

109. Librerías multimedia

109.1 Multimedia con Jetpack Compose

Documentación en AD versión actual [media3](#) en versión 1.0.0 que se corresponde a la versión de `ExoPlayer` 2.18.3.

En esencia, Media3 es una modernización y expansión de la conocida biblioteca ExoPlayer

109.1.1 Características:

- **Reproductor Unificado:** Media3 ofrece un reproductor unificado para todo tipo de medios. Ya sea que estés transmitiendo desde la web, reproduciendo archivos locales o trabajando con DASH (Dynamic Adaptive Streaming over HTTP), Media3 lo maneja todo.
- **Soporte para diferentes formatos:** Soporta una gran variedad de formatos de audio y video, incluyendo MP4, MP3, WAV, AAC, MKV, FLV, y muchos más. Además, es compatible con formatos de streaming como HLS y DASH.
- **Personalización y Control:** Te da mucho control y personalización. Puedes crear tu propia interfaz de usuario para el reproductor o usar los componentes proporcionados por la biblioteca.
- **Eficiencia y Rendimiento:** Media3 es eficiente en el uso de recursos del sistema, lo que es crucial para aplicaciones que manejan mucho contenido multimedia.
- **Fácil de Integrar:** Al ser parte del ecosistema Jetpack, Media3 se integra bien con otras bibliotecas de Jetpack y sigue las mejores prácticas de Android.
- **Actualizaciones y Mantenimiento:** Dado que es un producto de Google, puedes esperar actualizaciones regulares y un buen mantenimiento, asegurando que esté al día con las últimas tendencias y tecnologías.

109.2 Introducción [media3](#)

En una arquitectura simple tendríamos de un reproductor multimedia y un interfaz gráfico para controlar la reproducción.

Esta arquitectura permite una conectividad limitada.

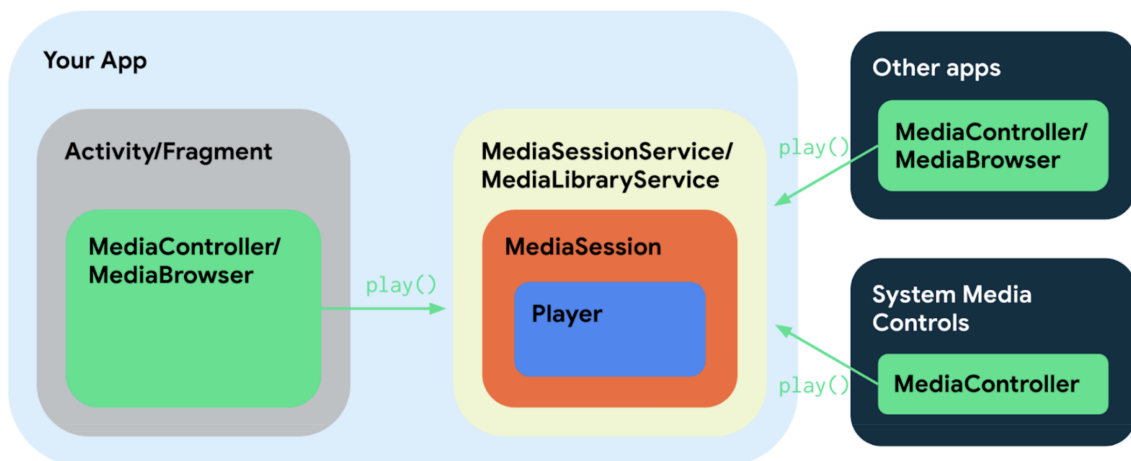
Al añadir una nueva arquitectura con una `sesión multimedia` nos permite tener un abanico amplio de integraciones:

- Desde Android 11 , la reproducción se refleja en el Media Control Cox incluido en Android y permite controlar la reproducción y volumen en el panel de configuración rápida.
- Controlar el reproductor desde Wear OS
- Desde Google Assistan con controles de voz
- Procesamiento automático de eventos externos , como pulsar el mando de tv android

Para trabajar con ExoPlayer, normalmente utilizas varias bibliotecas que dependen de tus necesidades específicas de reproducción de medios. Aquí te doy una idea general de las librerías más comunes y sus usos:

- Librería básica de ExoPlayer: Es el núcleo de ExoPlayer e incluye funcionalidades básicas de reproducción de medios. `com.google.android.exoplayer:exoplayer-core:2.X.X`
- UI: Componentes de la interfaz de usuario para controlar la reproducción de medios, como botones de play/pausa, barra de progreso, etc. `com.google.android.exoplayer:exoplayer-ui:2.X.X`
- Dash: Soporte para streaming adaptativo DASH (Dynamic Adaptive Streaming over HTTP). `com.google.android.exoplayer:exoplayer-dash:2.X.X`
- HLS: Soporte para streaming HLS (HTTP Live Streaming). `com.google.android.exoplayer:exoplayer-hls:2.X.X`
- SmoothStreaming: Soporte para Microsoft Smooth Streaming. `com.google.android.exoplayer:exoplayer-smoothstreaming:2.X.X`
- Extensiones: ExoPlayer también ofrece extensiones para soportar formatos o funcionalidades adicionales, como decodificación de hardware, reproducción de VR, etc. Estas se añaden según sea necesario.

Componentes media3



109.3 Reproductor multimedia

Un reproductor multimedia es un componente de tu app que permite la reproducción de archivos multimedia. En Media3, encontrarás lo siguiente:

- **Player** : Player es una interfaz que define las capacidades tradicionales de alto nivel para un reproductor multimedia, como la capacidad de reproducir, pausar y buscar.
- **ExoPlayer** : ExoPlayer es la implementación predeterminada de la interfaz Player en Media3.

109.4 Sesión multimedia

Una sesión multimedia proporciona una forma universal de interactuar con un reproductor multimedia. Esto permite que una app anuncie la reproducción de contenido multimedia a fuentes externas y reciba solicitudes de control de reproducción de fuentes externas. En Media3, encontrarás lo siguiente:

- **MediaSession** : Las sesiones multimedia permiten que tu app interactúe con MediaController. Por lo general, la clase MediaController se usa para enviar comandos desde fuera de la app, por ejemplo, desde otras apps o el propio sistema. Los comandos se envían al Player subyacente del MediaSession asociado. n un reproductor de audio o video. *Publicitan* la reproducción de contenido multimedia de forma externa y reciben comandos de reproducción de fuentes externas.
- **MediaSessionService** : El MediaSessionService contiene una sesión multimedia y su reproductor asociado en un servicio independiente del Activity principal de tu app para facilitar la reproducción en *segundo plano*.
- **MediaController** : Por lo general, la clase MediaController se usa para enviar comandos desde fuera de la app, por ejemplo, desde otras apps o el propio sistema. Los comandos se envían al Player subyacente del MediaSession asociado. La clase MediaController implementa la interfaz Player, pero cuando se llama a un método, el comando se envía al MediaSession conectado. Las apps cliente, como Asistente de Google, pueden usar MediaController para controlar la reproducción en una sesión conectada.
- **MediaLibraryService** : Un MediaLibraryService es similar a un MediaSessionService, con la excepción de que incluye API adicionales para que puedas entregar tu biblioteca de contenido a apps cliente. MediaBrowser
- **MediaBrowser** La clase MediaBrowser permite al usuario navegar por la biblioteca de contenido de una app de música y seleccionar qué elementos reproducir.

109.5 Componentes de la IU

PlayerView Una View predeterminada para mostrar los controles de video y reproducción. Se conecta a ExoPlayer, MediaController o cualquier otro Player personalizado.

109.6 Componentes para editar multimedia

- **Transformer** : Usa la clase Transformer para iniciar y detener las transformaciones y para verificar las actualizaciones del progreso de una transformación en ejecución.
- **Effects** : Un objeto Effects es una colección de efectos de audio y video que se aplican a un elemento multimedia.
- **EditedMediaItem** : Una EditedMediaItem representa un elemento multimedia que se debe procesar y las ediciones que se le aplican. Puedes usar ExoPlayer para obtener una vista previa de los efectos agregados a un elemento multimedia antes de comenzar el proceso de exportación.

109.7 Ejemplo

109.7.1 1. Crear directorio para guardar la musica:

```
->res
|----> raw
```

109.7.2 2. Dependencias

[Media3 Exoplayer](#) es una extensión de la ampliamente utilizada biblioteca ExoPlayer, aporta capacidades avanzadas de reproducción de medios a aplicaciones Android. En este proyecto, Media3 ExoPlayer se emplea para manejar la reproducción de pistas musicales.

```
implementation ("com.google.accompanist:accompanist-systemuicontroller:0.33.2-alpha")
implementation ("androidx.media3:media3-exoplayer:1.1.1")
```

109.8 Otro ejemplo

1. Añadir dependencias.

```
dependencies {
    val media3_version = "1.0.0-rc01"

    // For media playback using ExoPlayer
    implementation("androidx.media3:media3-exoplayer:$media3_version")

    // For DASH playback support with ExoPlayer
    implementation("androidx.media3:media3-exoplayer-dash:$media3_version")
    // For HLS playback support with ExoPlayer
    implementation("androidx.media3:media3-exoplayer-hls:$media3_version")
    // For RTSP playback support with ExoPlayer
    implementation("androidx.media3:media3-exoplayer-rtsp:$media3_version")
    // For ad insertion using the Interactive Media Ads SDK with ExoPlayer
    implementation("androidx.media3:media3-exoplayer-ima:$media3_version")

    // For loading data using the Cronet network stack
    implementation("androidx.media3:media3-datasource-cronet:$media3_version")
    // For loading data using the OkHttp network stack
    implementation("androidx.media3:media3-datasource-okhttp:$media3_version")
    // For loading data using librtmp
    implementation("androidx.media3:media3-datasource-rtmp:$media3_version")

    // For building media playback UIs
    implementation("androidx.media3:media3-ui:$media3_version")
    // For building media playback UIs for Android TV using the Jetpack Leanback
    library
    implementation("androidx.media3:media3-ui-leanback:$media3_version")

    // For exposing and controlling media sessions
    implementation("androidx.media3:media3-session:$media3_version")

    // For extracting data from media containers
    implementation("androidx.media3:media3-extractor:$media3_version")

    // For integrating with Cast
    implementation("androidx.media3:media3-cast:$media3_version")

    // For scheduling background operations using Jetpack Work's WorkManager with
    ExoPlayer
    implementation("androidx.media3:media3-exoplayer-workmanager:$media3_version")
}
```

```
// For transforming media files
implementation("androidx.media3:media3-transformer:$media3_version")

// Utilities for testing media components (including ExoPlayer components)
implementation("androidx.media3:media3-test-utils:$media3_version")
// Utilities for testing media components (including ExoPlayer components) via
Robolectric
implementation("androidx.media3:media3-test-utils-robolectric:$media3_version")

// Common functionality for media database components
implementation("androidx.media3:media3-database:$media3_version")
// Common functionality for media decoders
implementation("androidx.media3:media3-decoder:$media3_version")
// Common functionality for loading data
implementation("androidx.media3:media3-datasource:$media3_version")
// Common functionality used across multiple media libraries
implementation("androidx.media3:media3-common:$media3_version")
}
```

¿Fue útil esta página?

