



DESARROLLO DE APLICACIONES MÓVILES AVANZADAS



“USO DE AUDIO EN ANDROID”

Resultados de Aprendizaje:

- Comprender los elementos necesarios para iniciarse en el desarrollo de aplicaciones basadas en Android.
- Desarrollar proyectos haciendo uso de librerías avanzadas que proporciona Android.

INDICE

INTRODUCCIÓN	3
CREANDO NUESTRO PROYECTO	3
PRÁCTICA POR REALIZAR	11
RUBRICA DE EVALUACIÓN	11

INTRODUCCIÓN

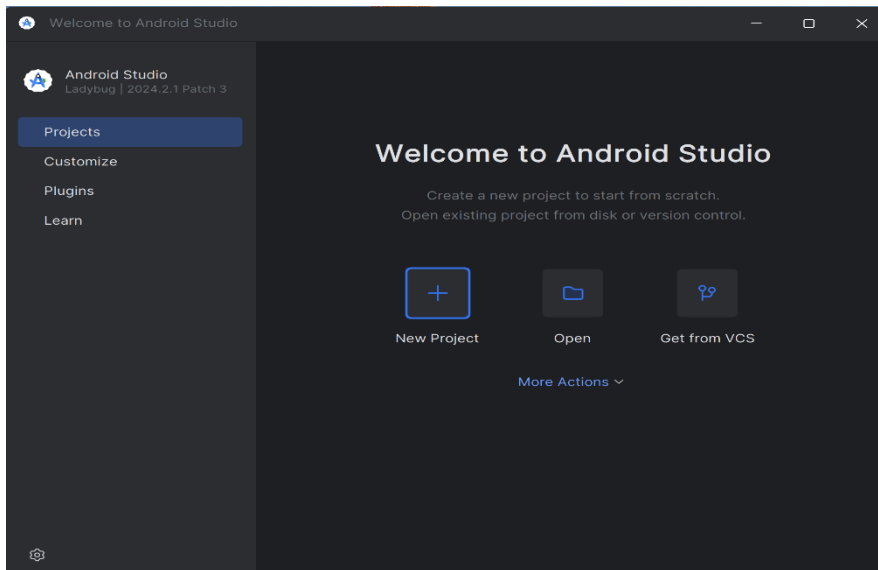
Para desarrollar esta práctica usted deberá tener los siguientes paquetes de software:

- Android Studio Versión 4 o superior
- Equipo Móvil con Android (Puede ser Tablet o Smartphone) con **modo desarrollador habilitado**.

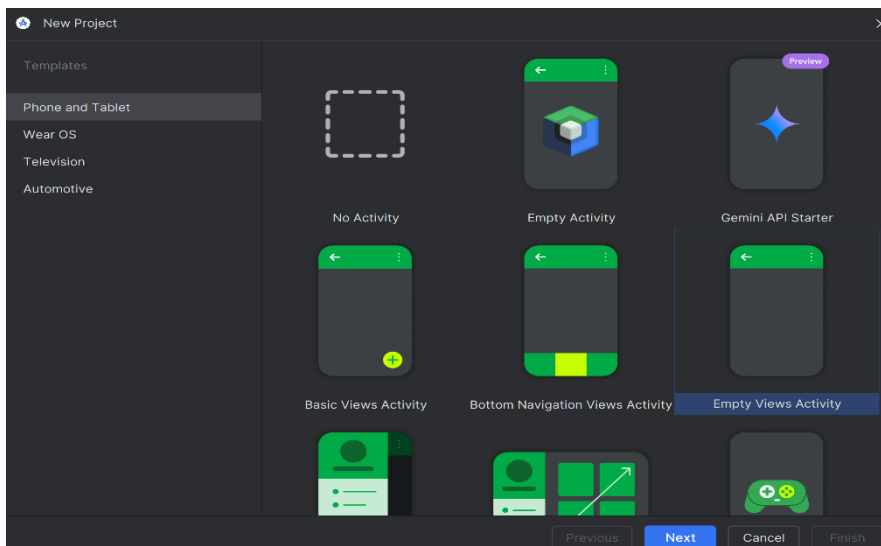
(Ver <https://www.youtube.com/watch?v=wLJS8CKo95o>)

CREANDO NUESTRO PROYECTO

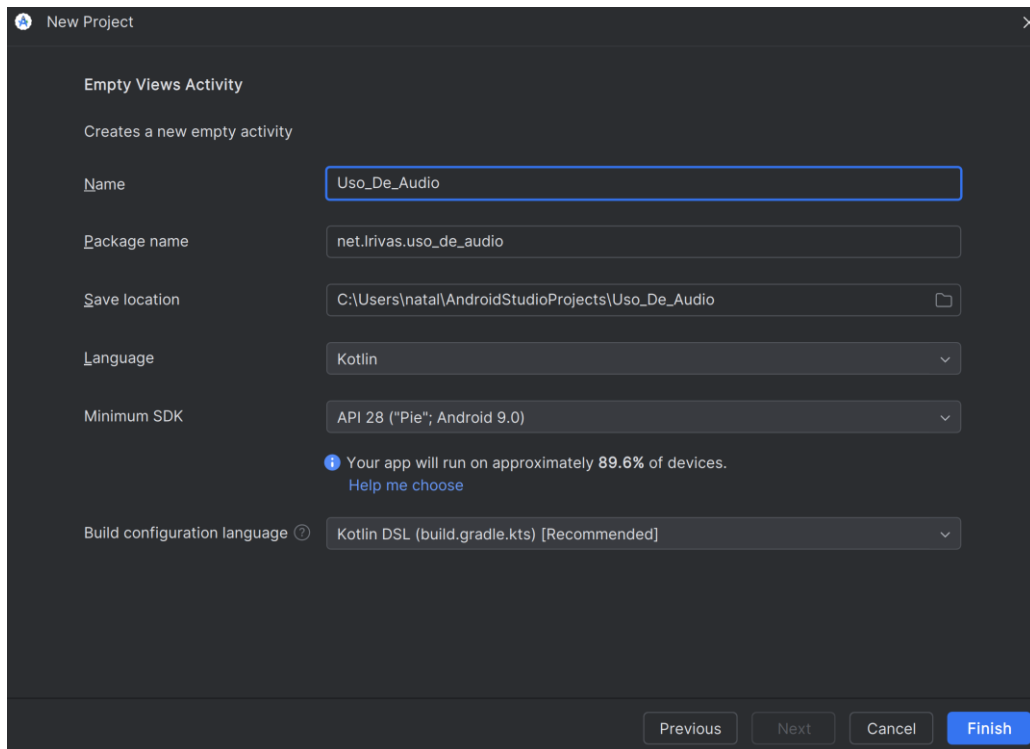
1. Iniciamos nuestro Android Studio.



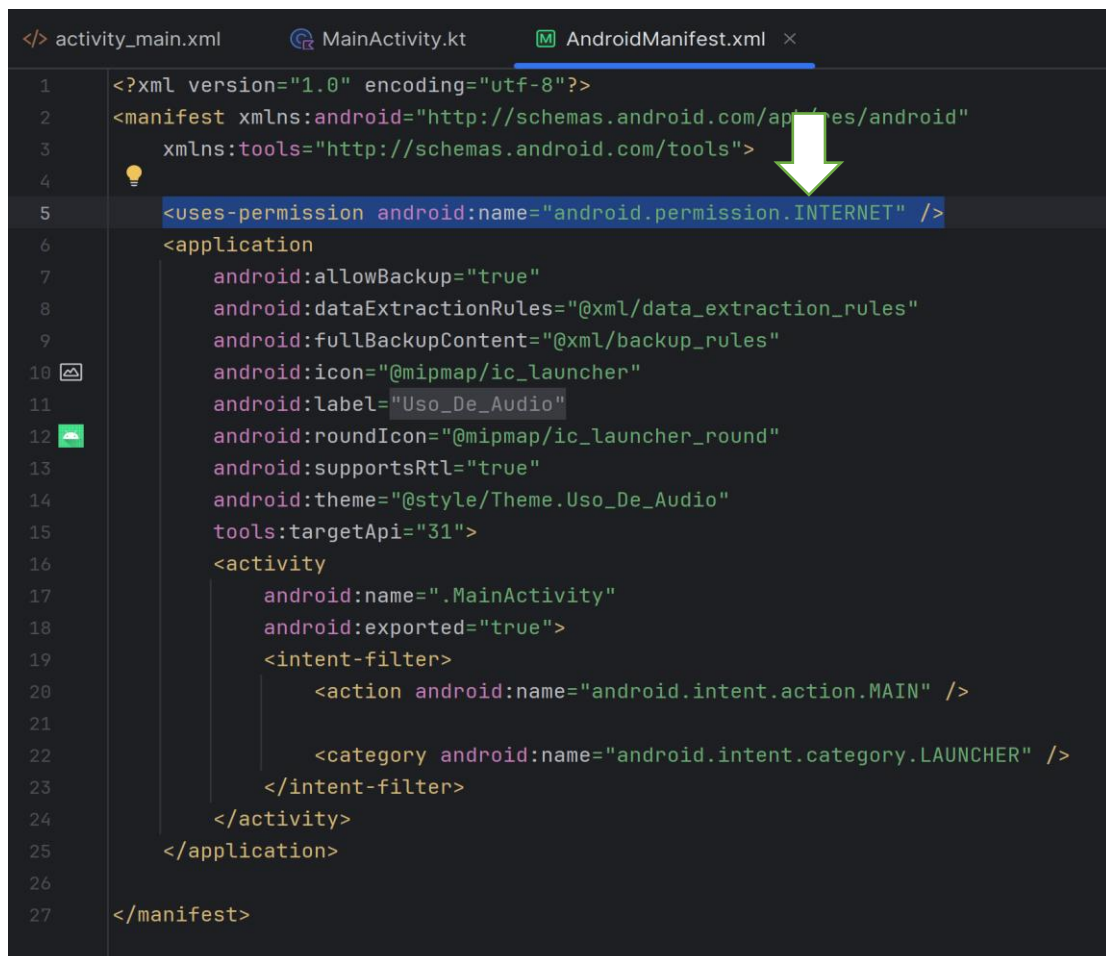
2. Elegimos la plantilla **Empty Views Activity**.



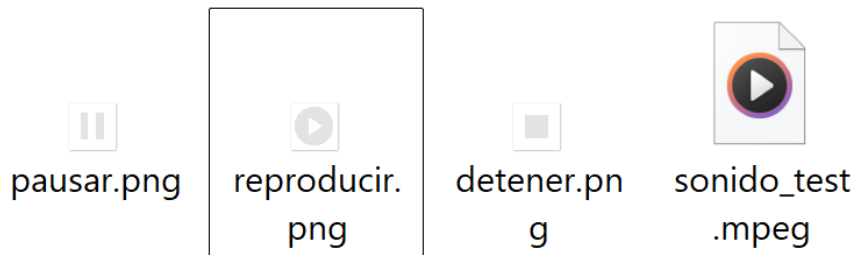
3. Luego establezca las siguientes propiedades al proyecto:
 - a. Nombre: **Uso_De_Audio**
 - b. Package Name: **net.lrivas.uso_de_audio**
 - c. Lenguaje: **Kotlin**
 - d. Minimum Api: **33**



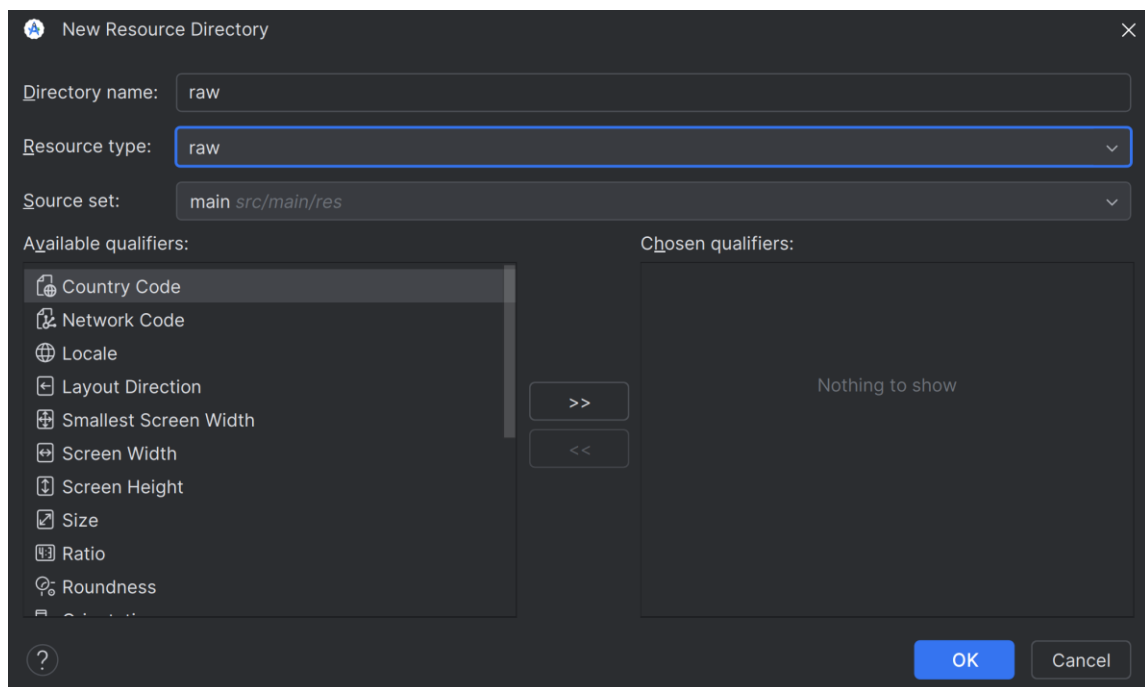
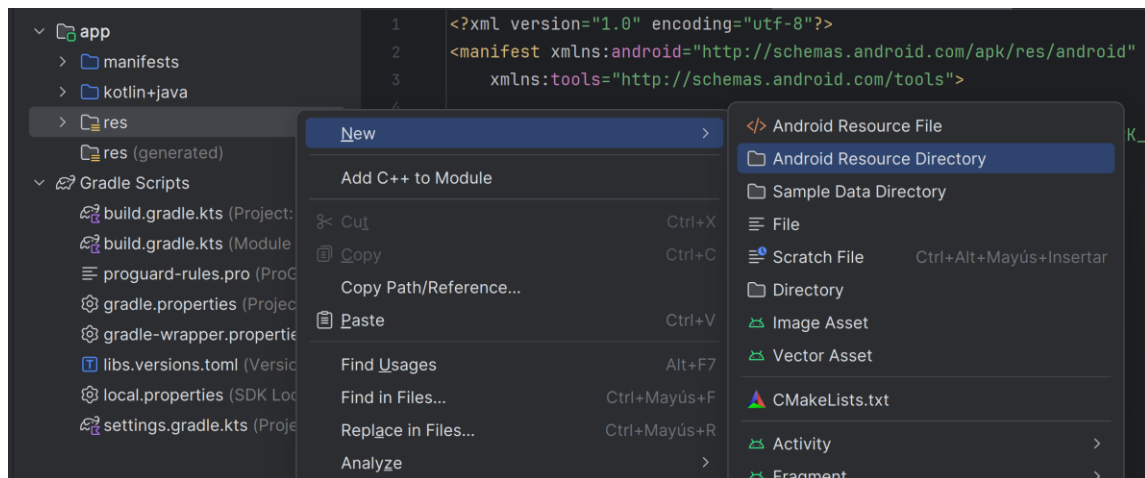
4. Implemente los permisos de acceso a internet de la aplicación, esto servirá para poder acceder de manera remota a un audio alojado en un servidor.



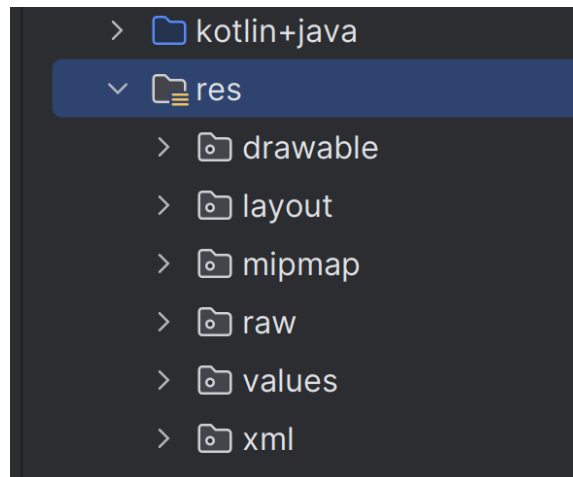
5. Descargue y descomprima el archivo **Practica 6 – Recursos.zip**.



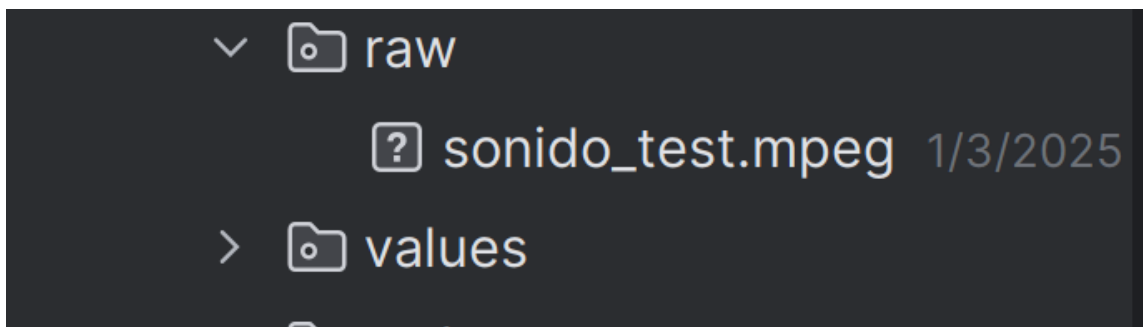
6. Cree un nuevo directorio de recursos con el nombre de **raw** dentro de la carpeta **res**. Tal cual como se aprecia en la siguiente secuencia de pantallas.



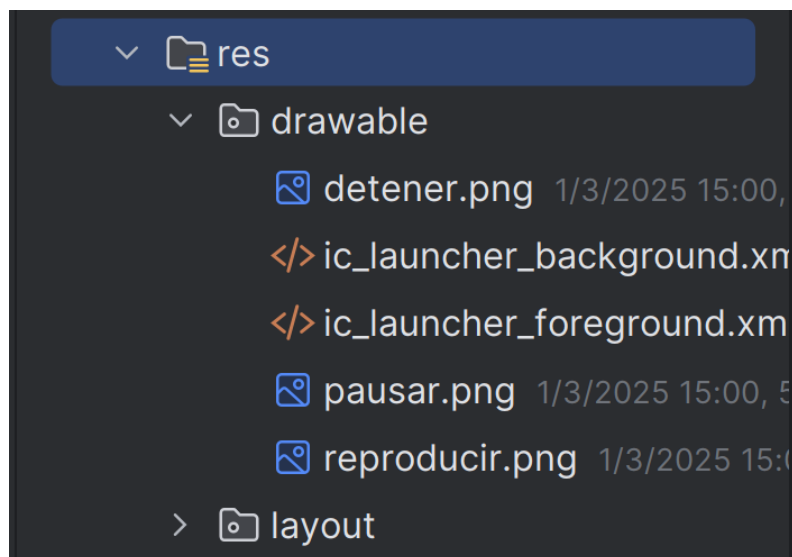
7. Deberá quedar así, nuestro directorio.



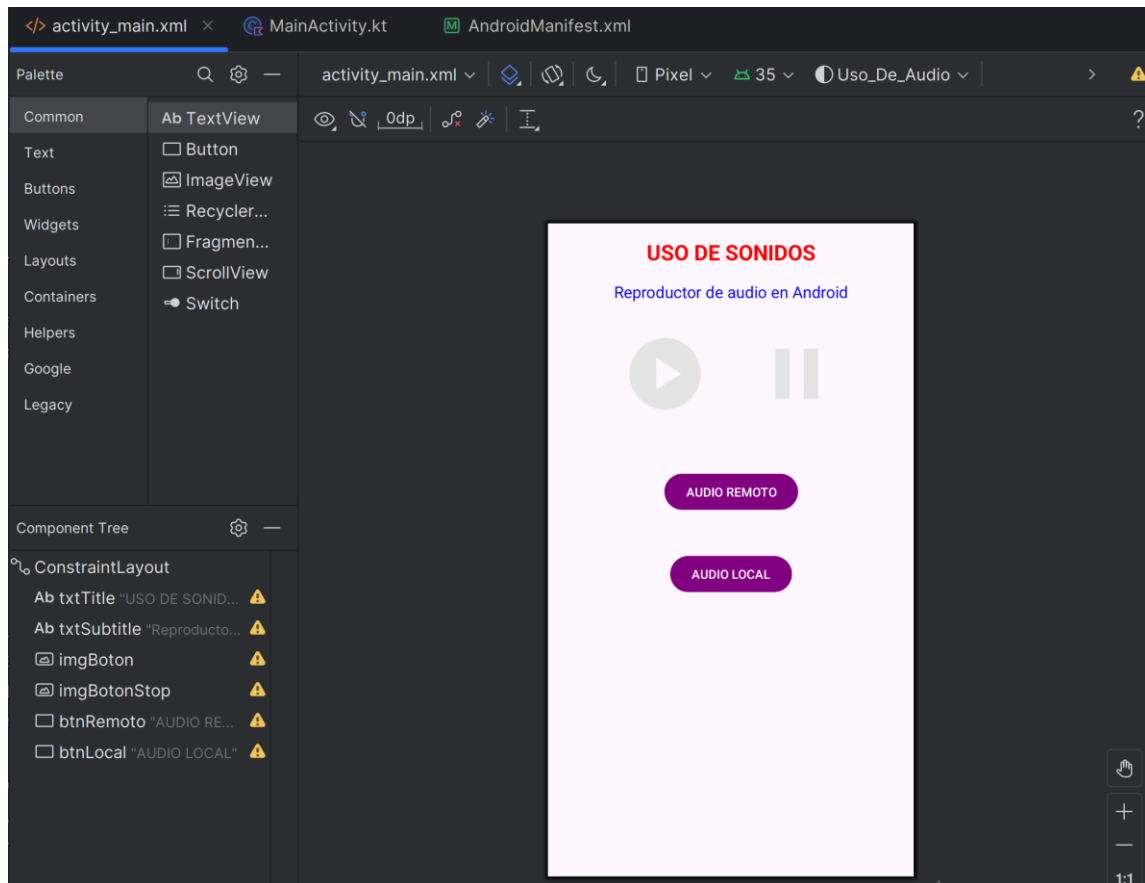
8. Ahora vamos a copiar el archivo sonido_test.mpeg dentro de la carpeta raw, quedando así.



9. Luego agregamos las 3 imágenes a la carpeta **drawable**, quedando así:



10. Ahora crearemos la siguiente interfaz gráfica en el archivo **activity_main.xml**.



11. Luego agregamos las siguientes líneas de código:

Declaramos los objetos a nivel de clase

```
class MainActivity : AppCompatActivity() {

    private var reproductorMedia: MediaPlayer? = null
    private var fuenteAudioEstablecida = false
```

Dentro del evento **onCreate** de la clase vamos a declarar las referencias de cada uno de los controles de la interfaz gráfica con las variables a utilizar en la clase.

```
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_main)

    val btnReproducir = findViewById<ImageButton>(R.id.imgBoton)
    val btnDetener = findViewById<ImageButton>(R.id.imgBotonStop)
    val btnRemoto = findViewById<Button>(R.id.btnRemoto)
    val btnLocal = findViewById<Button>(R.id.btnLocal)

    reproductorMedia = MediaPlayer()
```

Por último, vamos a colocar el evento clic para los botones.

btnLocal

```
btnLocal.setOnClickListener {
    try {
        if (reproductorMedia?.isPlaying == true) {
            reproductorMedia?.stop()
        }
        reproductorMedia?.reset()
        try {
            val idRecurso = R.raw.sonido_test
            reproductorMedia?.setDataSource(context, this, android.net.Uri.parse("android.resource://$packageName/$idRecurso"))

            reproductorMedia?.setOnPreparedListener {
                fuenteAudioEstablecida = true
                mostrarMensaje(mensaje: "Audio local listo para reproducir")
            }
            reproductorMedia?.setOnErrorListener { _, _, _ ->
                fuenteAudioEstablecida = false
                mostrarMensaje(mensaje: "Error al cargar el audio local")
                false
            }
            mostrarMensaje(mensaje: "Cargando audio local...")
            reproductorMedia?.prepare()
        } catch (e: Exception) {
            mostrarMensaje(mensaje: "Error al acceder al archivo de audio local")
        }
    } catch (e: Exception) {
        mostrarMensaje(mensaje: "Error al cargar el audio local")
    }
}
```

Ahora vamos a configurar el botón para acceder de manera remoto a un audio que se encuentra alojado en un servidor, la url del recurso es esta.

https://tonosmovil.net/wp-content/uploads/tonosmovil.net_himno_champions_league.mp3

btnRemoto

```
btnRemoto.setOnClickListener {
    try {
        if (!hayConexionInternet()) {
            mostrarMensaje(mensaje: "No hay conexión a internet")
            return@setOnClickListener
        }
        if (reproductorMedia?.isPlaying == true) {
            reproductorMedia?.stop()
        }
        reproductorMedia?.reset()
        val urlAudio = "https://tonosmovil.net/wp-content/uploads/tonosmovil.net_himno_champions_league.mp3"
        reproductorMedia?.setDataSource(urlAudio)
        reproductorMedia?.setOnPreparedListener {
            fuenteAudioEstablecida = true
            mostrarMensaje(mensaje: "Audio remoto listo para reproducir")
        }
        reproductorMedia?.setOnErrorListener { _, _, _ ->
            fuenteAudioEstablecida = false
            mostrarMensaje(mensaje: "Error al cargar el audio remoto")
            false
        }
        mostrarMensaje(mensaje: "Cargando audio remoto...")
        reproductorMedia?.prepareAsync()
    } catch (e: Exception) {
        mostrarMensaje(mensaje: "Error al configurar el audio remoto")
    }
}
```

Y por último los eventos clic para las imágenes que contienen los botones.

btnDetener

```

btnDetener.setOnClickListener {
    try {
        if (reproductorMedia?.isPlaying == true) {
            reproductorMedia?.stop()
            reproductorMedia?.reset()
            fuenteAudioEstablecida = false
            mostrarMensaje( mensaje: "Reproducción detenida")
        } else {
            mostrarMensaje( mensaje: "No hay audio reproduciéndose")
        }
    } catch (e: Exception) {
        mostrarMensaje( mensaje: "Error al detener el audio")
    }
}

```

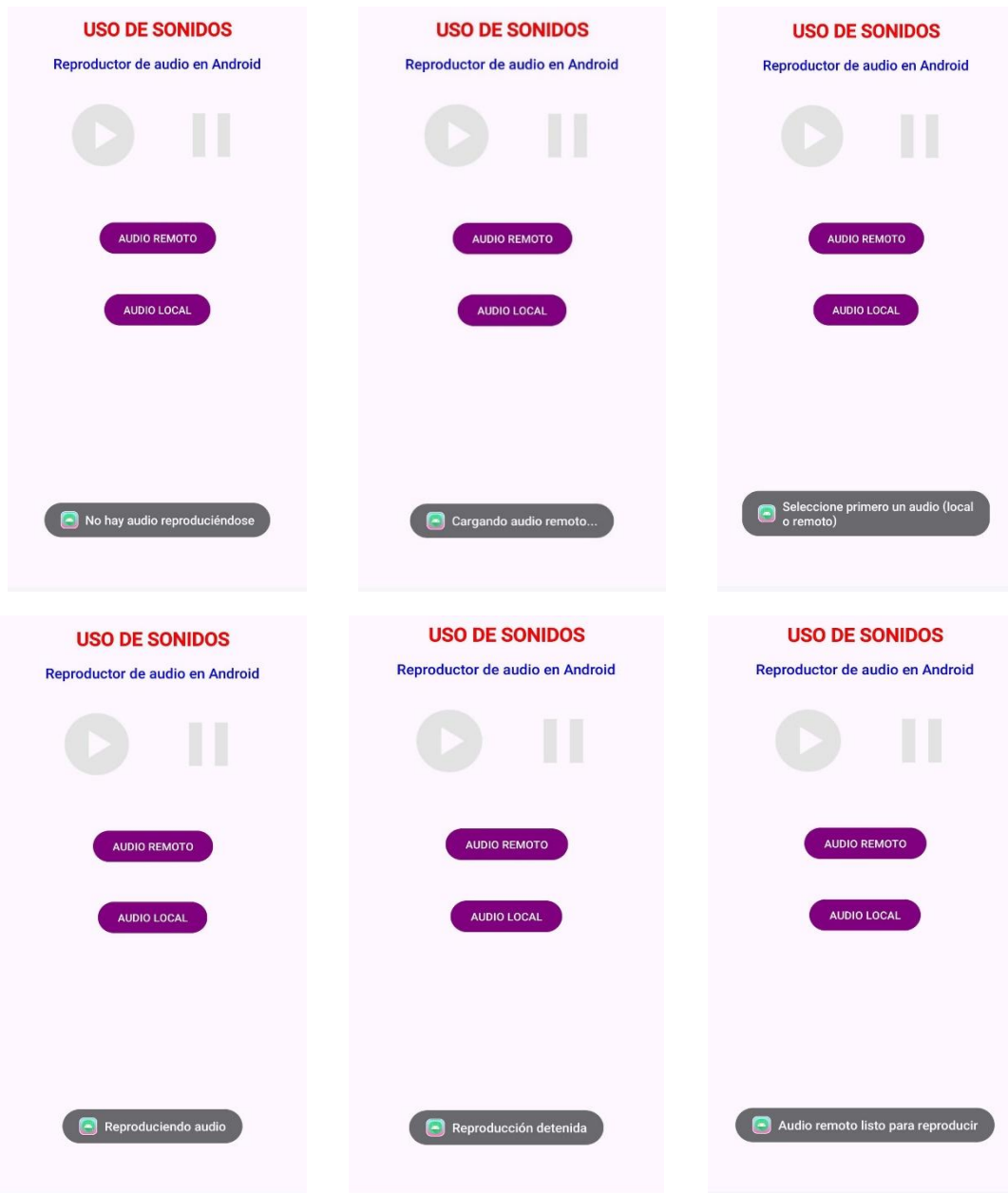
btnReproducir

```

btnReproducir.setOnClickListener {
    try {
        if (reproductorMedia?.isPlaying == true) {
            mostrarMensaje( mensaje: "El audio ya se está reproduciendo")
        } else if (fuenteAudioEstablecida) {
            reproductorMedia?.start()
            mostrarMensaje( mensaje: "Reproduciendo audio")
        } else {
            mostrarMensaje( mensaje: "Seleccione primero un audio (local o remoto)")
        }
    } catch (e: Exception) {
        mostrarMensaje( mensaje: "Error al reproducir el audio")
    }
}

```

12. Compile su aplicación, deberá cargarle de la siguiente manera:



PRÁCTICA POR REALIZAR

NOMBRE: Uso de Audios

Indicación: haciendo uso del contenido visto en la semana, genere un documento **PDF** utilizando la plantilla proporcionada por el docente, deberá contener las evidencias que respalden el siguiente requerimiento:

- A. Elabore una interfaz gráfica para un tutor virtual para niños. Entre las opciones a utilizar están (**quedará a decisión del estudiante cuál desarrollar**):
 - I. Aplicación para aprender los números en inglés y español.
 - II. Aplicación para aprender los colores en inglés y español.
 - III. Aplicación para aprender los nombres de animales en inglés y español.
- B. La aplicación deberá tener como **mínimo 20 elementos** (10 en inglés y 10 en español) cada elemento deberá reproducir un sonido **elaborado por el estudiante**.
- C. La aplicación deberá ser interactiva y utilizar recursos de imágenes que permitan dar un sentido más intuitivo a la aplicación.

FORMA DE ENTREGA: Se deberá enviar al buzón de tarea llamado **Uso de Librerías para Audio**, de manera individual.

TIPO DE ENTREGA: Enviar un documento en **PDF o DOCX** al final de la semana antes de las **23:59**.

RUBRICA DE EVALUACIÓN

Indicación: a continuación, se establecen los criterios de evaluación para la actividad de la semana.

#	CRITERIO	PTS.
1	Entrega el proyecto en la fecha establecida.	2.0
2	Diseña la interfaz principal de acuerdo con el requerimiento.	0.5
3	Hace uso de imágenes para representar los elementos de aprendizaje.	1.0
4	Contiene los 20 elementos de aprendizaje solicitado.	3.0
5	Los audios son propiedad intelectual del estudiante.	1.0
6	Elabora el ejemplo de la práctica.	1.0
7	Sube el proyecto a git y anexa el enlace en el documento.	0.5
8	El documento está ordenado según la plantilla y sin errores de ortografía.	1.0