MALLA REDDY UNIVERSITY
(Telangana State Private Universities Act No. 13 of 2020 &
G.O.Ms. No. 14, Higher Education (UE) Department)
Maisammaguda, Kompally,
Medchal - Malkajgiri District
Hyderabad - 500100,
Telangana State.
www.mallareddyuniversity.

Department Of Computer Science And Engineering

Mobile Application Development Project

Money Mapper

Name: Subhapreet Patro

Roll number: 2211CS010547

Yr/Sec: Group-3(III-year)

XML Code:

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:app="http://schemas.android.com/apk/res-auto"
   android:layout_width="match_parent"
   android:layout_height="match_parent"
   android:padding="20dp"
   android:background="@drawable/gradient_background">

   <LinearLayout
      android:id="@+id/title_section"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:orientation="horizontal"
      android:gravity="center_vertical"
      android:layout_marginBottom="16dp"
      android:layout_marginTop="150dp"
      app:layout_constraintTop_toTopOf="parent"
      app:layout_constraintStart_toStartOf="parent"
      app:layout_constraintEnd_toEndOf="parent">

      <ImageView
         android:layout_width="48dp"
         android:layout_height="48dp"
         android:src="@drawable/ic_logo"
         android:contentDescription="App Logo"
         android:layout_marginEnd="10dp" />

      <TextView
         android:layout_width="wrap_content"
         android:layout_height="wrap_content"
         android:text="Money Mapper"
         android:textSize="24sp"
         android:textStyle="bold"
         android:textColor="@color/black"/>
   </LinearLayout>

   <androidx.cardview.widget.CardView
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      app:cardCornerRadius="16dp"
```

```
app:cardElevation="6dp"
android:layout_marginTop="15dp"
android:padding="16dp"
app:cardBackgroundColor="@color/white"
app:layout_constraintTop_toBottomOf="@id/title_section"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintEnd_toEndOf="parent">

<LinearLayout
   android:layout_width="match_parent"
   android:layout_height="wrap_content"
   android:orientation="vertical"
   android:padding="10dp">

   <LinearLayout
      android:layout_width="match_parent"
      android:layout_height="wrap_content"
      android:orientation="horizontal"
      android:gravity="center"
      android:paddingBottom="10dp">

      <Spinner
         android:id="@+id/from_currency"
         android:layout_width="0dp"
         android:layout_height="34dp"
         android:layout_weight="1"
         android:background="@drawable/spinner_background" />

      <ImageView
         android:id="@+id/swap_button"
         android:layout_width="48dp"
         android:layout_height="48dp"
         android:src="@drawable/ic_exchange"
         app:tint="@color/black"
         android:layout_marginHorizontal="16dp"
         android:clickable="true"
         android:focusable="true"
         android:contentDescription="Swap Currencies"/>


      <Spinner
         android:id="@+id/to_currency"
         android:layout_width="0dp"
         android:layout_height="34dp"
```

```
            android:layout_weight="1"
            android:background="@drawable/spinner_background" />
        </LinearLayout>

        <com.google.android.material.textfield.TextInputLayout
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:hint="Enter amount"
          android:layout_marginTop="16dp">

          <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/amount_input"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textColor="@color/black"
            android:inputType="numberDecimal" />
        </com.google.android.material.textfield.TextInputLayout>

        <Button
          android:id="@+id/convert_button"
          android:layout_width="match_parent"
          android:layout_height="wrap_content"
          android:text="Convert"
          android:textSize="16sp"
          android:layout_marginTop="16dp"
          android:backgroundTint="@color/primary_color"
          android:textColor="@color/white"
          android:padding="12dp"
          android:drawableStart="@drawable/ic_exchange" />
      </LinearLayout>
    </androidx.cardview.widget.CardView>

</androidx.constraintlayout.widget.ConstraintLayout>
```

activity_result.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    android:padding="20dp"
```

```
android:background="@drawable/gradient_background">

<TextView
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:textSize="28sp"
   android:text="Converted Amount"
   android:textStyle="bold"
   android:textColor="@color/black"/>

<TextView
   android:id="@+id/result_text"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:textSize="24sp"
   android:text="Converted Amount"
   android:textStyle="bold"
   android:textColor="@color/black"/>
<RadioGroup
   android:id="@+id/time_filter_group"
   android:layout_width="wrap_content"
   android:layout_height="wrap_content"
   android:orientation="horizontal"
   android:layout_gravity="center"
   android:layout_marginTop="10dp"
   android:layout_marginBottom="7dp">

   <RadioButton
      android:id="@+id/rb_1d"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="1D"
      android:checked="true"
      android:textColor="@android:color/black" />

   <RadioButton
      android:id="@+id/rb_5d"
      android:layout_width="wrap_content"
      android:layout_height="wrap_content"
      android:text="5D"
      android:textColor="@android:color/black"/>

   <RadioButton
      android:id="@+id/rb_1m"
```

```
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="1M"
            android:textColor="@android:color/black"/>

        <RadioButton
            android:id="@+id/rb_1y"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="1Y"
            android:textColor="@android:color/black"/>

        <RadioButton
            android:id="@+id/rb_5y"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="5Y"
            android:textColor="@android:color/black"/>
    </RadioGroup>

    <com.github.mikephil.charting.charts.LineChart
        android:id="@+id/line_chart"
        android:layout_width="match_parent"
        android:layout_height="310dp"
        android:layout_marginTop="20dp"
        android:background="@android:color/transparent"/>

    <Button
        android:id="@+id/back_button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Go Back"
        android:layout_marginTop="20dp"
        android:backgroundTint="@color/primary_color"
        android:textColor="@color/white"/>
</LinearLayout>
```

Java Code:

MainActivity.java:

```
package com.example.currencyconverter;

import android.content.Intent;
import android.os.AsyncTask;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import org.json.JSONObject;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;

public class MainActivity extends AppCompatActivity {

    private Spinner fromCurrency, toCurrency;
    private EditText amountInput;
    private TextView resultText;
    private Button convertButton;
    private ImageView swapButton;
    private JSONObject exchangeRates;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        fromCurrency = findViewById(R.id.from_currency);
        toCurrency = findViewById(R.id.to_currency);
```

```
        amountInput = findViewById(R.id.amount_input);
        convertButton = findViewById(R.id.convert_button);
        swapButton = findViewById(R.id.swap_button);

        new FetchExchangeRates().execute("https://open.er-api.com/v6/latest/USD");

        convertButton.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                convertCurrency();
            }
        });

        swapButton.setOnClickListener(v -> {
            int fromPosition = fromCurrency.getSelectedItemPosition();
            int toPosition = toCurrency.getSelectedItemPosition();

            fromCurrency.setSelection(toPosition);
            toCurrency.setSelection(fromPosition);
        });
    }

    private class FetchExchangeRates extends AsyncTask<String, Void, String> {
        @Override
        protected String doInBackground(String... urls) {
            try {
                URL url = new URL(urls[0]);
                HttpURLConnection connection = (HttpURLConnection) url.openConnection();
                connection.setRequestMethod("GET");

                BufferedReader reader = new BufferedReader(new
InputStreamReader(connection.getInputStream()));
                StringBuilder response = new StringBuilder();
                String line;

                while ((line = reader.readLine()) != null) {
                    response.append(line);
                }

                reader.close();
                return response.toString();

            } catch (Exception e) {
                Log.e("API Error", "Failed to fetch exchange rates", e);
```

```
            return null;
        }
    }

    @Override
    protected void onPostExecute(String result) {
        if (result == null) {
            resultText.setText("Error: Unable to fetch exchange rates");
            return;
        }

        try {
            Log.d("API Response", result);
            JSONObject jsonResponse = new JSONObject(result);

            if (!jsonResponse.has("rates")) {
                resultText.setText("Error: Invalid API response");
                return;
            }

            exchangeRates = jsonResponse.getJSONObject("rates");

            List<String> currencyList = new ArrayList<>();
            Iterator<String> keys = exchangeRates.keys();

            while (keys.hasNext()) {
                currencyList.add(keys.next());
            }

            ArrayAdapter<String> adapter = new ArrayAdapter<>(MainActivity.this,
android.R.layout.simple_spinner_item, currencyList);

adapter.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);
            fromCurrency.setAdapter(adapter);
            toCurrency.setAdapter(adapter);
        } catch (Exception e) {
            Log.e("JSON Error", "Failed to parse API response", e);
            resultText.setText("Error: Failed to process exchange rates");
        }
    }
}
private void convertCurrency() {
    try {
        if (exchangeRates == null) {
```

```
            resultText.setText("Error: Exchange rates not loaded");
            return;
        }

        String from = fromCurrency.getSelectedItem().toString();
        String to = toCurrency.getSelectedItem().toString();
        String amountStr = amountInput.getText().toString();

        if (amountStr.isEmpty()) {
            Toast.makeText(MainActivity.this, "Please enter an amount",
Toast.LENGTH_SHORT).show();
            return;
        }

        double amount = Double.parseDouble(amountStr);

        if (!exchangeRates.has(from) || !exchangeRates.has(to)) {
            resultText.setText("Error: Invalid currency selection");
            return;
        }

        double fromRate = exchangeRates.getDouble(from);
        double toRate = exchangeRates.getDouble(to);

        double convertedAmount = (amount / fromRate) * toRate;
        String result = String.format("%.2f %s", convertedAmount, to);

        Intent intent = new Intent(MainActivity.this, ResultActivity.class);
        intent.putExtra("converted_amount", result);
        startActivity(intent);

      } catch (NumberFormatException e) {
            Toast.makeText(MainActivity.this, "Invalid amount entered",
Toast.LENGTH_SHORT).show();
      } catch (Exception e) {
            Log.e("Conversion Error", "Failed to convert currency", e);
            Toast.makeText(MainActivity.this, "Conversion failed",
Toast.LENGTH_SHORT).show();
      }
   }
}
```

ResultActivity.java:

```
package com.example.currencyconverter;

import android.graphics.Color;
import android.graphics.drawable.Drawable;
import android.os.Bundle;
import android.widget.Button;
import android.widget.RadioGroup;
import android.widget.TextView;
import androidx.appcompat.app.AppCompatActivity;
import com.github.mikephil.charting.charts.LineChart;
import com.github.mikephil.charting.components.Legend;
import com.github.mikephil.charting.components.XAxis;
import com.github.mikephil.charting.components.YAxis;
import com.github.mikephil.charting.data.Entry;
import com.github.mikephil.charting.data.LineData;
import com.github.mikephil.charting.data.LineDataSet;
import com.github.mikephil.charting.utils.Utils;
import java.util.ArrayList;

public class ResultActivity extends AppCompatActivity {
   private LineChart lineChart;
   private RadioGroup timeFilterGroup;

   @Override
   protected void onCreate(Bundle savedInstanceState) {
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_result);

      TextView resultText = findViewById(R.id.result_text);
      Button backButton = findViewById(R.id.back_button);
      lineChart = findViewById(R.id.line_chart);
      timeFilterGroup = findViewById(R.id.time_filter_group);

      String convertedAmount = getIntent().getStringExtra("converted_amount");
      resultText.setText(convertedAmount);

      setupChart(getMockData("1M"));

      timeFilterGroup.setOnCheckedChangeListener((group, checkedId) -> {
         String selectedTimeFrame;
         if (checkedId == R.id.rb_1d) {
            selectedTimeFrame = "1D";
```

```
        } else if (checkedId == R.id.rb_5d) {
            selectedTimeFrame = "5D";
        } else if (checkedId == R.id.rb_1m) {
            selectedTimeFrame = "1M";
        } else if (checkedId == R.id.rb_1y) {
            selectedTimeFrame = "1Y";
        } else {
            selectedTimeFrame = "5Y";
        }
        setupChart(getMockData(selectedTimeFrame));
    });

    backButton.setOnClickListener(v -> finish());
}

private void setupChart(ArrayList<Entry> dataValues) {
    LineDataSet lineDataSet = new LineDataSet(dataValues, "Currency Trend");
    lineDataSet.setColor(Color.parseColor("#FF6D72"));
    lineDataSet.setLineWidth(2f);
    lineDataSet.setDrawCircles(false);
    lineDataSet.setDrawValues(false);
    lineDataSet.setMode(LineDataSet.Mode.CUBIC_BEZIER);
    lineDataSet.setDrawFilled(true);

    if (Utils.getSDKInt() >= 18) {
        Drawable gradient = getResources().getDrawable(R.drawable.gradient_fill);
        lineDataSet.setFillDrawable(gradient);
    } else {
        lineDataSet.setFillColor(Color.parseColor("#FF6D72"));
    }

    LineData lineData = new LineData(lineDataSet);
    lineChart.setData(lineData);
    lineChart.invalidate();

    lineChart.setDrawGridBackground(false);
    lineChart.getDescription().setEnabled(false);
    lineChart.setTouchEnabled(true);
    lineChart.setPinchZoom(true);
    lineChart.setScaleEnabled(true);
    lineChart.setBackgroundColor(Color.BLACK);

    XAxis xAxis = lineChart.getXAxis();
    xAxis.setPosition(XAxis.XAxisPosition.BOTTOM);
```

```
    xAxis.setTextColor(Color.LTGRAY);
    xAxis.setDrawGridLines(false);
    xAxis.setLabelCount(6, true);


    YAxis leftAxis = lineChart.getAxisLeft();
    leftAxis.setTextColor(Color.LTGRAY);
    leftAxis.setDrawGridLines(true);
    leftAxis.enableGridDashedLine(10f, 10f, 0f);
    lineChart.getAxisRight().setEnabled(false);

    Legend legend = lineChart.getLegend();
    legend.setTextColor(Color.LTGRAY);
    legend.setEnabled(true);
}

private ArrayList<Entry> getMockData(String timeFrame) {
    ArrayList<Entry> data = new ArrayList<>();

    switch (timeFrame) {
        case "1D":
            data.add(new Entry(0, 86.2f));
            data.add(new Entry(6, 86.5f));
            data.add(new Entry(12, 86.7f));
            data.add(new Entry(18, 86.4f));
            data.add(new Entry(24, 86.5f));
            break;

        case "5D":
            for (int i = 0; i <= 5; i++) {
                data.add(new Entry(i, 86.0f + (float) Math.random()));
            }
            break;

        case "1M":
            for (int i = 0; i <= 30; i += 5) {
                data.add(new Entry(i, 86.5f + (float) Math.random()));
            }
            break;

        case "1Y":
            for (int i = 0; i <= 12; i++) {
                data.add(new Entry(i, 85.5f + (float) Math.random() * 2));
            }
```

```
            break;

          case "5Y":
            for (int i = 0; i <= 5; i++) {
              data.add(new Entry(i, 84.0f + (float) Math.random() * 3));
            }
            break;
        }
        return data;
    }
}
```

Output:

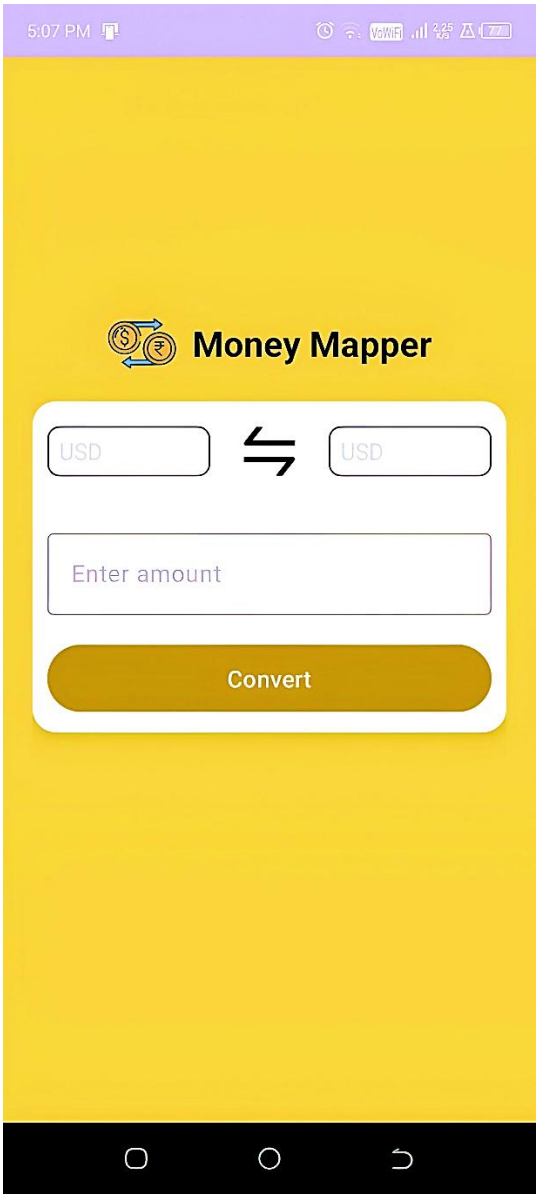

Figure 1: Default Screen

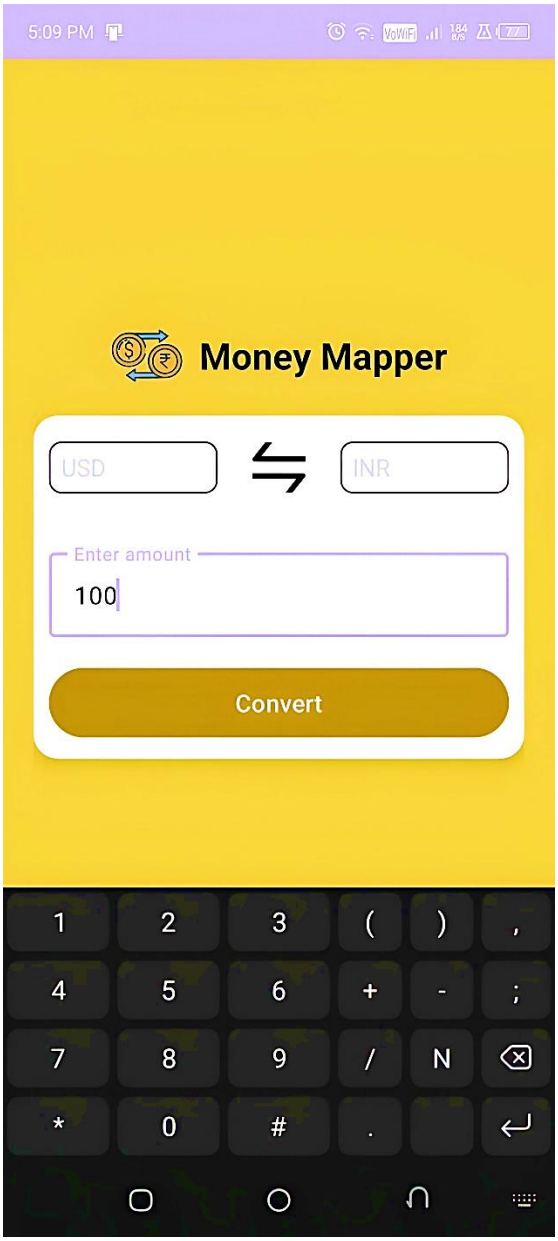

Figure 2: Entering Amount

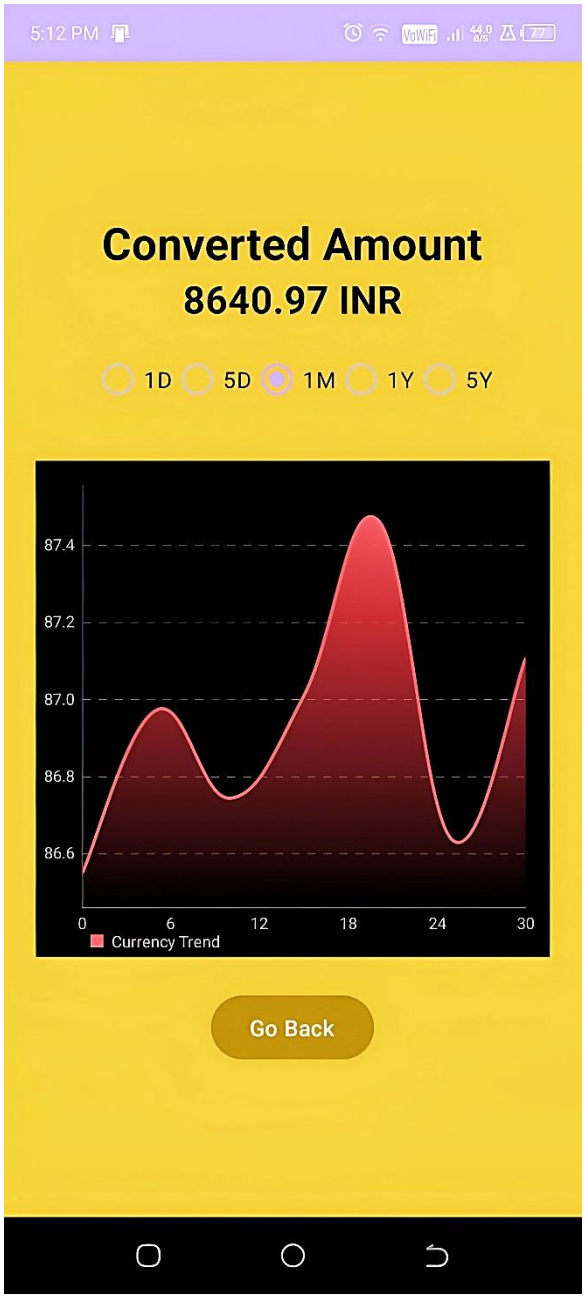

Figure 3: Conversion with graph