# VISVESVARAYA TECHNOLOGICAL UNIVERSITY
# BELAGAVI

*Mini Project Report on*

## "MeAn - Message Analyzer Application"

*Submitted in the partial fulfillment for the requirements of Mobile Application Development of 6th semester CSE requirement in the form of the Mini Project work*

*Submitted By*

**LIKITH S**                          USN: 1BY18CS081

**DEEP RAKESH TANK**          USN: 1BY18CS225

*Under the guidance of*

Dr. Vishwa Kiran S                          Mrs. Srivani P
Assistant Professor, BMSIT&M          Assistant Professor, BMSIT&M

Mrs. Vidya R                          Dr. Dhanalakshmi B K
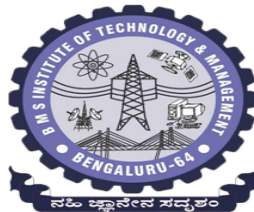Assistant Professor, BMSIT&M          Assistant Professor, BMSIT&M

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

# BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT
## YELAHANKA, BENGALURU - 560064.

2020-2021

# BMS INSTITUTE OF TECHNOLOGY & MANAGEMENT
## YELAHANKA, BENGALURU – 560064

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING



# CERTIFICATE

This is to certify that the Project work entitled **"MeAn - Message Analyzer Application"** is a bonafide work carried out by **Likith S (1BY18CS081) and Deep Rakesh Tank (1BY18CS225)** in partial fulfillment for *Mini Project* during the year 2020-2021. It is hereby certified that this project covers the concepts of *Mobile Application Development*. It is also certified that all corrections/suggestions indicated for Internal Assessment have been incorporated in this report.

**Signature of HOD with date**
Dr. BHUVANESHWARI.C.M

Prof & Head
CSE, BMSIT&M

**Signature of the Guide
with date**
Dr. Vishwa Kiran S
Assistant Professor
AIML, BMSIT&M

**Signature of the Guide
with date**
Mrs. Srivani P
Assistant Professor
CSE, BMSIT&M

**Signature of the Guide
with date**
Mrs. Vidya R
Assistant Professor
CSE, BMSIT&M

**Signature of the Guide
with date**
Dr. Dhanalakshmi B K
Assistant Professor
CSE, BMSIT&M

## INSTITUTE VISION

To emerge as one of the finest technical institutions of higher learning, to develop engineering professionals who are technically competent, ethical and environment friendly for betterment of the society.

## INSTITUTE MISSION

Accomplish stimulating learning environment through high quality academic instruction, innovation and industry-institute interface.

## DEPARTMENT VISION

To develop technical professionals acquainted with recent trends and technologies of computer science to serve as valuable resource for the nation/society.

## DEPARTMENT MISSION

Facilitating and exposing the students to various learning opportunities through dedicated academic teaching, guidance and monitoring.

## PROGRAM EDUCATIONAL OBJECTIVES

1. Lead a successful career by designing, analysing and solving various problems in the field of Computer Science & Engineering.
2. Pursue higher studies for enduring edification.
3. Exhibit professional and team building attitude along with effective communication.
4. Identify and provide solutions for sustainable environmental development.

# ACKNOWLEDGEMENT

# ABSTRACT

Natural language processing (NLP) is the ability of a computer program to understand human language as it is spoken and written - referred to as natural language. It is a component of artificial intelligence (AI).

NLP has existed for more than 50 years and has roots in the field of linguistics. It has a variety of real-world applications in a number of fields, including medical research, search engines, and business intelligence.

NLP enables computers to understand natural language as humans do. Whether the language is spoken or written, natural language processing uses artificial intelligence to take real-world input, process it, and make sense of it in a way a computer can understand. Just as humans have different sensors such as ears to hear and eyes to see, computers have programs to read and microphones to collect audio. And just as humans have a brain to process that input, computers have a program to process their respective inputs. At some point in processing, the input is converted to code that the computer can understand.

There are two main phases to natural language processing: data preprocessing and algorithm development.

Data preprocessing involves preparing and "cleaning" text data for machines to be able to analyze it. Preprocessing puts data in the workable form and highlights features in the text that an algorithm can work with.

# TABLE OF CONTENTS

1. ACKNOWLEDGEMENT

2. ABSTRACT

3. TABLE OF CONTENTS

# CHAPTER 1

# INTRODUCTION

## 1.1 Brief Introduction

A mobile application also referred to as a mobile app or simply an app is a computer program or software application designed to run on a mobile device such as a phone, tablet, or watch. Mobile applications often stand in contrast to desktop applications which are designed to run on desktop computers, and web applications that run in mobile web browsers rather than directly on the mobile device.

Apps were originally intended for productivity assistance such as email, calendar, and contact databases, but the public demand for apps caused rapid expansion into other areas such as mobile games, factory automation, GPS and location-based services, order-tracking, and ticket purchases so that there are now millions of apps available. Many apps require Internet access. Apps are generally downloaded from app stores, which are a type of digital distribution platform.

## 1.2 Motivation

Human language is astoundingly complex and diverse. We express ourselves in infinite ways, both verbally and in writing. Not only are there hundreds of languages and dialects, but within each language is a unique set of grammar and syntax rules, terms, and slang. When we write, we often misspell or abbreviate words, or omit punctuation. While supervised and unsupervised learning, and specifically deep learning, are now widely used for modeling human language, there's also a need for syntactic and semantic understanding and domain expertise that are not necessarily present in these machine learning approaches. NLP is important because it helps resolve ambiguity in language and adds useful numeric structure to the data for many downstream applications, such as speech recognition or text analytics.

## 1.3 Problem Statement

The aim of this application is to show a basic implementation of natural language processing. The message will be analyzed and the detected sentiments will be displayed in the form of a pie chart.

## 1.4 Android Operating System

Android is a mobile operating system based on a modified version of the Linux kernel and other open-source software, designed primarily for touchscreen mobile devices such as smartphones and tablets. Android is developed by a consortium of developers known as the Open Handset Alliance and commercially sponsored by Google. It was unveiled in November 2007, with the first commercial Android device, the HTC Dream, being launched in September 2008.

It is free and open-source software; its source code is known as Android Open Source Project (AOSP), which is primarily licensed under the Apache License. However, most Android devices ship with additional proprietary software pre-installed, most notably Google Mobile Services (GMS) which includes core apps such as Google Chrome, the digital distribution platform Google Play and the associated Google Play Services development platform.

About 70 percent of Android smartphones run Google's ecosystem; some with vendor-customized user interface and software suites, such as TouchWiz and later One UI by Samsung, and HTC Sense. Competing Android ecosystems and forks include Fire OS (developed by Amazon) or LineageOS. However, the "Android" name and logo are trademarks of Google which impose standards to restrict "uncertified" devices outside their ecosystem to use Android branding.

The source code has been used to develop variants of Android on a range of other electronics, such as game consoles, digital cameras, portable media players, PCs, and others, each with a specialized user interface. Some well-known derivatives include Android TV for televisions and Wear OS for wearables, both developed by Google. Software packages on Android, which use the APK format, are generally distributed through proprietary application stores

like Google Play Store, Samsung Galaxy Store, Huawei AppGallery, Cafe Bazaar, and GetJar, or open-source platforms like Aptoide or F-Droid.

Android has been the best-selling OS worldwide on smartphones since 2011 and on tablets since 2013. As of May 2021, it has over three billion monthly active users, the largest installed base of any operating system, and as of January 2021, the Google Play Store features over 3 million apps. The current stable version is Android 11, released on September 8, 2020.

## 1.5 Android Applications

An Android app is a software application running on the Android platform. Because the Android platform is built for mobile devices, a typical Android app is designed for a smartphone or a tablet PC running on the Android OS.

Although an Android app can be made available by developers through their websites, most Android apps are uploaded and published on the Android Market, an online store dedicated to these applications. The Android Market features both free and priced apps.

Android apps are written in the Java programming language and use Java core libraries. They are first compiled to Dalvik executables to run on the Dalvik virtual machine, which is a virtual machine specially designed for mobile devices.

Developers may download the Android software development kit (SDK) from the Android website. The SDK includes tools, sample code, and relevant documents for creating Android apps.

Novice developers who simply want to play around with Android programming can make use of the App Inventor. Using this online application, a user can construct an Android app as if putting together pieces of a puzzle.

# Chapter 2

# LITERATURE SURVEY

## 2.1 History of Mobile Applications

Most mobile devices are sold with several apps bundled as pre-installed software, such as a web browser, email client, calendar, mapping program, and an app for buying music, other media, or more apps. Some pre-installed apps can be removed by an ordinary uninstall process, thus leaving more storage space for desired ones. Where the software does not allow this, some devices can be rooted to eliminate the undesired apps.

Apps that are not preinstalled are usually available through distribution platforms called app stores. These may be operated by the owner of the device's mobile operating system, such as the App Store (iOS) or Google Play Store; by the device manufacturers, such as the Galaxy Store and Huawei AppGallery; or by third parties, such as the Amazon Appstore and F-Droid.

Usually, they are downloaded from the platform to a target device, but sometimes they can be downloaded to laptops or desktop computers. Apps can also be installed manually, for example by running an Android application package on Android devices.

Some apps are freeware, while others have a price, which can be upfront or a subscription. Some apps also include microtransactions and/or advertising. In any case, the revenue is usually split between the application's creator and the app store. The same app can, therefore, cost a different price depending on the mobile platform.

The official US Army iPhone app presents the service's technology news, updates, and media in a single place

Mobile apps were originally offered for general productivity and information retrieval, including email, calendar, contacts, the stock market, and weather information. However, public demand and the availability of developer tools drove rapid expansion into other

categories, such as those handled by desktop application software packages. As with other software, the explosion in the number and variety of apps made discovery a challenge, which in turn led to the creation of a wide range of review, recommendation, and curation sources, including blogs, magazines, and dedicated online app-discovery services. In 2014 government regulatory agencies began trying to regulate and curate apps, particularly medical apps. Some companies offer apps as an alternative method to deliver content with certain advantages over an official website.

With a growing number of mobile applications available at app stores and the improved capabilities of smartphones, people are downloading more applications to their devices. Usage of mobile apps has become increasingly prevalent across mobile phone users. A May 2012 comScore study reported that during the previous quarter, more mobile subscribers used apps than browsed the web on their devices: 51.1% vs. 49.8% respectively. Researchers found that usage of mobile apps strongly correlates with user context and depends on the user's location and time of the day. Mobile apps are playing an ever-increasing role within healthcare and when designed and integrated correctly can yield many benefits.

Market research firm Gartner predicted that 102 billion apps would be downloaded in 2013 (91% of them free), which would generate $26 billion in the US, up 44.4% on 2012's US$18 billion. By Q2 2015, the Google Play and Apple stores alone generated $5 billion. An analyst report estimates that the app economy creates revenues of more than €10 billion per year within the European Union, while over 529,000 jobs have been created in 28 EU states due to the growth of the app market.

## 2.2 Related Work

MonkeyLearn, Lexalytics, Brandwatch, Social Searcher, MeaningCloud, Talkwalker's Quick Search, Rosette, Repustate, Clarabridge, Aylien, Social Mention, Critical Mention, Awario, Hootsuite Insights are some of the commercially available sentiment analysis tools in the market

# Chapter 3

# SYSTEM REQUIREMENTS

## 3.1 Software Requirements

Software requirements deal with defining software resource requirements and prerequisites that need to be installed on a computer to provide optimal functioning of an application.

The following are the software requirements for the application:

· Operating System: Android OS 7.1 & above

· Development Environment: Android Studio

## 3.2 Hardware Requirements

The most common set of requirements defined by any operating system or software application is the physical computer resources, also known as hardware.

· RAM: minimum 2GB (>2GB recommended)

· Secondary Storage: 20Mb

# Chapter 4

# SYSTEM DESIGN

## Flowchart

A flowchart is a visual representation of the sequence of steps and decisions needed to perform a process. Each step in the sequence is noted within a diagram shape. Steps are linked by connecting lines and directional arrows. The flowchart shown depicts how the proposed system works in different steps sequentially.
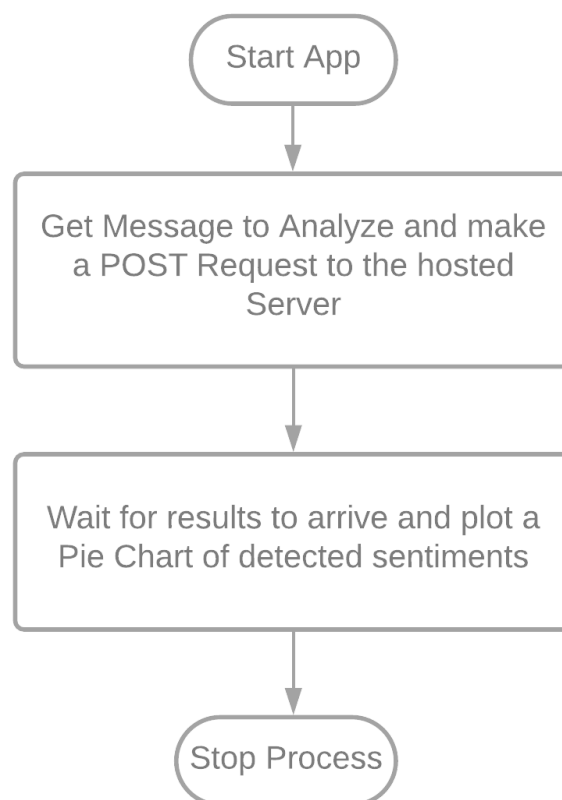
# Chapter 5

# IMPLEMENTATION

## 5.1 Java Code

### 5.1.1 IntroActivty.java

```java
package in.bmsit.sixthsem.mean;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;

import android.os.Bundle;

import android.os.Handler;

import android.view.WindowManager;

public class IntroActivity extends AppCompatActivity {

    Handler h = new Handler();

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_intro);

        getSupportActionBar().hide();//hide action bar


getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManager.LayoutParams.FLAG_FULLSCREEN); //hide status bar

        h.postDelayed(func,3000);

    }

    private final Runnable func = new Runnable() {

        @Override

        public void run() {
```

```java
        openMain();

    }

};

public void openMain() {

    Intent openmain = new Intent(IntroActivity.this,
MainActivity.class);

    startActivity(openmain);

    finish();

}

}
```

## 5.1.2 MainActivty.java

```java
package in.bmsit.sixthsem.mean;

import android.app.ProgressDialog;

import android.content.ClipData;

import android.content.ClipboardManager;

import android.content.Context;

import android.content.Intent;

import android.os.AsyncTask;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.view.inputmethod.InputMethodManager;

import android.widget.Button;

import android.widget.EditText;

import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.android.volley.Request;

import com.android.volley.RequestQueue;
```

```java
import com.android.volley.Response;

import com.android.volley.toolbox.JsonObjectRequest;

import com.android.volley.toolbox.Volley;

import org.json.JSONException;

import org.json.JSONObject;


public class MainActivity extends AppCompatActivity {

    EditText ptxt;

    Button btnpst, bClr, bAnlz;

    ClipboardManager clipboardManager;

    ProgressDialog p;

    String resultData = null;

    int count = 0;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_main);

        getSupportActionBar().hide();//hide action bar

        bAnlz = findViewById(R.id.bAnalyse);

        bClr = findViewById(R.id.bClear);

        ptxt = findViewById(R.id.txtShow);

        btnpst = findViewById(R.id.btnShow);

        clipboardManager =
(ClipboardManager)getSystemService(Context.CLIPBOARD_SERVICE);

        btnpst.setOnClickListener(new View.OnClickListener() {

            @Override

            public void onClick(View v) {
```

```java
            ClipData pData =
clipboardManager.getPrimaryClip();

            ClipData.Item item = pData.getItemAt(0);

            String txtpaste = item.getText().toString();

            ptxt.setText(txtpaste);

            Toast.makeText(getApplicationContext(),"Data
Pasted from Clipboard",Toast.LENGTH_SHORT).show();

        }

    });



    bAnlz.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View v) {

            closeKeyboard();

            AsyncTaskAnalyze asyncTask=new
AsyncTaskAnalyze();

asyncTask.execute("https://mean-senti.herokuapp.com/predict");

        }

    });

    bClr.setOnClickListener(new View.OnClickListener() {

        @Override

        public void onClick(View v) {

            ptxt.setText("");

        }

    });

}

private class AsyncTaskAnalyze extends AsyncTask<String,
String, String> {

    @Override
```

```java
    protected void onPreExecute() {

        super.onPreExecute();

        p = new ProgressDialog(MainActivity.this);

        p.setMessage("Please Wait...");

        p.setIndeterminate(false);

        p.setCancelable(false);

        p.show();

    }



    @Override

    protected String doInBackground(String... strings) {

        try {

            String url = new String(strings[0]);

            RequestQueue requestQueue =
Volley.newRequestQueue(MainActivity.this);

            JSONObject postData = new JSONObject();

            try {

                String content = new String();

                content = ptxt.getText().toString();

                postData.put("text", content);

            } catch (JSONException e) {

                e.printStackTrace();

            }


            JsonObjectRequest jsonObjectRequest = new
JsonObjectRequest(Request.Method.POST, url, postData, new
Response.Listener<JSONObject>() {

                @Override
```

```java
                public void onResponse(JSONObject response)
{

                resultData = response.toString();

                count--;

            }

        }, error -> resultData = error.toString());

        requestQueue.add(jsonObjectRequest);

        count++;

    } catch (Exception e) {

        e.printStackTrace();

    }

    while(count !=0 ){

        Log.d("Waiting","waiting");

    }

    return resultData;

}


    @Override

    protected void onPostExecute(String result) {

        super.onPostExecute(result);

        if(resultData != null){

            p.hide();

            Intent intent = new Intent(MainActivity.this,
AnalyseActivity.class);

            intent.putExtra("result",resultData);

            resultData = null;

            startActivity(intent);

        } else{
```

```java
            p.show();

        }

    }

}

    private void closeKeyboard()

    {

        View view = this.getCurrentFocus();

        if(view!=null)

        {

            InputMethodManager imm =
(InputMethodManager)getSystemService(Context.INPUT_METHOD_SERVICE);

            imm.hideSoftInputFromWindow(view.getWindowToken(),
0);

        }

    }

}
```

## 5.1.3 AnalyseActivty.java

```java
package in.bmsit.sixthsem.mean;

import androidx.appcompat.app.AppCompatActivity;

import android.graphics.Color;

import android.os.Bundle;

import android.widget.TextView;

import com.razerdp.widget.animatedpieview.AnimatedPieView;

import com.razerdp.widget.animatedpieview.AnimatedPieViewConfig;

import com.razerdp.widget.animatedpieview.data.SimplePieInfo;

public class AnalyseActivity extends AppCompatActivity {

    String result;
```

```java
TextView res;

String[] data;

String s1,s2,s3,s4;

@Override

protected void onCreate(Bundle savedInstanceState) {

    result = getIntent().getStringExtra("result");

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_analyse);

    getSupportActionBar().hide();//hide action bar

    res = findViewById(R.id.res);

    s1 = result.replace('"', ' ');

    s2 = s1.replace("{", "");

    s3 = s2.replace("}", "");

    s4 = s3.replace(":", ",");

    data = s4.split(",");

    drawPie();

}


public void drawPie()

{

    AnimatedPieView mAnimatedPieView =
findViewById(R.id.pie);

    AnimatedPieViewConfig config = new
AnimatedPieViewConfig();

    for(int i=0; i<data.length; i+=2) {

        config.startAngle(-90)

                .addData(new
SimplePieInfo(Integer.parseInt(data[i+1]), randomColor(),
data[i]))
```

```java
.drawText(true).duration(2000).textSize(30);

        }

        mAnimatedPieView.applyConfig(config);

        mAnimatedPieView.start();

    }



    public int randomColor() {

        int r = (int) (0xff * Math.random());

        int g = (int) (0xff * Math.random());

        int b = (int) (0xff * Math.random());

        return Color.rgb(r, g, b);

    }

}
```

# 5.2 XML Code

## 5.2.1 activity_intro.xml

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

<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"

    xmlns:app="http://schemas.android.com/apk/res-auto"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout_width="match_parent"

    android:layout_height="match_parent"

    android:background="#ffffff"


app:layout_behavior="@string/hide_bottom_view_on_scroll_behavi
or"
```

```xml
    tools:context=".IntroActivity">


    <pl.droidsonroids.gif.GifImageView

        android:layout_width="255dp"

        android:layout_height="310dp"

        android:layout_alignParentStart="true"

        android:layout_alignParentTop="true"

        android:layout_alignParentEnd="true"

        android:layout_alignParentBottom="true"

        android:layout_marginStart="50dp"

        android:layout_marginTop="150dp"

        android:layout_marginEnd="50dp"

        android:layout_marginBottom="150dp"

        android:src="@drawable/meanintro"

        app:layout_constraintBottom_toBottomOf="parent"

        tools:layout_editor_absoluteX="53dp" />


</RelativeLayout>
```

## 5.2.2 activity_main.xml

```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"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout_width="match_parent"

    android:layout_height="match_parent"
```

```xml
    android:background="@drawable/gradient4"

    tools:context=".MainActivity">

<TextView

        android:id="@+id/textView"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:fontFamily="@font/fontdemi"

        android:text="Message Analyser"

        android:textColor="#FFE07D"

        android:textSize="12pt"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent"

        app:layout_constraintVertical_bias="0.144" />


<Button

        android:id="@+id/bAnalyse"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:fontFamily="@font/fontdemi"

        android:text="Analyse"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintHorizontal_bias="0.718"

        app:layout_constraintStart_toStartOf="parent"
```

```xml
        app:layout_constraintTop_toTopOf="parent" />


    <Button
        android:id="@+id/btnShow"

        android:layout_width="250dp"

        android:layout_height="wrap_content"

        android:fontFamily="@font/fontdemi"

        android:text="Paste Clipboard Data"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintHorizontal_bias="0.509"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent"

        app:layout_constraintVertical_bias="0.409" />


    <Button
        android:id="@+id/bClear"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:fontFamily="@font/fontdemi"

        android:text="Clear"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintHorizontal_bias="0.284"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent" />
```

```xml
    <EditText
        android:id="@+id/txtShow"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:ems="10"

        android:fontFamily="@font/fontdemi"

        android:hint="Type/paste text"

        android:textAlignment="center"

        android:textColor="#f1f2f6"

        android:textColorHint="#57DFE4EA"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintHorizontal_bias="0.488"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent"

        app:layout_constraintVertical_bias="0.32" />


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

### 5.2.3 activity_analyse.xml

```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"

    xmlns:tools="http://schemas.android.com/tools"

    android:layout_width="match_parent"

    android:layout_height="match_parent"
```

```xml
    android:background="#D5D9DA"

    tools:context=".AnalyseActivity">


    <com.razerdp.widget.animatedpieview.AnimatedPieView

        android:id="@+id/pie"

        android:layout_width="363dp"

        android:layout_height="286dp"

        android:layout_marginBottom="140dp"

        android:fontFamily="@font/fontdemi"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toBottomOf="@+id/textView2"

        app:layout_constraintVertical_bias="0.546" />


    <TextView

        android:id="@+id/res"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:fontFamily="@font/fontregular"

        android:text=""

        android:textAlignment="center"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintHorizontal_bias="0.498"

        app:layout_constraintStart_toStartOf="parent"
```

---

```xml
        app:layout_constraintTop_toTopOf="parent"

        app:layout_constraintVertical_bias="0.823" />


    <TextView

        android:id="@+id/textView2"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:text="Result"

        android:fontFamily="@font/fontdemi"

        android:textSize="15pt"

        android:textColor="#5C5C5C"

        app:layout_constraintBottom_toBottomOf="parent"

        app:layout_constraintEnd_toEndOf="parent"

        app:layout_constraintHorizontal_bias="0.498"

        app:layout_constraintStart_toStartOf="parent"

        app:layout_constraintTop_toTopOf="parent"

        app:layout_constraintVertical_bias="0.119" />



</androidx.constraintlayout.widget.ConstraintLayout>
```
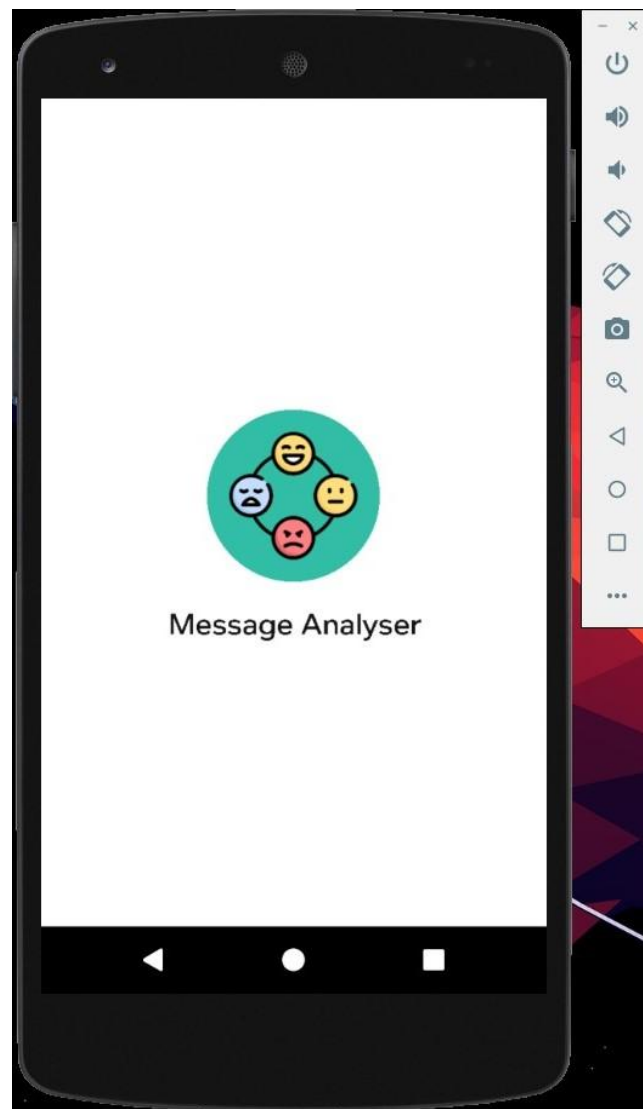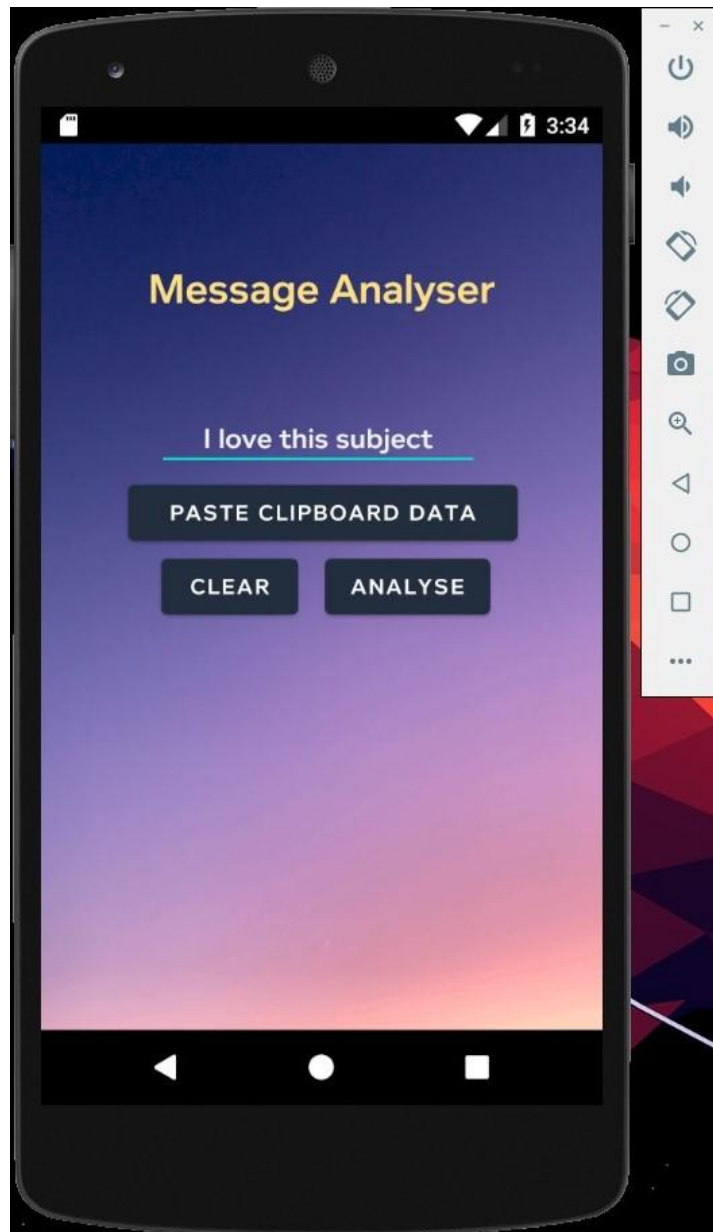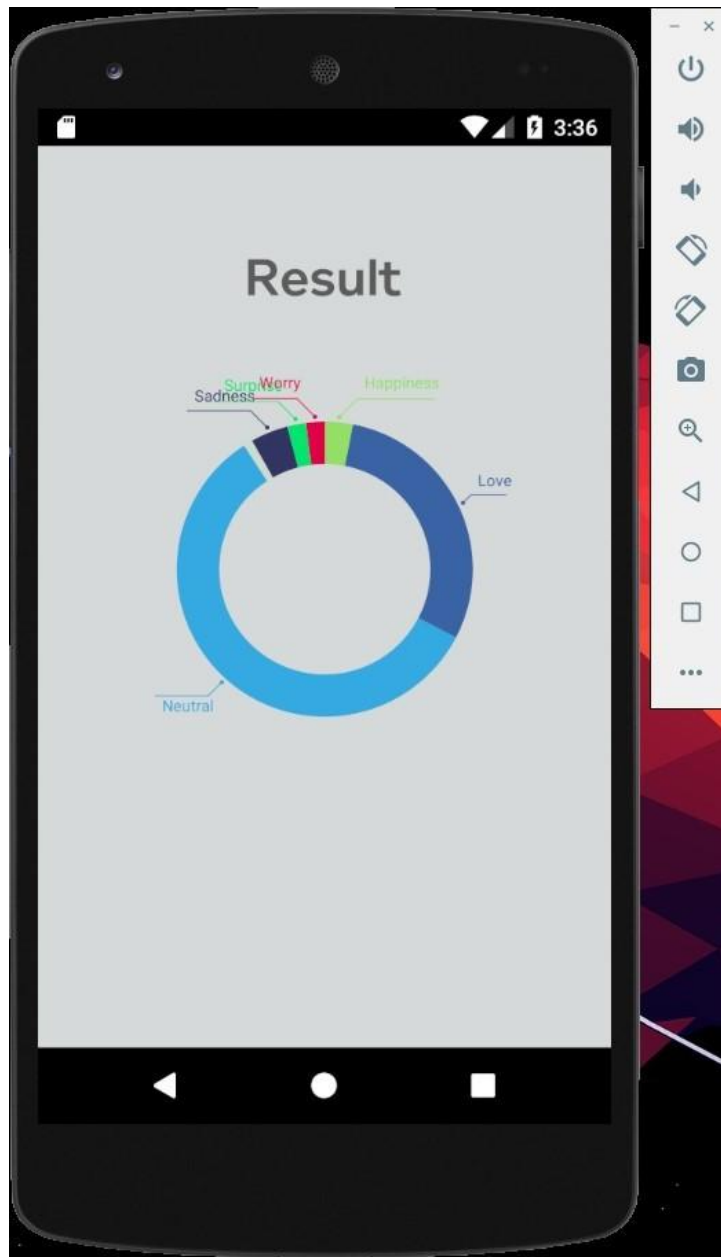
# CHAPTER 6

# RESULTS



**Fig 1**

**Welcome Screen showing logo**

**Fig 2**

**The main Screen having an input field and buttons to paste messages from the clipboard, clear the input field, and analyze the message**

**Fig 3**

**The Result Screen showing the Pie Chart of sentiments detected**

# Chapter 7

# CONCLUSION AND FUTURE ENHANCEMENTS

As a result of a deeper and better understanding of the feelings, emotions, and sentiments of a brand or organization's key, high-value audiences, members of these audiences will increasingly receive experiences and messages that are personalized and directly related to their wants and needs. Rather than segment markets based on age, gender, income, and other surface demographics, organizations can further segment based on how their audience members actually feel about the brand or how they use social media. While some people shudder at the thought of companies learning more about them, more exact targeting means that, in the near future, we will no longer be scratching our heads wondering why we see advertisements for products we'd never dream of purchasing. In other words, the spray-and-pray advertising tactics are almost put to rest and there will be a time when every marketing message we see will be relevant and useful to us. Sentiment analysis is going to be a large contributing factor towards achieving this vision.

# BIBLIOGRAPHY

1. https://developer.android.com/reference/android/content/Context

2. https://docs.python.org/3/

3. https://developer.android.com/training/volley

4. https://devcenter.heroku.com/

5. https://www.nltk.org/

6. https://keras.io/