



# **Guía de uso “Librería Picasso”**

Walter Martín Lopes



## ¿Qué es la librería Picasso?

La librería Picasso en Android es una herramienta para la carga y manipulación de imágenes en aplicaciones.

Desarrollada por Square, es muy valorada por su simplicidad y eficiencia. Permite cargar imágenes desde la web o recursos locales, y ofrece funcionalidades como el ajuste de tamaño, la rotación y el almacenamiento en caché.

Es especialmente útil en aplicaciones donde se manejan muchas imágenes, ayudando a reducir el uso de memoria y mejorar el rendimiento.



## Guía de uso

Explicaremos su funcionamiento usando de ejemplo la práctica entregable del tema 9, mostrando paso a paso cómo implementar esta librería.

En primer lugar debemos agregar la dependencia de Picasso en nuestro archivo *'build.gradle'(Module:app)* en la sección de dependencias:

```
//Librería para Retrofit 2
implementation("com.squareup.retrofit2:retrofit:2.9.0")
//Librería GSON para el tratamiento y conversión de datos JSON
implementation("com.squareup.retrofit2:converter-gson:2.9.0")
//Librería OkHttp para simplificar la construcción de peticiones HTTP
implementation("com.squareup.okhttp3:okhttp:4.12.0")
//Librería para utilizar corrutinas en Kotlin (peticiones HTTP en segundo plano)
implementation("org.jetbrains.kotlinx:kotlinx-coroutines-android:1.7.3")
//Librería Picasso
implementation("com.squareup.picasso:picasso:2.71828")
//=====
```



## Guía de uso

Debemos asegurarnos de tener el permiso de Internet en nuestro *'AndroidManifest.xml'* si vamos a cargar imágenes desde una URL como es nuestro caso:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">
    <!-- Damos permisos para Internet -->
    <uses-permission android:name="android.permission.INTERNET" />
    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
```



## Guía de uso

Seguidamente simplemente tenemos que hacer uso de la librería en nuestro código Kotlin usando la función ***Picasso.get()*** cargaremos la url de la imagen con el ***.load()*** y asignaremos el **ImageView** con el ***.into()***, opcionalmente podremos asignar imágenes por defecto como placeholder y en caso de que hubiera un error al cargar la imagen.

```
private fun mostrarDatos(libro: Libro?) {  
    GlobalScope.launch(Dispatchers.Main) { this: CoroutineScope  
        // Hago visible los componentes del layout  
        mostrarComponentesLibroConsultado()  
        //Setteo  
        titulo.setText(libro?.titulo)  
        autor.setText(libro?.autor)  
        genero.setText(libro?.genero)  
        publicacion.setText(libro?.publicacion)  
        precio.setText(libro?.precio + "€")  
  
        //Muestro imagen con la libreria Picasso  
        Picasso.get()  
            .load(libro?.imagen)//Cargamos la url de la imagen del campo imagen de nuestro objeto Libro  
            .placeholder(R.drawable.libro) // Imagen de placeholder  
            .error(R.drawable.libro) // Imagen de error  
            .into(imagen) //Asignamos el ImageView  
    }  
}
```

## Guía de uso

Como resultado final, observamos que la imagen se carga correctamente desde una URL absoluta de internet.

