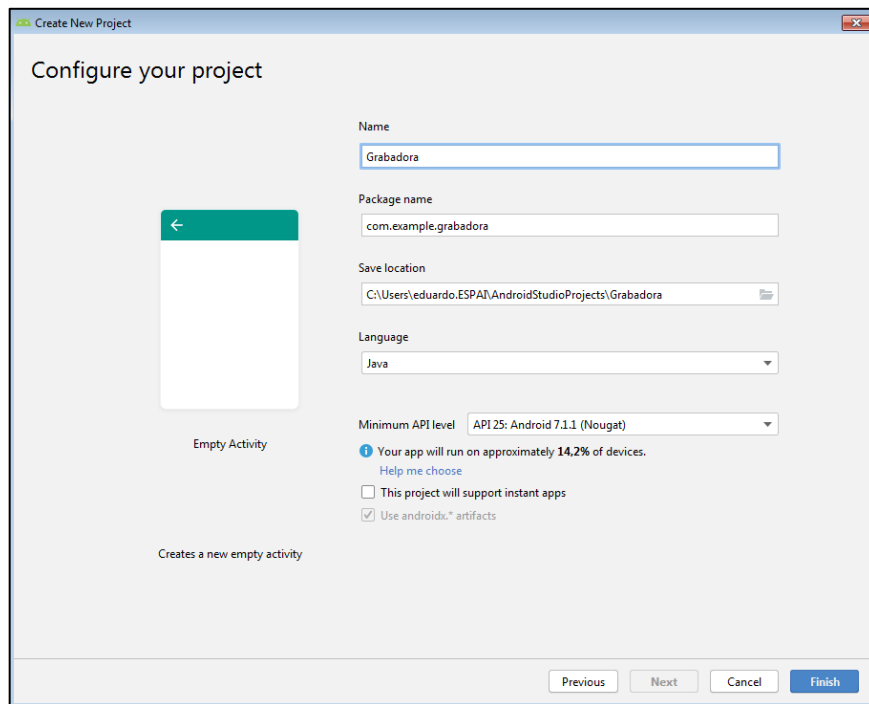


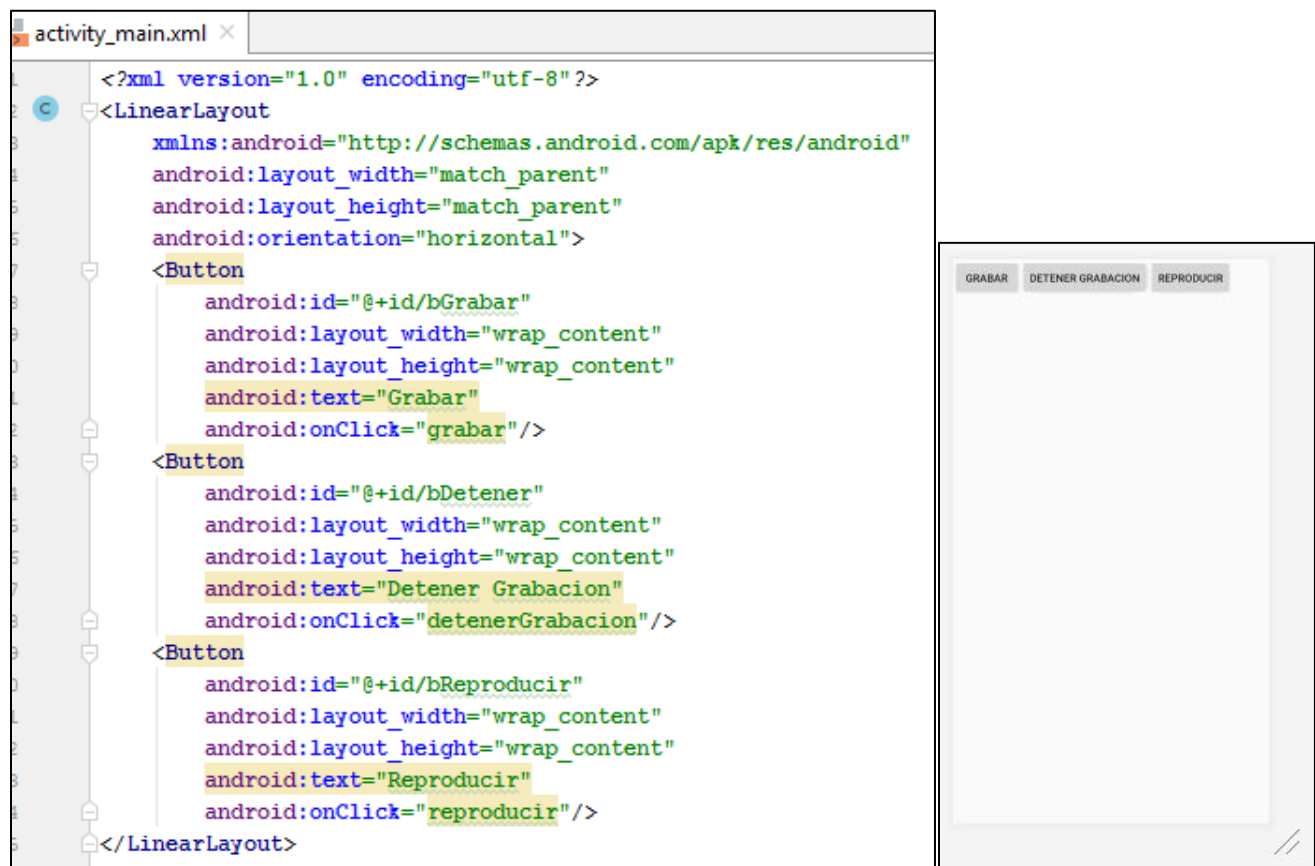
PRACTICA 12: GRABADORA

Parte I: Grabación de audio utilizando MediaRecorder

Paso 1. Crea un nuevo proyecto con nombre *Grabadora*. El nombre del paquete ha de ser *com.example.grabadora*.



Paso 2. Reemplaza el *Layoutactivity_main.xml* por el siguiente código:



Paso 3. Reemplaza el código de la actividad por:

```
public class MainActivity extends AppCompatActivity {

    private static final String LOG_TAG = "Grabadora";
    private MediaRecorder mediaRecorder;
    private MediaPlayer mediaPlayer;

    private static String fichero;

    @Override public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        fichero = this.getExternalFilesDir( type: null).getAbsolutePath()+"/audio.3gp";
    }

    public void grabar(View v) {
        if (ActivityCompat.checkSelfPermission( context: this, Manifest.permission.RECORD_AUDIO)
            != PackageManager.PERMISSION_GRANTED) {
            ActivityCompat.requestPermissions( activity: this, new String[] { Manifest.permission.RECORD_AUDIO },
                requestCode: 10);
        } else {
            grabar2();
        }
    }

    @Override public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions,
        @NonNull int[] grantResults) {
        super.onRequestPermissionsResult(requestCode, permissions, grantResults);
        if (requestCode == 10) {
            if (grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                grabar2();
            } else {
                //User denied Permission.
            }
        }
    }
}
```

```
public void grabar2() {
    mediaRecorder = new MediaRecorder();
    mediaRecorder.setOutputFile(fichero);
    mediaRecorder.setAudioSource(MediaRecorder.AudioSource.MIC);
    mediaRecorder.setOutputFormat(MediaRecorder.OutputFormat.THREE_GPP);
    mediaRecorder.setAudioEncoder(MediaRecorder.AudioEncoder.AMR_NB);
    try {
        mediaRecorder.prepare();
    } catch (IOException e) {
        Log.e(LOG_TAG, msg: "Fallo en grabación");
    }
    mediaRecorder.start();
}

public void detenerGrabacion(View view) {
    mediaRecorder.stop();
    mediaRecorder.release();
}

public void reproducir(View view) {
    mediaPlayer = new MediaPlayer();
    try {
        mediaPlayer.setDataSource(fichero);
        mediaPlayer.prepare();
        mediaPlayer.start();
    } catch (IOException e) {
        Log.e(LOG_TAG, msg: "Fallo en reproducción");
    }
}
}
```

Comenzamos declarando dos objetos de las clases **MediaRecorder** y **MediaPlayer** que serán usados para la grabación y reproducción respectivamente. También se declaran dos **String**: La constante **LOG_TAG** será utilizada como etiqueta para identificar nuestra aplicación en el **fichero** de Log. La variable **fichero** identifica en nombre del fichero donde se guardará la grabación. Este fichero se almacenará en una carpeta especialmente creada para nuestra aplicación. NOTA: En la unidad 8 se introducirá la gestión de ficheros. En el método **onCreate()** no se realiza ninguna acción especial.

A continuación se han introducido tres métodos que serán ejecutados al pulsar los botones de nuestro Layout. El significado de cada uno de los métodos que se invocan acaba de ser explicado.

Paso 4. Abre **AndroidManifest.xml** y agrega los permisos de **RECORD_AUDIO** y **WRITE_EXTERNAL_STORAGE**.

```
AndroidManifest.xml x
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.grabadora">

    <uses-permission android:name="android.permission.RECORD_AUDIO"/>
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>

</manifest>
```

Paso 5. Ejecuta de nuevo la aplicación y verifica el resultado.

Funciona correctamente!!!

Parte II: Grabación de audio utilizando MediaRecorder (II)

La aplicación anterior resulta algo confusa de utilizar. Sería más sencillo si los botones que no pudiéramos utilizar en un determinado momento estuvieran deshabilitados. Para conseguirlo vamos a utilizar el método **Button.setEnabled(boolean)**. Veamos cómo conseguirlo:

Paso 1. Declara las siguientes tres variables al principio de la Actividad:

```
public class MainActivity extends AppCompatActivity {

    private Button bGrabar, bDetener, bReproducir;

    private static final String LOG_TAG = "Grabadora";
    private MediaRecorder mediaRecorder;
    private MediaPlayer mediaPlayer;

}
```

Paso 2. En el método **onCreate()** inicializa estos tres botones. Además comenzaremos deshabilitando dos de los botones que al principio de la aplicación no deben ser pulsados:

```
@Override public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    fichero = this.getExternalFilesDir(null).getAbsolutePath()+"/audio.3gp";

    bGrabar= (Button) findViewById(R.id.bGrabar);
    bDetener= (Button) findViewById(R.id.bDetener);
    bReproducir= (Button) findViewById(R.id.bReproducir);
    bDetener.setEnabled(false);
    bReproducir.setEnabled(false);
}
```

Paso 3. En el método `grabar2()` añade el siguiente código:

```
public void grabar2() {  
    bGrabar.setEnabled(false);  
    bDetener.setEnabled(true);  
    bReproducir.setEnabled(false);  
  
    mediaRecorder = new MediaRecorder();  
    mediaRecorder.setOutputFile(fichero);  
    mediaRecorder.setAudioSource(MediaRecorder.AudioSource.MIC);  
    mediaRecorder.setOutputFormat(MediaRecorder.OutputFormat.THREE_GPP);  
    mediaRecorder.setAudioEncoder(MediaRecorder.AudioEncoder.AMR_NB);  
    try {  
        mediaRecorder.prepare();  
    } catch (IOException e) {  
        Log.e(LOG_TAG, msg: "Fallo en grabación");  
    }  
    mediaRecorder.start();  
}
```

Paso 4. En el método `detenerGrabacion()` añade el siguiente código:

```
public void detenerGrabacion(View view) {  
    bGrabar.setEnabled(true);  
    bDetener.setEnabled(false);  
    bReproducir.setEnabled(true);  
  
    mediaRecorder.stop();  
    mediaRecorder.release();  
}
```

Paso 5. Ejecuta de nuevo la aplicación y verifica el resultado.

