VISVESVARAYA TECHNOLOGICAL UNIVERSITY

Jnana Sangama, Belagavi – 590014.

MINI PROJECT REPORT

ON

"Clipboard options to add font style, size, color, boldness and underline features"

Submitted in partial fulfilment for the requirement of 6th semester for the

Degree of Bachelor of Engineering in

COMPUTER SCIENCE & ENGINEERING

For the academic year 2021-22

SUBMITTED BY:

PRIYA (1DB19CS111)
RAJKUMAR(1DB19CS116)
SAIF ALI(1DB19CS127)
VAISHALEE(1DB19CS156)

Under the guidance of:

Prof. RAKHEEBA TASEEN

Assistant Professor,

Dept. of CSE

DON BOSCO INSTITUTE OF TECHNOLOGY, BENGALURU-560074

DON BOSCO INSTITUTE OF TECHNOLOGY

Kumbalagodu, Bengaluru-560074

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Mini Project Report entitled **"Clipboard options to add font style, size, color, boldness and underline features"** is a bonafide Mini Project work carried out by **PRIYA KUMARI (1DB19CS111), RAJKUMAR MANNA (1DB19CS116), SAIF ALI(1DB19CS127)** and **VAISHALEE KUMARI (1DB19CS156)** in partial fulfillment of '6th' semester for the Degree of **Bachelorof Engineeringin Computer Science and Engineering** of Visvesvaraya Technological University, Belagavi, during the academic year 2021-22. It is certified that all corrections/suggestions indicated for Internal Assessments have beenincorporated with the degree mentioned.

Project Guide

Prof. Rakheeba Taseen
Assistant Professor
Dept. of CSE
DBIT, Bangalore

Head of Department

Prof. Shiva Kumar
Assistant Professor &
Head of Department
Dept. of CSE
DBIT, Bangalore

Sl. No	Name of The Examiner Signature with Date	
1		
2		

DON BOSCO INSTITUTE OF TECHNOLOGY

Kumbalagodu, Bengaluru -560074

DECLARATION

PRIYA KUMARI, RAJKUMAR MANNA, SAIF ALI, VAISHALEE KUMARI, students of sixth semester B.E, Department of Computer Science and Engineering, Don Bosco Institute of Technology, Kumbalagodu, Bengaluru, declare, that Mini Project Work entitled **"Options to Add font style, size, color, boldness and underline features"** has been carried out by and submitted in partial fulfillment of the requirement of 6th semester 2021-22. The matter embodied in this reporthas been submitted to any university or institute for the award of any other degree or diploma.

Place: Bengaluru

Date:

ACKNOWLEDGEMENT

At the various stages in making the mini project, a number of people have given me invaluable comment on the manuscript. I take this opportunity to express my deepest gratitude and appreciation to all those whohelped me directly or indirectly towards the successful completion of this mini project.

I would like to thank **Principal Dr. Umashankar, Don Bosco Institute Of Technology** for his support though out this mini project.

I express my whole hearted gratitude to **Prof. Shivakumar,** who is our respectable **Associate Professor** and **Head OfDept. Department Of Computer Science And Engineering**. I wish to acknowledge for her valuable help and encouragement.

In this regard, a heartfelt gratitude to guide **Prof. Rakheeba Taseen, Assistant Professor Of Department Of Computer Science And Engineering**, for her timely advice on the mini project and regular assistance throughout the mini project work. I would also like to thank the staff members of **Department Of Computer Science And Engineering** for their co-operation.

ABSTRACT

The main aim and objective of this project was to develop a application where the user can easily copy and paste text in any instance of time. This project report consists of basic idea of copying and pasting text. Android provides the clipboard framework for copying and pasting different types of data. The data could be character, number, special character and also in words. Android provides the library of ClipboardManager and ClipData to use the copying and pasting framework. In order to use clipboard framework, you need to put data into clip object, and then put that object into system wide clipboard. Then the object entered is copied on pressing the copy button. The entered text is pasted when the paste button is pressed. Hence this application is helpful for copying and pasting in an effective manner. An extended feature is also added to this application. That extended feature is the character count, number count and word count of the pasted text. The character count will count the number of characters present in the pasted text and displays the number of character. Likewise, the number count displays the count of number present in the pasted text and the word count will display the count of the word in the pasted text.

TABLE OF CONTENTS

Chapter-1

INTRODUCTION

In recent years, the emergence of smart phones has changed the definition of mobile phones. Phone is no longer just a communication tool, but also an essential part of the people's communication and daily life. Various applications added unlimited fun for people's lives. It is certain that the future of the network will be the mobile terminal. Now the Android system in the electronics market is becoming more and more popular, especially in the smart phone market. Because of the open source, some of the development tools are free, so there are plenty of applications generated. This greatly inspired the people to use the Android system.In addition, it provides a very convenient hardware platform for developers so that they can spend less effort to realize their ideas. After studying some previous Android applications andaccess to large amounts of materials, we utilize the Java language, the Eclipse platform, Android ADT and the Android SDK to develop these three mobile applications. Thesesystems have a nice interface and smooth operation. These Apps won't steal any personal information, but can exclude useless information and bring a wonderful user experience.

1.1 Android:

Android is a mobile operating system (OS) currently developed by Google, based on the Linux kernel and designed primarily for touch screen mobile devices such as smart phones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. In addition to touch screen devices, Google has further developedAndroid TV for televisions, Android Auto for cars, and Android Wear for wristwatches, each with a specialized user interface. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics. Initially developed by Android, Inc., which Google bought in 2005, Android was unveiled in 2007, along with the founding of the Open Handset Alliance – a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. As of July 2013, the Google Play store has had over one million Android applications ("apps") published, and over 50 billion applications downloaded. AnApril– May 2013 survey of mobile application developers found that 71% of developers create applications for Android, and a 2015 survey found that 40% of full- time professional developers see Android as their priority target platform.

Fig 1.1 Android image

1.2 Software Development Kit:

A software development kit (SDK or "devkit") is typically a set of software development tools. that allows the creation of applications for a certain software package, software framework, hardware platform, computer system, video game console, operating system, or similar development platform. To create applications, you have to download this software development kit. For example, if you want to create an Android app you require an SDK with java programming, for iOS apps you require an iOS SDK with swift language, and to develop MS Windows apps you require the.net language. There are also SDKs that are installed in apps to provide analytics and data about activity. Prominent examples include Google and Facebook.

1.3 Android Studio:

Android Studio is an integrated development environment (IDE) for developing for theAndroid platform. It was announced on May 16, 2013 at the Google I/O conference. Android Studio is freely available under the Apache License 2.0. Android Studio was in early access preview stage starting from version

0.1 in May 2013, then entered beta stage starting from version 0.8 which was released in June 2014. The first stable build was released in December 2014, starting from version 1.0 to currently Arctic Fox Based on JetBrains' IntelliJ IDEA software, Android Studio is designed specifically for Android development. It is available for download on Windows, Mac OS X and Linux, and replaced Eclipse Android Development Tools (ADT) as Google's primary IDE for native Android application development.

Fig 1.3 Android Studio

1.4 Android Manifest:

The AndroidManifest.xml file *contains information of your package*, including components of the application such as activities, services, broadcast receivers, content providers etc.

It performs some other tasks also:

- It is responsible to protect the application to access any protected parts by providing the permissions.
- It also declares the android api that the application is going to use.
- It lists the instrumentation classes. The instrumentation classes provide profiling and other information. This information is removed just before the application is published.

1.5 Main Activity:

The Main Activity File

The main activity code is a Java file MainActivity.java. This is the actual application file which ultimately gets converted to a Dalvik executable and runs your application. Following is the default code generated by the application wizard for Hello World! application − package

```
com.example.helloworld;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle; public class MainActivity
extends AppCompatActivity {
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
  }}
```

Here, R.layout.activity_main refers to the activity_main.xml file located in the res/layout folder. The onCreate() method is one of many methods that are figured when an activity is loaded.

Chapter-2

PROBLEM STATEMENT AND OBJECTIVES

2.1 Problem Statement:

Develop an application that makes use of the clipboard framework for copying and pasting of the text. The activity consist of the two Edit Text controls and two Buttons to trigger the copy and paste functionality .

2.2 Objectives:

- To design an xml code that can add font style, size, color, boldness and underline features.
- To code a java program that can add font style, size, color, boldness and underline features.
- Adding additional features to improve system and user interface by adding font style, size, color, boldness and underline features.
- Testing the code for all its intended functionalities by a android device or avd.

Chapter-3

SYSTEM REQUIREMENT

3.1 HARDWARE REQUIREMENT:

Minimum RAM :- 8 GB or more.
Processor :- Intel core i5 8th Gen.
Disk Space :- 20 GB of available disk space.
Display :- 1280 x 800 minimum screen resolution.

3.2 SOFTWARE REQUIREMENT:

Operating System :- 64-bit Microsoft Windows 8/10.
Emulator :- Pixel 4 api 30.
Target SDK version :- 30.
Language Used :- xml , java.
JDK version :- java development kit 16.

Chapter-4

STEPS FOR EXECUTION

4.1 RUN ON EMULATOR:

1) Run the Android Studio on laptop or desktop.

2) Create a project and enter the source codes of xml and java.

3) In Android Studio, create an Android Virtual Device (AVD) that the emulator can use to install and run your app.
   In the toolbar, select your app from the run/debug configurations drop-down menu.

5) From the target device drop-down menu, select the AVD that you want to run your app on.

6) Click Run . Android Studio installs the app on the AVD and starts the emulator.

Chapter-5

SOURCE CODES

XML CODE:

5.1 Activitymain.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"
    tools:context=".MainActivity">


    <EditText
        android:id="@+id/mytext"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="38dp"
        android:layout_marginBottom="41dp"
        android:ems="10"
        android:hint="Enter any text"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toTopOf="@+id/button2"
        app:layout_constraintEnd_toEndOf="@+id/button" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginStart="16dp"
        android:layout_marginBottom="43dp"
        android:text="paste"
        app:layout_constraintBottom_toTopOf="@+id/spinner2"
        app:layout_constraintStart_toStartOf="@+id/spinner2" />
```

```
<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginEnd="1dp"
    android:text="view"
    app:layout_constraintBaseline_toBaselineOf="@+id/button"
    app:layout_constraintEnd_toEndOf="@+id/spinner" />

 <Spinner
    android:id="@+id/spinner"
    android:layout_width="0dp"
    android:layout_height="55dp"
    android:layout_marginStart="16dp"
    android:layout_marginEnd="39dp"
    android:layout_marginBottom="245dp"
    android:entries="@array/color"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toStartOf="@+id/spinner2"
    app:layout_constraintStart_toStartOf="parent" />

 <Spinner
    android:id="@+id/spinner2"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="6dp"
    android:layout_marginEnd="16dp"
    android:entries="@array/size"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toEndOf="@+id/spinner"
    app:layout_constraintTop_toTopOf="@+id/spinner" />

 <Spinner
    android:id="@+id/spinner3"
    android:layout_width="170dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="1dp"
    android:layout_marginTop="44dp"
    android:entries="@array/style"
```

```
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/spinner" />

  <EditText
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="42dp"
    android:text="TextView"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />


  <ToggleButton
    android:id="@+id/Bold"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="1dp"
    android:layout_marginBottom="19dp"
    android:text="ToggleButton"
    app:layout_constraintBottom_toTopOf="@+id/under"
    app:layout_constraintStart_toStartOf="@+id/under" />

  <ToggleButton
    android:id="@+id/under"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="30dp"
    android:layout_marginBottom="38dp"
    android:text="ToggleButton"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

5.2 Main Activity:

```
package com.example.clipboard1

import android.graphics.Color
import android.graphics.Paint
import android.graphics.Typeface
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.view.View
import android.widget.*
import androidx.core.content.res.ResourcesCompat
import com.google.android.material.snackbar.Snackbar

class MainActivity : AppCompatActivity() {

    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val tv = findViewById<EditText>(R.id.textView2)
        val ed1 = findViewById<EditText>(R.id.mytext)
        val b1 = findViewById<Button>(R.id.button)
        val b2 = findViewById<Button>(R.id.button2)
        val b3 = findViewById<ToggleButton>(R.id.Bold)
        val b4 = findViewById<ToggleButton>(R.id.under)


        b1.setOnClickListener {
            val contextView = findViewById<View>(R.id.button)
            val snack = Snackbar.make(contextView, "Text Pasted",
Snackbar.LENGTH_INDEFINITE)
            snack.show()

            tv.setText(ed1.text.toString())
    }
        b2.setOnClickListener{
            val contextView = findViewById<View>(R.id.button2)
            val snack = Snackbar.make(contextView, "Text Copied",
```

```
Snackbar.LENGTH_INDEFINITE)
    snack.show()
}
b3.setOnCheckedChangeListener{_,isChecked ->
    if (isChecked){
    tv.setTypeface(ed1.typeface, Typeface.BOLD);
        }
    else{
        tv.setTypeface(ed1.typeface, Typeface.NORMAL);


    }
}
b4.setOnCheckedChangeListener{_,isChecked ->
    if (isChecked){
        tv.setPaintFlags(ed1.getPaintFlags() or Paint.UNDERLINE_TEXT_FLAG)
    }
    else{
        tv.setPaintFlags(ed1.paintFlags or Paint.ANTI_ALIAS_FLAG);


    }
}
val sp1 = findViewById<Spinner>(R.id.spinner)
sp1.onItemSelectedListener = object : AdapterView.OnItemSelectedListener {
    override fun onItemSelected(p0: AdapterView<*>?, p1: View?, p2: Int, p3: Long) {
        val pos = sp1.selectedItemPosition
        when (pos) {
            1 -> tv.setTextColor(Color.RED)
            2 -> tv.setTextColor(Color.BLACK)
            3 -> tv.setTextColor(Color.GREEN)
            4 -> tv.setTextColor(Color.BLUE)
        }
    }

    override fun onNothingSelected(p0: AdapterView<*>?) {
        TODO("Not yet implemented")
    }
```

```
}

    val sp2 = findViewById<Spinner>(R.id.spinner2)
    sp2.onItemSelectedListener = object : AdapterView.OnItemSelectedListener {
        override fun onItemSelected(p0: AdapterView<*>?, p1: View?, p2: Int, p3: Long) {
            var pos = sp2.selectedItemPosition
            when (pos) {
                1 -> tv.setTextSize(20f)
                2 -> tv.setTextSize(25f)
                3 -> tv.setTextSize(30f)
                4 -> tv.setTextSize(35f)
            }
        }

        override fun onNothingSelected(p0: AdapterView<*>?) {
            TODO("Not yet implemented")
        }

    }

    val sp3 = findViewById<Spinner>(R.id.spinner3)
    sp3.onItemSelectedListener = object : AdapterView.OnItemSelectedListener {
        override fun onItemSelected(p0: AdapterView<*>?, p1: View?, p2: Int, p3: Long) {
            val pos = sp3.selectedItemPosition
            when (pos) {
                1 -> tv.typeface = ResourcesCompat.getFont(this,R.font.lacto)
                2 -> tv.typeface = ResourcesCompat.getFont(this,R.font.lacktoblack)
                3 -> tv.typeface = ResourcesCompat.getFont(this,R.font.lacktoital)
                4 -> tv.typeface = ResourcesCompat.getFont(this,R.font.lactooo)
            }
        }
          override fun onNothingSelected(p0: AdapterView<*>?) {
            TODO("Not yet implemented")
        }

    }

    }
}
```

Chapter-6

SNAPSHOTS

6.1 Main Page:

This is the main page of the clipboard application. Here we can add font style, size, color, boldness and underline features as shown in Fig 6.1.

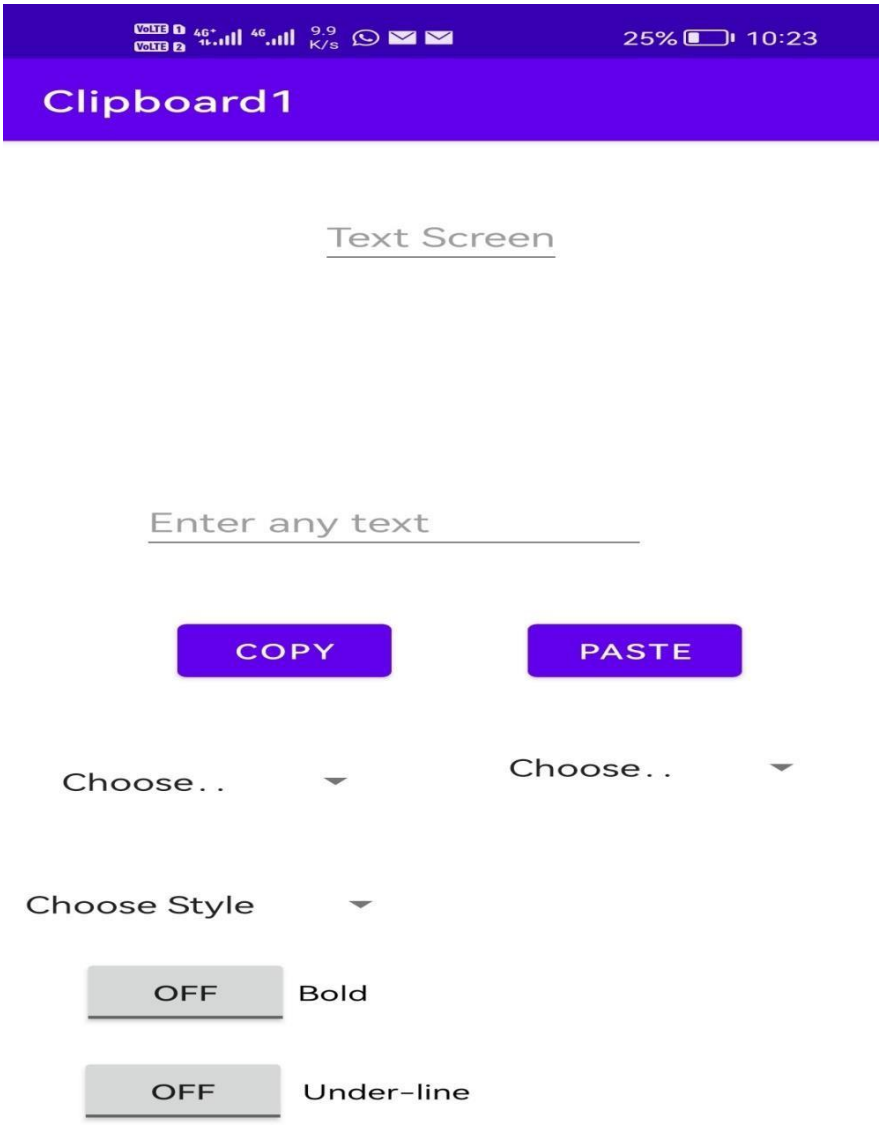

Fig 6.1 Snapshot displays main page

6.2 Enter Text Copy :

This is the page where the text has been entered which is to be copied. We can enter any character, number or specialcharacter as shown in Fig 6.2.

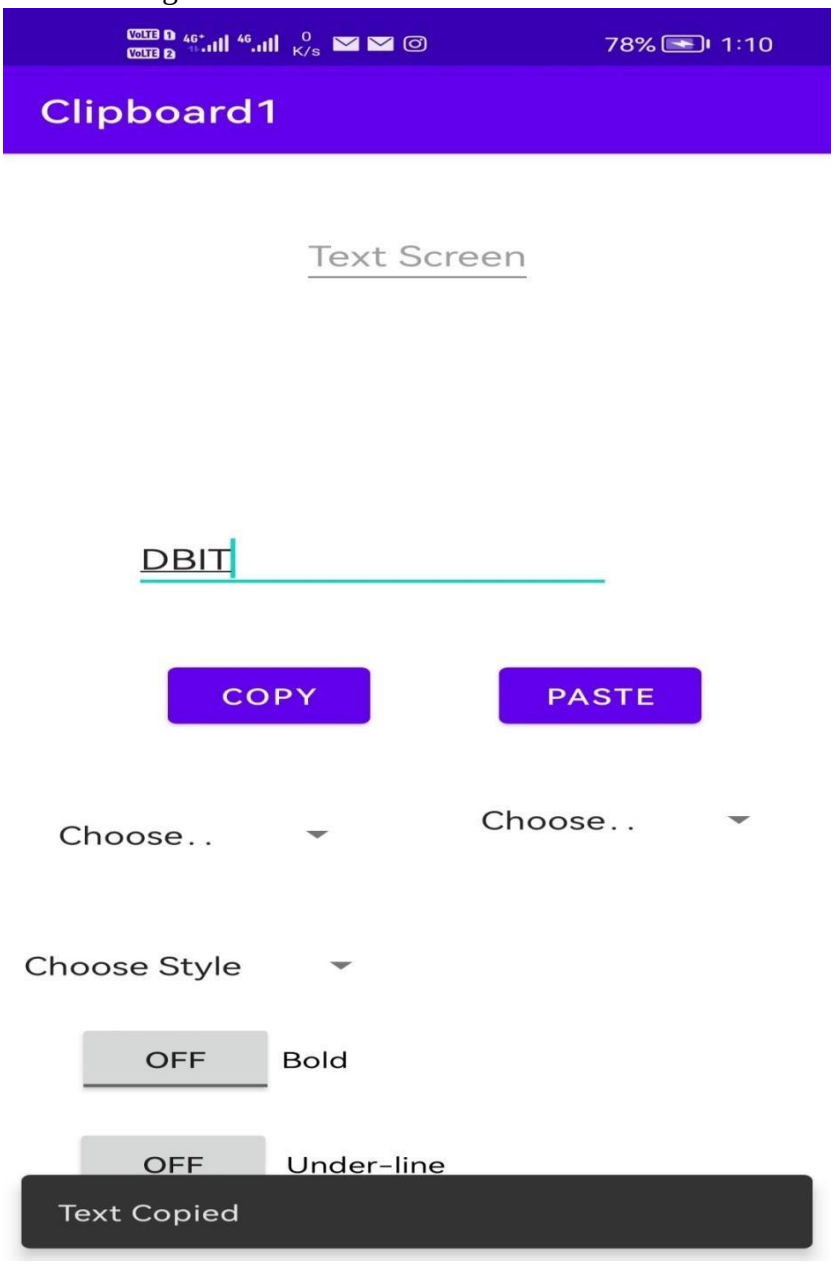

Fig 6.2 Snapshot displays the text entered

6.3 Enter Text Paste :

The text entered will be pasted on pressing the paste button. A simple toast message 'Text Pasted' is appeared sayingthat the message is pasted as shown in Fig 6.3.

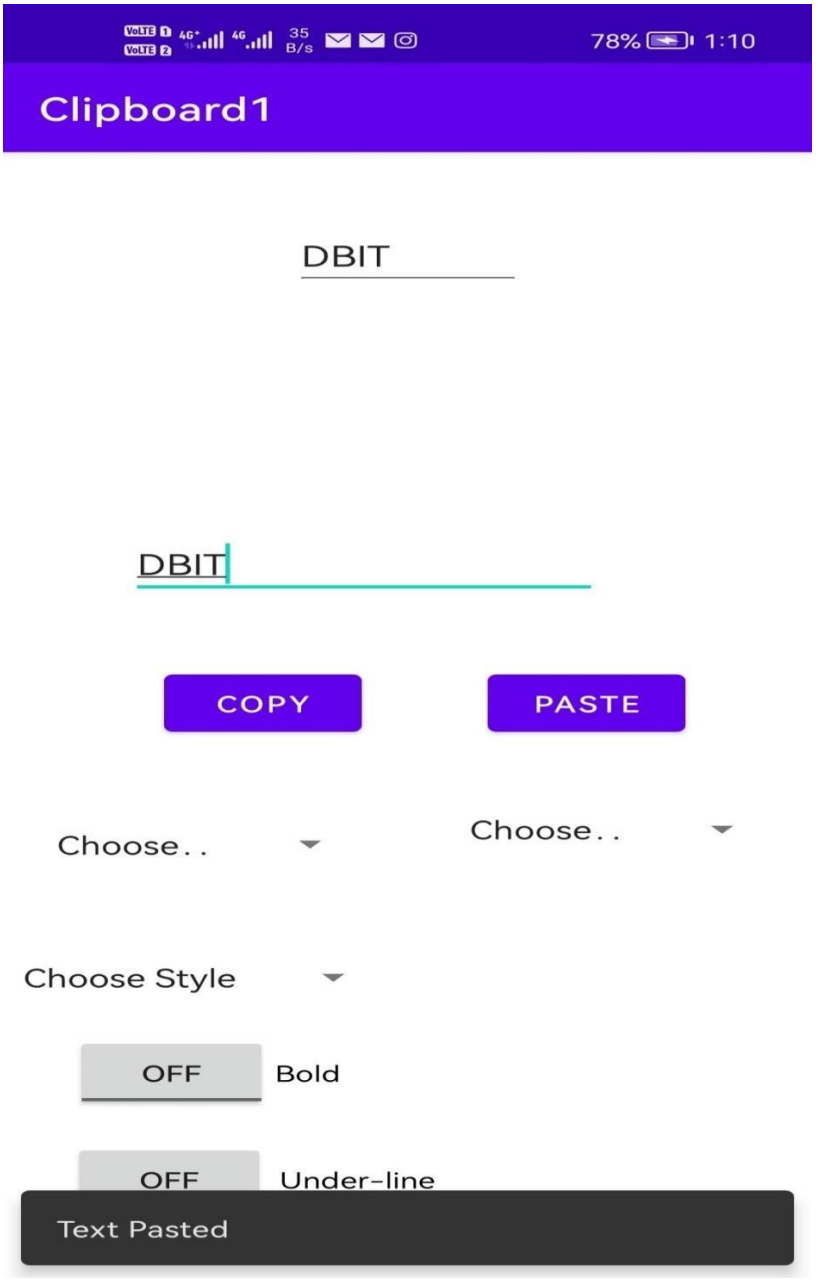

Fig 6.3 Snapshot displays text copied

6.4 Enter Text:

A simple toast message is appeared saying that display red color message entered as shown in Fig 6.4.

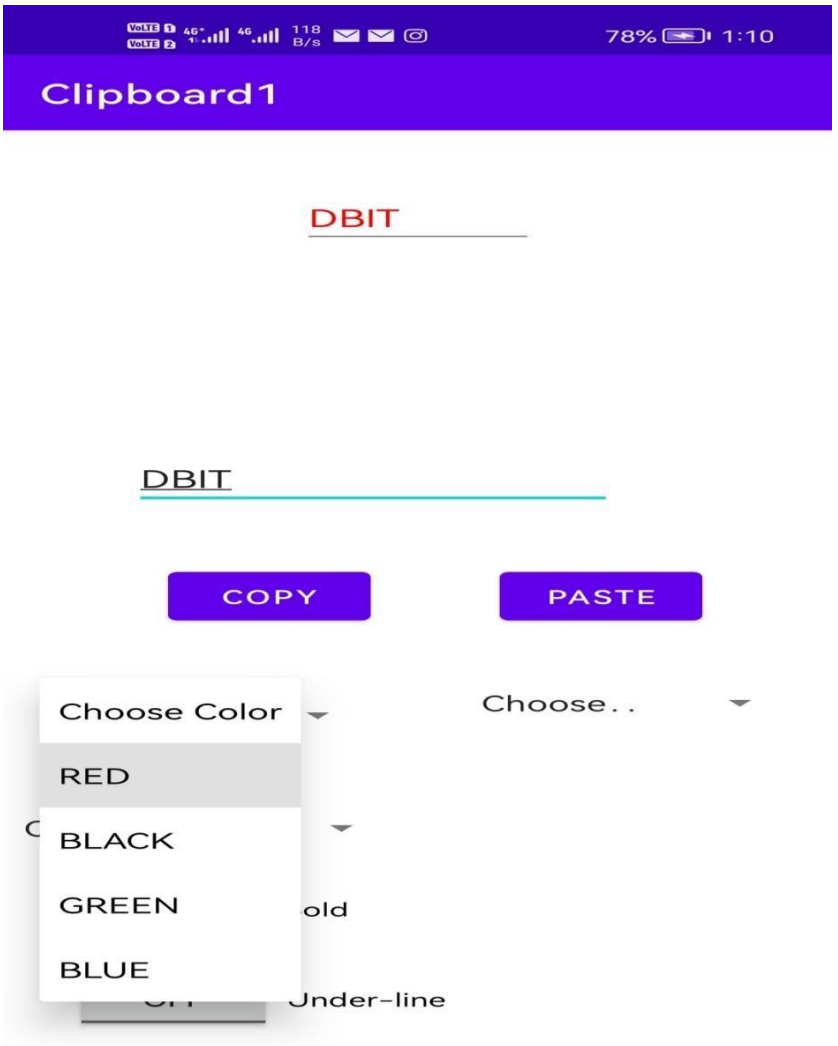

Fig 6.4 Snapshot displays text pasted

6.5 Font style, color, underline

Here the character count, number count and word count of the entered text is counted and is displayed as shown in Fig6.5.

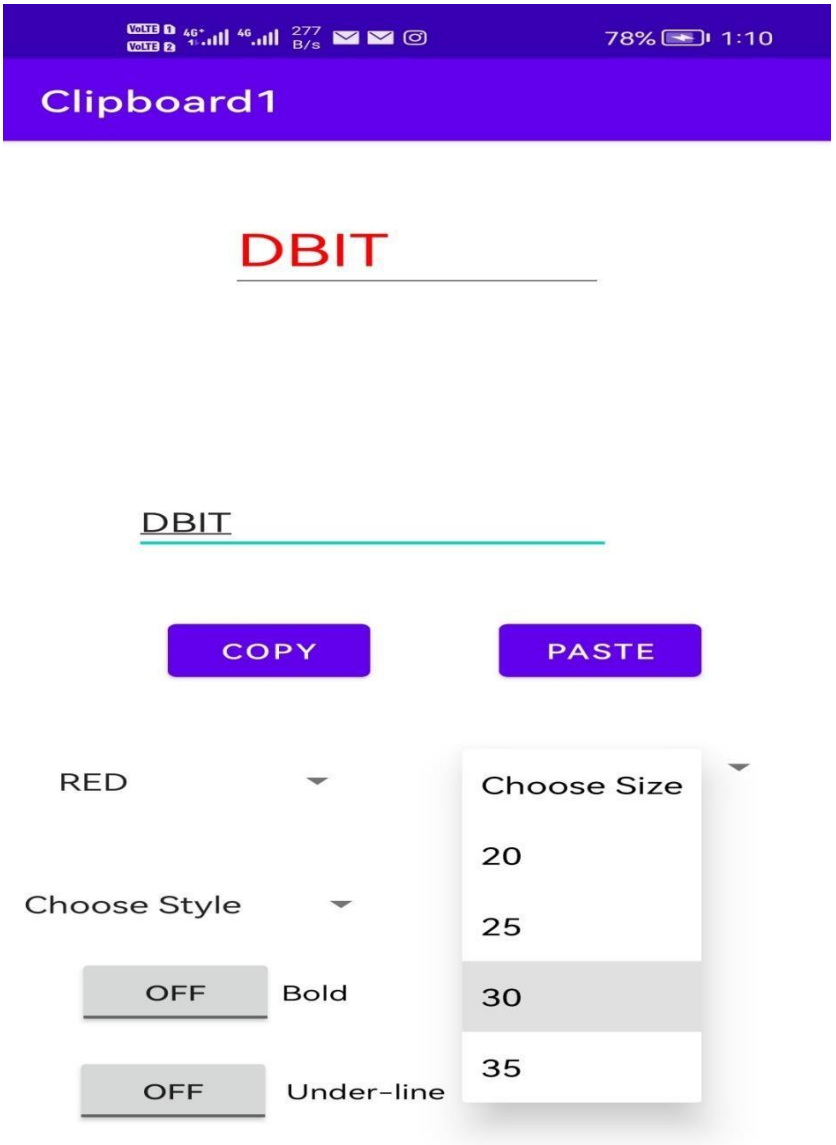

Fig 6.5 Snapshot displays Font style, color, underline

Chapter - 7

CONCLUSION AND FUTURE ENHANCEMENT

Android as a full, open and free mobile device platform, with its powerful function and good user experience rapidly developed into the most popular mobile operating system. This report gives an overview of the different challenges and issues faced in android app development. The experience of developing an android app is quite challenging, motivating as well as satisfying.

Chapter- 8

REFERENCE

Reference Book:

Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017.

Website:

[1]. YouTube : https://www.youtube.com/
[2].https://developer.android.com/studio
[3].https://github.com/
[4].https://www.gitbook.com/book/google-developer-training/android-developer-
[5].fundamentals-course- concepts/details
[6].https://developer.android.com/reference/android/os/Messagingappliaction