Intent intent=new Intent(MainActivity22.this , MainActivity20.class);
startActivity(intent);
}
});
b6.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
Intent intent=new Intent(MainActivity22.this , MainActivity21.class);
startActivity(intent);
}
});
pieChart = findViewById(R.id.pieChart);
setupPieChart();
```

```
loadPieChartData();
}
private void setupPieChart() {
pieChart.setDrawHoleEnabled(true);
pieChart.setUsePercentValues(false);
pieChart.setEntryLabelTextSize(16);
pieChart.setEntryLabelColor(Color.BLACK);
pieChart.setCenterText("VIT FACULTIES");
pieChart.setCenterTextSize(24);
pieChart.getDescription().setEnabled(false);
Legend l = pieChart.getLegend();
l.setVerticalAlignment(Legend.LegendVerticalAlignment.TOP);
l.setHorizontalAlignment(Legend.LegendHorizontalAlignment.RIGHT);
l.setOrientation(Legend.LegendOrientation.VERTICAL);
l.setDrawInside(false);
l.setEnabled(true);
}
private void loadPieChartData() {
ArrayList<PieEntry> entries = new ArrayList<>();
entries.add(new PieEntry(38, "SCSE"));
entries.add(new PieEntry(20,"SAS"));
entries.add(new PieEntry(23,"SASL"));
entries.add(new PieEntry(15,"SEEE"));
entries.add(new PieEntry(9,"SMEC"));
entries.add(new PieEntry(6,"OTHER"));
ArrayList<Integer> colors = new ArrayList<>();
for (int color: ColorTemplate.COLORFUL_COLORS) {
colors.add(color);
}
for ( int color: ColorTemplate.VORDIPLOM_COLORS) {
colors.add(color);
}
PieDataSet dataSet = new PieDataSet(entries,"TEACHERS");
```

```
dataSet.setColors(colors);
PieData data = new PieData(dataSet);
data.setDrawValues(true);
data.setValueFormatter(new PercentFormatter(pieChart));
data.setValueTextSize(14f);
data.setValueTextColor(Color.BLACK);
 pieChart.setData(data);
 pieChart.invalidate();
}
}
```

4.6 FACULTY SEARCH

```
public class MainActivity extends AppCompatActivity
{
TextView tv1, tv2;
Button b,B1,B2;
EditText E,E2;
@Override
protected void onCreate(Bundle savedInstanceState)
{
super.onCreate(savedInstanceState);
setContentView(R.layout.activity_main);
final String[] people = new String[]{"A. Sampath Kumar", "Abhay Vidyarthi",
"Abhishek Verma", "Amita Mahor", "Amrut Shrikant Mulay", "Anand Motwani",
"Anant Kant Shukla", "Anirban Bhowmick", "Anita Yadav", "Ankur Beohar",
"Ashish Kumar Sahu", "Ashish Tripathi", "Baseera A.", "Bhakti Parashar", "Bhumika
Girishkumar Choksi", "Chandan Kumar Behera", "Charles Richard", "Deepika
Masand", "Divya Haridas", "Dhruv Sharma", "Duggineni Karthik", "G L Balaji", "H.
```

Azath", "Harihara Padhy", "J. Amudhavel ", "Jitendra Kumar Tandelkar", "K. Venkatachalam", "Kanchanlata Kashyap", "Krishna Kumar", "L. Shakkeera", "M. Dinesh Babu", "Maheshwar R", "Mamta Agrawal", "Manas Kumar Mishra", "Manikandan Kalimuthu", "Manoj Acharya", "Mayank Gupta", "Muneeswaran V", "Nageswara Guptha", "Neha Choubey", "Nella Anveshkumar", "Nilamadhab Mishra", "Pallabi Sarkar", "Paras Jain", "Pon Harshavardhanan", "Pradeep Kumar Kayshap", "Praveen Lalwani", "Pritesh Vishwasrao Bansod", "Priyanka Singh", "Pushpdant Jain", "Pushpindra Singh", "Rahul Kottath", "Reena Jain", "Ribu Mathew", "Roja Rani Ilka", "Sandip Mal", "Saravanan J.", "Sathish Kumar R", "Sharad Chandra Tripathi", "Shince V Joseph", "Shishir K Shandilya", "Shriram R.", "Shubhash Chandra Patel", "Sounthar Rajan", "Suganya E.", "Sumit Mittal", "Suresh Dara", "Tushar Choudhary", "V. Pandimurugan", "Venkat Prasad Padhya", "Venkatesh T.", "Vinod Bhatt", "Virendra Kushwah", "Y. Sharmasth Vali", "Yogesh Shukla","Rakesh R"};

final String[] phoneNumbers = new String[]{"9894721222", "8765773284", "7586852491", "9425019572", "8830091723", "9894512300 ", "9345522103 ", "9547155428", "9977588551", "9893383443", "9506347438", "7579287442", "9698960667", "9826722177", "7016527953", "8847872575", "7708342512", "9425602587", "9930594727", "8535034245", "8124709555", "7974537024", "8120008102", "9668780860", "9597788658", "9406784831", "8610569054", "9826733258", "7584826299", "9701336323", "6380043405", "9655212300", "9425027637", "9956250356", "9952387877", "9131096990", "7722993939", "9842541667", "9843255706", "9713606045", "9503132874", "9437169510", "6294524861", "9963023330", "9937704985", "9389634514", "8087181373", "9933979757", "8765330670", "6301800673", "9893273243", "8591515682", "8989982847", "9003397713", "7987253273", "9583085832", "9047240141", "7667281338", "7697867027", "9938165145", "9009972032", "7358194673", "7905407827", "9786066776", "9884970535", "9318325748", "9007400537", "9752005705", "8249897681", "8310597038", "7747973581", "9826143220 ", "7415869616", "9557404420", "9479877102","9840768026"};

final String[] email = new String[]{"sampath.kumar@vitbhopal.ac.in", "abhay.vidyarthi@vitbhopal.ac.in", "abhishek.verma@vitbhopal.ac.in", "amita.mahor@vitbhopal.ac.in", "amrut.shrikant@vitbhopal.ac.in",

"anand.motwani@vitbhopal.ac.in ", "anantkant.shukla@vitbhopal.ac.in",
"anirban.bhowmick@vitbhopal.ac.in", "anita.yadav@vitbhopal.ac.in",
"ankur.beohae@vitbhopal.ac.in", "ashish.kumar@vitbhopal.ac.in",
"ashish.tripathi@vitbhopal.ac.in", "baseera.a@vitbhopal.ac.in",
"bhakti.parashar@vitbhopal.ac.in", "bhumika.choksi@vitbhopal.ac.in",
"chandank.behera@vitbhopal.ac.in", "charles.richard@vitbhopal.ac.in",
"deepika.masandvitbhopal.ac.in", "divya.haridas@vitbhopal.ac.in",
"dhruv.sharma@vitbhopal.ac.in", "duggineni.karthik@vitbhopal.ac.in",
"balajigl@vitbhopal.ac.in", "azah.h@vitbhopal.ac.in ", "dean.sasl@vitbhopal.ac.in",
"amuthavel.j@vitbhopal.ac.in", "Hitendra.kumar@vitbhopal.ac.in",
"venkatachalam.k@vitbhopal.ac.in", "kanchan.k@vitbhopal.ac.in",
"krishnakumar@vitbhopal.ac.in", "shakkeera.l@vitbhopal.ac.in ",
"dineshbabu.m@vitbhopal.ac.in", "maheswar.n@vitbhopal.ac.in",
"mamta.agrawal@vitbhopal.ac.in", "manaskumar.mishra@vitbhopal.ac.in",
"manikandan.k@vitbhopal.ac.in", "manoj.acharya@vitbhopal.ac.in",
"mayank.gupta@vitbhopal.ac.in", "muneeswaran@vitbhopal.ac.in",
"nageswara.guptha@vitbhopal.ac.in", "neha.choubey@vitbhopal.ac.in",
"nella.anveshkumar@vitbhopal.ac.in", "nilamadhab.mishra@vitbhopal.ac.in",
"pallabi.sarkar@vitbhopal.ac.in", "paras.jain@vitbhopal.ac.in",
"pon.harshavardhanan@vitbhopal.ac.in", "priyanka.singh@vitbhopal.ac.in",
"praveen.lalwani@vitbhopal.ac.in", "priteshvishwasrao.b@vitbhopal.ac.in",
"priyanka.singh@vitbhopal.ac.in", "pushpdant.jain@vitbhopal.ac.in",
"pushpinder.singh@vitbhopal.ac.in", "rahul.kottath@vitbhopal.ac.in",
"reena.jain@vitbhopal.ac.in ", "ribu.mathew@vitbhopal.ac.in",
"rojaran.ika@vitbhopal.ac.in", "sandip.mal@vitbhopal.ac.in",
"saravanan.j@vitbhopal.ac.in", "sathish.kumar@vitbhopal.ac.in",
"sharadchandra.tripathi@vitbhopal.ac.in", "shincey.joseph@vitbhopal.ac.in",
"shishirkumar.mishra@vitbhopal.ac.in", "shriram.r@vitbhopal.ac.in",
"subhath.patel@vithhopal.ac.in ", "s.sountharrajan@vitbhopal.ac.in",
"suganya.e@vitbhopal.ac.in", "sumit.mittal@vitbhopal.ac.in",
"suresh.dara@vitbhopal.ac.in", "tushar.choudhary@vitbhopal.ac.in",
"pandi.murugan@vitbhopal.ac.in", "venkat.prasad@vitbhopal.ac.in",
"venkatesh.t@vitbhopal.ac.in", "vinod.bhatt@vitbhopal.ac.in ",

```
"virendra.kushwah@vitbhopal.ac.in", "sharmasth.vali@vithbopal.ac.in ",
"yogesh.shukla@vitbhopal.ac.in","rakesh.r@vitbhopal.ac.in"};
ActivityCompat.requestPermissions(MainActivity.this,new
String[]{Manifest.permission.SEND_SMS},
PackageManager.PERMISSION_GRANTED);
ActivityCompat.requestPermissions(MainActivity.this,new
String[]{Manifest.permission.CALL_PHONE},1);
E = (EditText) findViewById(R.id.editText);
E2 = (EditText) findViewById(R.id.editText2);
tv1 = (TextView) findViewById(R.id.textView1);
tv2 = (TextView) findViewById(R.id.textView2);
b = (Button) findViewById(R.id.button);
B1 = (Button) findViewById(R.id.button2);
B2 = (Button) findViewById(R.id.button3);
b.setOnClickListener(new View.OnClickListener()
{
@Override
public void onClick(View view)
{
for (int i = 0; i < people.length; i++) {
if (people[i].equals(E.getText().toString())) {
tv1.setText(phoneNumbers[i]);
tv2.setText(email[i]);
}}}});
B2.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
for (int i = 0; i < people.length; i++) {
if (people[i].equals(E.getText().toString())) {
SmsManager smsManager = SmsManager.getDefault();
smsManager.sendTextMessage(phoneNumbers[i], null,
E2.getText().toString(),null,null);
}}}});
```

```
B1.setOnClickListener(new View.OnClickListener() {
@Override
public void onClick(View view) {
for (int i = 0; i < people.length; i++) {
if (people[i].equals(E.getText().toString())) {
String number1=phoneNumbers[i];
String dial="tel:"+number1;
startActivity(new Intent(Intent.ACTION_CALL, Uri.parse(dial)));
}}}});}}
```

OBSERVATION

Figure 2: GUI of CURRICULUM

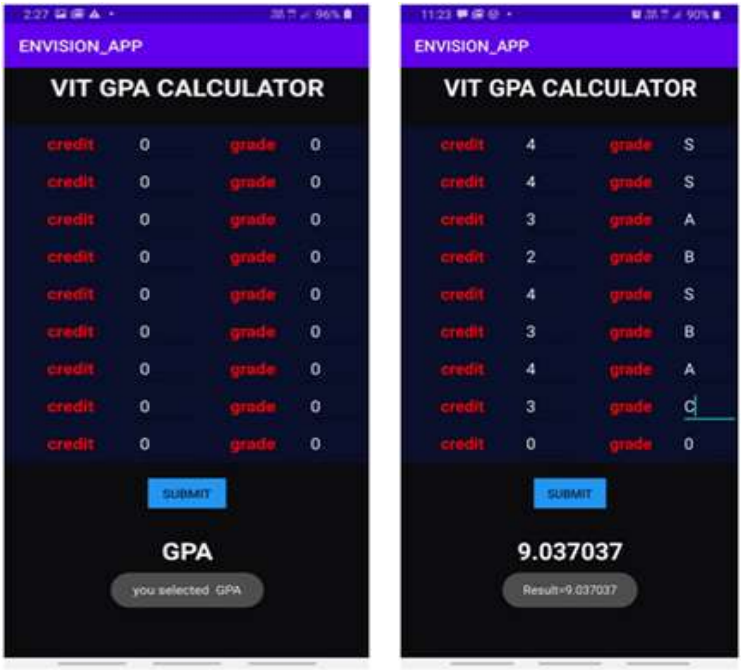


Figure 3: GUI of GPA CALCULATOR

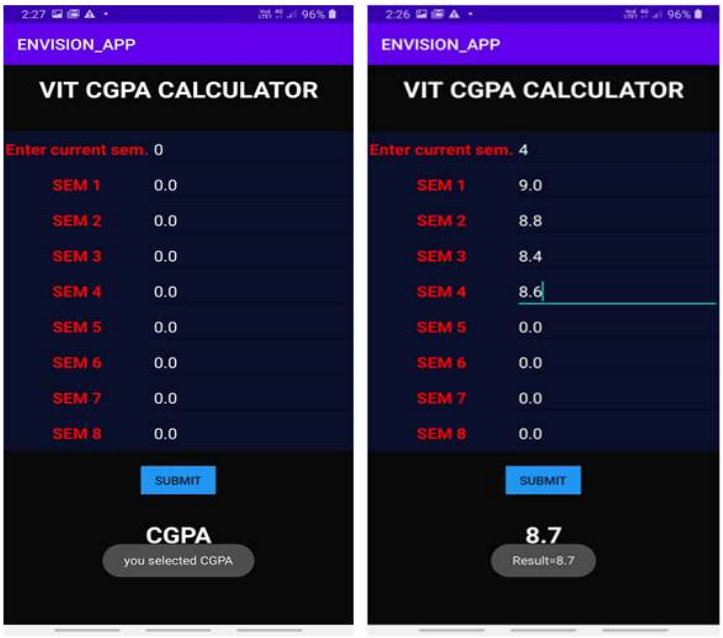


Figure 4: GUI of CGPA CALCULATOR

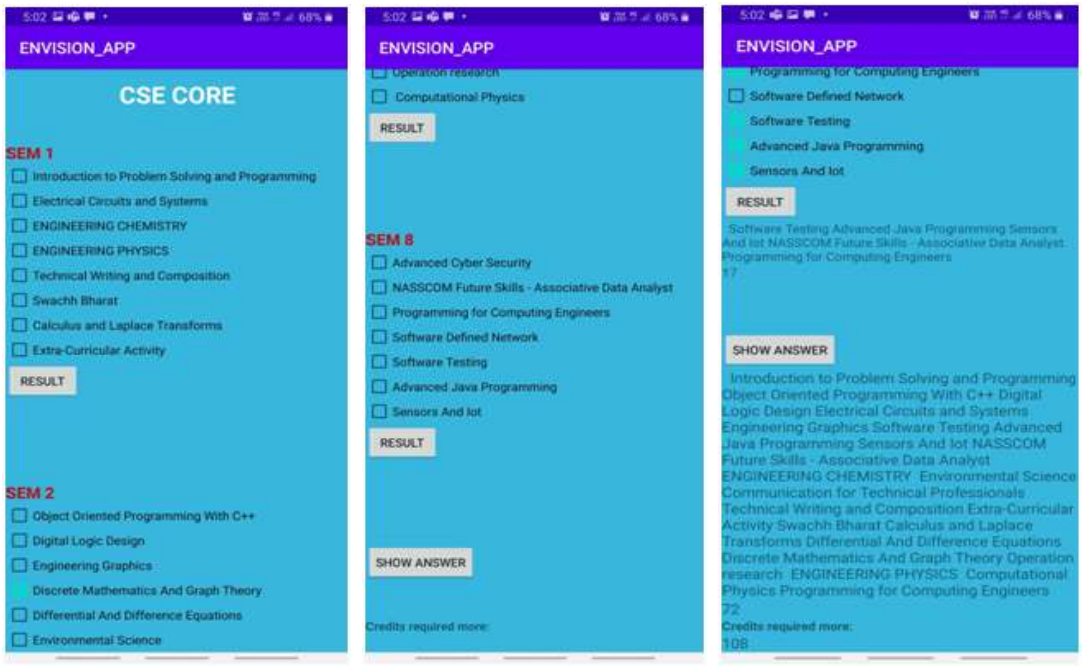


Figure 5: GUI of CREDITS Calculation

Figure 6: GUI of FACULTY Slots

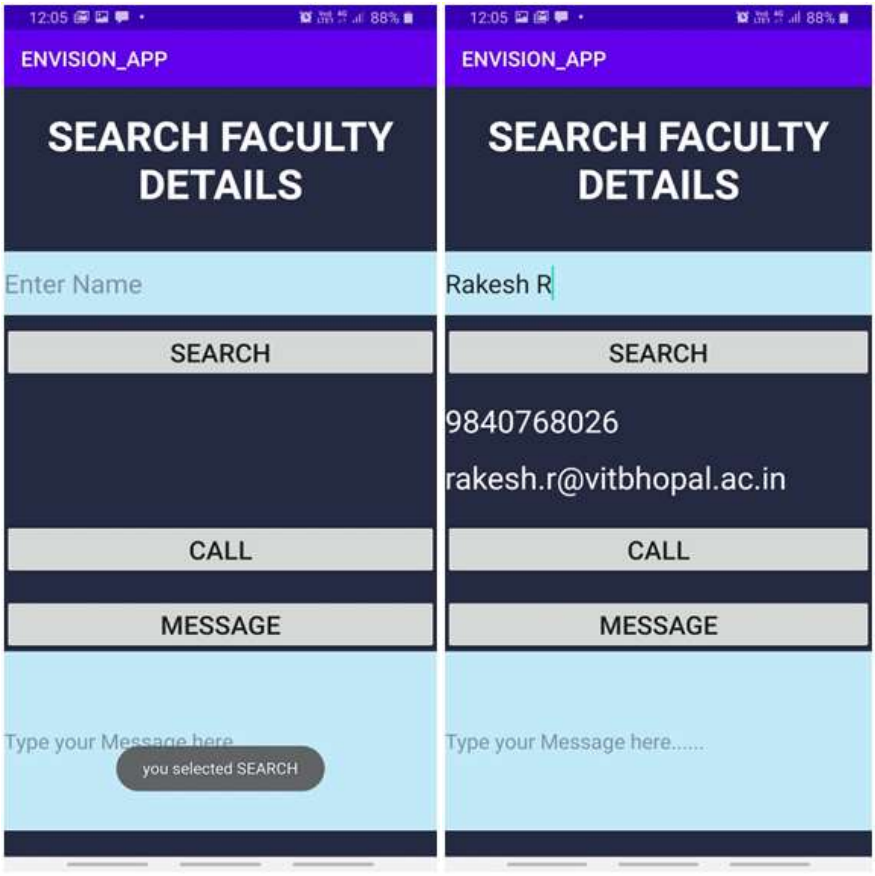


Figure 7: GUI of SEARCH Faculty

FUTURE ENHANCEMENT AND CONCLUSION

5.1 LIMITATIONS OF EXISTING WORK

- **Gives assessment not exactness**

I. While the information is exact in foreseeing the circumstances, the perception of similar just gives the assessment.

II. It without a doubt is anything but difficult to change over the robust and protracted information into simple pictorial configuration yet such a portrayal of data may prompt theoretical ends now and then.

- **One – sided**

I. The essential arrangement of information representation occurs with the human interface, which means the information that turns out to be the base of perception can be one-sided.

II. The individual bringing the information for the equivalent may just think about the significant part of the information or the information that requirements center and may reject the remainder of the information which may prompt one-sided results.

- **Absence of help**

I. One of the downsides of information perception is that it can't help, which means an alternate gathering of the crowd may decipher it in an unexpected way.

- **Inappropriate plan issue**

I. On the off chance that information perception is viewed as such a correspondence. At that point, it must be certifiable in clarifying the reason.

II. In the event that the plan isn't legitimate, at that point, this can prompt disarray in correspondence.

- **Wrong engaged individuals can skip center messages**

I. One of the issues with information perception is however it could be logical its clearness in clarification is totally subject to the focal point of its crowd.

5.2 FUTURE SCOPE OF DATA VISUALIZATION

Far more than ever has been achieved in the discipline of data visualization, as we are seeing cheaper and more reliable ways to store data. This is the reason why careers like data scientist are booming.

Data visualization is the obvious answer to turning all that data into actual usable information that will drive a business forward.

Data visualization is no more art. With emerging cognitive frameworks, multidimensional imaging and intelligence; data visualization is opening new horizons in being able to visualize large chunks of complex data.

Data visualization helps data to be understood in visually interactive forms such as patterns, correlations, graphics and so on. It delivers a better understanding of the business states and in developing patterns that provides solutions and insights.

5.3 CONCLUSION

- This report started with a brief introduction of data visualization including the objectives and brief of existing work.
- Later we discussed about our proposed work which consisted of our proposed approach on ENVISION app and the methodology used.
- We also discussed the software hardware used for bringing the idea into practical work and its advantages.
- A detailed explanation of how data visualization works on ENVISION was given through codes, an architecture diagram and GUI of our program.
- At last, we have concluded with the limitations of our work and scope of data visualization in future.