

# Estructura del proyecto

En los siguientes apartados describiremos los elementos principales de esta estructura.

Lo primero que debemos distinguir son los conceptos de **proyecto** y **módulo**. La entidad **proyecto** es única, y engloba a todos los demás elementos. Dentro de un proyecto podemos incluir varios **módulos**, que pueden representar aplicaciones distintas, versiones diferentes de una misma aplicación, o distintos componentes de un sistema (aplicación móvil, aplicación servidor, librerías, etc.). En la mayoría de los casos, trabajaremos con un proyecto que contendrá un sólo módulo correspondiente a nuestra aplicación principal. Por ejemplo en este caso que estamos creando tenemos el proyecto `HelloWorld` que contiene al módulo `app` que contendrá todo el software de la aplicación de ejemplo.

A continuación describiremos los contenidos principales de nuestro módulo principal.

Carpeta	Descripción
<code>/res/drawable/</code>	<p>Contiene las imágenes y otros elementos gráficos usados por la aplicación. Para poder definir diferentes recursos dependiendo de la resolución y densidad de la pantalla del dispositivo se suele dividir en varias subcarpetas:</p> <ul style="list-style-type: none"><li><code>/drawable</code> (recursos independientes de la densidad de pantalla)</li><li><code>/drawable-ldpi</code> (densidad baja)</li><li><code>/drawable-mdpi</code> (densidad media)</li><li><code>/drawable-hdpi</code> (densidad alta)</li><li><code>/drawable-xhdpi</code> (densidad muy alta)</li><li><code>/drawable-xxhdpi</code> (densidad muy muy alta)</li></ul>
<code>/res/mipmap/</code>	<p>Contiene los iconos de lanzamiento de la aplicación (el icono que aparecerá en el menú de aplicaciones del dispositivo) para las distintas densidades de pantalla existentes. Al igual que en el caso de las carpetas <code>/drawable</code>, se dividirá en varias subcarpetas dependiendo de la densidad de pantalla:</p> <ul style="list-style-type: none"><li><code>/mipmap-mdpi</code></li><li><code>/mipmap-hdpi</code></li><li><code>/mipmap-xhdpi</code></li></ul>

<code>/mipmap-xxhdpi</code>	
<code>/res/layout/</code>	<p>Contiene los ficheros de definición XML de las diferentes pantallas de la interfaz gráfica. Para definir distintos <b>layouts</b> dependiendo de la orientación del dispositivo se puede dividir también en subcarpetas:</p> <p><code>/layout</code> (vertical)</p> <p><code>/layout-land</code> (horizontal)</p>
<code>/res/menu/</code>	Contiene la definición XML de los menús de la aplicación.
<code>/res/values/</code>	<p>Contiene otros ficheros XML de recursos de la aplicación, como por ejemplo cadenas de texto (<code>strings.xml</code>), estilos (<code>styles.xml</code>), colores (<code>colors.xml</code>), arrays de valores (<code>arrays.xml</code>), tamaños (<code>dimens.xml</code>), etc.</p>

## Carpeta `/app/src/main/java`

Esta carpeta contendrá todo el código fuente de la aplicación, clases auxiliares, etc. Inicialmente, Android Studio creará por nosotros el código básico de la pantalla (actividad o activity) principal de la aplicación, que recordemos que en nuestro caso era **MainActivity**, y siempre bajo la estructura del paquete java definido durante la creación del proyecto.

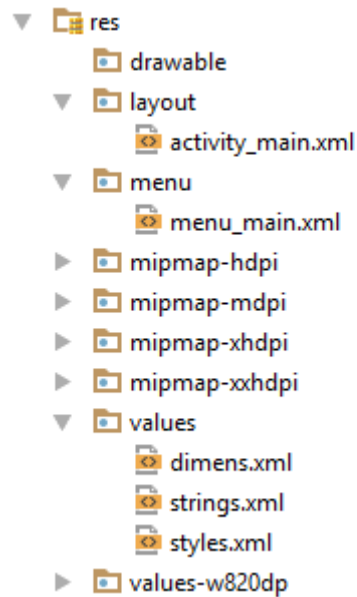


## Carpeta `/app/src/main/res/`

Contiene todos los ficheros de recursos necesarios para el proyecto: imágenes, layouts, cadenas de texto, etc. Los diferentes tipos de recursos se pueden distribuir entre las siguientes subcarpetas.

No todas estas carpetas tienen por qué aparecer en cada proyecto Android, tan sólo las que se necesiten.

Como ejemplo, para un proyecto nuevo Android como el que hemos creado, tendremos por defecto los siguientes recursos para la aplicación:



Como se puede observar, existen algunas carpetas en cuyo nombre se incluye un sufijo adicional, como por ejemplo **values-w820dp**. Estos, y otros sufijos, se emplean para definir recursos independientes para determinados dispositivos según sus características. De esta forma, por ejemplo, los recursos incluidos en la carpeta **values-w820dp** se aplicarían sólo a pantallas con más de 820dp de ancho, o los incluidos en una carpeta llamada **values-v11** se aplicarían tan sólo a dispositivos cuya versión de Android sea la 3.0 (API 11) o superior. Al igual que estos sufijos **-w** y **-v** existen otros muchos para referirse a otras características del terminal.

Entre los recursos creados por defecto cabe destacar los **layouts**, en nuestro caso sólo tendremos por ahora el llamado **activity\_main.xml**.

## Fichero /app/src/main/AndroidManifest.xml

Contiene la definición en XML de muchos de los aspectos principales de la aplicación, como por ejemplo su identificación (nombre, icono, etc.), sus componentes (pantallas, servicios, etc.), o los permisos necesarios para su ejecución.

## Fichero `/app/build.gradle`

Contiene información necesaria para la compilación del proyecto, por ejemplo la versión del SDK de Android utilizada para compilar, la mínima versión de Android que soportará la aplicación, referencias a las librerías externas utilizadas, etc.

En un proyecto pueden existir varios ficheros `build.gradle`, para definir determinados parámetros a distintos niveles. Por ejemplo, en nuestro proyecto podemos ver que existe un fichero `build.gradle` a nivel de proyecto, y otro a nivel de módulo dentro de la carpeta `app/`. El primero de ellos definirá parámetros globales a todos los módulos del proyecto, y el segundo sólo tendrá efecto para cada módulo en particular.