



Título del Proyecto:

FULLMEDIAPLAYER

Autor:

Rico Portalés, Carlos

Director:

Carbonell Frasquet, Vicente

**TESINA PARA LA
OBTENCIÓN DEL TÍTULO DE:**

**Máster en Desarrollo de
Aplicaciones sobre Dispositivos
Móviles**

Septiembre del 2016



Contenido

Título del Proyecto:	1
Autor:	1
Director:	1
Máster en Desarrollo de Aplicaciones sobre Dispositivos Móviles.....	1
Introducción	4
Descripción del problema	4
Objetivos	4
Motivación	5
Tecnologías utilizadas	5
Arquitectura de la aplicación	7
JSON, listas de reproducción.....	7
Esquema del diseño	8
Código Java.....	9
Recursos utilizados.....	12
Capturas de pantalla	14
Conclusiones	25

Introducción

Descripción del problema

Este proyecto ha sido elaborado como trabajo fin del Master en desarrollo de aplicaciones sobre dispositivos móviles.

El proyecto pretende desarrollar una aplicación que permita reproducir audio y video en streaming desde Internet, así como cualquier fichero de audio y video almacenado en nuestro dispositivo.

Todas las tecnologías utilizadas han sido bajo la plataforma Android, dado que la misma es la más extendida en el mundo de las aplicaciones para dispositivos móviles, y proporciona multitud de herramientas gratuitas para su desarrollo.

Objetivos

El principal objetivo perseguido por la aplicación, como ya se ha mencionado con anterioridad, es poder reproducir desde la misma tanto video y audio desde streaming, así como video y audio en local.

Sin embargo, existen otros objetivos adicionales que han sido perseguidos para hacer más operativa la aplicación. Entre ellos podemos destacar:

1. Otorgar a la aplicación la capacidad de trabajar con listas de reproducción en formato JSON que podemos encontrar publicadas en Internet.
2. La aplicación debe permitir que el usuario de la misma pueda generar sus propias listas de reproducción.
3. Debe presentar la capacidad de poder trabajar con Google Drive para preservar copias de seguridad de las listas de reproducción almacenadas en el dispositivo.
4. La aplicación debe estar preparada para poder ser traducida a cualquier idioma.

Motivación

Dado que en mi vida profesional, nunca había tenido oportunidad de trabajar con aplicaciones relacionadas con audio y vídeo, y en el entorno de los dispositivos móviles este tipo de elementos (audio y vídeo) son utilizados de forma muy extendida, me resultaba interesante la aplicación que he desarrollado.

Otro motivo que me ha llevado a desarrollar esta aplicación, ha sido la necesidad de utilizar las comunicaciones con Internet, así como la utilización del api de Google Drive.

Tecnologías utilizadas

Para una mejor comprensión del funcionamiento de la aplicación, en esta sección exponemos una breve explicación de algunas de las tecnologías utilizadas:

- JSON/GSON

JSON (JavaScript Object Notation) es un formato utilizado actualmente para el intercambios de datos, JSON permite definir los datos con una sintaxis especialmente diseñada para identificar y gestionar los datos. JSON surge como alternativa al lenguaje XML. Una de las mayores ventajas que tiene el uso de JSON es que puede ser interpretado fácilmente por cualquier lenguaje de programación. Por lo tanto, puede ser usado para el intercambio de información entre distintas tecnologías.

Dadas las características del lenguaje JSON, actualmente en Internet podemos encontrar listas de reproducción basadas en este lenguaje. Dado que el formato interno de las listas de reproducción no están estandarizados es posible encontrar ficheros en Internet que no puedan ser correctamente interpretados por la aplicación.

La librería GSON (Google JSON) es una librería de código abierto diseñada para el lenguaje de programación Java que nos permite la serialización y deserialización entre objetos JAVA y objetos JSON, de forma fácil y eficiente.



- Google Drive REST v.2

Google Drive es un servicio de alojamiento de archivos introducido por Google y que de forma gratuita a todos aquellos usuarios que disponen de una cuenta de correo Gmail disponen de una cuota de almacenamiento de 15Gb.

REST, REpresentational State Transfer, es un tipo de arquitectura de desarrollo web que se apoya totalmente en el estándar HTTP. El término REST actualmente se usa para describir cualquier interfaz entre sistemas que utilice directamente HTTP para obtener datos o indicar la ejecución de operaciones sobre los datos, en cualquier formato (XML, JSON, etc) sin las abstracciones adicionales de los protocolos basados en patrones de intercambio de mensajes, como por ejemplo SOAP.

Google Drive REST es una API ofrecida de forma gratuita por Google que nos permite integrar en nuestras aplicaciones, de una forma simple, el acceso a Google Drive a través de solicitudes de tipo REST.

- Librería ZXING

El proyecto ZXing es un proyecto open-source que ofrece soporte para la lectura y decodificación para la gran mayoría de códigos de barras, códigos BIDI o QR en múltiples plataformas.

En nuestro caso, nos proporciona código Java para generar y decodificar QRcodes. Proporciona a su vez una aplicación de decodificación para Android, Barcode Scanner, que puede ser llamada y utilizada desde nuestra aplicación.

Arquitectura de la aplicación

JSON, listas de reproducción

Para el desarrollo de la aplicación no ha sido utilizado ningún modelo de datos propio, dado que uno de sus principales objetivos era poder interpretar ficheros de tipo JSON (listas de reproducción) que habitualmente se encuentran en la red Internet.

La aplicación almacena en un directorio denominado **FullMediaPlayer**, todos los ficheros de datos que utiliza para su ejecución (listas de reproducción y archivos de imagen), de forma que la app durante su ejecución consulta este directorio con el objetivo de trabajar con las listas de reproducción.

Dado que no existe un estándar sobre el formato de estos ficheros, nuestra aplicación trabajará con ficheros que cumplan con el siguiente esquema JSON:

```
{
  "name": "",
  "author": "",
  "url": "",
  "parental": "",
  "info": "",
  "stations":
    [
      {
        "name": "",
        "image": "",
        "url": ""
      }, ...
    ]
}
```

Veamos los distintos campos, y aquellos que siempre se deben encontrar para poder ser considerada una lista válida.

- name. Representa el nombre de la lista.
- autor. Contendrá el nombre del autor, este nombre puede estar vacío.
- url. Debe contener la dirección url desde la cual descargamos la lista. En una lista creada por el usuario esta dirección se encuentra vacía.
- parental. Campo opcional de tipo booleano que informa si la lista debe ser o no bloqueada en caso de activar el control parental.
- info. Campo opcional de texto que contendrá información adicional asociada a la lista.
- stations. Vector que contendrá la lista de medios asociados a la lista.
- stations.name. Nombre del medio
- stations.image. Url desde la cual descargar la imagen asociada al medio.
- stations.url. Url desde la cual podemos acceder al contenido.

La representación JAVA de las listas de reproducción, se pueden encontrar en los ficheros de la aplicación:

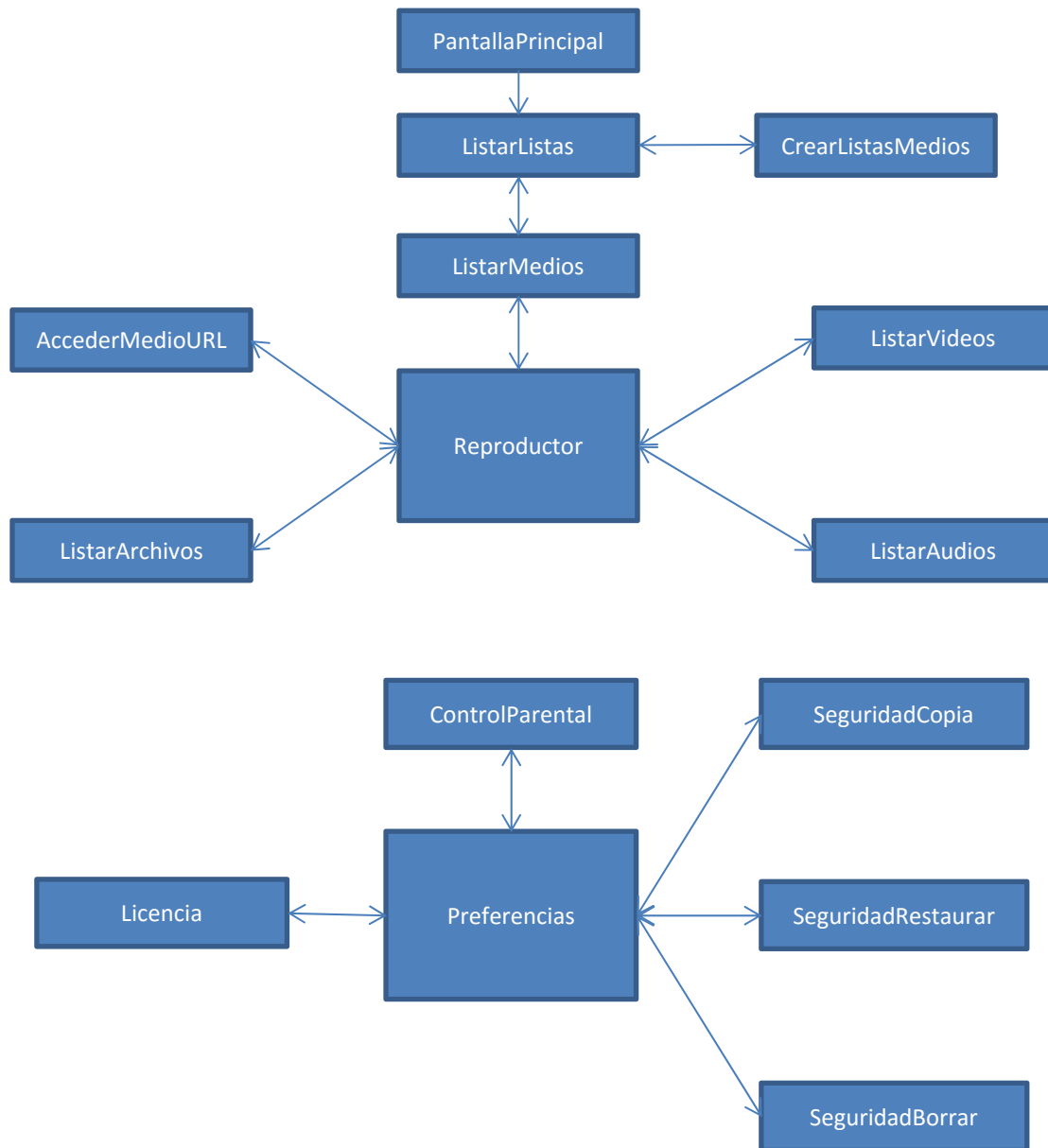
- ListaMedios.java
- Medio.java

Esquema del diseño

La aplicación se encuentra dividida en las siguientes secciones principales:

- ListarListas
- ListarVideos
- ListarAudios
- ListarArchivos
- AccederMedioURL
- Preferencias

La aplicación dispone de un un Drawer Layout que visualiza un menú principal que nos permite acceder desde cualquier sección principal a otra sección. Por ello, y para simplificar el esquema de diseño no se representarán las transiciones a las secciones principales.



Código Java

- **AccederMedioURL.** Este fichero representa el código asociado a la actividad visualizar medio desde URL. Cabe destacar al igual que ocurre con las listas de reproducción cargadas desde internet que antes de mandar al reproductor el medio a reproducir, se comprueba si dicho medio es realmente accesible.
- **AdaptadorEntradaSistema.** Este fichero crea un adaptador para un RecyclerView que es utilizado en la actividad que permite listar los archivos de audio y video almacenados en nuestro dispositivo.

- **AdaptadorListas.** Este fichero crea un adaptador para un RecyclerView que es utilizado por la actividad que visualiza las listas cargadas en nuestra aplicación.
- **AdaptadorMedios.** Este fichero crea un adaptador para un RecyclerView que es utilizado por todas aquellas actividades que listan elementos a reproducir (ListarMedios, ListarVideos, ListarAudios y CrearListaMedios)
- **AppCompatPreferenceActivity.** Esta clase de código abierto permite crear una actividad de tipo preferencias que contenga una ActionBar. Es utilizada con el objetivo de mantener un diseño lo más unificado posible entre todas las partes de la aplicación.
- **ControlParental.** Esta actividad visualizará un cuadro de diálogo que nos permite habilitar o deshabilitar el control parental dentro de la aplicación.
- **CrearListasMedios.** Este fichero representa la actividad que permite crear y almacenar una lista de reproducción personalizada.
- **EntradaSistemaArchivo.** Esta clase representa la modelización de una entrada dentro del sistema de archivos del dispositivo. Esta clase es utilizada por la actividad **ListarArchivos** y el adaptador **AdaptadorEntradaSistema**.
- **InfoGlobal.** Esta clase es utilizada con el objetivo de definir los datos globales de la aplicación bajo el patrón Singleton.
- **JSONParser.** Este fichero define el mecanismo que permite a partir de un fichero de tipo JSON obtener una cadena de texto con su contenido.
- **Libreria.** Este fichero define enumerados, vectores, variables y funciones que son utilizados en distintas partes del código.
- **Licencia.** Esta actividad visualizará la licencia del programa.
- **ListaMedios.** Esta clase representa la modelización de una lista de reproducción dentro de nuestro código. Según ya se ha comentado en un apartado anterior, nuestras listas de reproducción contendrán los siguientes campos:
 - name. Nombre de la lista
 - author. Nombre del autor de la lista.



- url. URL desde la cual se descarga la lista de reproducción.
 - nameFile. Nombre del fichero con el que se almacena la lista de reproducción en nuestro sistema de archivos.
 - info. Información adicional asociada a la lista de reproducción.
 - parental. Establece si la lista debe ser de acceso libre o limitado cuando este activado el control parental.
 - stations. Lista de medios a reproducir, estos medios se encuentran modelizados en el fichero **Medio**.
-
- **ListarArchivos**. Esta actividad nos permite navegar por el sistema de archivos de nuestro dispositivo, mostrándonos únicamente aquellos ficheros de audio y video que pueden ser reproducidos desde el sistema Android. En caso de seleccionar alguno de los ficheros este será reproducido por la aplicación.
 - **ListarAudios**. Esta actividad permite visualizar todos los audios almacenados en nuestro dispositivo consultando el objeto **MediaStore**.
 - **ListarListas**. Este fichero contiene la actividad que visualizará y nos permitirá trabajar con las listas de reproducción.
 - **ListarMedios**. Este fichero contiene la actividad que visualizará los elementos contenidos dentro de una lista de reproducción.
 - **ListarVideos**. De forma similar a como lo hace la actividad **ListarAudios**, esta actividad visualizará los videos almacenados en nuestro dispositivo.
 - **Medio**. Esta clase modeliza un elemento a reproducir, este medio a reproducir puede bien provenir de una lista de reproducción, o bien a partir de las listas obtenidas a partir del objeto **MediaStore**.
 - **PantallaPrincipal**. Esta actividad, siendo la principal de nuestra app, tiene como único objetivo mostrar un mensaje de espera mientras se están cargando en memoria las listas de reproducción almacenadas en el dispositivo.
 - **Preferencias** y **PreferenciasFragment**. Estos dos ficheros son utilizados para crear la actividad de configuración.
 - **ReproductorWeb**. Representa la actividad que permite mediante un objeto del tipo **WebView** permite reproducir los distintos elementos de audio y video.

- **SeguridadBorrar.** Esta actividad es la responsable de borrar todos los archivos de copia de seguridad en **Google Drive**.
- **SeguridadCopia.** Esta actividad es la responsable de realizar una copia de seguridad de todos los ficheros de listas de reproducción almacenados en nuestro dispositivo sobre **Google Drive**.
- **SeguridadRestaurar.** Esta actividad es la responsable de recuperar las copias de seguridad en Google Drive realizadas mediante la anterior actividad.

Recursos utilizados

- Drawable
 - **fondo.png.** Esta imagen se utiliza como fondo en la cabecera del **Drawer Layout**.
 - **generic256.png.** Esta imagen representa una carpeta, utilizada en el adaptador **AdaptadorEntradaSistema**.
 - **ic_exit_to_app_black_18dp.png.** Esta imagen es utilizada en el menú principal para acompañar a la opción de **Salir**.
 - **ic_folder_black_18dp.png.** Esta imagen es utilizada en el menú principal para acompañar a la opción de **Archivo local**.
 - **ic_list_black_18dp.png.** Esta imagen es utilizada en el menú principal para acompañar a la opción de **Lista reproducción**.
 - **ic_more_vert_black_18dp.png.** Esta imagen es utilizada dentro del layout **elemento_lista** para indicar que el elemento tiene un menú asociado al realizar un toque largo sobre el mismo.
 - **ic_music_note_black_18dp.png.** Esta imagen es utilizada en el menú principal para acompañar a la opción de **Audio local**.
 - **ic_music_video_black_24dp.png.** Esta imagen es utilizada por el **AdaptadorMedios** para representar aquellos medios que no disponen de una imagen propia.
 - **ic_settings_applications_black_18dp.png.** Esta imagen es utilizada en el menú principal para acompañar a la opción de **Configuración**.
 - **ic_videocam_black_18dp.png.** Esta imagen es utilizada en el menú principal para acompañar a la opción de **Video local**.
 - **ic_web_black_18dp.png.** Esta imagen es utilizada en el menú principal para acompañar a la opción de **Medio desde url**.
 - **musicfile256.pgn.** Esta imagen representa un fichero de audio, utilizada en el adaptador **AdaptadorEntradaSistema**



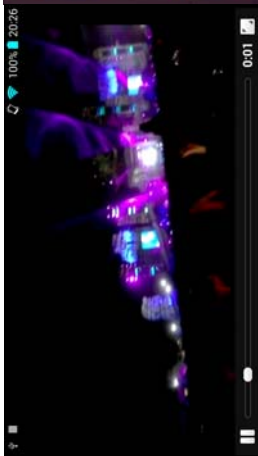
- **videofile256.png**. Esta imagen representa un fichero de video, utilizada en el adaptador **AdaptadorEntradaSistema**
- **Layout**
 - **activity_acceder_medio_url**. Vistas asociadas a la actividad **AccederMedioURL**.
 - **activity_control_parental**. Vista asociada a la actividad **ControlParental**.
 - **activity_crear_listas_medios, content_crear_listas_medios**. Vistas asociadas a la actividad **CrearListasMedios**.
 - **activity_licencia**. Vista asociada a la actividad **Licencia**.
 - **activity_listar_archivos, content_listar_archivos, app_bar_listar_archivos**. Vistas asociadas a la actividad **ListarArchivos**.
 - **activity_listar_audios, content_listar_audios, app_bar_listar_audios**. Vistas asociadas a la actividad **ListarAudios**.
 - **activity_listar_listas, content_listar_listas, app_bar_listar_listas**. Vistas asociadas a la actividad **ListarListas**.
 - **activity_listar_medios, content_listar_medios, app_bar_listar_medios**. Vistas asociadas a la actividad **ListarMedios**.
 - **activity_listar_videos, content_listar_videos, app_bar_listar_videos**. Vistas asociadas a la actividad **ListarVideos**.
 - **activity_pantalla_principal**. Vista asociada a la actividad **PantallaPrincipal**.
 - **activity_reproductor_web**. Vista asociada a la actividad **ReproductorWeb**.
 - **dialog_lista_nueva**. Cuadro de diálogo utilizado en la actividad **CrearListasMedios** para introducir el nombre y el autor de la nueva lista.
 - **dialog_medio_nuevo**. Cuadro de diálogo utilizado en la actividad **CrearListasMedios** para ir agregando elementos a la lista.
 - **dialog_medio_url**. Cuadro de diálogo utilizado para introducir la URL desde la cual cargaremos un elemento para ser reproducido, utilizado en la actividad **AccederMedioURL**.
 - **dialog_parental**. Cuadro de diálogo utilizado por la actividad **ControlParental** para habilitar y deshabilitar el control parental.
 - **elemento_entrada_sistema_archivo**. Vista utilizada por el adaptador **AdaptadorEntradaSistema**.
 - **elemento_lista**. Vista utilizada por el adaptador **AdaptadorListas**.
 - **elemento_medio**. Vista utilizada por el adaptador **AdaptadorMedios**.
 - **nav_header_main**. Vista utilizada para la cabecera del **Drawer Layout**.
- **Menu**
 - **drawer_navegacion**. Menú principal de la aplicación accesible desde el **Drawer Layout**.
 - **menu_crear_listas_medios**. Menú de la actividad **CrearListasMedios**.
 - **menu_fab_listar_lista**. Menú asociado al botón **fab** de la actividad **ListarLista**.
 - **menu_item_listar_lista**. Menú asociado a los elementos creados por el adaptador **AdaptadorListas**.
 - **menu_listar_audio**. Menú asociado a la actividad **ListarAudios**.
 - **menu_listar_listas**. Menú asociado a la actividad **ListarListas**.

- **menu_listar_medio**. Menú asociado a la actividad **ListarMedios**.
 - **menu_listar_video**. Menú asociado a la actividad **ListarVideos**.
- **Mimap**
 - **ic_launcher.png**. Icono asociado a la aplicación.
- **Values**
 - **colors**. Definición de los colores principales de la aplicación.
 - **dimens**. Definición de algunas medidas utilizadas en la definición de los layouts.
 - **strings**. Cadenas de texto utilizadas en la aplicación. Existe una versión para castellano y otra para inglés.
 - **styles**. Definición de los estilos utilizados en la aplicación.
- **Xml**
 - **preferences**. Menú asociado a la configuración de la aplicación.

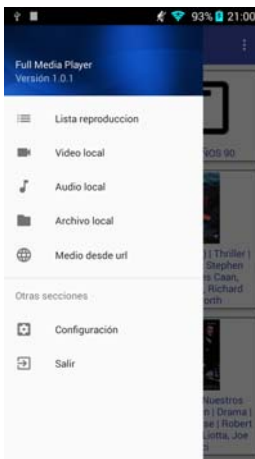
Capturas de pantalla



Primera pantalla de la aplicación que se visualizará mientras se recuperan las listas de reproducción desde los ficheros almacenados en el dispositivo.



Pantalla de reproducción de videos y audio, accesible desde varias secciones de la aplicación.



Menú principal de la aplicación que nos permite de una forma rápida cambiar entre las secciones principales de las mismas. Estas secciones son:

- Listas de reproducción. Permite trabajar con listas de reproducción.
- Video local. Recupera los videos almacenados en el dispositivo.
- Audio local. Recupera los audios almacenados en el dispositivo.

SECCIÓN LISTA DE REPRODUCCION



Pantalla principal de la sección de la lista de reproducción. En la parte central de la pantalla disponemos de las listas que actualmente están disponibles en nuestro dispositivo.

Si sobre una lista realizamos un toque corto accederemos a listar los elementos de la lista.



Menú disponible si realizamos un toque largo sobre una de las listas. Las opciones que disponemos son:

- Eliminar lista. Esta opción no pedirá una confirmación antes de eliminar la lista.
- Compartir link. Permite enviar el link asociado a la lista mediante cualquier aplicación que responda a **Intent.ACTION_SEND**.
- Compartir lista. Igual que en el caso anterior pero se enviará el fichero de la lista completo.
- Sincronizar lista. Permite recargar desde la fuente original de Internet la lista.



Esta pantalla se visualizará cuando tengamos activado el control parental y presionemos sobre una lista para visualizar su contenido y la misma se encuentra protegida.

Solo en caso de introducir el PIN correcto podremos acceder a consultar el contenido de la lista.

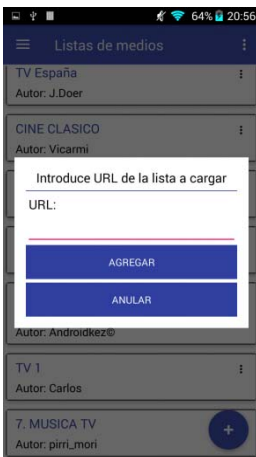


El menú asociado a esta actividad dispone de los siguientes elementos:

- Crear lista. Permite al usuario crear una nueva lista personalizada a partir de url's.
- Eliminar listas. Permite borrar todas las listas en una sola operación.
- Recargar listas. Permite volver a cargar las listas disponibles en nuestro dispositivo. Puede ser una buena opción si pasamos la aplicación a segundo plano y añadimos directamente sobre la carpeta FullMediaPlayer un nuevo fichero de tipo lista de reproducción.
- Sincronizar listas. Busca nuevas versiones de los ficheros en Internet.



El botón **fab** dispone de un menú asociado que nos permitirá añadir nuevas listas. Para ello disponemos de dos opciones. En las siguientes ilustraciones mostramos las pantallas que visualizaremos al acceder a las distintas opciones.



Pantalla asociada a la opción **añadir lista medio desde URL**.



Pantalla asociada a la opción **añadir lista medio desde QR**.



Pantalla que visualizaremos cuando seleccionemos la opción de **Crear lista**.



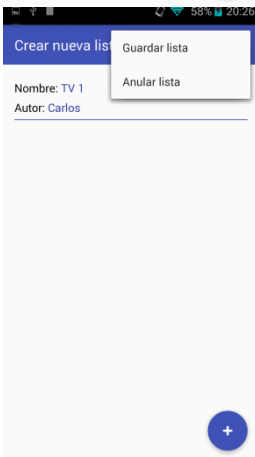
Una vez introducida en la pantalla anterior el nombre de la lista y el autor pasaremos a esta pantalla donde podremos ir agregando medios si presionemos sobre el botón **fab**.



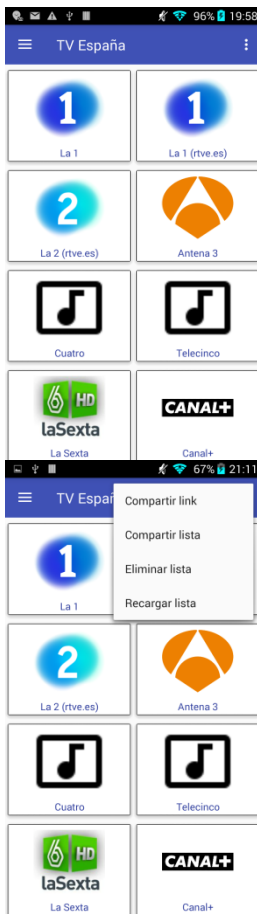
Cada vez que presionamos sobre el botón **fab** visualizaremos la siguiente pantalla que nos permitirá añadir nuevos elementos a la lista.



Aspecto que visualizaremos según añadamos nuevos medios.



Menú que permitirá salir de esta actividad, bien guardando la nueva lista o no.

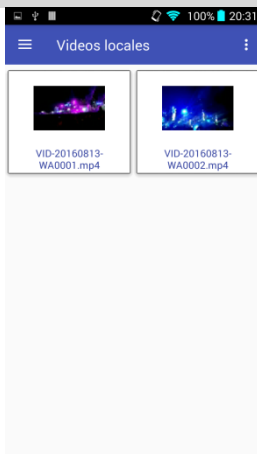


Pantalla que visualiza los medios asociados a una lista de reproducción.

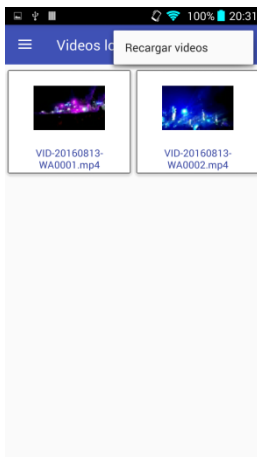
Si presionamos sobre cualquiera de los medios, si este está disponible, se iniciará la reproducción del mismo.

Menú asociado a la actividad que visualiza una lista de reproducción, y que permite realizar una serie de operaciones que ya hemos comentado con anterioridad.

SECCIÓN LISTA DE VIDEOS LOCALES

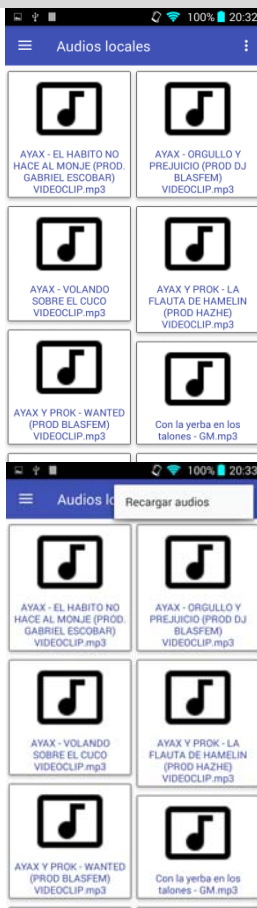


Esta es la pantalla que se visualizará cuando accedemos a la sección de videos locales, al igual que en las listas de reproducción cuando presionamos sobre un medio éste será reproducido.



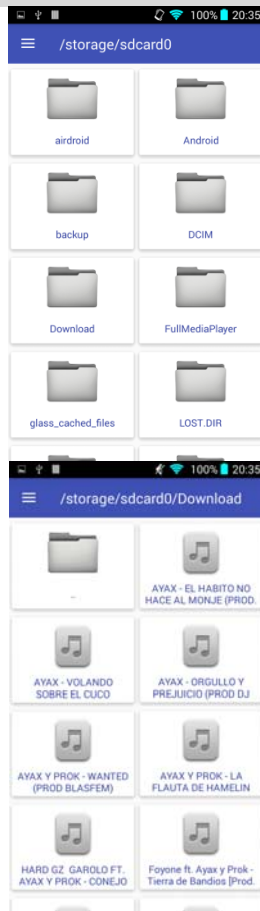
El menú asociado a esta aplicación permite recargar la lista de videos locales.

SECCIÓN LISTA DE AUDIOS LOCALES



Esta sección se comporta del mismo modo que la sección vista anteriormente. La única diferencia es que trabajamos con ficheros de audio.

SECCIÓN LISTA DE ARCHIVOS LOCALES



Pantalla inicial que se visualizará cuando accedemos a la sección de lista de archivos locales. Al presionar sobre una carpeta podremos acceder a su interior y visualizar su contenido.

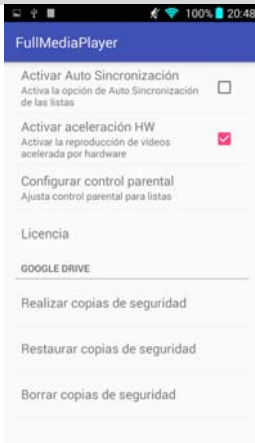
Una vez ubicado sobre una carpeta que no resida en la raíz, veremos que disponemos de una carpeta para poder subir un nivel. Para hacer un sistema más fácil de utilizar, únicamente se visualizarán los ficheros de tipo audio y video compatible con el sistema Android.

SECCIÓN MEDIO DESDE URL



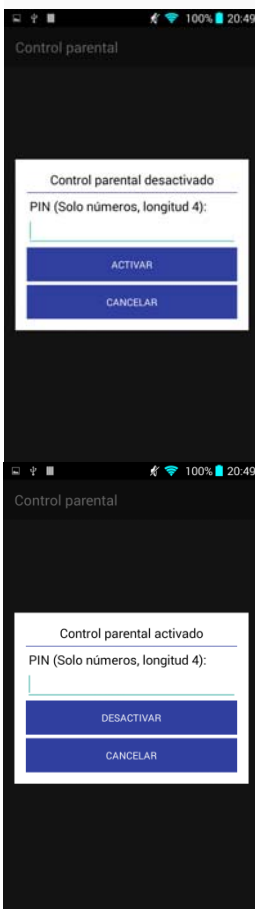
Pantalla que nos permitirá introducir la URL del medio que deseamos reproducir desde Internet.

SECCIÓN CONFIGURACIÓN



Pantalla principal asociada a la configuración desde la cual podemos:

- Activar la auto sincronización con el objetivo de ir comprobando de forma constante si existen actualizaciones de los ficheros en Internet.
- Activar o desactivar la reproducción hardware



Pantalla que visualizaremos cuando presionemos sobre la opción **Configurar control parental** y este se encuentre desactivado.

Pantalla que visualizaremos cuando presionemos sobre la opción **Configurar control parental** y este se encuentre activado.



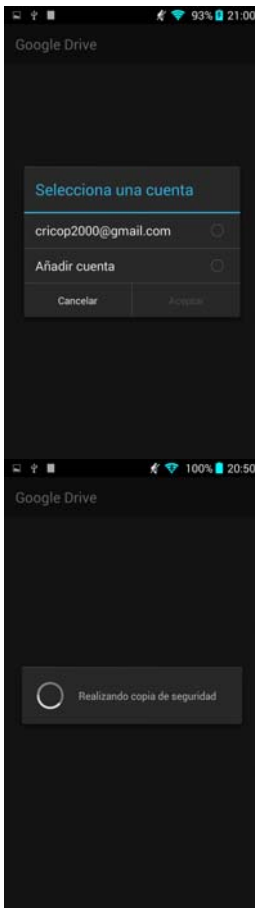
Vista para informar sobre la licencia de software.

Este software ha sido creado con un objetivo educativo, por ello los archivos fuentes pueden ser abiertos para su libre estudio, permitiéndose modificaciones y mejoras del mismo.

Cualquier modificación o mejora deberá ser publicada para su libre estudio y mejora. En dicha publicación deberá mantenerse el nombre del autor del código original y la declaración de derechos de autor en el código.

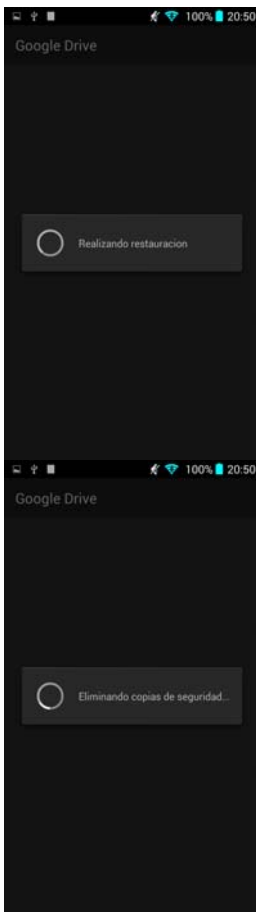
No está permitido el licenciar bajo un uso comercial el código o parte de él mismo.

Autor: Carlos Rico Portalés



Pantalla que visualizaremos cuando accedemos por primera vez a cualquier operación relacionada con Google Drive donde deberemos o bien seleccionar una cuenta de Gmail o dar de alta una nueva.

Pantalla que visualizaremos mientras se realizan las copias de seguridad de las listas de reproducción.



Pantalla que visualizaremos mientras se restauran las copias de seguridad de las listas de reproducción.

Pantalla que visualizaremos mientras se borran las copias de seguridad de las listas de reproducción.

Conclusiones

En esta primera fase se ha creado una aplicación que cumple con los objetivos inicialmente planteados:

- La aplicación permite la reproducción de vídeos y audios tanto en streaming desde Internet como de archivos almacenados en nuestro dispositivo.
- La aplicación nos permite tanto crear sus propias listas de reproducción como descargar listas de reproducción publicadas por Internet.
- La aplicación haciendo uso de Google Drive mantiene copias de seguridad de las listas de reproducción. Esta opción facilita que un mismo usuario que disponga de varios dispositivos comparta entre los mismos sus listas de reproducción.
- Todos los mensajes de la interfaz del usuario han sido parametrizados para poder ser fácilmente traducibles a cualquier idioma.

Aunque bien es cierto que estos objetivos iniciales han sido cubiertos, no es menos cierto que existen una serie de retos que podrían haber sido incorporados como son:

- La capacidad de poder editar las listas de reproducción y sus contenidos.
- Soportar listas de reproducción que contengan grupos de medios.
- Crear listas de reproducción donde se puedan incorporar medios locales.
- Dado su finalidad de reproductor multimedia, también podría ser deseable incorporar no únicamente audio y vídeo sino también imágenes fotográficas.
- Sería muy interesante la posibilidad de extender la aplicación para trabajar con dispositivos externos del tipo ChromeCast y MiraCast.
- Etc.

Por todo ello, puedo concluir que el proyecto en general ha sido una buena experiencia a la hora de poner en práctica los conocimientos adquiridos durante el Master. Por otro lado, los retos planteados serán incorporados en las futuras versiones de esta app.