

Reproductor de vídeo

VideoR

Código fuente

El código fuente se encuentra en la dirección github del proyecto:

<https://github.com/aaronperez/VideoR>

Funcionamiento de la aplicación

VideoR es una aplicación pasiva. Simplemente se al intentar abrir un archivo de video, se ofrece para reproducirlo.

Menús

La aplicación no dispone de menú.

Implementación

Manifiesto

En *AndroidManifest* usamos el permiso de lectura en el almacenamiento externo y el de internet. Además modificamos el intent-filter quitando el launcher añadiendo el action para que al intentar el dispositivo abrir un archivo de video ofrezca nuestra aplicación.

```
<intent-filter>
  <action android:name="android.intent.action.VIEW" />
  <data android:mimeType="video/*"/>
  <category android:name="android.intent.category.DEFAULT" />
</intent-filter>
```

Recursos

Drawable

Aquí hemos añadido 1 imagen que servirá de icono a la app.

Layout

El layout de la actividad principal contiene un *VideoView* a pantalla completa (*activity_main.xml*).

Código fuente

Usamos *MediaController* para manejar el video además de las variables *control* (int) para guardar el avance del video y *data* (uri) para saber que archivo estamos reproduciendo.

Clase Main

El método *onCreate ()* visualiza en el Layout principal e inicia *VideoView* y llama al metodo *play()* pasandole *data* y *control*.

El método *play (Uri data, int control)* obtiene el path de la *Uri* y se la da al *VideoView*, inicializa el *MediaController* y se lo pasa al *VideoView*. Por último llevamos el video al tiempo que marque la variable *control* (por defecto 0) e iniciamos el video.

```
public void play (Uri data, int control){
    vv.setVideoPath(data.getPath());
    mc = new MediaController(this);
    mc.setAnchorView(vv);
    vv.setMediaController(mc);
    vv.seekTo(control);
    vv.start();
}
```

El método *onSaveInstanceState ()* guarda el valor de *data* y obtiene el avance del video (*control=vv.getCurrentPosition()*) y lo guarda en *control*. Posteriormente *onRestoreInstanceState* recuperamos los valores de *data* y de *control* y volvemos a llamar a *play()*.

```
@Override
protected void onRestoreInstanceState(Bundle savedInstanceState) {
    super.onRestoreInstanceState(savedInstanceState);
    control=savedInstanceState.getInt("control");
    data.parse(savedInstanceState.getString("data"));
    play(data,control);
}
```