Aplicaciones Multimedia

## Desarrollo de Aplicaciones Multiplataforma

Programación multimedia y dispositivos móviles



## **Actividad**

Aplicaciones Multimedia.

## Objetivos

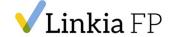
- Desarrollar Aplicaciones para teléfono móvil.
- Aplicar elementos multimedia.
- Reproducir audio y video.



## ¿Cómo lo hago?

- 1. Rellena los datos que se piden en la tabla "Antes de empezar".
- 2. Haz uso de fuentes comunes como Arial, Calibri, Times New Roman etc.
- 3. Utiliza el color negro para desarrollar tus respuestas y usa otros colores para destacar contenidos o palabras que creas necesario resaltar.
- 4. Recuerda entregar la actividad en formato PDF a no ser que el profesor o profesora indique lo contrario.
- 5. Recuerda nombrar el archivo siguiendo estas indicaciones:
  - Ciclo\_Módulo o crédito\_Tema\_ACT\_número actividad\_Nombre y apellido
    - Ejemplo: AF\_M01\_T01\_ACT\_01\_Maria Garcia

Antes de empezar		
Nombre	Jacobo	
Apellidos	Montero Gil	
Módulo/Crédito	M05	
UF (solo ciclos LOE)		
Título de la actividad	MediaPlayer	





#### Actividad 3

### MediaPlayer

Desarrollar una aplicación Android para reproducir archivos de vídeo almacenados en la memoria del teléfono o dispositivo Android:

- Dependiendo del tipo de dispositivo, utilizar la memoria interna o la memoria externa (tarjeta SD card).
- Crear una carpeta llamada *videos* y almacenar en ella varios archivos de vídeo
- Presentar por pantalla una lista de los archivos de vídeo disponibles en dicha carpeta y permitir al usuario elegir uno de ellos y reproducirlo.
- Agregar los clásicos controles de Play, Pause y Stop.

La App debe intuitiva y fácil de usar y debe evitar cualquier comportamiento inestable por información errónea o no proporcionada.





olucio	Ŕη	6
1.	Funcionalidades de la aplicación	6
2.	El código	7
•	Android Manifest.xml	7
•	MainActivy.java	8
•	Pantalla Reproductor. java	. 10
•	activity_main.xml	. 13
•	activity_pantalla_reproductor.xml	. 14
•	activity_pantalla_reproductor.xml(land)	. 16
3.	Fiemplos de funcionamiento	. 18



# Solución

## 1. Funcionalidades de la aplicación.

Para la realización de la práctica he utilizado el framework de acceso a almacenamiento (SAF), implementándolo en el método : private void showFileChooser(). Esta opción me permite acceder a todo el almacenamiento filtrando por tipos de archivos(video/\*), evitando además que se intente abrir un tipo de archivo no soportado.

Una vez seleccionado el archivo se inicia una nueva activity y se comienza la reproducción de forma automática, al realizar una pulsación sobre la pantalla se muestran los botones de reproducción y si pulsamos nuevamente se ocultan.

En los botones se ha implementado la funcionalidad correspondiente a cada uno: start, pause y stop.



## 2. El código

### AndroidManifest.xml

\*Implemento la barra de navegación para retroceder desde la pantalla de reproducción a la pantalla de selección.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"</pre>
  package="com.example.mediaplayer">
  <application
    android:allowBackup="true"
   android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundlcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/Theme.AppCompat.DayNight"
 <activity android:name=".MainActivity">
      <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
      </intent-filter>
    </activity>
    <activity android:name=".PantallaReproductor">
      android:parentActivityName="com.example.mediaplayer.MainActivity
      <meta-data
        android:name="android.support.PARENT_ACTIVITY"
        android:value="com.example.mediaplayer.MainActivity"/>
    </activity>
  </application>
</manifest>
```



## MainActivy.java

```
import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.net.Uri;
import android.os.Bundle;
import android.os.FileUtils;
import android.provider.DocumentsContract;
import android.util.Log;
import android.widget.ListView;
import android.widget.Toast;
import java.lang.reflect.Field;
public class MainActivity extends AppCompatActivity {
 //Iniciamos la variables necesarias
  public static final String EXTRA_MESSAGE = "com.example.mediaplayer.MESSAGE";
  private static final int FILE_SELECT_CODE = 0;
 //Sobrecargamos el metodo y llamos a showFileChooser(); para que inicie el selector de
archivos del
 //frameworf SAF
  @Override
  protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    showFileChooser();
```



```
//Se implementa el acceso a almacenamiento
 private void showFileChooser() {
   //Se implementa desde un Intent el acceso al almacenamiento
   Intent intent = new Intent(Intent.ACTION_GET_CONTENT);
   intent.setType("*/*");
    //Filtro los contenidos que por tipo video
   String[] mimetypes = { "video/*"};
   intent.putExtra(Intent.EXTRA_MIME_TYPES, mimetypes);
   intent. add Category (Intent. \textit{CATEGORY\_OPENABLE});
   //Se realiza una captura en caso de no exista un Gestor de archivos
   try {
     startActivityForResult(
          Intent.createChooser(intent, "Select a File to Upload"),
          FILE_SELECT_CODE);
   } catch (android.content.ActivityNotFoundException ex) {
     // Potentially direct the user to the Market with a Dialog
     Toast.makeText(this, "Please install a File Manager.",
          Toast.LENGTH_SHORT).show();
 //Se sobrecarga el método para recibir el uri del archivo seleccionado
 @Override
 protected void onActivityResult(int requestCode, int resultCode, Intent data) {
   switch (requestCode) {
     case FILE SELECT CODE:
        if (resultCode == RESULT_OK) {
          //Se obtiene el uri del archivo
          Uri uri = data.getData();
          //Se envía el uri a la pantalla de reproduccion con un intent
         Intent intent = new Intent(this,PantallaReproductor.class);
          intent.putExtra(EXTRA_MESSAGE,uri.toString());
          //Se inicia la pantalla de reproduccion
          startActivity(intent);
        break;
   super.onActivityResult(requestCode, resultCode, data);
```



## PantallaReproductor.java

```
package com.example.mediaplayer;
import androidx.appcompat.app.AppCompatActivity;
import androidx.constraintlayout.widget.ConstraintLayout;
import android.app.ProgressDialog;
import android.content.Intent;
import android.content.pm.ActivityInfo;
import android.content.res.Resources;
import android.graphics.PixelFormat;
import android.media.MediaPlayer;
import android.net.Uri;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.MotionEvent;
import android.view.SurfaceHolder;
import android.view.SurfaceView;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.MediaController;
import android.widget.ProgressBar;
import android.widget.RelativeLayout;
import android.widget.Toast;
import android.widget.VideoView;
public class PantallaReproductor extends AppCompatActivity {
  //Inicio las variables
  private VideoView mVideoView;
  String modo="Start";
  int sel=0;
    super.onCreate(savedInstanceState);
    //obtengo la vista
    mVideoView = (VideoView) findViewById(R.id.surface_view);
    setContentView(R.layout.activity_pantalla_reproductor);
    //Establezco el area de reproduccion
    MediaController mediaController= new MediaController(this);
    mediaController.setAnchorView(mVideoView);
     //Obtengo el uri a reproducir
    Intent intent = getIntent();
    String message = intent.getStringExtra(MainActivity.EXTRA_MESSAGE);
    //Se lo envío al area de reproduccion
    mVideoView.setVideoPath(String.valueOf(Uri.parse(message)));
    //Inicio la reproduccion
    mVideoView.start();
    //Envío un mensaje para que el usuario perciba el inicio de la reproducción
    Toast toast1 =
         Toast.makeText(getApplicationContext(),
             "Reproduccion iniciada", Toast.LENGTH_SHORT);
    toast1.show();
```



```
//Inicio los botones y los listeners de los mismos
Button buttonStart = findViewById(R.id.button4);
buttonStart.setVisibility(View.INVISIBLE);
buttonStart.setOnClickListener(new View.OnClickListener() {
  public void onClick(View v) {
    mVideoView.start();
    Toast toast1 =
         Toast.makeText(getApplicationContext(),
             "Reproduccion iniciada", Toast.LENGTH_SHORT);
    toast1.show();
});
  Botón stop
Button buttonStop = findViewById(R.id.button5);
buttonStop.setVisibility(View.INVISIBLE);
buttonStop.setOnClickListener(new View.OnClickListener() {
  public void onClick(View v) {
    //Al tocar stop paro la reproduccion y la pongo a 0
    mVideoView.stopPlayback();
    mVideoView.resume();
    Toast toast1 =
         Toast.makeText(getApplicationContext(),
              "Reproduccion parada", Toast.LENGTH_SHORT);
    toast1.show();
//Botón pause
Button buttonPause = findViewByld(R.id.button6);
buttonPause.setVisibility(View.INVISIBLE);
buttonPause.setOnClickListener(new View.OnClickListener() {
  public void onClick(View v) {
    //Al tocar pause si está en Start paro la reproduccion
    if(modo.equals("Start")){
      mVideoView.pause();
      Toast toast1 =
           Toast.makeText(getApplicationContext(),
               "Reproduccion en pause", Toast.LENGTH_SHORT);
      toast1.show();
      modo="pause";
       //Si está en pause la inicio
    }else if(modo.equals("pause")) {
      mVideoView.start();
      Toast toast1 =
           Toast.makeText(getApplicationContext(),
               "Reproduccion iniciada", Toast.LENGTH_SHORT);
      toast1.show();
      modo="Star";
```

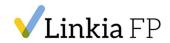


```
// Al tocar la pantalla muestro u oculto los botones
  mVideoView.setOnTouchListener(new View.OnTouchListener()
    @Override
    public boolean onTouch(View v, MotionEvent event) {
      if(sel==0){
         buttonStart.setVisibility(View.VISIBLE);
         buttonStop.setVisibility(View.VISIBLE);
         buttonPause.setVisibility(View.VISIBLE);
         sel=1;
      }else if(sel==1){
         buttonStart.setVisibility(View.INVISIBLE);
         buttonStop.setVisibility(View.INVISIBLE);
         buttonPause.setVisibility(View.INVISIBLE);
         sel=0;
      return false;
  });
```



#### activity\_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
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:backgroundTint="#655757"
    tools:context=".MainActivity">
</androidx.constraintlayout.widget.ConstraintLayout>
```



#### activity pantalla reproductor.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout 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" xmlns:tools=
 android:layout_width="match_parent"
 android:layout_height="match_parent"
      android:orientation="vertical"
 tools:context=".MainActivity">
       <\! and roidx. constraint layout. widget. Constraint Layout
              android:id="@+id/constraintLayout2" android:layout_width="match_parent"
               android:layout_height="match_parent"
               android:layout_marginStart="4dp"
             android:layout_marginstart= 4up
android:layout_marginEnd="4dp"
android:layout_marginEnd="4dp"
android:layout_marginBottom="4dp"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
               app:layout_constraintHorizontal_bias="0.166"
               app:layout_constraintStart_toStartOf="parent"
               app: layout\_constraintTop\_toTopOf = "parent"
               app:layout_constraintVertical_bias="0.229">
                      android:id="@+id/surface_view"
                      android:layout_width="fill_parent"
android:layout_height="fill_parent"
                      android:gravity="center_horizontal" android:orientation="vertical"
                      app: layout\_constraintBottom\_toBottomOf = "parent"
                     app:layout_constraintEnd_toEndOf="parent" app:layout_constraintHorizontal_bias="0.0" app:layout_constraintStart_toStartOf="parent"
                      app:layout_constraintTop_toTopOf="parent" app:layout_constraintVertical_bias="0.0" />
       </androidx.constraintlayout.widget.ConstraintLayout>
```





```
<androidx.constraintlayout.widget.ConstraintLayout
android:id="@+id/constraintLayout"</pre>
  android:layout_width="382dp'
  android:layout_height="149dp"
  android:layout_marginEnd="16dp"
  app:layout_constraintBottom_toBottomOf="parent" app:layout_constraintEnd_toEndOf="parent"
  app:layout_constraintHorizontal_bias=
  app:layout_constraintStart_toStartOf="parent"
  app:layout_constraintTop_toTopOf="parent"
  app:layout_constraintVertical_bias="0.975">
  <Button
     android:id="@+id/button4"
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
     android:layout_marginStart="16dp"
     android:layout_marginTop="127dp"
     android:layout_marginEnd="273dp"
     android:backgroundTint="#474242"
     android:text="@string/Start"
    app:layout_constraintEnd_toEndOf="parent" app:layout_constraintEnd_toEndOf="parent" app:layout_constraintHorizontal_bias="0.0"
     app:layout_constraintStart_toStartOf="parent"
     app:layout_constraintTop_toTopOf="parent"
     app:layout_constraintVertical_bias="1.0" />
     android:id="@+id/button5"
     android:layout_width="wrap_content"
android:layout_height="wrap_content"
    android:layout_marginTop="127dp"
     android:layout_marginEnd="139dp"
     android:backgroundTint="#474242"
     android:text="@string/Stop"
     app: layout\_constraintBottom\_toBottomOf = "parent"
     app:layout_constraintEnd_toEndOf="parent" app:layout_constraintHorizontal_bias="0.0"
     app:layout_constraintStart_toStartOf="parent" app:layout_constraintTop_toTopOf="parent"
     app:layout_constraintVertical_bias="1.0" />
  <Button
     android:id="@+id/button6"
    android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="283dp"
     android:layout_marginTop="127dp"
android:layout_marginEnd="6dp"
     android:backgroundTint="#474242"
     android:text="@string/Pause"
    app:layout_constraintEnd_toEndOf="parent" app:layout_constraintEnd_toEndOf="parent" app:layout_constraintHorizontal_bias="0.0" app:layout_constraintStart_toStartOf="parent"
     app:layout_constraintTop_toTopOf="parent" app:layout_constraintVertical_bias="1.0" />
</androidx.constraintlayout.widget.ConstraintLayout>
```



activity pantalla reproductor.xml(land)

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"</p>
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:orientation="horizontal"
tools:context=".MainActivity">
   <androidx.constraintlayout.widget.ConstraintLayout
      android:id="@+id/constraintLayout2"
      android:layout_width="match_parent"
     android:layout_height="match_parent"
android:layout_marginStart="4dp"
android:layout_marginTop="4dp"
     android:layout_marginFop= 4dp"
android:layout_marginBottom="4dp"
app:layout_constraintBottom_toBottomOf="parent"
      app:layout_constraintEnd_toEndOf="parent"
      app: layout\_constraintHorizontal\_bias = "0.804"
      app:layout_constraintStart_toStartOf="parent"
     app:layout_constraintTop_toTopOf="parent" app:layout_constraintVertical_bias="0.057">
      <VideoView
         android:id="@+id/surface_view"
         android:layout_width="fill_parent"
         android:layout_height="fill_parent"
        android:layout_neignt= IIII_pareIII
android:layout_marginStart="8dp"
android:layout_marginEnd="8dp"
android:layout_marginBottom="8dp"
android:gravity="center_horizontal"
android:orientation="horizontal"
app:layout_constraintBottom_toBottomOf="parent"
         app:layout_constraintEnd_toEndOf="parent"
         app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent" app:layout_constraintTop_toTopOf="parent" app:layout_constraintVertical_bias="0.0" />
   </androidx.constraintlayout.widget.ConstraintLayout>
```



```
<\! and roidx. constraint layout. widget. Constraint Layout
  android:id="@+id/constraintLayout"
  android:layout_width="180dp"
  android:layout_height="294dp"
  android:layout_marginStart="595dp"
  android:layout_marginEnd="14dp"
  android:layout_marginBottom="40dp"
  app:layout_constraintBottom_toBottomOf="parent"
  app:layout constraintEnd toEndOf="parent"
  app:layout_constraintStart_toStartOf="parent"
  app:layout_constraintTop_toTopOf="parent">
  <Button
     android:id="@+id/button4"
    android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="52dp"
     android:layout_marginTop="32dp"
     android:layout_marginEnd="55dp"
    android:layout_marginBottom='29dp"
android:layout_marginBottom='29dp"
android:backgroundTint="#474242"
android:text="@string/Start"
app:layout_constraintBottom_toTopOf="@+id/button5"
     app:layout_constraintEnd_toEndOf="parent"
     app:layout_constraintHorizontal_bias="1.0"
     app:layout_constraintStart_toStartOf="parent"
     app:layout_constraintTop_toTopOf="parent" />
  <Button
     android:id="@+id/button5"
     android:layout_width="wrap_content"
     android:layout_height="wrap_content"
android:layout_marginStart="52dp"
     android:layout_marginTop="29dp"
     android:layout_marginEnd="55dp"
     android:layout_marginBottom="39dp"
     android:backgroundTint="#474242"
android:text="@string/Stop"
app:layout_constraintBottom_toTopOf="@+id/button6"
     app:layout_constraintStart_toStartOf="parent" app:layout_constraintStart_toStartOf="parent"
     app:layout_constraintTop_toBottomOf="@+id/button4"/>
  <Button
     android:id="@+id/button6"
    android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="49dp"
     android:layout_marginTop="39dp"
android:layout_marginEnd="58dp"
     android:layout_marginBottom="44dp"
    android:dydd_magmbtton= 444p
android:backgroundTint="#474242"
android:text="@string/Pause"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintEnd_toEndOf="parent"
     app:layout_constraintTop_toBottomOf="@+id/button5" />
</androidx.constraintlayout.widget.ConstraintLayout>
```



## 3. Ejemplos de funcionamiento

