

```

package com.example.clap;

import android.Manifest;
import android.content.pm.PackageManager;
import android.media.MediaRecorder;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ImageButton;
import android.widget.Spinner;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import java.io.IOException;

public class MainActivity extends AppCompatActivity {

    final int RECORD_AUDIO = 0;
    ImageButton btnStart;
    Spinner spinDuration;
    int clipDuration = 0;
    LongOperation recordAudioSync = null;
    String[] durationItems = {"5", "10", "15", "20"};

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        initializeHandles();
        ArrayAdapter<String> spinDurationAdapter = new ArrayAdapter<>(this,
        android.R.layout.simple_spinner_item, durationItems);

        spinDurationAdapter.setDropDownViewResource(android.R.layout.simple_spinner_d
        ropdown_item);
        spinDuration.setAdapter(spinDurationAdapter);
        btnStart.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                clipDuration =
                Integer.parseInt(spinDuration.getSelectedItem().toString());
                if (ActivityCompat.checkSelfPermission(MainActivity.this,
                Manifest.permission.RECORD_AUDIO) != PackageManager.PERMISSION_GRANTED) {
                    ActivityCompat.requestPermissions(MainActivity.this, new
                    String[] {Manifest.permission.RECORD_AUDIO}, RECORD_AUDIO);
                } else {
                    recordAudioSync = new LongOperation();
                    recordAudioSync.execute("");
                }
            }
        });
    }

    private void initializeHandles() {

```

```

        btnStart = findViewById(R.id.start);
        spinDuration = findViewById(R.id.duration);
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        if(recordAudioSync != null && recordAudioSync.getStatus() !=
AsyncTask.Status.FINISHED) {
            recordAudioSync.done();
            recordAudioSync.cancel(true);
        }
    }

    @Override
    protected void onPause() {
        super.onPause();
        if(recordAudioSync != null && recordAudioSync.getStatus() !=
AsyncTask.Status.FINISHED) {
            recordAudioSync.done();
            recordAudioSync.cancel(true);
        }
    }

    private class LongOperation extends AsyncTask<String, Void, String> {

        MediaRecorder recorder;
        int clapDetectedNumber;

        @Override
        protected String doInBackground(String... strings) {
            recordAudio();
            return "" + clapDetectedNumber;
        }

        @Override
        protected void onPreExecute() {
            btnStart.setImageResource(R.drawable.mic_on);
            btnStart.setEnabled(false);
        }

        @Override
        protected void onPostExecute(String s) {
            Toast.makeText(MainActivity.this, "You Clapped: " + s + " times",
Toast.LENGTH_SHORT).show();
            btnStart.setEnabled(true);
            btnStart.setImageResource(R.drawable.mic_off);
        }

        private void recordAudio() {
            recorder = new MediaRecorder();
            recorder.setAudioSource(MediaRecorder.AudioSource.MIC);
            recorder.setOutputFormat(MediaRecorder.OutputFormat.THREE_GPP);
            recorder.setAudioEncoder(MediaRecorder.AudioEncoder.AMR_NB);
            recorder.setOutputFile("/data/data/" + getPackageName() +
"/music.3gp");
            int startAmplitude = 0;

```

```

        int finishAmplitude;
        int amplitudeThreshold = 18000;
        int counter = 0;
        try {
            recorder.prepare();
            recorder.start();
            startAmplitude = recorder.getMaxAmplitude();
        } catch (IOException e) {
            e.printStackTrace();
        }
        do{
            if(isCancelled()){
                break;
            }
            counter++;
            waitSome();
            finishAmplitude = recorder.getMaxAmplitude();
            if(finishAmplitude >= amplitudeThreshold){
                clapDetectedNumber++;
            }
        }while(counter < (clipDuration * 4));
        done();
    }

    private void done() {
        if(recorder != null){
            recorder.stop();
            recorder.release();
        }
    }

    private void waitSome() {
        try {
            Thread.sleep(250);
        } catch (InterruptedException e) {
            e.printStackTrace();
        }
    }
}
}
}

```

MANIFEST

```

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <uses-permission android:name="android.permission.RECORD_AUDIO" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"

```

```

        android:theme="@style/Theme.ClapCounter">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>

</manifest>

```

THEMES

```

<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Theme.ClapCounter"
parent="Theme.MaterialComponents.DayNight.DarkActionBar">
        <!-- Primary brand color. -->
        <item name="colorPrimary">@color/purple_500</item>
        <item name="colorPrimaryVariant">@color/purple_700</item>
        <item name="colorOnPrimary">@color/white</item>
        <!-- Secondary brand color. -->
        <item name="colorSecondary">@color/teal_200</item>
        <item name="colorSecondaryVariant">@color/teal_700</item>
        <item name="colorOnSecondary">@color/black</item>
        <!-- Status bar color. -->
        <item name="android:statusBarColor"
tools:targetApi="1">?attr/colorPrimaryVariant</item>
        <!-- Customize your theme here. -->
    </style>
</resources>

```

STRING

```

<resources>
    <string name="app_name">ClapCounter</string>
</resources>

```

COLOR

```

<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="purple_200">#FFBB86FC</color>
    <color name="purple_500">#FF6200EE</color>
    <color name="purple_700">#FF3700B3</color>
    <color name="teal_200">#FF03DAC5</color>
    <color name="teal_700">#FF018786</color>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
</resources>

```

THEMES NIGHT

```

<resources xmlns:tools="http://schemas.android.com/tools">
    <!-- Base application theme. -->
    <style name="Theme.ClapCounter"
parent="Theme.MaterialComponents.DayNight.DarkActionBar">
        <!-- Primary brand color. -->
        <item name="colorPrimary">@color/purple_200</item>
        <item name="colorPrimaryVariant">@color/purple_700</item>
        <item name="colorOnPrimary">@color/black</item>
        <!-- Secondary brand color. -->
        <item name="colorSecondary">@color/teal_200</item>
        <item name="colorSecondaryVariant">@color/teal_200</item>
        <item name="colorOnSecondary">@color/black</item>
        <!-- Status bar color. -->
        <item name="android:statusBarColor"
tools:targetApi="1">?attr/colorPrimaryVariant</item>
        <!-- Customize your theme here. -->
    </style>
</resources>

```

ACTIVITY

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:gravity="center"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <ImageButton
        android:id="@+id/start"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="@drawable/mic_off" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Select Recording Duration (in seconds)"
        android:textSize="18sp" />

    <Spinner
        android:id="@+id/duration"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />

</LinearLayout>

```

FOR

```

<vector xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:aapt="http://schemas.android.com/aapt"
    android:width="108dp"

```

```

        android:height="108dp"
        android:viewportWidth="108"
        android:viewportHeight="108">
        <path android:pathData="M31,63.928c0,0 6.4,-11 12.1,-13.1c7.2,-2.6 26,-
1.4 26,-1.4l38.1,38.1L107,108.928l-32,-1L31,63.928z">
            <aapt:attr name="android:fillColor">
                <gradient
                    android:endX="85.84757"
                    android:endY="92.4963"
                    android:startX="42.9492"
                    android:startY="49.59793"
                    android:type="linear">
                        <item
                            android:color="#44000000"
                            android:offset="0.0" />
                        <item
                            android:color="#00000000"
                            android:offset="1.0" />
                    </gradient>
                </aapt:attr>
            </path>
        <path
            android:fillColor="#FFFFFF"
            android:fillType="nonZero"
            android:pathData="M65.3,45.828l3.8,-6.6c0.2,-0.4 0.1,-0.9 -0.3,-1.1c-
0.4,-0.2 -0.9,-0.1 -1.1,0.3l-3.9,6.7c-6.3,-2.8 -13.4,-2.8 -19.7,0l-3.9,-6.7c-
0.2,-0.4 -0.7,-0.5 -1.1,-0.3C38.8,38.328 38.7,38.828
38.9,39.228l3.8,6.6C36.2,49.428 31.7,56.028 31,63.928h46C76.3,56.028
71.8,49.428 65.3,45.828zM43.4,57.328c-0.8,0 -1.5,-0.5 -1.8,-1.2c-0.3,-0.7 -
0.1,-1.5 0.4,-2.1c0.5,-0.5 1.4,-0.7 2.1,-0.4c0.7,0.3 1.2,1
1.2,1.8C45.3,56.528 44.5,57.328 43.4,57.328L43.4,57.328zM64.6,57.328c-0.8,0 -
1.5,-0.5 -1.8,-1.2s-0.1,-1.5 0.4,-2.1c0.5,-0.5 1.4,-0.7 2.1,-0.4c0.7,0.3
1.2,1 1.2,1.8C66.5,56.528 65.6,57.328 64.6,57.328L64.6,57.328z"
            android:strokeWidth="1"
            android:strokeColor="#00000000" />
    </vector>

```

BACK

```

<?xml version="1.0" encoding="utf-8"?>
<vector xmlns:android="http://schemas.android.com/apk/res/android"
    android:width="108dp"
    android:height="108dp"
    android:viewportWidth="108"
    android:viewportHeight="108">
    <path
        android:fillColor="#3DDC84"
        android:pathData="M0,0h108v108h-108z" />
    <path
        android:fillColor="#00000000"
        android:pathData="M9,0L9,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path

```

```
        android:fillColor="#00000000"
        android:pathData="M19,0L19,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M29,0L29,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M39,0L39,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M49,0L49,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M59,0L59,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M69,0L69,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M79,0L79,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M89,0L89,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M99,0L99,108"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,9L108,9"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,19L108,19"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,29L108,29"
```

```
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,39L108,39"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,49L108,49"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,59L108,59"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,69L108,69"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,79L108,79"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,89L108,89"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M0,99L108,99"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M19,29L89,29"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M19,39L89,39"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M19,49L89,49"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
    <path
        android:fillColor="#00000000"
        android:pathData="M19,59L89,59"
        android:strokeWidth="0.8"
        android:strokeColor="#33FFFFFF" />
```



```
<path
    android:fillColor="#00000000"
    android:pathData="M19,69L89,69"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M19,79L89,79"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M29,19L29,89"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M39,19L39,89"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M49,19L49,89"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M59,19L59,89"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M69,19L69,89"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
<path
    android:fillColor="#00000000"
    android:pathData="M79,19L79,89"
    android:strokeWidth="0.8"
    android:strokeColor="#33FFFFFF" />
</vector>
```

