


Open · Opened 3 days ago by  **Rubén Montero**

¡Mostrando vídeos!

Resumen

- Añadiremos un `VideoView` a nuestra actividad `VideoActivity.java`
- Con la URL de vídeo que recibimos en la tarea anterior, reproduciremos un vídeo

Descripción

[VideoView](#) es una clase extremadamente útil en Android que permite reproducir un vídeo obtenido de una URL o de un *recurso* propio de la *app*.

La tarea

Añade un `<VideoView>` a `activity_video.xml`:

```
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <VideoView
        android:id="@+id/video_view"
        android:layout_width="match_parent"
        android:layout_height="match_parent"/>

</androidx.constraintlayout.widget.ConstraintLayout>
```

(Como ves, se usa `match_parent`. Esto deformará el vídeo para ajustarlo al contenedor. Nos vale, ya que esperamos que el vídeo tenga una relación de aspecto parecida a la pantalla. También es útil jugar con `wrap_content` y `constraints` según qué escenario).

Luego, en `VideoActivity.java`, **añade** un atributo tipo `VideoView` e **inicialízalo** con `findViewById` en `onCreate`, como ya estamos acostumbrados:

```
public class VideoActivity extends AppCompatActivity {
    public static final String INTENT_CLIP_ID = "CLIP_ID";
    public static final String INTENT_CLIP_URL = "CLIP_URL";

    private VideoView videoView; // Nuevo atributo

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_video);
        Intent intent = getIntent();
        int clipId = intent.getIntExtra(VideoActivity.INTENT_CLIP_ID, -1);
        String clipUrl = intent.getStringExtra(VideoActivity.INTENT_CLIP_URL);
        Toast.makeText(this, "videoID: " + clipId, Toast.LENGTH_LONG).show();

        // VideoView
        videoView = findViewById(R.id.video_view);
    }
}
```

A continuación, ¡basta con invocar `setVideoUri` para reproducir un vídeo de Internet asociado a una URL! **Hazlo** así:

```
// VideoView
videoView = findViewById(R.id.video_view);
videoView.setVideoURI(Uri.parse(clipUrl));
videoView.start();
```

Puedes lanzar la *app* y verificar que cualquier vídeo se ve correctamente. Quizás la pantalla del emulador se quede en negro bastante rato (incluso 4-5 minutos) si el ancho de banda de la red es muy reducido para descargar el vídeo. Sé paciente. ¡Seguramente verás tu vídeo!

Si giras la pantalla...

Es posible que el vídeo se vea demasiado *deformado* al girar la pantalla.

Vamos a optar por *deshabilitar* el cambio de orientación. Forzaremos a que la pantalla siempre se muestre *apaisada* (vista **landscape**).

En el *manifiesto*, **añade** `screenOrientation` a la etiqueta de nuestra *activity*. **Así**:

```
<activity
    android:name=".VideoActivity"
    android:screenOrientation="landscape" />
```

Esto *obligará* a los usuarios a girar la pantalla cuando se inicie la actividad. ¡Pruébalo!

Por último

Sube tus cambios al repositorio en un nuevo *commit*.



Rubén Montero @ruben.montero changed milestone to [%Sprint 2](#) 3 days ago