

JAYPEE INSTITUTE OF INFORMATION TECHNOLOGY



MINOR PROJECT

FINAL EVALUATION

PC Remote

(RFCOMM PROTOCOL)



GROUP MEMBERS:-

AYUSH KAUSHIK(13103679)

AMIT SACHDEVA(13103677)

AYUSH BANSAL(13103664)

LAKSHAY GOYAL(13103726)

SUBMITTED TO:-

MRS. GAGANDEEP KAUR

MRS. SANGEETA MITTAL

ABSTRACT:

The main objective of this project is to calibrate a tool that enables the user to use various features of computer via mobile phone. This tool finds a vast scope of uses in educational institutions and corporations.

Especially while giving presentations user may want to control the computer from a distance.

This tool actually makes our mobile phone a remote controller.

INTRODUCTION:

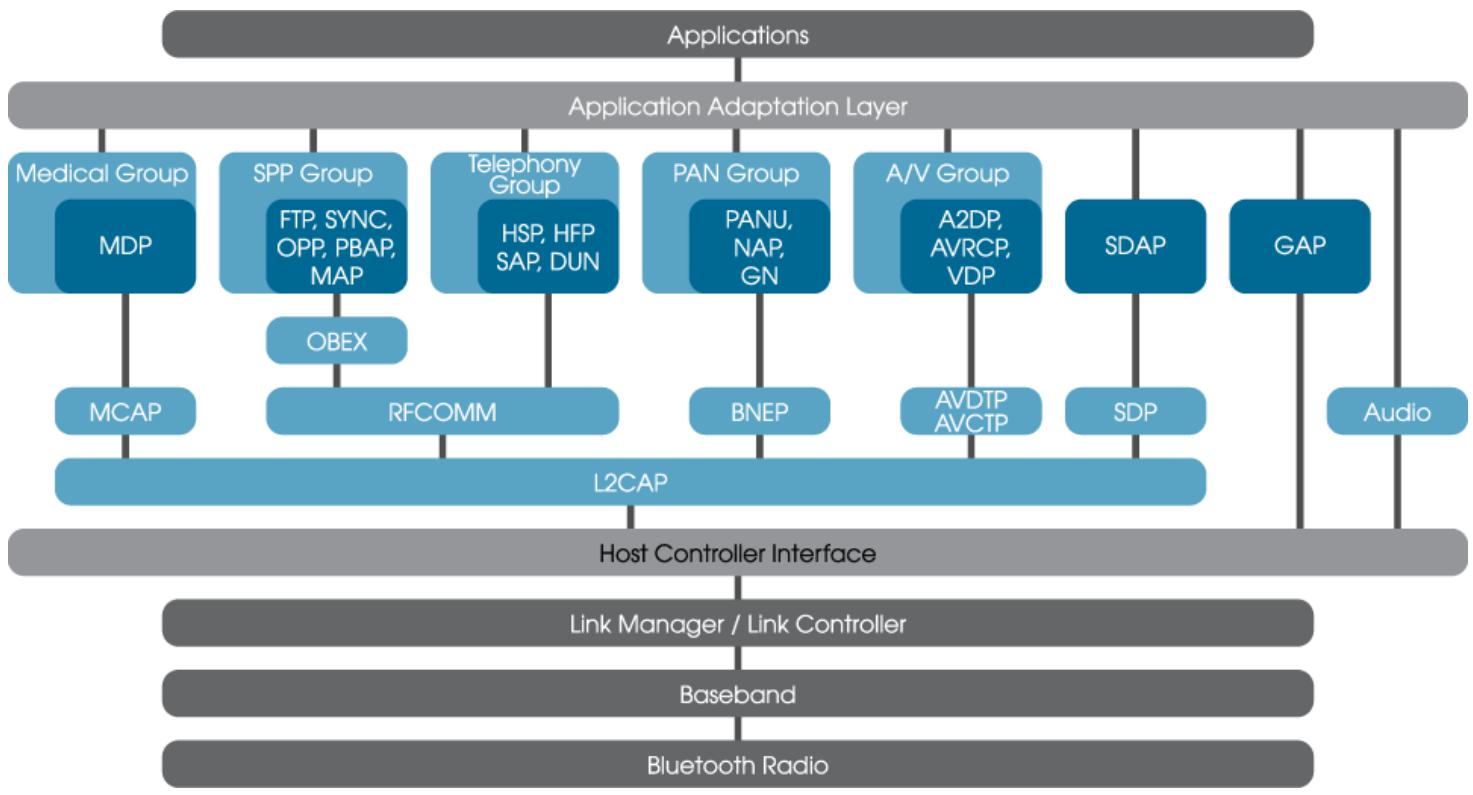
For the implementation of this tool a server program should be running on the computer. On mobile phone an app must be installed. The app lets you connect the phone with computer via Bluetooth. All the data is sent from the phone via Bluetooth.

Implemented features:

- Cursor controller : User can control the cursor of PC by just swiping his fingers on the touchscreen of his phone.

- Music Player : The app accesses the music folder of the PC (by default) and fetches the list of all music files. The user can select any of the items of the list and can either play, pause, stop or switch to next or previous music file.
- PPT tool : The app fetches the list of ppt files from the PC and lets the user choose which presentation is to be played. It lets the user to switch to next or previous slide.
- Virtual Keyboard:
- **GUI based implementation.**

RFCOMM PROTOCOL BLUETOOTH



The RFCOMM protocol emulates the serial cable line settings and status of an RS-232 serial port and is used for providing serial data transfer. RFCOMM connects to the lower layers of the *Bluetooth* protocol stack through the L2CAP layer.

Example Products

Here are a few examples of the types of devices that you might find using the RFCOMM:

- Printer
- Modem
- PC
- Laptop

Getting Technical

By providing serial-port emulation, RFCOMM supports legacy serial-port applications while also supporting the OBEX protocol among others. RFCOMM is a subset of the ETSI TS 07.10 standard, along with some *Bluetooth* specific adaptations.

The RFCOMM protocol supports up to 60 simultaneous connections between two *Bluetooth* devices. The number of connections that can be used simultaneously in a *Bluetooth* device is implementation-specific.

The RFCOMM protocol emulates the serial cable line settings and status of an RS-232 serial port and is used for providing serial data transfer. RFCOMM connects to the lower layers of the *Bluetooth* protocol stack through the L2CAP layer.

Example Products

Here are a few examples of the types of devices that you might find using the RFCOMM:

- Printer
- Modem
- PC
- Laptop

Getting Technical

By providing serial-port emulation, RFCOMM supports legacy serial-port applications while also supporting the OBEX protocol among others. RFCOMM is a subset of the ETSI TS 07.10 standard, along with some *Bluetooth* specific adaptations.

The RFCOMM protocol supports up to 60 simultaneous connections between two *Bluetooth* devices. The number of connections that can be used simultaneously in a *Bluetooth* device is implementation-specific.

For the purposes of RFCOMM, a complete communication path involves two applications running on different devices (the communication endpoints) with a communication segment between them. The figure above shows the complete communication path. (In this context, the term application may mean other things than end-user application; e.g. higher layer protocols or other services acting on behalf of end-user applications.)

RFCOMM is intended to cover applications that make use of the serial ports of the devices in which they reside. In the simple configuration, the communication segment is a *Bluetooth* link from one device to another (direct connect), see the figure to the left. Where the communication segment is another network, *Bluetooth* wireless technology is used for the path between the device and a network connection device like a modem. RFCOMM is only concerned with the connection between the devices in the direct connect case, or between the device and a modem in the network case.

RFCOMM can support other configurations, such as modules that communicate via *Bluetooth*wireless technology on one side and provide a wired interface on the other side, as shown in the figure below. These devices are not really modems but offer a similar service. They are therefore not explicitly discussed here.

Division Of Work:

Amit Sachdeva and Ayush Kaushik:

- Bluetooth Chat App Server
- PC Client
- PC Server

Ayush Bansal and Lakshay Goyal:

- PC Remote Android App
- PC Remote Android App Server
- Bluetooth Chat App Client

BACKGROUND STUDY AND FINDINGS:

- Python Documentation
- Study material on file server
- developers.android.com
- androidhive.com
- stackoverflow.com
- codesmith.com
- pipy.com
- github.com

REQUIREMENT ANALYSIS:

Software:

- Install python 2.7.1 (minimum)
- Need systems with current updated software installed
- Android Studio

Libraries Import:

- Win32api
- Win32com
- VLC
- Pyautogui
- Functools
- Pybluez
- Android.bluetooth.bluetoothadapter
- Android.bluetooth.bluetoothdevice
- Java.net.socket
- Java.io.bufferedReader
- Java.io.bufferedWriter
- Java.net.InetAddress
- Java.io.OutputStream
- Java.io.InputStream
- Java.io.OutputStreamWriter
- Java.io.InputStreamReader

Hardware:

- Android device (minimum API 15)
- Computer
- Bluetooth antenna (already installed in phone and computer)

User Requirements:

- Install python, all necessary libraries must be installed
- Android Studio
- Knowledge about socket programming (java and python)

Functional requirements:

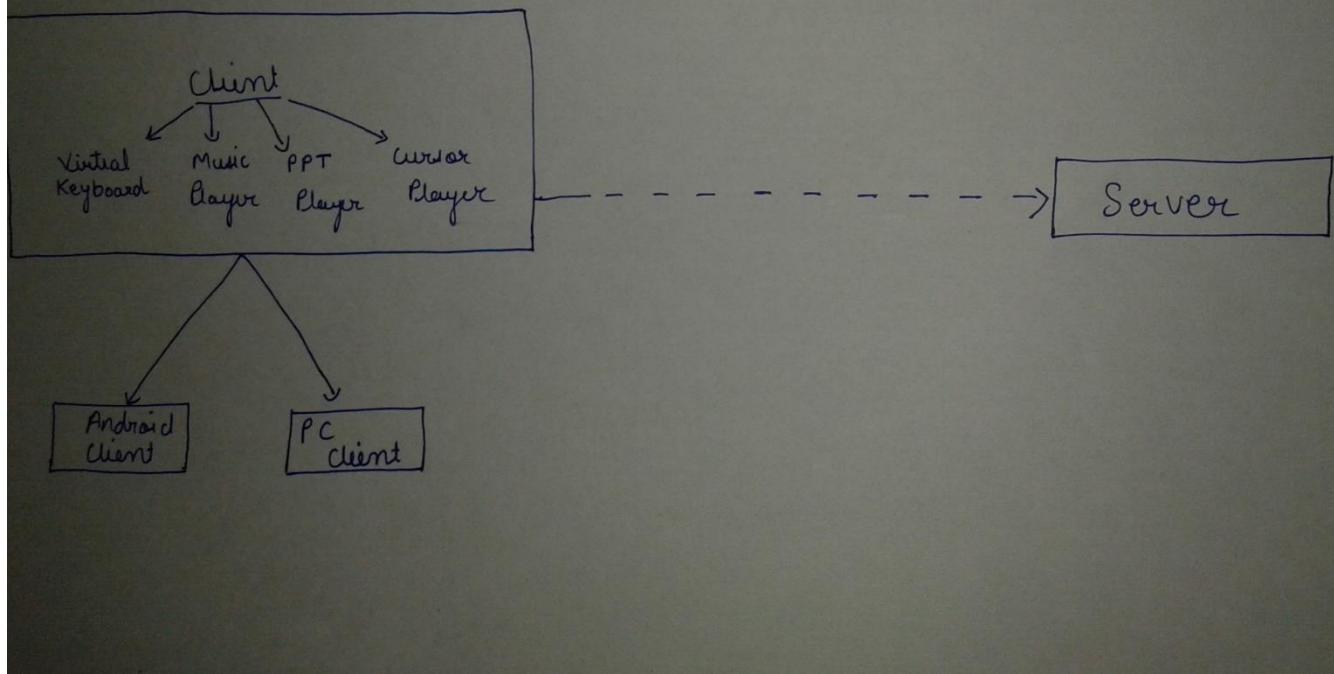
- Touch Pad
- Playlist of songs
- Virtual keyboard
- PPT List
- Bluetooth Search

Non-functional Requirements:

- 8 GB RAM for smooth functioning of Android studio
- Compatible bluetooth drivers for strong connection
- VLC Music Player
- MS Powerpoint
- Android
- PC

DESIGNING:

UML Diagram



```

74 appServerProper - Copy.py - W:\3rdYearSem6\Python\MirrorProject\FINAL PROJECT COE\appServerProper - Copy.py
File Edit Format Run Options Windows Help
return rv

def ppt():
    global pptlist
    global flag3
    global presenti
    getFolderPlaylists1('A:\\ppt',1)
    while True:
        data=conn.recv(1024)
        k=data.split(" ")
        print k
        if int(k[2])<len(pptlist) and int(k[1])==1 and flag3==0 and presenti==0:
            flag3=1
            f=int(k[2])-1
            r=0
            h='A:\\ppt\\'+pptlist[f]
            print h
            app = win32com.client.Dispatch("PowerPoint.Application")
            presentation = app.Presentations.Open(h, ReadOnly=1)
            slide_count = len(presentation.Slides)
            presentation.SlideShowSettings.Run()
            presenti=presenti+1
        elif int(k[2])<len(pptlist) and int(k[1])==1 and presenti!=0:
            if flag3==1:
                del presentation
                app.Quit()
        flag3=1
        r=0
        f=int(k[2])-1
        h='A:\\ppt\\'+pptlist[f]
        print h
        app = win32com.client.Dispatch("PowerPoint.Application")
        presentation = app.Presentations.Open(h, ReadOnly=1)
        slide_count = len(presentation.Slides)
        presentation.SlideShowSettings.Run()
        presenti=presenti+1
        if int(k[1])==5:
            if flag3==1:
                del presentation
                app.Quit()
        break
    if int(k[0])==1 and int(k[1])==1 and flag3==1 and r<slide_count:
        presentation.SlideShowWindow.View.Next()
        r=r+1
    elif int(k[0])==2 and int(k[1])==1 and flag3==1 and r<slide_count:
        presentation.SlideShowWindow.View.Previous()
        r=r-1
Ln: 8 Col: 10
3:23 AM 13-May-16

```

File Edit Format Run Options Windows Help

```

p.play()
p=p
value=r
flag4=1
elif int(data3[0])==1 and int(data3[1])==3 and flag4==1:
    p.play()
elif int(data3[0])==2 and int(data3[1])==3 and flag4==1:
    p.pause()
elif int(data3[0])==3 and int(data3[1])==3 and kk>0 and flag4==1:
    p.stop()
    kk=kk-1
    m="A:\\WhatsAppAudio\\"+muslist[kk]
    print m
    pvvic.MediaPlayer(m)
    p=p
    p.play()
    value=r
elif int(data3[0])==4 and int(data3[1])==3 and kk>0 and flag4==1:
    p.stop()
    kk=kk-1
    m="A:\\WhatsAppAudio\\"+muslist[kk]
    print m
    pvvic.MediaPlayer(m)
    p=p
    p.play()
    value=r
elif int(data3[0])==5 and int(data3[1])==3 and flag4==1:
    p.stop()
if int(data3[1])!=3:
    if flag4==1:
        p.stop()
    break

keybo=""
def keyboard():
    global keybo
    z=""
    while True:
        data=conn.recv(1024)

        data1=data.split("!")
        print data1
        if data1[1]==str(5):
            keybo=""
            break
        if data1[0]==str(1000):
            z=z+keybo
Ln: 8 Col: 10
3:23 AM 13-May-16

```

PARTIAL IMPLEMENTATION

FPClient - Copy.py - W:\3rdYearSem6\Python\MirrorProject\FINAL PROJECT COE\FPClient - Copy.py

```
import os
import socket
import sys, time
import Tkinter as tk
import tkMessageBox
from Tkinter import *
from functools import partial
import vlc
import pickle, win32gui, pyautogui
from socket import *

host='192.168.43.217'
port=5081
addr=(host, port)
s = socket(AF_INET, SOCK_DGRAM)
s.connect((host, port))
#           MUSIC
value=0
i1=0
p1=0
root=0
frame=0
mainflag=2
print addr
def onFrameConfigure(canvas):
    '''Reset the scroll region to encompass the inner frame'''
    canvas.configure(scrollregion=canvas.bbox("all"))
def Tkinter():
    global root
    global frame
    root = tk.Tk()
    root.title("MUSIC PLAYER")
    w = Label(root, text="MUSIC PLAYER", font=("Helvetica", 16), fg="RED")
    w.pack()
    root.geometry("700x600+100+100")

##SCROLLBAR
    canvas = tk.Canvas(root, borderwidth=0, background="#eeeeee")
    frame = tk.Frame(canvas, background="#eeeeee")
    vsb = tk.Scrollbar(root, orient="vertical", command=canvas.yview)
    canvas.configure(yscrollcommand=vsb.set)

    vsb.pack(side="right", fill="y")
    canvas.pack(side="right", fill="both", expand=True)
    canvas.create_window((4,4), window=frame, anchor="nw")

    frame.bind("<Configure>", lambda event, canvas=canvas: onFrameConfigure(canvas))

###PRINTING
root.mainloop()
```

Minor_II - [C:\Users\Admin\Desktop\Minor_II] - [app] - ...\\app\\src\\main\\java\\com\\lightthefuture\\minor_ii\\MainActivity.java - Android Studio 1.4.1

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

Project: Minor_II app src main java com.lightthefuture minor_ii MainActivity.java

Project Structure: DeviceList IconifiedText IconifiedTextListAdapter IconifiedTextView Keyboard ListViewPptAdapter MainActivity ClientThread MainActivity menuoptions menuoptions1 Mouse mousePad.java mousePad Pointer SocketConnection WMP

captures: Build Variants: Favorites:

```
Intent intent = new Intent(MainActivity.this,menuoptions.class);
startActivity(intent);

b1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent intent = new Intent(MainActivity.this,mousePad.class);
        startActivity(intent);
    }
});

public Socket setConnection(String IP, String Port) {
    AlertDialog.Builder builder = new AlertDialog.Builder(this);
    Socket socket = null;
    try {
        socket= new Socket(IP, Integer.parseInt(Port));
    } catch (NumberFormatException e) {
        showError(builder,"Verifica el puerto");
    } catch (UnknownHostException e) {
        showError(builder,"Error al conectar con el servidor");
    } catch (IOException e) {
        showError(builder,"Error al conectar con el servidor");
    }
}
```

TODO: 0: Android Monitor: 0: Messages: Terminal: Event Log: Gradle Console: 147:16 CRLF: UTF-8: Context: <no context>

Gradle build finished in 41s 788ms (26 minutes ago)

12:57 13-05-2016

Minor_II - [C:\Users\Admin\Desktop\Minor_II] - [app] - ...app\src\main\java\com\lightthefuture\minor_ii\menuoptions.java - Android Studio 1.4.1

```
import android.view.View;
import android.widget.Button;
import android.widget.ListView;

import java.io.IOException;

/**
 * Created by Deepanshu on 5/6/2016.
 */
public class menuoptions extends AppCompatActivity {

    Button b1,b2,b3,b4,b5;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        super.onCreate(savedInstanceState);
        setContentView(R.layout.content_menuoptions);
        IconifiedTextListAdapter item = new IconifiedTextListAdapter(this);

        item.addIconifiedText("PowerPoint", ResourcesCompat.getDrawable(getResources(), R.drawable.powerpoint_icon, null));
        item.addIconifiedText("Windows Media Player", ResourcesCompat.getDrawable(getResources(), R.drawable.wmp_icon, null));
        item.addIconifiedText("Mouse Emulator", ResourcesCompat.getDrawable(getResources(), R.drawable.mouse_icon, null));
        item.addIconifiedText("Exit", ResourcesCompat.getDrawable(getResources(), R.drawable.exit, null));
        setListAdapter(item);

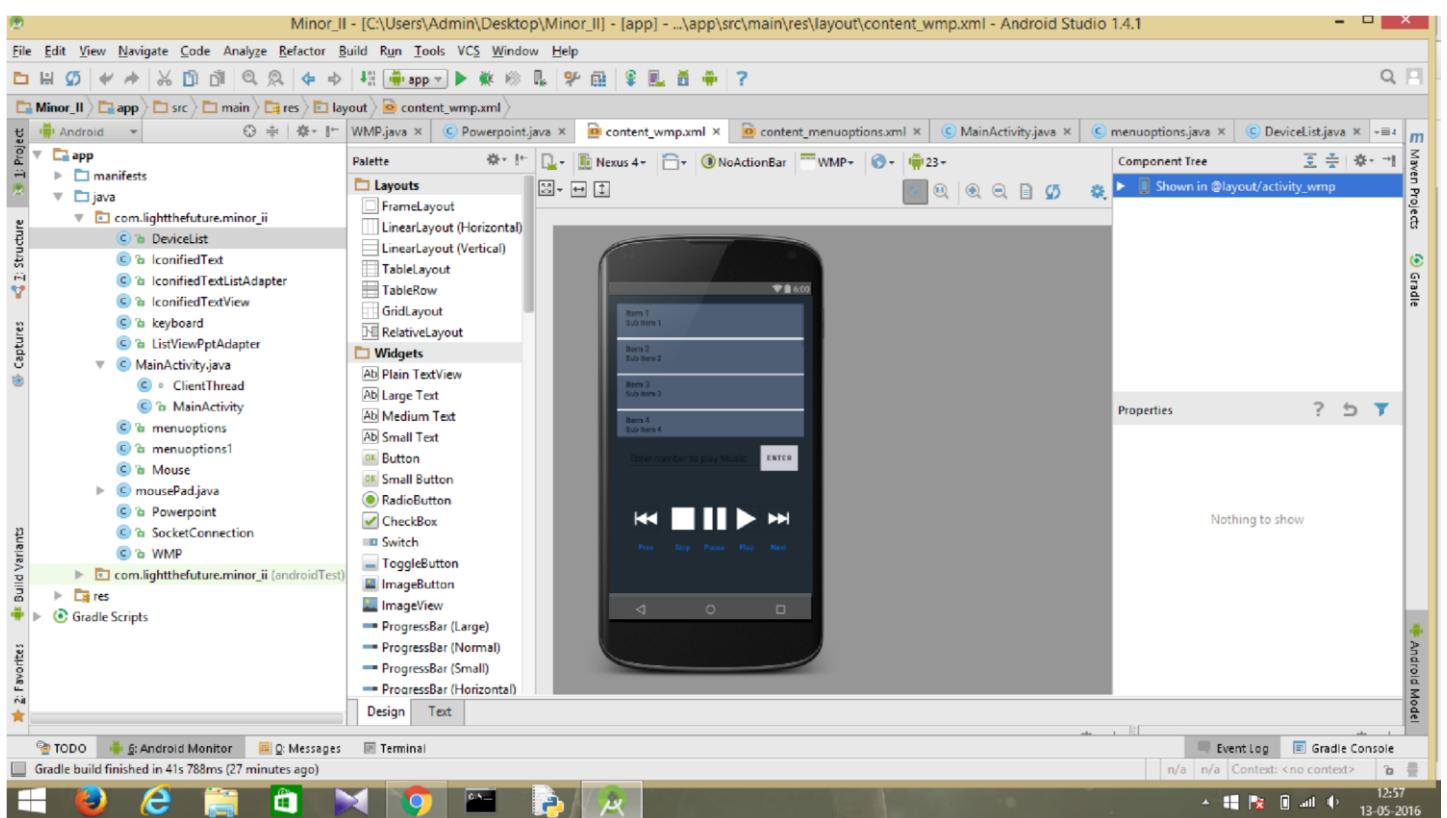
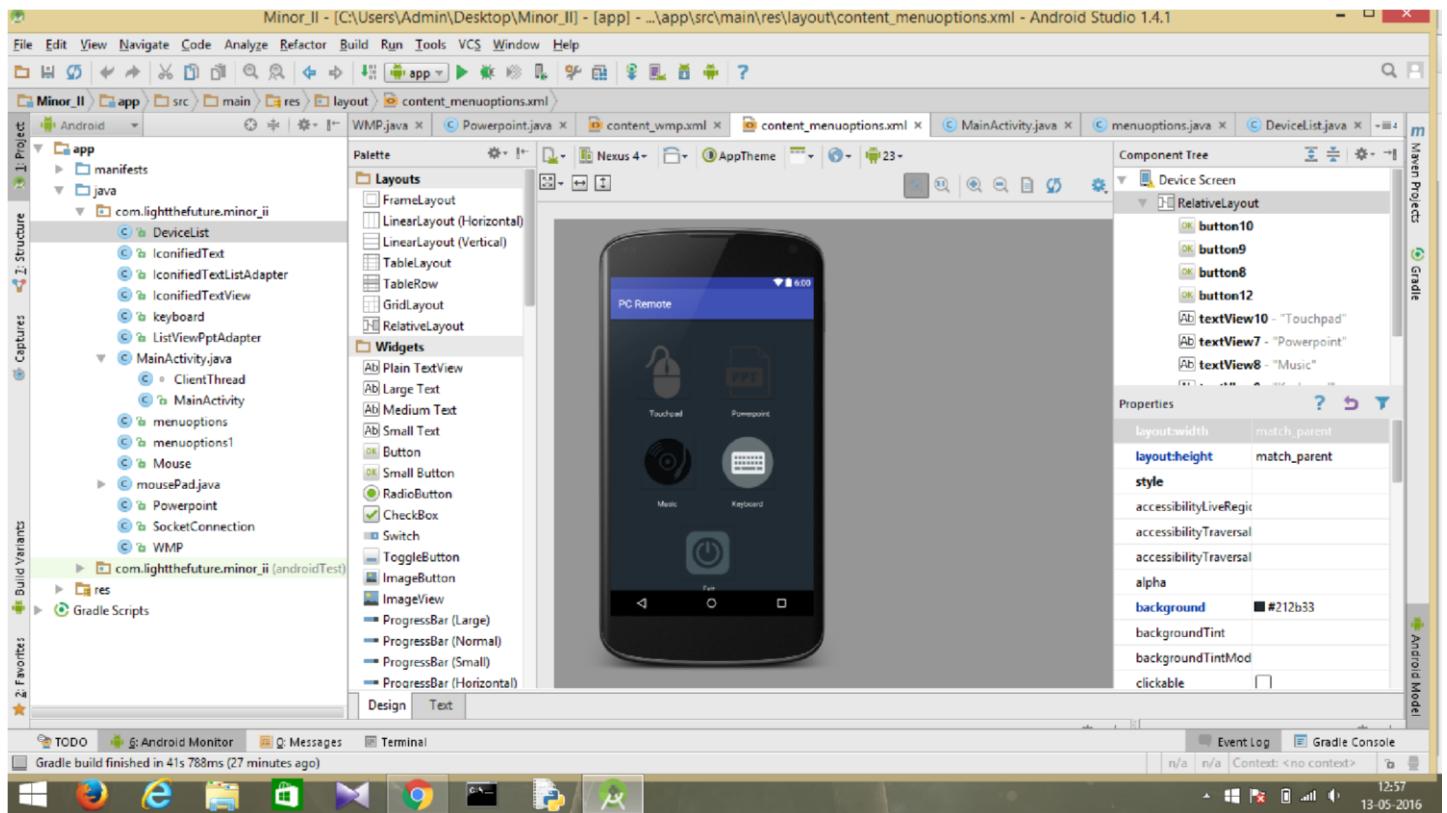
        b1 = (Button) findViewById(R.id.button8);
        b2 = (Button) findViewById(R.id.button9);
        b3 = (Button) findViewById(R.id.button10);
        b4 = (Button) findViewById(R.id.button11);
        b5= (Button)findViewById(R.id.button12);
        b1.setOnClickListener((v) -> {
            try{
                ClientThread.bw.write("2" + "\n"); //Mouse
            }catch (IOException e){
                e.printStackTrace();
            }
        });
    }
}
```

Minor_II - [C:\Users\Admin\Desktop\Minor_II] - [app] - ...app\src\main\java\com\lightthefuture\minor_ii\DeviceList.java - Android Studio 1.4.1

```
public void onActivityResult(int requestCode, int resultCode, Intent data) {
    if(requestCode == ENABLE_BT_REQUEST_CODE) {
        // Bluetooth successfully enabled!
        if(resultCode == Activity.RESULT_OK) {
            Toast.makeText(getApplicationContext(), "Bluetooth is now enabled..!!" +
                    "\n" + "Scanning for Bluetooth devices...", Toast.LENGTH_SHORT).show();

            // Make local device discoverable by other devices
            makeDiscoverable();

            // To discover remote Bluetooth devices
            discoverDevices();
        } else { // RESULT_CANCELED as user refused or failed to enable Bluetooth
            Toast.makeText(getApplicationContext(), "Bluetooth is not enabled.", Toast.LENGTH_SHORT).show();
        }
    }
    else if(requestCode == DISCOVERABLE_BT_REQUEST_CODE) {
        if(resultCode == DISCOVERABLE_DURATION) {
            Toast.makeText(getApplicationContext(), "Your device is now discoverable by other devices for " +
                    DISCOVERABLE_DURATION + " seconds", Toast.LENGTH_SHORT).show();
        }
    }
}
```



Minor_II - [C:\Users\Admin\Desktop\Minor_II] - [app] - ...\\app\\src\\main\\java\\com\\lightthefuture\\minor_II\\WMP.java - Android Studio 1.4.1

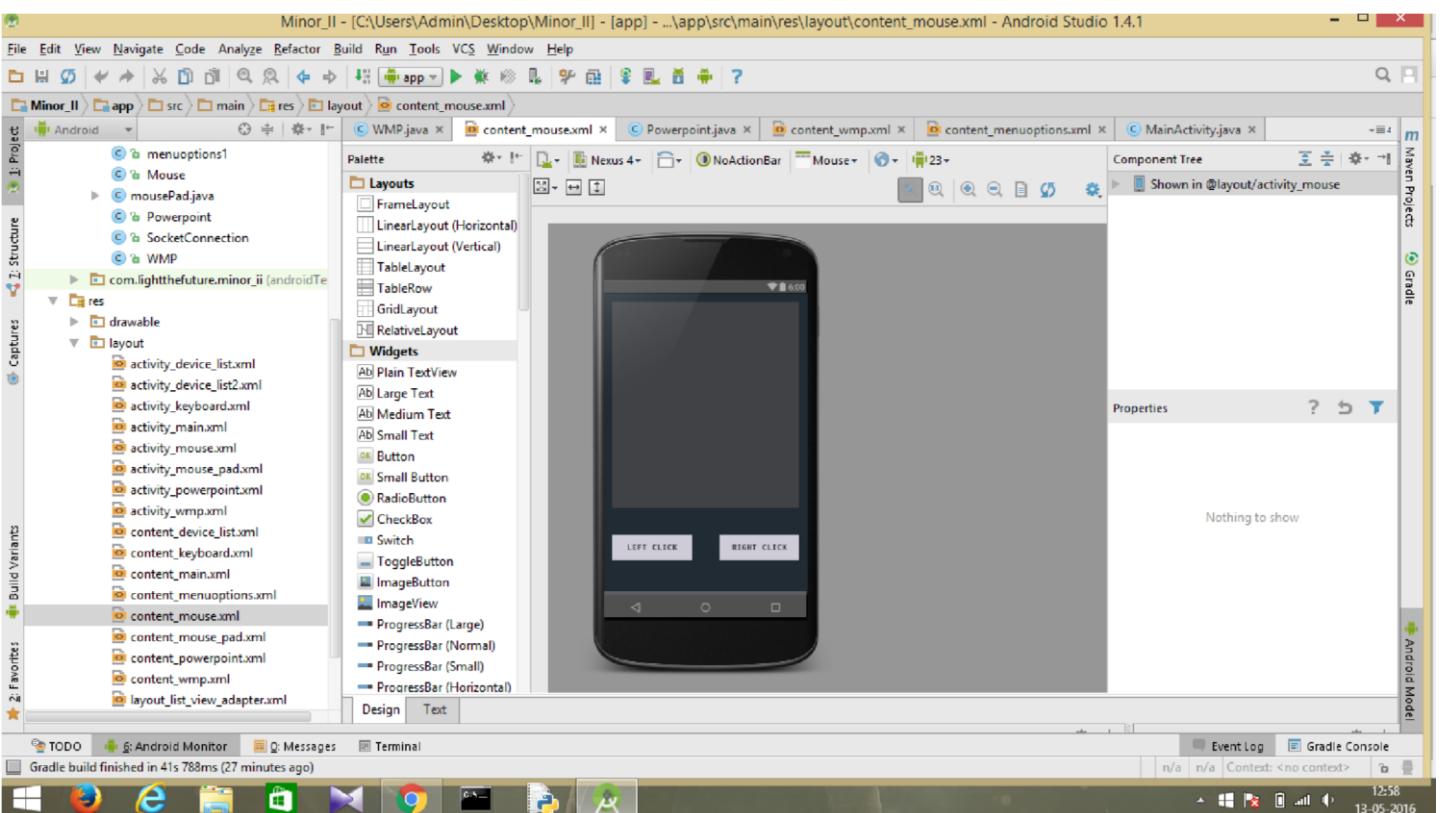
```

super.onCreate(savedInstanceState);
setContentView(R.layout.content_wmp);
AddData();
listView = (ListView) findViewById(R.id.ListView44);
adapter = new ListViewPptAdapter(this, strings);
listView.setAdapter(adapter);
edit=(EditText)findViewById(R.id.editText44);
Button b1=(Button)findViewById(R.id.playmusic);
b1.setOnClickListener((View v) ->
{
    s1 = edit.getText().toString();
    System.out.println("EditText:" + s1);
    try {
        ClientThread.bw.write("n" + " " + "3" + " " + s1);
        ClientThread.bw.flush();
    } catch (Exception exception) {
        exception.printStackTrace();
    } finally {
        //Closing the socket
        try {

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
});
System.out.println("edittext:" + s1);

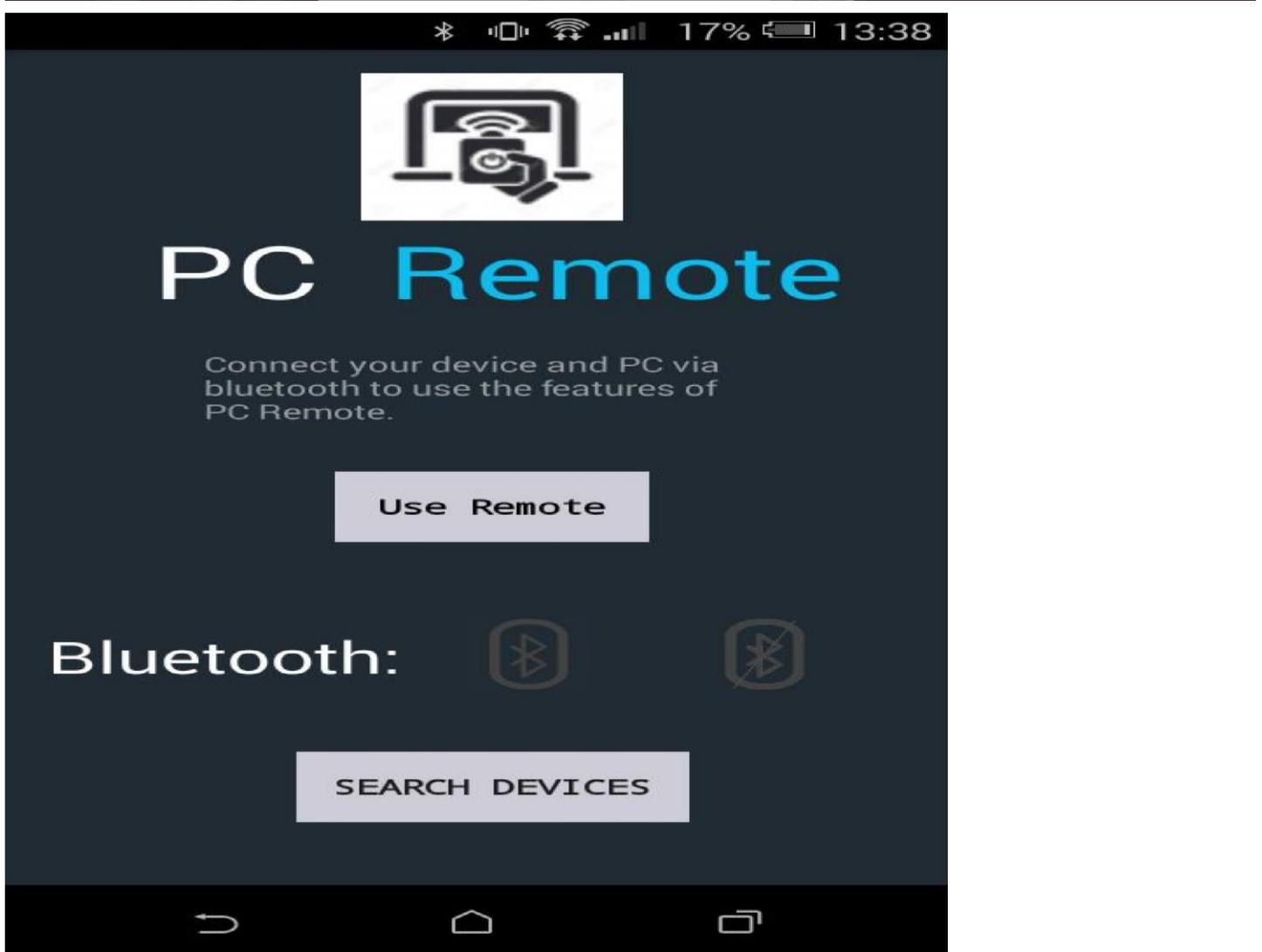
public void pressNext(View v) {
    try {
}
}

```



TESTING:

```
74 Python 2.7.6 Shell
File Edit Shell Debug Options Windows Help
Introduction.ppt
Monte Carlo.ppt
Project Manager.ppt
Project Organization.ppt
Project Planning.ppt
Project Risk ii.ppt
Project Selection.ppt
A:\ppt\Monte Carlo.ppt
A:\ppt\Project Manager.ppt
HAHA
AMIT
AMIT SACHDEVA
AMIT SACHDEVAI
HAHAH !aaw
HAHAH !aaw JAHFBUJED
309.91125 162.9866666667
data
JAI MATA DI
405.53125 322.56
data
JAI MATA DI
395.28625 260.266666667
data
JAI MATA DI
347.47625 307.2
data
JAI MATA DI
432 370
data
right click
332.9625 243.2
data
JAI MATA DI
331 167
data
left Click
Lr: 89 Col: 4
```



* 15% 13:42

Discoverability On

SAMSUNG

18:67:B0:72:09:C0

SAMSUNG

18:67:B0:72:09:C0

lucky's

74:51:BA:E8:DB:F3



* 15% 13:42



Touchpad



Powerpoint



Music

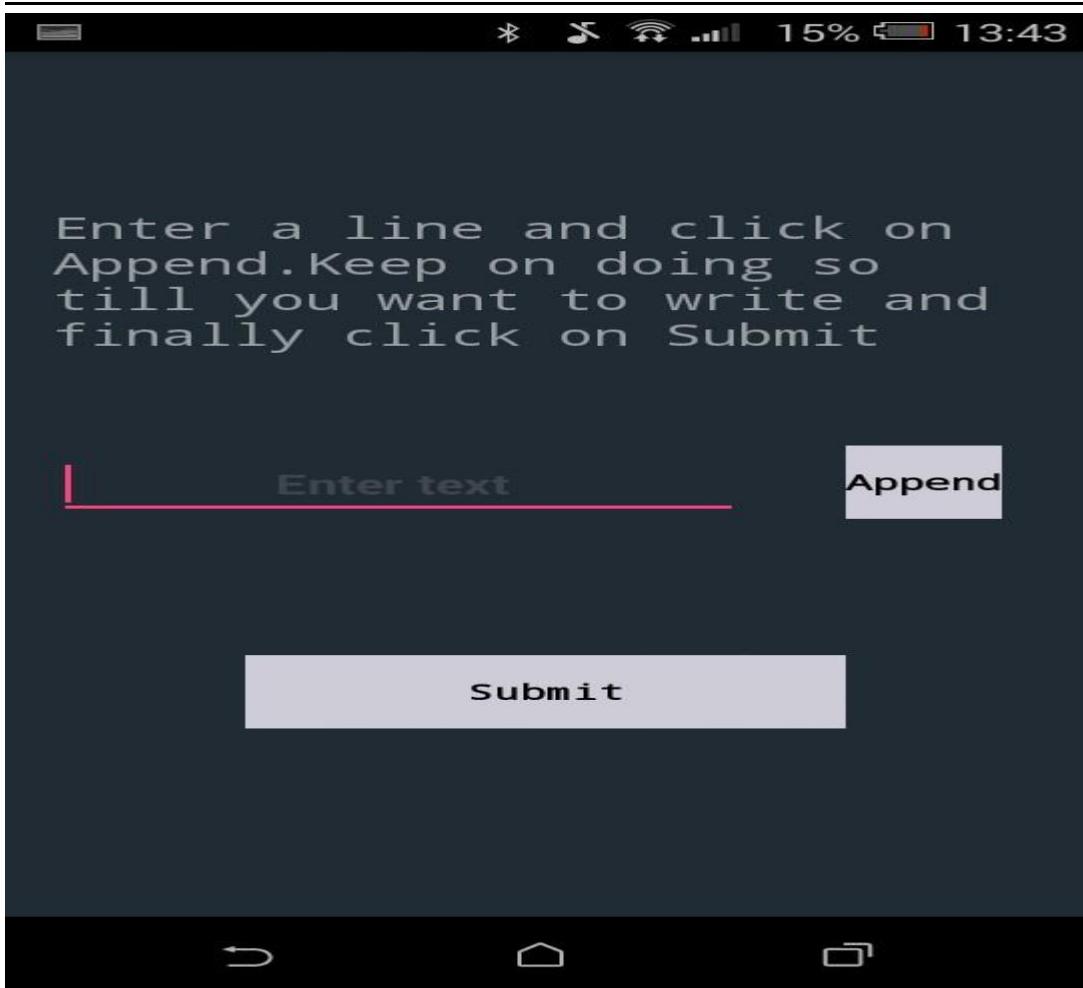
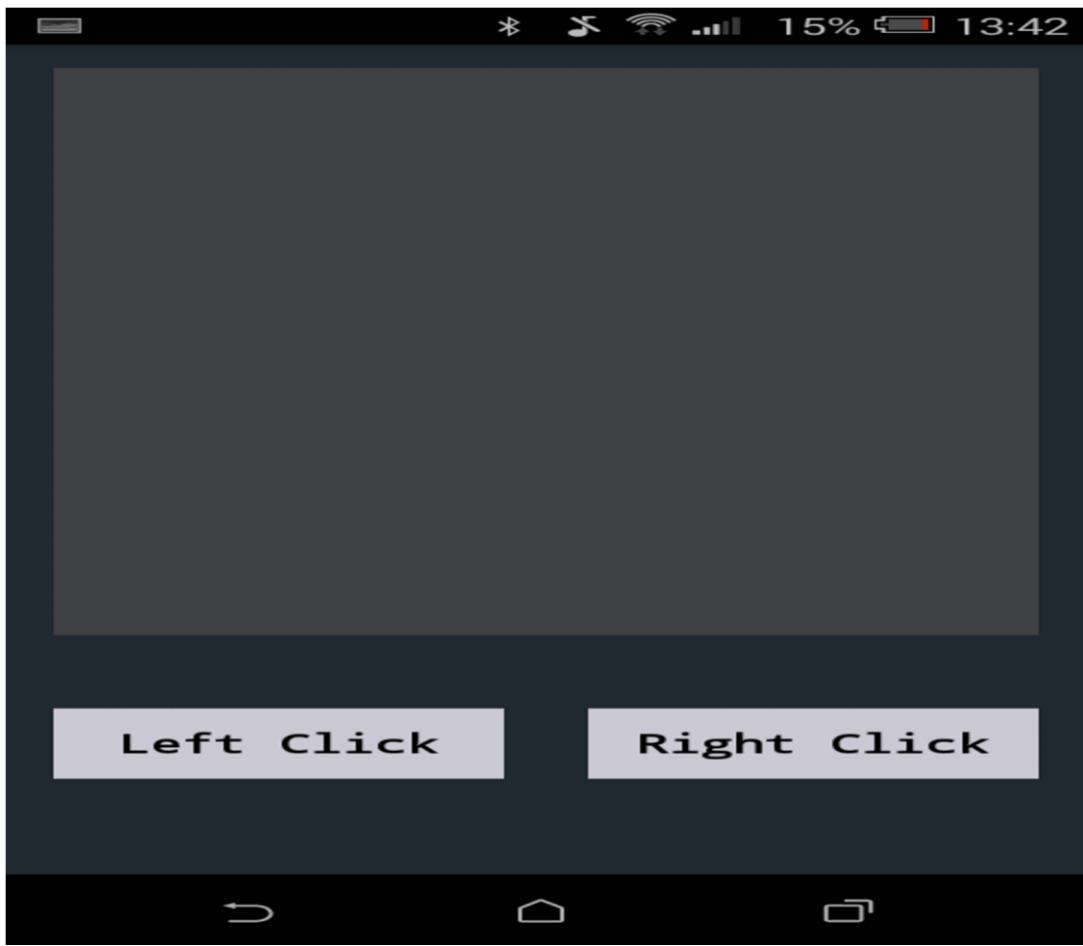


Keyboard

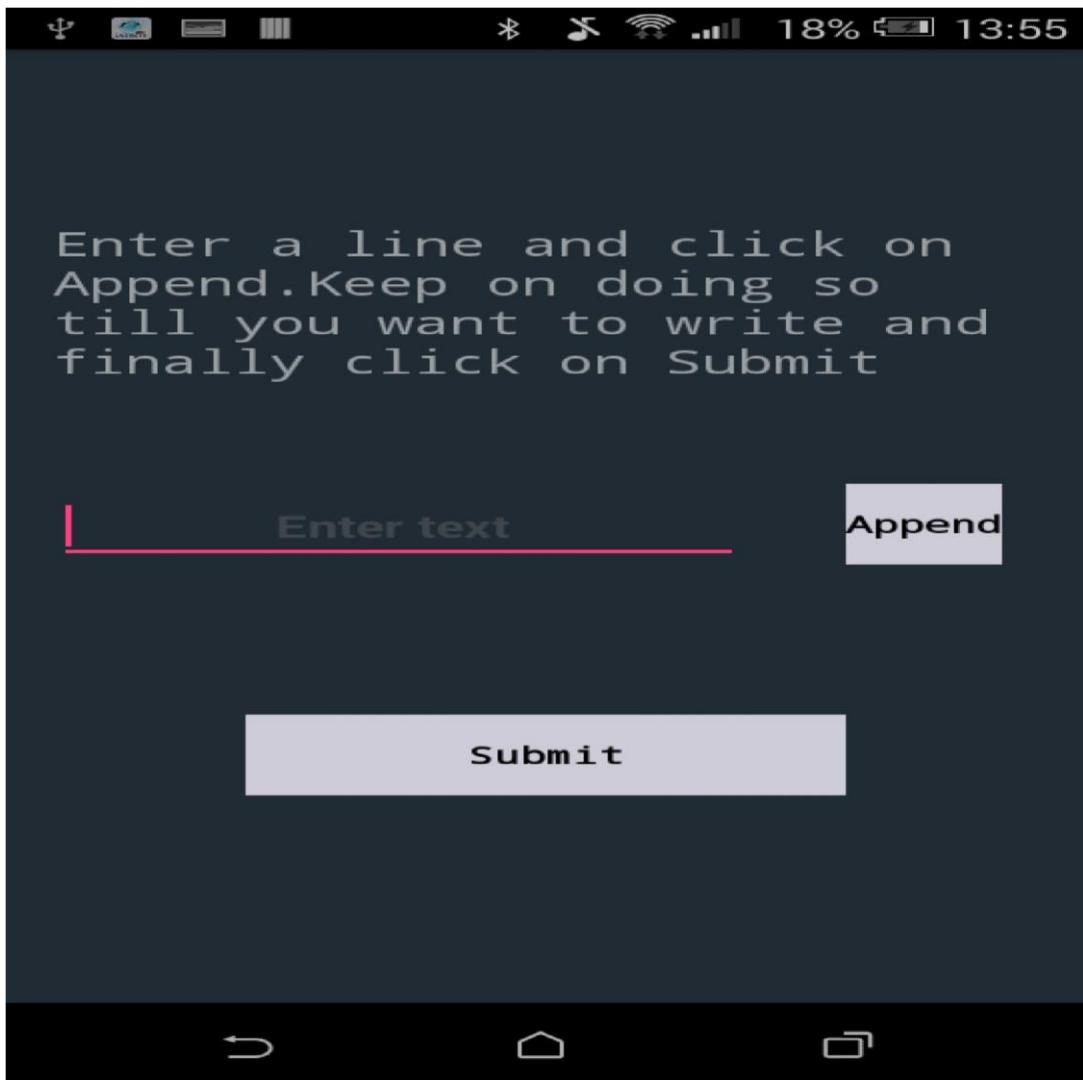


Exit









REFERENCES:

- https://www.youtube.com/watch?v=la3oI4K7_vI
- <https://www.youtube.com/watch?v=og9-X4xIxQ>
 - <http://stackoverflow.com/questions/28609136>
- <http://www.ibm.com/developerworks/linux/tutorials/lpysocks/l-pysocks-pdf.pdf>

FUTURE SCOPE:

This project has immense use in future.

By using PC Remote, one can control Cursor, Keyboard, Mouse, Music Player and PPT of the PC. So, it can be used in professional meetings while giving presentations and also it can be used by teachers and professors while giving lectures. The project can be extended further to share the screen of PC in the app, providing options to download the PPT and music files.

GANTT CHART:

6.3 Project Management | GANNT Chart

	Dec	Jan	Feb	Mar	Apr	May
Requirements						
Design						
Implementation						
Varification						
Maintenance						