

Actividad | 1 | Instalación de Android Studio y Diseño de aplicación.

Ingeniería del conocimiento.

Ingeniería en Desarrollo de Software.



TUTOR: Eduardo Israel Castillo García.

ALUMNO: Carlos Ariel Nicolini

FECHA: 31/03/2025

Índice

Introducción	3
Descripción	4
Justificación	5
Desarrollo.....	6
• Interfaz.....	6
• Codificación	13
• Prueba de aplicación.....	22
Conclusión.....	24
Referencias.....	25

Introducción

Android Studio es el entorno de desarrollo integrado (IDE) oficial que se usa en el desarrollo de apps para Android. Basado en el potente editor de código y las herramientas para desarrolladores de IntelliJ IDEA, Android Studio ofrece aún más funciones que mejoran tu productividad cuando compilas apps para Android, como las siguientes:

- Un sistema de compilación flexible basado en Gradle.
- Un emulador rápido y cargado de funciones.
- Un entorno unificado donde puedes desarrollar para todos los dispositivos Android.
- Ediciones en vivo para actualizar elementos componibles en emuladores y dispositivos físicos, en tiempo real.
- Integración con GitHub y plantillas de código para ayudarte a compilar funciones de apps comunes y también importar código de muestra.
- Variedad de marcos de trabajo y herramientas de prueba.
- Herramientas de Lint para identificar problemas de rendimiento, usabilidad y compatibilidad de versiones, entre otros.
- Compatibilidad con C++ y NDK.
- Compatibilidad integrada con Google Cloud Plataform, que facilita la integración con Google Cloud Messaging y App Engine.

Descripción

Contextualización:

El buen fin se acerca, por lo que se le solicita a un ingeniero en desarrollo de software el realizar una aplicación móvil en Android. Esta debe permitir manejar una galería de ringtones.

Actividad:

Instalar Android Studio y configurarlo para su correcto uso. Una vez hecho esto, se deberá de crear una aplicación que al descargarse contenga 10 audios en formato .mp3.

Este trabajo me da la oportunidad de seguir utilizando este programa espectacular pero que pide bastantes recursos a nivel máquina y conocimiento. El seguir utilizando Android Studio me da la oportunidad de seguir aprendiendo de su funcionamiento y los espectaculares resultados que da.

Al estar realizando este trabajo, mi hijo de pregunto qué estaba haciendo y le explique, el me dio la idea de hacerlo sobre el juego que jugamos juntos, lo cual me gusto, hice la presentación y las canciones con estricta supervisión de él.

Espero que este trabajo cumpla con las expectativas solicitadas.

Justificación

En esta actividad se nos pide realizar una aplicación en Android Studio para manejar una galería de ringtones, en mi caso realice la temática con los Helldrivers, que es un juego donde la humanidad vive bajo una “democracia dirigida”, por lo cual esta aplicación y la música ahí compartida es la única que Supertierra (así se llama el planeta tierra en esa realidad) que está avalada y es permitida.

Se trabajara como se mencionó anteriormente con la aplicación Android Studio, la cual ya había utilizado en una materia anterior por tal motivo no realice la instalación, pero si configure el emulador con un dispositivo para las pruebas y de la cual se adjunta capturas.

Este programa y este modo de desarrollar es muy impresionante, si es verdad que se necesita mucha practica y tener los conocimientos de los métodos, su estilo y su aparente simpleza hace que uno se divierta trabajando con él y poder realizar las pruebas en el emulador son espectaculares, aunque no voy a mentir, cuando uno lo ejecuta y ve error no se siente nada lindo, pero por suerte tiene una explicación muy detallada de los errores y donde los está cometiendo uno para su remediación.

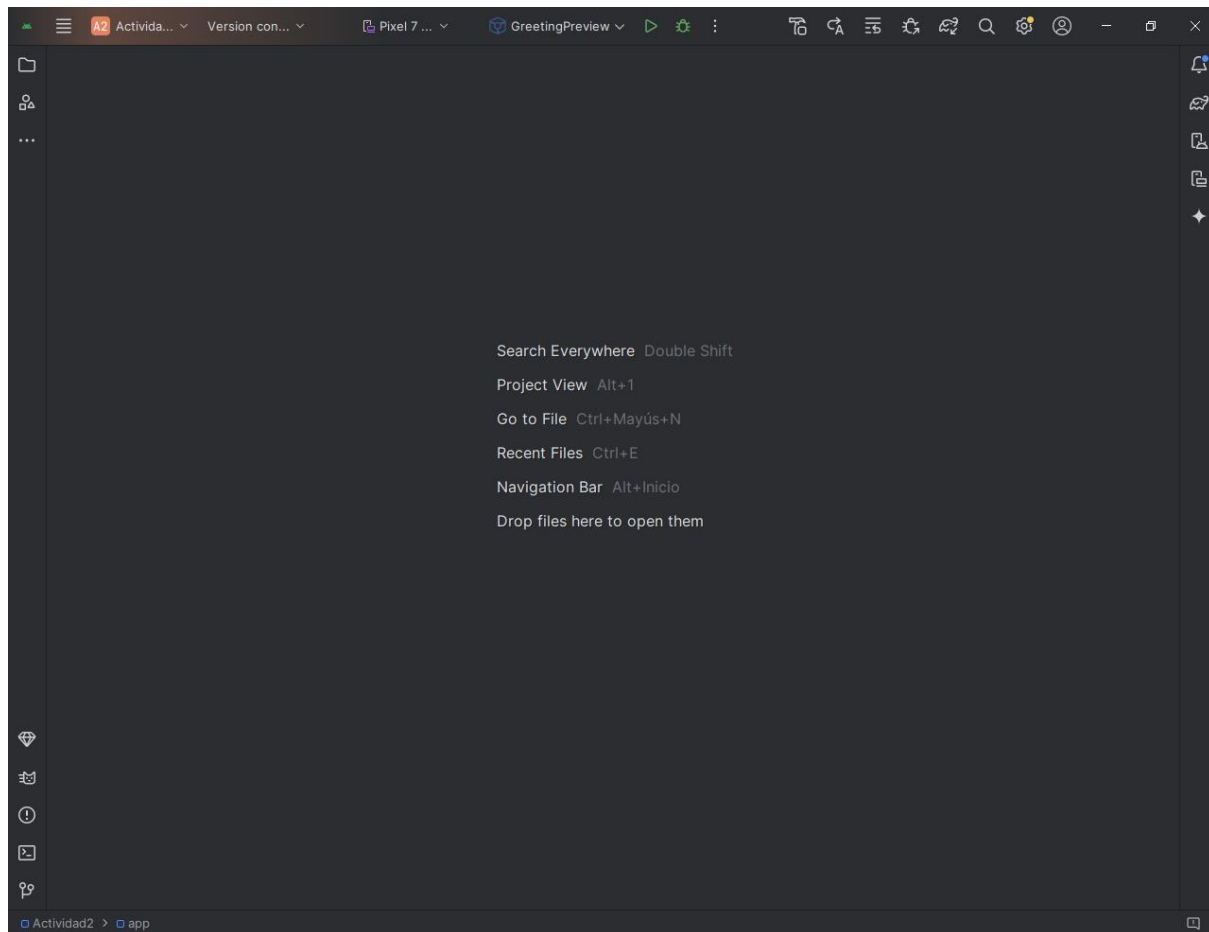
Este trabajo fue subido al siguiente enlace de GitHub

<https://github.com/CarlosNico/Desarrollo-de-Aplicaciones-M-viles-II>

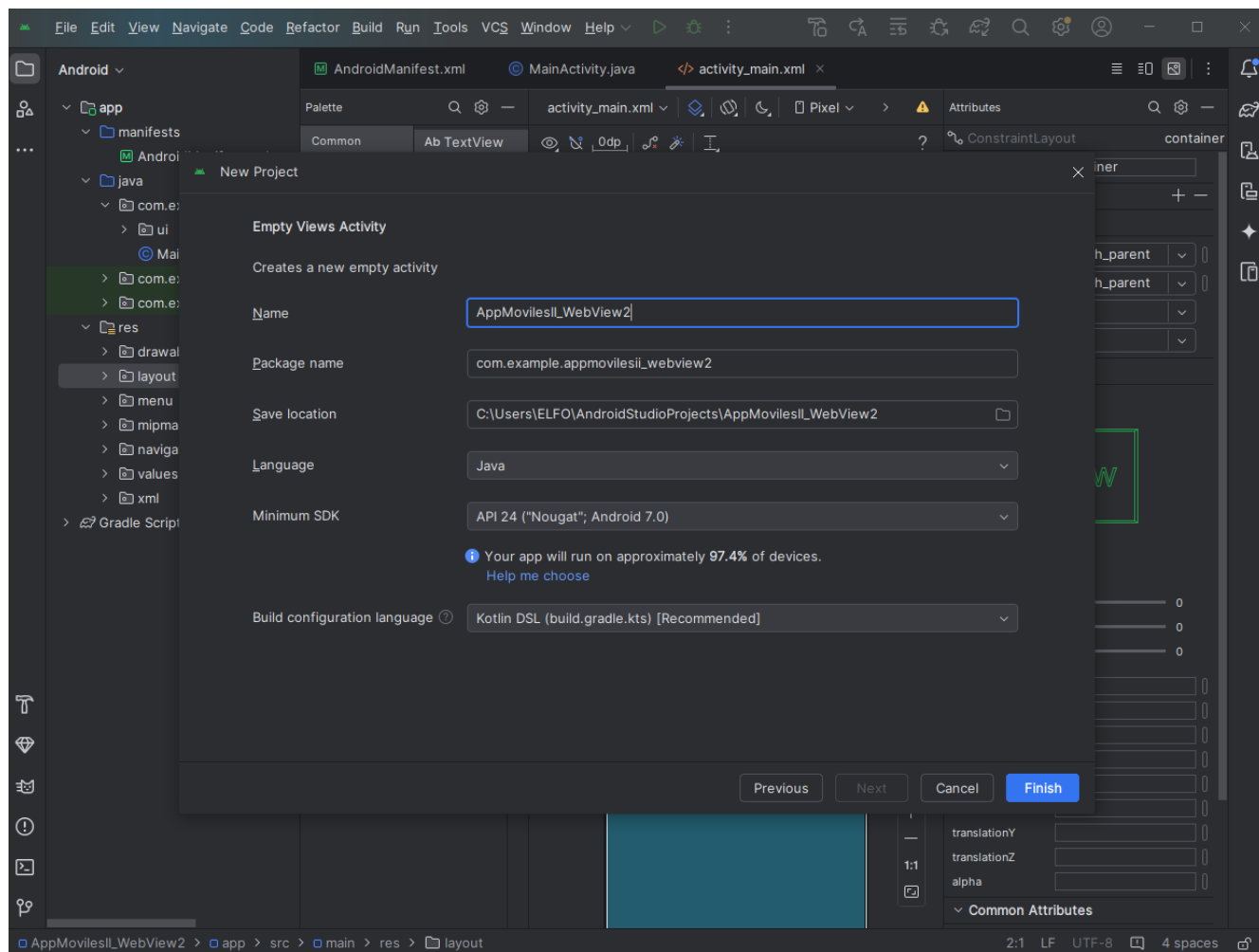
Desarrollo

Interfaz

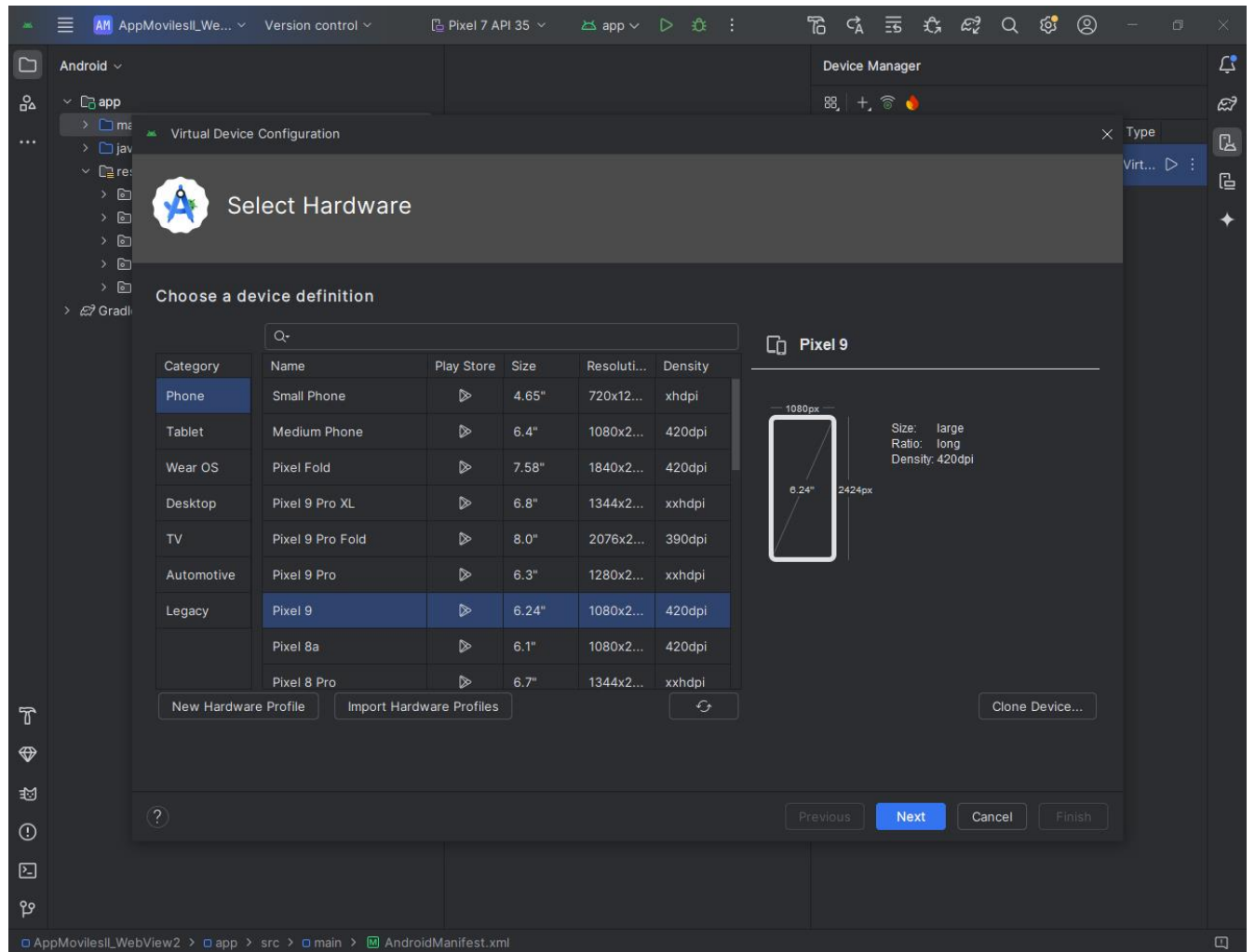
Para esta parte del ejercicio es necesario realizar la instalación del software Android Studio el cual ya teníamos instalado de trabajos anteriores. Se pasa a mostrar imagen del programa instalado.

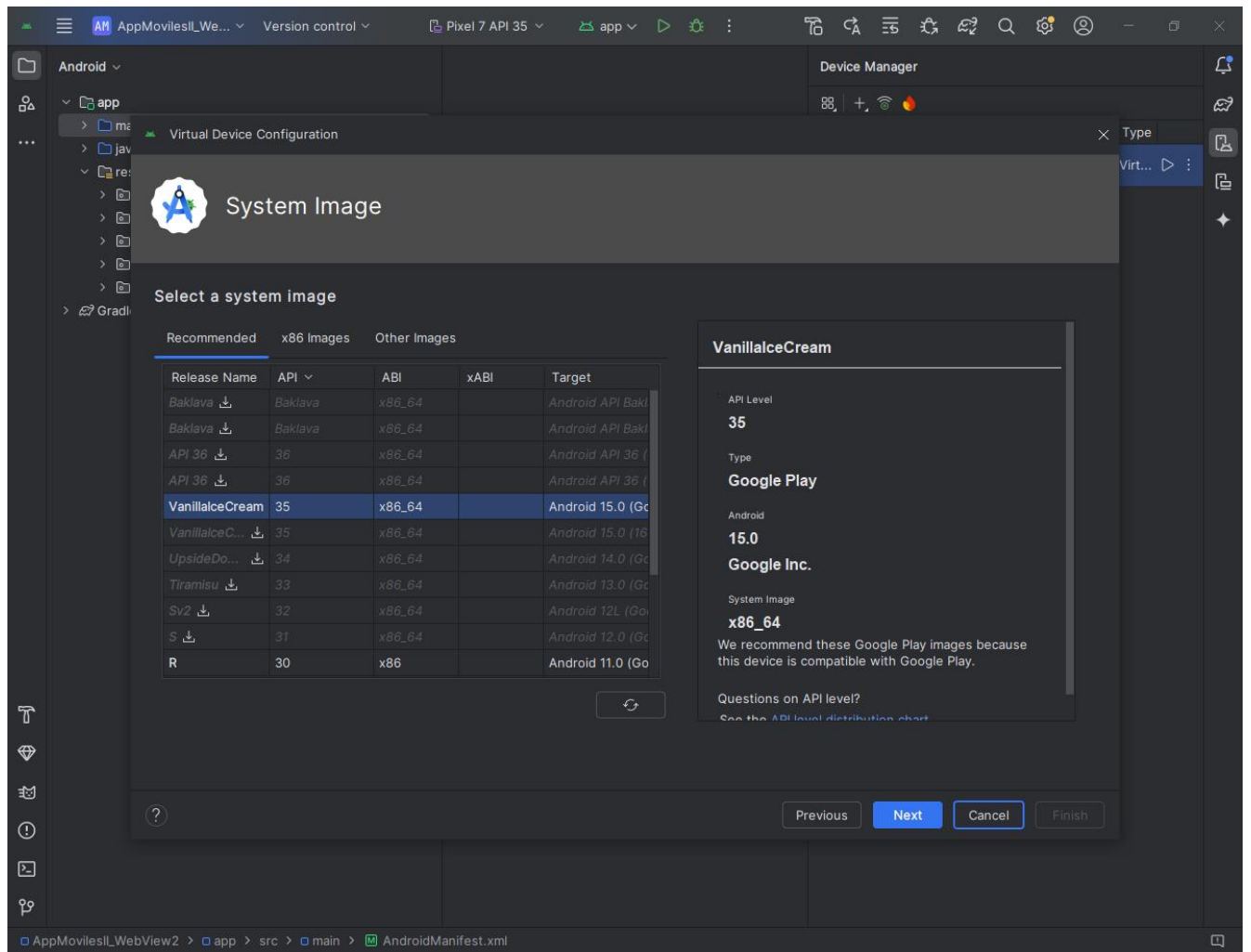


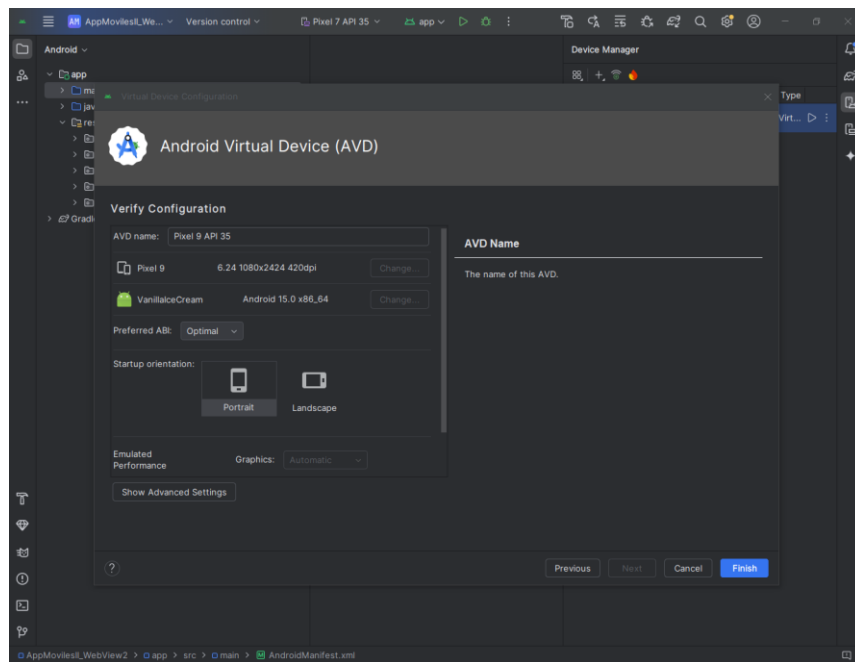
A continuación crearemos un proyecto nuevo, el cual llamaremos AppMovilesII_WebView2, utilizando el lenguaje de Java y las configuraciones como se muestran en la imagen.



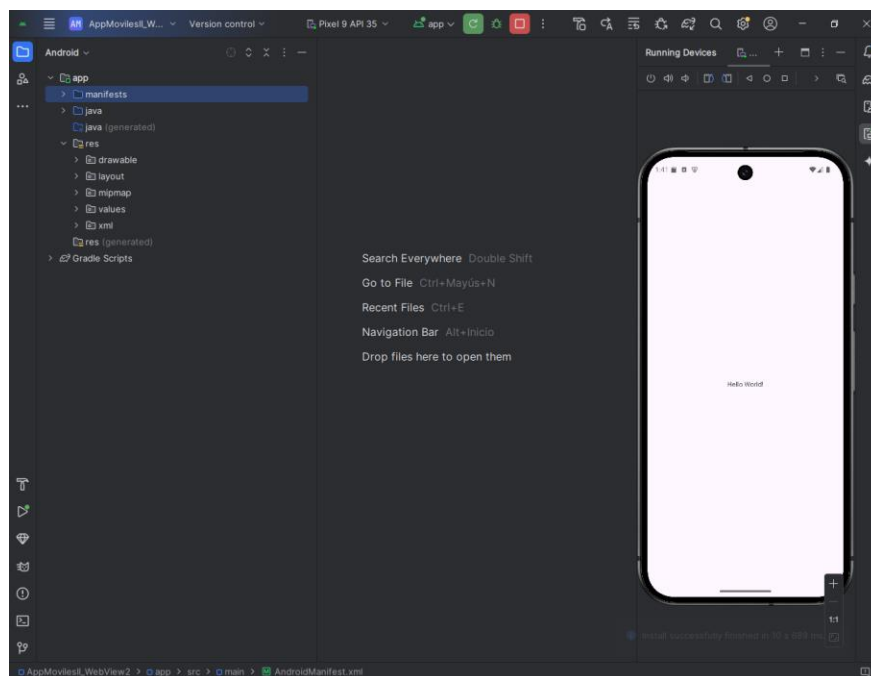
A continuación procedemos a configurar un dispositivo en el emulador.



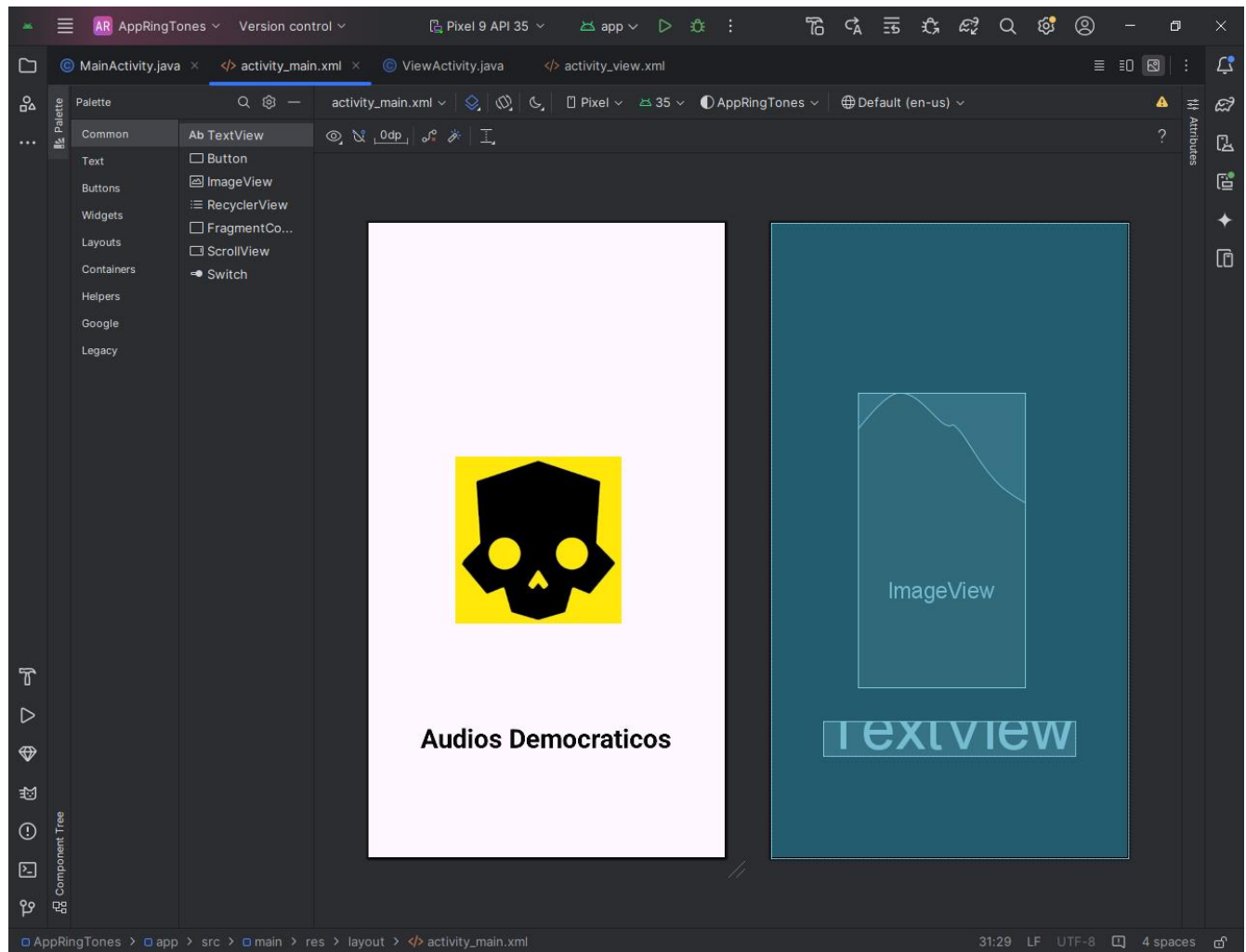




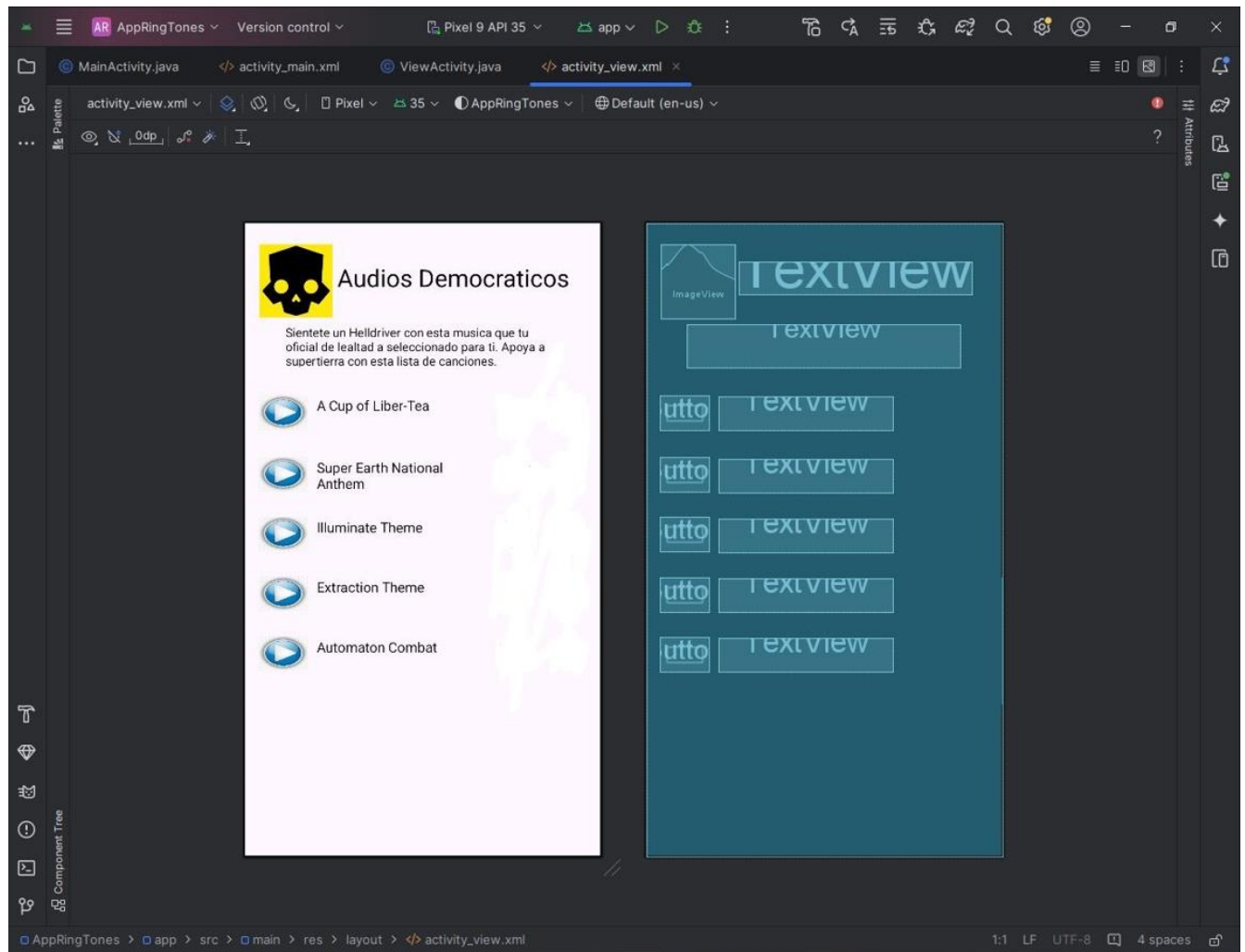
Una vez configurado se testea si es funcional, como se muestra a continuación el dispositivo está funcionando de manera correcta.



A continuación mostramos como quedo realizada nuestra Splash Screen (activity_main.xml), la cual se compone gráficamente de un icono y un texto.



A continuación mostramos como quedo realizada nuestra pantalla de la aplicación (activity_view.xml), la cual se compone gráficamente de un icono (en la parte superior izquierda), un texto a modo de título, un texto largo a modo de introducción y luego los botones con los nombres de las canciones.



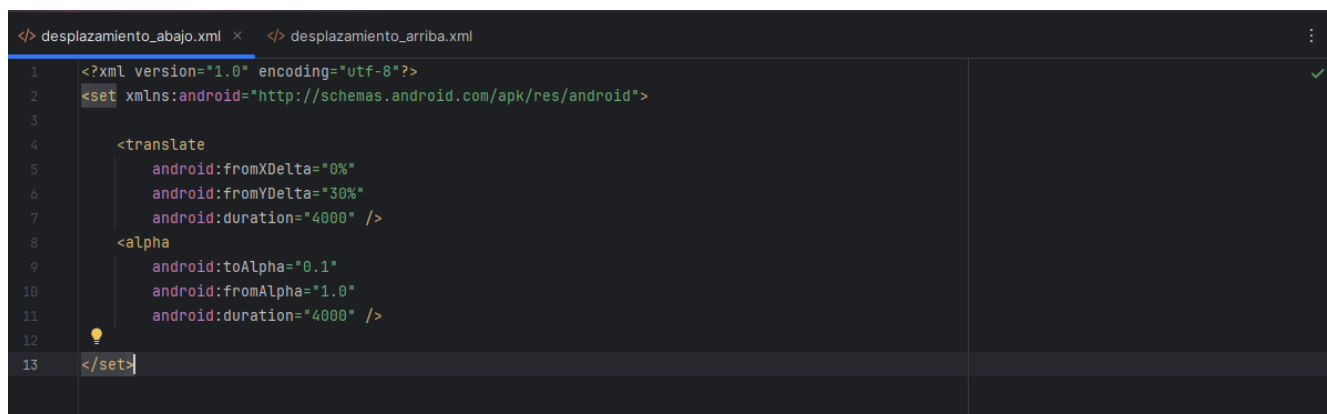
Codificación

A continuación mostraremos las imágenes de los códigos realizados y una breve explicación de que realizan como fue visto en la clase.

Animaciones:

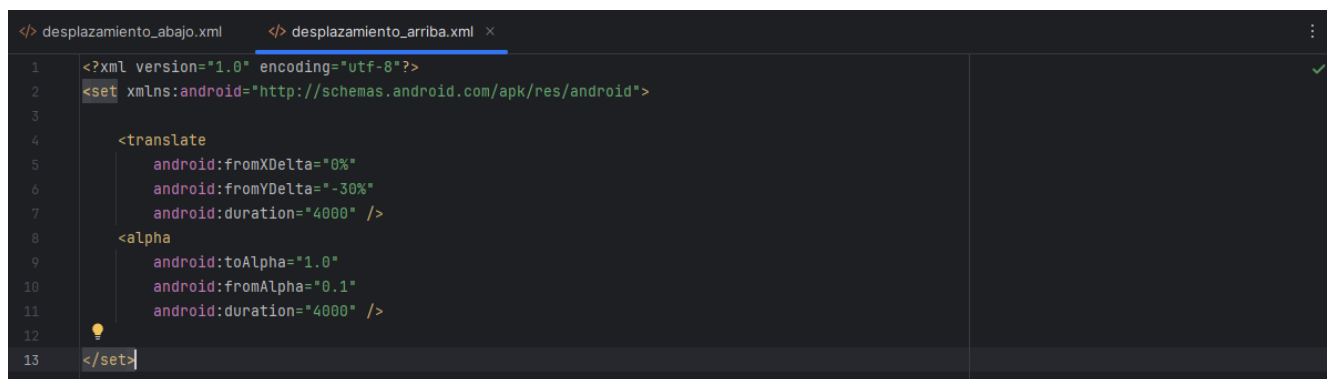
Como se vio en la clase se realizan dos animaciones para el Splash screen, lo cual hace lo siguiente:

- desplazamiento hacia abajo: Aparezca un 30% más debajo de su posición normal, se desplace hacia arriba hasta su lugar original, se desvanezca lentamente mientras sube y todo sucede en 4 segundos.



```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate
        android:fromXDelta="0%"
        android:fromYDelta="30%"
        android:duration="4000" />
    <alpha
        android:toAlpha="0.1"
        android:fromAlpha="1.0"
        android:duration="4000" />
</set>
```

- Desplazamiento hacia arriba : Aparezca desde arriba, bajando a su posición original, se desvanezca hacia adentro, pasando de casi invisible a completamente visible y todo ocurre durante 4 segundos.



```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android">
    <translate
        android:fromXDelta="0%"
        android:fromYDelta="-30%"
        android:duration="4000" />
    <alpha
        android:toAlpha="1.0"
        android:fromAlpha="0.1"
        android:duration="4000" />
</set>
```

En el MainActivity.java configuramos lo siguiente:

- Se oculta la barra de estado para que la app se muestre a pantalla completa.
- Se establece el archivo de diseño XML (activity_main.xml) como la interfaz gráfica.
- Se cargan las animaciones para los dos objetos de animación de Splash Screen (el icono y el texto).
- Encuentra un TextView y un ImageView en el layout por sus IDs y les aplica las animaciones cargadas.
- Después de 6 segundos, lanza el ViewActivity y cierra el Splash Screen.

```

1  package com.example.appringtones;
2
3  import android.content.Intent;
4  import android.os.Bundle;
5  import androidx.appcompat.app.AppCompatActivity;
6  import android.widget.TextView;
7  import android.widget.ImageView;
8  import android.view.WindowManager;
9  import android.view.animation.Animation;
10 import android.view.animation.AnimationUtils;
11 import android.os.Handler;
12
13
14 public class MainActivity extends AppCompatActivity {
15
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN, WindowManager.LayoutParams.FLAG_FULLSCREEN);
20         setContentView(R.layout.activity_main);
21
22         //Agregar animaciones: de arriba a abajo
23         Animation animacion1 = AnimationUtils.loadAnimation(context: this, R.anim.desplazamiento_arriba);
24         Animation animacion2 = AnimationUtils.loadAnimation(context: this, R.anim.desplazamiento_arriba);
25
26         //Agregar animaciones: de abajo a arriba
27         //Animation animacion1 = animationUtils.loadAnimation(this,R.anim.desplazamiento_abajo);
28         //Animation animacion2 = animationUtils.loadAnimation(this,R.anim.desplazamiento_abajo);
29
30         TextView audiotextview = findViewById(R.id.audiotextview);
31         ImageView logoimageview = findViewById(R.id.logoimageview);
32
33         audiotextview.setAnimation(animacion1);
34         logoimageview.setAnimation(animacion2);
35
36         new Handler().postDelayed(new Runnable() {
37             @Override
38             public void run() {
39                 Intent intent = new Intent(packageContext: MainActivity.this, ViewActivity.class);
40                 startActivity(intent);
41                 finish();
42             }
43         }, delayMillis: 6000);
44     }
45 }
46

```

Activity_main.xml

En el configuramos para que realice lo siguiente:

- Crea una imagen con su ID, le definimos su tamaño.
- La imagen esta ubicada en la carpeta mipmap como fuente
- Esta dentro de un ConstraintLayout y se posiciona centrada dentro del padre.
- Crea un texto con un ID, con un texto que dice “Audios Democraticos”, le definimos el color de texto, tamaño y estilo.
- Se posiciona debajo de la imagen y está centrado horizontalmente y anclado al borde inferior.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:id="@+id/main"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".MainActivity">
9
10     <ImageView
11         android:id="@+id/logoimageview"
12         android:layout_width="192dp"
13         android:layout_height="339dp"
14         android:layout_marginStart="100dp"
15         android:layout_marginTop="80dp"
16         android:layout_marginEnd="100dp"
17         android:layout_marginBottom="80dp"
18         app:layout_constraintBottom_toBottomOf="parent"
19         app:layout_constraintStart_toEndOf="parent"
20         app:layout_constraintStart_toStartOf="parent"
21         app:layout_constraintTop_toTopOf="parent"
22         app:srcCompat="@mipmap/logoh" />
23

```

```

23
24     <TextView
25         android:id="@+id/audiotextview"
26         android:layout_width="wrap_content"
27         android:layout_height="wrap_content"
28         android:layout_marginTop="26dp"
29         android:layout_marginBottom="106dp"
30         android:fontFamily="sans-serif"
31         android:text="Audios Democraticos"
32         android:textColor="#090909"
33         android:textSize="30sp"
34         android:textStyle="bold"
35         app:layout_constraintBottom_toBottomOf="parent"
36         app:layout_constraintEnd_toEndOf="parent"
37         app:layout_constraintHorizontal_bias="0.496"
38         app:layout_constraintStart_toStartOf="parent"
39         app:layout_constraintTop_toBottomOf="@+id/logoimageview"
40         tools:ignore="MissingConstraints" />
41 </androidx.constraintlayout.widget.ConstraintLayout>

```

Activity_view.xml

En el configuramos para que realice lo siguiente:

- Se muestra una imagen del tamaño definido, en el borde izquierdo y superior.
- Se muestra un texto a modo de título alineado hacia la derecha en la parte superior.
- Se muestra un texto descriptivo largo como introducción que está centrado y un poco más debajo de los elementos anteriores.
- Se agrega un botón con su ID, que tiene un icono (@drawable/play). Este mismo con diferente posición se utiliza en total 5 veces para los 5 botones.
- Se muestra un Textview que muestra el nombre de la canción, está centrado y alineado con el eje vertical. Este mismo con diferente posición se utiliza en total 5 veces para los 5 nombres de canciones que se piden.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:id="@+id/main"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".ViewActivity">
9
10     <ImageView
11         android:id="@+id/imageView"
12         android:layout_width="85dp"
13         android:layout_height="85dp"
14         app:layout_constraintBottom_toBottomOf="parent"
15         app:layout_constraintEnd_toEndOf="parent"
16         app:layout_constraintHorizontal_bias="0.051"
17         app:layout_constraintStart_toStartOf="parent"
18         app:layout_constraintTop_toTopOf="parent"
19         app:layout_constraintVertical_bias="0.037"
20         app:srcCompat="@mipmap/logo" />
21
22     <TextView
23         android:id="@+id/textView"
24         android:layout_width="wrap_content"
25         android:layout_height="wrap_content"
26         android:text="Audios Democraticos"
27         android:textColor="#090909"
28         android:textSize="28sp"
29         app:layout_constraintBottom_toBottomOf="parent"
30         app:layout_constraintEnd_toEndOf="parent"
31         app:layout_constraintHorizontal_bias="0.750"
32         app:layout_constraintStart_toStartOf="parent"
33         app:layout_constraintTop_toTopOf="parent"
34         app:layout_constraintVertical_bias="0.063" />

```



```

35
36     <TextView
37         android:id="@+id/textView2"
38         android:layout_width="316dp"
39         android:layout_height="50dp"
40         android:text="Sientete un Helldriver con esta musica que tu oficial de lealtad a seleccionado para ti... Apoya a supertierra con esta"
41         android:textColor="#090909"
42         app:layout_constraintBottom_toBottomOf="parent"
43         app:layout_constraintEnd_toEndOf="parent"
44         app:layout_constraintHorizontal_bias="0.493"
45         app:layout_constraintStart_toStartOf="parent"
46         app:layout_constraintTop_toTopOf="parent"
47         app:layout_constraintVertical_bias="0.171" />
48

```

Canción 1

```

<Button
    android:id="@+id/play1"
    android:layout_width="57dp"
    android:layout_height="40dp"
    android:foreground="@drawable/play"
    android:text="Button"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.045"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.288" />

<TextView
    android:id="@+id/cancion1"
    android:layout_width="201dp"
    android:layout_height="39dp"
    android:text="A Cup of Liber-Tea"
    android:textColor="#090909"
    android:textSize="16sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.395"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.288" />

```

Canción 2

```

<Button
    android:id="@+id/play2"
    android:layout_width="57dp"
    android:layout_height="40dp"
    android:foreground="@drawable/play"
    android:text="Button"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.045"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.391" />

<TextView
    android:id="@+id/cancion2"
    android:layout_width="201dp"
    android:layout_height="39dp"
    android:text="Super Earth National Anthem"
    android:textColor="#090909"
    android:textSize="16sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.395"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.392" />

```

Canción 3

```

<Button
    android:id="@+id/play3"
    android:layout_width="57dp"
    android:layout_height="40dp"
    android:foreground="@drawable/play"
    android:text="Button"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.045"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.491" />

<TextView
    android:id="@+id/cancion3"
    android:layout_width="201dp"
    android:layout_height="39dp"
    android:text="Illuminate Theme"
    android:textColor="#090909"
    android:textSize="16sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.395"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.492" />

```

Canción 4

```

<Button
    android:id="@+id/play4"
    android:layout_width="57dp"
    android:layout_height="40dp"
    android:foreground="@drawable/play"
    android:text="Button"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.045"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.591" />

<TextView
    android:id="@+id/cancion4"
    android:layout_width="201dp"
    android:layout_height="39dp"
    android:text="Extraction Theme"
    android:textColor="#090909"
    android:textSize="16sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.395"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.592" />

```

Canción 5

```
<Button
    android:id="@+id/play5"
    android:layout_width="57dp"
    android:layout_height="40dp"
    android:foreground="@drawable/play"
    android:text="Button"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.045"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.691" />

<TextView
    android:id="@+id/cancion5"
    android:layout_width="201dp"
    android:layout_height="39dp"
    android:text="Automaton Combat"
    android:textColor="#090909"
    android:textSize="16sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.395"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="0.692" />
```

ViewActivity.java

En el configuramos para que realice lo siguiente:

- Define que es una actividad que se extiende a una pantalla dentro de la app
- Se declaran los 5 botones para la reproducción de canciones.
- Se declaran 5 objetos mediaplayer, que cada uno cargara un audio diferente.
- Establece el diseño de la interfaz activity_view.xml.
- Conecta cada botón de la interfaz con su respectivo ID definido ya en el archivo XML.
- Carga los archivos de música desde la carpeta raw.
- Le da su funcionalidad a cada botón, si el audio sonando, lo pausa y muestra un Toast con el mensaje Pausa, si esta pausado o detenido lo reproduce y muestra un mensaje de Play.

```
package com.example.appringtones;

import android.media.MediaPlayer;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class ViewActivity extends AppCompatActivity {

    2 usages
    Button play1, play2, play3, play4, play5;

    4 usages
    MediaPlayer mp1, mp2, mp3, mp4, mp5;

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

        play1 = (Button)findViewById(R.id.play1);
        play2 = (Button)findViewById(R.id.play2);
        play3 = (Button)findViewById(R.id.play3);
        play4 = (Button)findViewById(R.id.play4);
        play5 = (Button)findViewById(R.id.play5);

        mp1 = MediaPlayer.create(context: this, R.raw.libertea);
        mp2 = MediaPlayer.create(context: this, R.raw.anthem);
        mp3 = MediaPlayer.create(context: this, R.raw.illuminate);
        mp4 = MediaPlayer.create(context: this, R.raw.extraction);
        mp5 = MediaPlayer.create(context: this, R.raw.automaton);
    }
}
```

```

play1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(mp1.isPlaying()) {
            mp1.pause();
            Toast.makeText(context: ViewActivity.this, text: "Pausa", Toast.LENGTH_SHORT).show();
        }else{
            mp1.start();
            Toast.makeText(context: ViewActivity.this, text: "Play", Toast.LENGTH_SHORT).show();
        }
    }
});
play2.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(mp2.isPlaying()) {
            mp2.pause();
            Toast.makeText(context: ViewActivity.this, text: "Pausa", Toast.LENGTH_SHORT).show();
        }else{
            mp2.start();
            Toast.makeText(context: ViewActivity.this, text: "Play", Toast.LENGTH_SHORT).show();
        }
    }
});

```

```

play3.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(mp3.isPlaying()) {
            mp3.pause();
            Toast.makeText(context: ViewActivity.this, text: "Pausa", Toast.LENGTH_SHORT).show();
        }else{
            mp3.start();
            Toast.makeText(context: ViewActivity.this, text: "Play", Toast.LENGTH_SHORT).show();
        }
    }
});
play4.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(mp4.isPlaying()) {
            mp4.pause();
            Toast.makeText(context: ViewActivity.this, text: "Pausa", Toast.LENGTH_SHORT).show();
        }else{
            mp4.start();
            Toast.makeText(context: ViewActivity.this, text: "Play", Toast.LENGTH_SHORT).show();
        }
    }
});

```

```

play5.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        if(mp5.isPlaying()) {
            mp5.pause();
            Toast.makeText(context: ViewActivity.this, text: "Pausa", Toast.LENGTH_SHORT).show();
        }else{
            mp5.start();
            Toast.makeText(context: ViewActivity.this, text: "Play", Toast.LENGTH_SHORT).show();
        }
    }
});
}
}

```

Prueba de aplicación

En esta parte probamos la aplicación realizada y adjuntaremos imágenes de su comportamiento. Al iniciar la aplicación aparecerá un Splash Screen con una animación (imagen 1), tras lo cual nos llevara a la página de la aplicación (imagen 2). En ella tenemos 5 canciones con sus botones de play y el título de cada una de ellas. Al presionar el botón play se reproducirá la canción y nos mostrara un mensaje de texto que indicara que está reproduciéndose (imagen 3), al presionar nuevamente en el botón de play la canción se detendrá y nos mostrara un mensaje de Pause (imagen 4).



Imagen 1

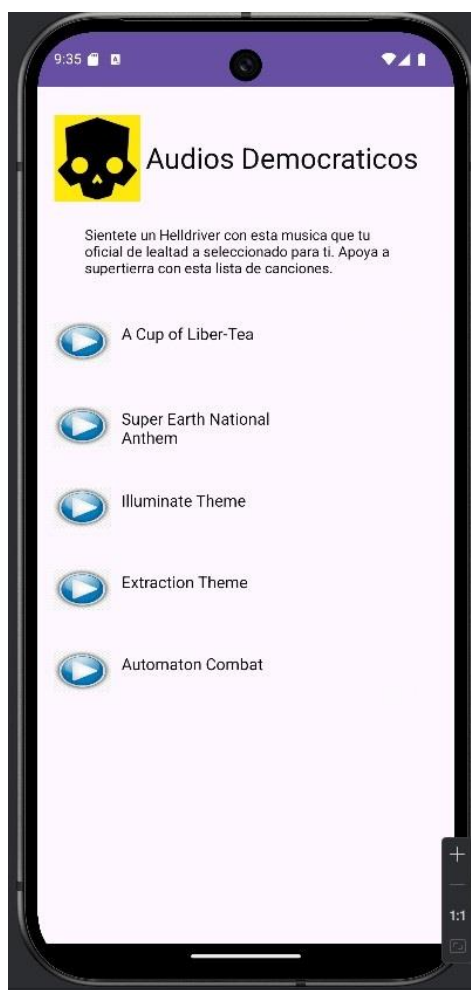


Imagen 2

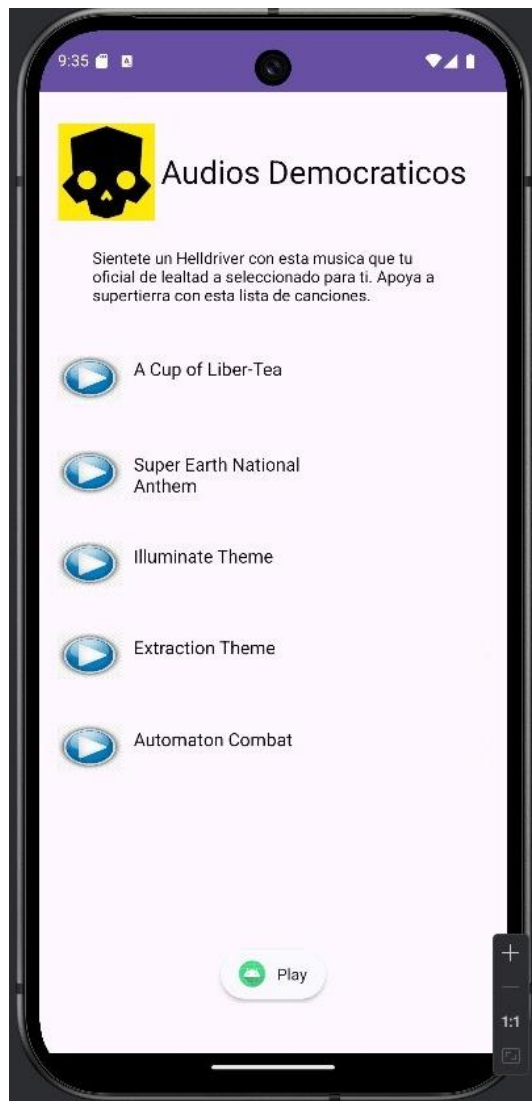


Imagen 3

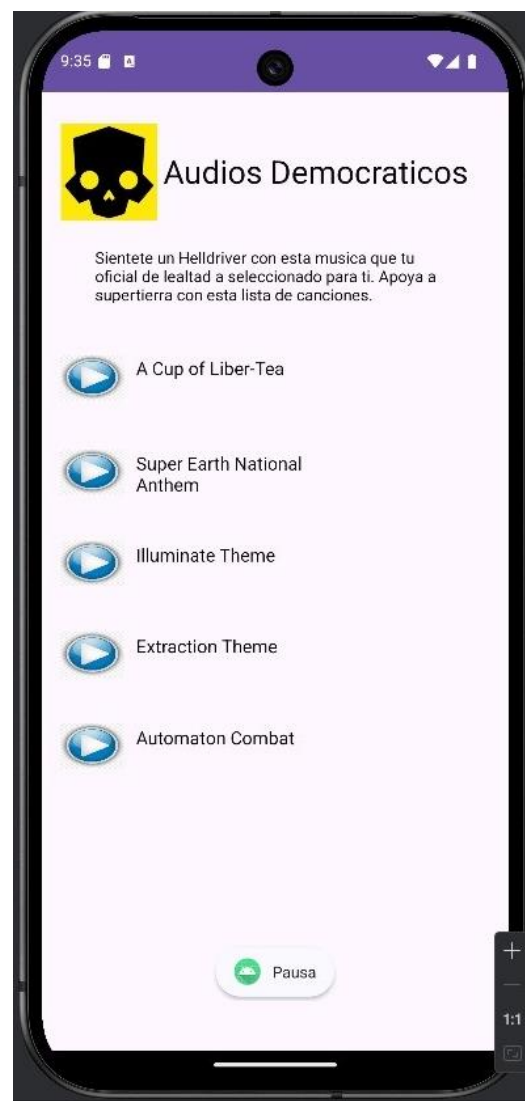


Imagen 4

Conclusión

El conocimiento y la utilización de herramientas como Android Studio son muy necesarias en el ámbito laboral hoy en día, pero este tipo de conocimiento que estamos aprendiendo clase con clase no también se pueden utilizar de muchas maneras. El crear esta aplicación con un Splash screen y poder realizar que al presionar un botón se reproduzcan canciones me da un conocimiento que no tenía y que tiene muchísimas utilidades en mi día a día, puedo hacer una aplicación para mostrar documentación del departamento, reproducir audios con explicaciones de procesos, etc. La utilización del Android studio, dominarlo y programar en él es una herramienta muy útil de tener en el currículo a la hora de poder desarrollar herramientas innovadoras de muy fácil funcionamiento pero de alcances inimaginables.

La clase y la explicación del profesor como siempre han sido impresionantes y de muchísimo provecho para mi persona. La pase muy bien desarrollando esta actividad y me divertí muchísimo porque aunque estaba haciendo las tareas también pude tener un tiempo con mi hijo en desarrollarla y que él me ayude a diseñar ciertos aspectos de ella. Quien sabe, puede que aunque esta chico ahora esto sea algo que pueda el también utilizar.

Este trabajo fue subido al siguiente enlace de GitHub

<https://github.com/CarlosNico/Desarrollo-de-Aplicaciones-M-viles-II>

Referencias

Introducción a Android Studio. (n.d.). Android Developers. Retrieved April 20, 2025, from

<https://developer.android.com/studio/intro?hl=es-419>

Helldivers 2. (n.d.). Logopedia; Fandom, Inc. Retrieved April 20, 2025, from

https://logos.fandom.com/wiki/Helldivers_2