Complete Android Application Code

Login Activity: XML

```xml
<?xml version = "1.0" encoding = "utf-8"?>
<RelativeLayout xmlns:android =
"http://schemas.android.com/apk/res/android"
    xmlns:tools = "http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height = "match_parent"
    tools:context="com.example.maids1.MainActivity">

    <TextView android:text = "MAIDS"
        android:layout_width="wrap_content"
        android:layout_height = "wrap_content"
        android:id = "@+id/textview"
        android:textSize = "75dp"
        android:layout_alignParentTop = "true"
        android:layout_centerHorizontal = "true" />

    <TextView
        android:layout_width = "wrap_content"
        android:layout_height = "wrap_content"
        android:text = "Meis Intrusion Detection System Remote
Control"
        android:id = "@+id/textView"
        android:layout_below = "@+id/textview"
        android:layout_centerHorizontal = "true"
        android:textSize = "35dp" />

    <EditText
        android:layout_width = "wrap_content"
        android:layout_height = "wrap_content"
        android:id = "@+id/editText"
        android:hint = "Enter Name"
        android:focusable = "true"
        android:textColorHighlight = "#ff7eff15"
        android:textColorHint = "#ffff25e6"
        android:layout_marginTop = "6dp"
        android:layout_below = "@+id/imageView"
        android:layout_alignParentLeft = "true"
        android:layout_alignParentStart = "true"
        android:layout_alignParentRight = "true"
        android:layout_alignParentEnd = "true" />

    <ImageView
```

```xml
    android:id="@+id/imageView"
    android:layout_width="match_parent"
    android:layout_height="314dp"
    android:layout_below="@+id/textView"
    android:layout_marginTop="4dp"
    android:src="@drawable/a22a" />

<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    android:ems="10"
    android:id="@+id/editText2"
    android:layout_below="@+id/editText"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:layout_alignRight="@+id/editText"
    android:layout_alignEnd="@+id/editText"
    android:textColorHint="#ffff299f"
    android:hint="Password" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Attempts Left:"
    android:id="@+id/textView2"
    android:layout_below="@+id/editText2"
    android:layout_alignParentLeft="true"
    android:layout_alignParentStart="true"
    android:textSize="25dp" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="New Text"
    android:id="@+id/textView3"
    android:layout_alignTop="@+id/textView2"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:layout_alignBottom="@+id/textView2"
    android:layout_toEndOf="@+id/textview"
    android:textSize="25dp"
    android:layout_toRightOf="@+id/textview" />

<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
```

```
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="-3dp"
        android:layout_marginLeft="-3dp"
        android:layout_marginBottom="15dp"
        android:layout_toEndOf="@+id/textview"
        android:layout_toRightOf="@+id/textview"
        android:text="Cancel" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="4dp"
        android:layout_marginRight="4dp"
        android:layout_marginBottom="15dp"
        android:layout_toStartOf="@+id/textview"
        android:layout_toLeftOf="@+id/textview"
        android:text="login" />

</RelativeLayout>
```

Login Activity: Code

```java
package com.example.maids1;

import android.app.Activity;
import android.graphics.Color;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;
import android.content.Intent;
import com.example.maids1.myapplication.R;


public class MainActivity extends Activity {

    Button b1, b2;
    EditText ed1, ed2;
    TextView tx1;
    int counter = 3;

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

        b1 = findViewById(R.id.button);
        ed1 = findViewById(R.id.editText);
        ed2 = findViewById(R.id.editText2);

        b2 = findViewById(R.id.button2);
        tx1 = findViewById(R.id.textView3);
        tx1.setVisibility(View.GONE);

        b1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if (ed1.getText().toString().equals("admin") &&
ed2.getText().toString().equals("admin")) {
                    Toast.makeText(getApplicationContext(),
                            "Login in...",
Toast.LENGTH_SHORT).show();
                    Intent intent = new Intent(v.getContext(),
Main3Activity.class);
```

```java
                        startActivity(intent);

                } else {
                        Toast.makeText(getApplicationContext(),
"Wrong Credentials", Toast.LENGTH_SHORT).show();

                        tx1.setVisibility(View.VISIBLE);
                        tx1.setBackgroundColor(Color.RED);
                        counter--;
                        tx1.setText(Integer.toString(counter));

                        if (counter == 0) {
                            b1.setEnabled(false);
                        }
                    }
                }
            });

        b2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                finish();
            }
        });
    }
}
```

Data Visualization & Action Control Activity: 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"
    tools:context="com.example.maids1.Main3Activity">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="70dp">

        <Button
            android:id="@+id/button1"
            android:layout_width="154dp"
            android:layout_height="76dp"
            android:layout_marginLeft="19dp"
            android:text="Fetch Database Information" />

        <Button
            android:id="@+id/button2"
            android:layout_width="154dp"
            android:layout_height="76dp"
            android:layout_marginLeft="19dp"
            android:layout_toRightOf="@+id/button1"
            android:text="Fetch Remote Photo" />

        <Button
            android:id="@+id/button6"
            android:layout_width="154dp"
            android:layout_height="76dp"
            android:layout_marginLeft="19dp"
            android:layout_toRightOf="@+id/button2"
            android:text="Display Image Locally" />


        <Button
            android:id="@+id/button3"
            android:layout_width="154dp"
            android:layout_height="76dp"
            android:layout_marginLeft="19dp"
            android:layout_toRightOf="@+id/button6"
            android:text="Test Device LEDs" />
```

```xml
        <Button
            android:id="@+id/button4"
            android:layout_width="154dp"
            android:layout_height="76dp"
            android:layout_marginLeft="19dp"
            android:text="Activate MAIDS"
            android:layout_toRightOf="@+id/button3"/>

        <Button
            android:id="@+id/button5"
            android:layout_width="154dp"
            android:layout_height="76dp"
            android:layout_marginLeft="19dp"
            android:text="Deactivate MAIDS"
            android:layout_toRightOf="@+id/button4"
     />

    </RelativeLayout>

    <RelativeLayout
        android:layout_height="match_parent"
        android:layout_width="match_parent">

        <WebView
            android:id="@+id/webview"
            android:layout_width="match_parent"
            android:layout_height="300dp"


            android:layout_marginTop="80dp"
            android:layout_marginEnd="-3dp"
            android:layout_marginRight="-3dp">
        </WebView>


        <ImageView
            android:id="@+id/imageView2"
            android:layout_width="match_parent"
            android:layout_height="300dp"
            android:layout_below="@+id/webview"
            android:src="@drawable/b6" />

    </RelativeLayout>
</RelativeLayout>
```

Data Visualization & Action Control Activity: Code

```java
package com.example.maids1;

import android.annotation.SuppressLint;
import android.app.Activity;
import android.graphics.Bitmap;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.StrictMode;
import android.view.View;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.webkit.WebViewClient;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.Toast;

import com.example.maids1.myapplication.R;
import com.jcraft.jsch.Channel;
import com.jcraft.jsch.ChannelExec;
import com.jcraft.jsch.ChannelSftp;
import com.jcraft.jsch.JSch;
import com.jcraft.jsch.Session;

import java.io.BufferedInputStream;
import java.io.BufferedOutputStream;
import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileOutputStream;
import java.io.OutputStream;
import java.util.Properties;


public class Main3Activity extends Activity {

    private Button mButton1;
    private Button mButton2;
    private Button mButton3;
    private Button mButton4;
    private Button mButton5;
    private Button mButton6;
    private WebView webView;
    private ImageView imageView;
    private Bitmap bimage = null;
    private String username = "pi";
    private String password = "srgawain2264";
```

```java
    private String host = "192.168.0.10";
    private int port = 22;
    private static JSch jsch = new JSch();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main3);
        if (android.os.Build.VERSION.SDK_INT > 9) {
            StrictMode.ThreadPolicy policy = new
StrictMode.ThreadPolicy.Builder().permitAll().build();
            StrictMode.setThreadPolicy(policy);
        }

        mButton1 = findViewById(R.id.button1);
        mButton1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(getApplicationContext(),
                        "Retrieving DB Information...",
Toast.LENGTH_SHORT).show();
                WebView htmlWebView =
findViewById(R.id.webview);
                htmlWebView.setWebViewClient(new
CustomWebViewClient());
                WebSettings webSetting =
htmlWebView.getSettings();
                webSetting.setJavaScriptEnabled(true);
                webSetting.setDisplayZoomControls(true);
                htmlWebView.loadUrl("https://singular-gar-
5555.dataplicity.io/maidsintrusion.php");
            }
        });


        mButton2 = findViewById(R.id.button2);
        mButton2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                try {
                    Toast.makeText(getApplicationContext(),
                            "Retrieving REMOTE Intrusion
Photo...", Toast.LENGTH_SHORT).show();
                    getRemoteFile();
                    Toast.makeText(getApplicationContext(),
                            "REMOTE Intrusion Photo
Retrieved...", Toast.LENGTH_SHORT).show();
```

```java
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });

    mButton6 = findViewById(R.id.button6);
    mButton6.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            try {

                Toast.makeText(getApplicationContext(),
                        "Displaying Intrusion Image...",
Toast.LENGTH_SHORT).show();
                ImageView image =
findViewById(R.id.imageView2);
                image.setImageResource(R.drawable.lion3);

            } catch (Exception e) {
                e.printStackTrace();
            }
        }

    });


    mButton3 = findViewById(R.id.button3);
    mButton3.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Toast.makeText(getApplicationContext(),
                    "Testing LEDs Remotely...",
Toast.LENGTH_LONG).show();
            // Do something
            @SuppressLint("StaticFieldLeak") final
AsyncTask<Integer, Void, Void> execute = new AsyncTask<Integer,
Void, Void>() {
                @Override
                protected Void doInBackground(Integer...
params) {
                    try {

testLedsCommand(username,password,host,port);
                    } catch (Exception e) {
                        e.printStackTrace();
                    }
```

```java
                        return null;
                    }
                }.execute(1);

                Toast.makeText(getApplicationContext(),
                        "LEDs Test Completed...",
Toast.LENGTH_LONG).show();
            }
        });

        mButton4 = findViewById(R.id.button4);
        mButton4.setOnClickListener(new View.OnClickListener() {
            @SuppressLint("StaticFieldLeak")
            @Override
            public void onClick(View view) {
                Toast.makeText(getApplicationContext(),
                        "Activating MAIDS Remotely...",
Toast.LENGTH_SHORT).show();
                // Do something
                final AsyncTask<Integer, Void, Void> execute =
new AsyncTask<Integer, Void, Void>() {
                    @Override
                    protected Void doInBackground(Integer...
params) {
                        try {

maidsOnCommand(username,password,host,port);
                        } catch (Exception e) {
                            e.printStackTrace();
                        }
                        return null;
                    }
                }.execute(1);
            }
        });

        mButton5 = findViewById(R.id.button5);
        mButton5.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Toast.makeText(getApplicationContext(),
                        "Deactivating MAIDS Remotely...",
Toast.LENGTH_SHORT).show();
                // Do something
                @SuppressLint("StaticFieldLeak") final
AsyncTask<Integer, Void, Void> execute = new AsyncTask<Integer,
Void, Void>() {
```

```java
                    @Override
                    protected Void doInBackground(Integer...
params) {
                            try {

maidsOffCommand(username,password,host,port);
                            } catch (Exception e) {
                                e.printStackTrace();
                            }
                            return null;
                    }
                }.execute(1);
            }
        });
    }

    class CustomWebViewClient extends WebViewClient {
        @Override
        public boolean shouldOverrideUrlLoading(WebView view,
String url) {
            view.loadUrl(url);
            return true;
        }
    }

    public String testLedsCommand(String username, String
password, String host, int port)
            throws Exception {
        //JSch jsch = new JSch();
        Session session = jsch.getSession(username, host, port);
        session.setPassword(password);
        // Avoid asking for key confirmation
        Properties prop = new Properties();
        prop.put("StrictHostKeyChecking", "no");
        session.setConfig(prop);
        session.connect();
        // SSH Channel
        ChannelExec channelssh = (ChannelExec)
session.openChannel("exec");
        ByteArrayOutputStream baos = new
ByteArrayOutputStream();
        channelssh.setOutputStream(baos);
        // Execute command
        channelssh.setCommand("python3 testleds.py");
        channelssh.connect();
        channelssh.disconnect();
        return baos.toString();
```

```java
    }

    public String maidsOnCommand(String username, String
password, String hostname, int port)
            throws Exception {
        //JSch jsch = new JSch();
        Session session = jsch.getSession(username, hostname,
port);
        session.setPassword(password);
        // Avoid asking for key confirmation
        Properties prop = new Properties();
        prop.put("StrictHostKeyChecking", "no");
        session.setConfig(prop);
        session.connect();
        // SSH Channel
        ChannelExec channelssh = (ChannelExec)
session.openChannel("exec");
        ByteArrayOutputStream baos = new
ByteArrayOutputStream();
        channelssh.setOutputStream(baos);
        // Execute command
        channelssh.setCommand("python3
maids_final_python_code_22102019_bak1.py");
        channelssh.connect();
        channelssh.disconnect();
        return baos.toString();
    }

    public void maidsOffCommand(String username, String
password, String hostname, int port)
            throws Exception {
        //JSch jsch = new JSch();
        Session session = jsch.getSession(username, hostname,
port);
        session.setPassword(password);
        // Avoid asking for key confirmation
        Properties prop = new Properties();
        prop.put("StrictHostKeyChecking", "no");
        session.setConfig(prop);
        session.connect();
        // SSH Channel
        ChannelExec channelssh = (ChannelExec)
session.openChannel("exec");
        ByteArrayOutputStream baos = new
ByteArrayOutputStream();
        channelssh.setOutputStream(baos);
```

```java
        // Execute command
        channelssh.setCommand("maidsoff.py");
        channelssh.connect();
        channelssh.disconnect();
        return baos.toString();
    }


    public void getRemoteFile() throws Exception {

        String REMOTEDIR = "/home/pi/webcam/";
        String REMOTEFILE = "image.jpg";
        String LOCALDIR = "/home/maids1/...";

        //String command1="scp
pi@192.168.0.12:/home/pi/webcam/image.jpg
c:\\Users\\cfpm\\Desktop";

        try{
            //JSch jsch = new JSch();
            Session session = jsch.getSession(username, host,
port);
            session.setPassword(password);
            java.util.Properties config = new
java.util.Properties();
            config.put("StrictHostKeyChecking", "no");
            session.setConfig(config);
            session.connect();
            Toast.makeText(getApplicationContext(),
                    "Connected to Remote system...",
Toast.LENGTH_SHORT).show();
            Channel channel=session.openChannel("sftp");
            channel.connect();
            ChannelSftp channelSftp = (ChannelSftp) channel;
            channelSftp.cd(REMOTEDIR);
            Toast.makeText(getApplicationContext(),
                    "Downloading File...",
Toast.LENGTH_SHORT).show();
            channelSftp.get(REMOTEFILE, LOCALDIR);
            Toast.makeText(getApplicationContext(),
                    "File Downloaded!",
Toast.LENGTH_SHORT).show();
            channelSftp.exit();
            session.disconnect();

            /* byte[] buffer = new byte[1024];
```

```
            BufferedInputStream bis = new
BufferedInputStream(channelSftp.get(REMOTEFILE));
            File newFile = new
File("/home/maids1/Downloads/image.jpg");
            OutputStream os = new FileOutputStream(newFile);
            BufferedOutputStream bos = new
BufferedOutputStream(os);
            int readCount;
            while ((readCount = bis.read(buffer)) > 0) {
                Toast.makeText(getApplicationContext(),
                    "Writing Fike...",
Toast.LENGTH_SHORT).show();
                bos.write(buffer, 0, readCount);
            }
            bis.close();
            bos.close(); */
        }catch(Exception e){
            e.printStackTrace();
        }

    }

    }
```