

# Ejercicio 2: Estructura de un proyecto en Android

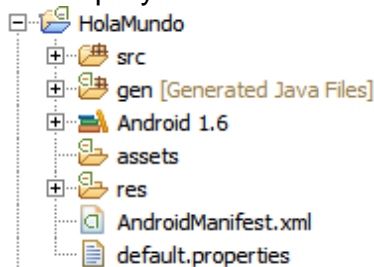
**En este segundo ejercicio vamos a aprender como está estructurado Android y como se debe programar.**

## 2.1 Estructura de un proyecto Android

Para empezar a comprender cómo se construye una aplicación Android vamos a echar un vistazo a la estructura general de un proyecto tipo.

Cuando creamos un nuevo proyecto Android en Eclipse se genera automáticamente la estructura de carpetas necesaria para poder generar posteriormente la aplicación. Esta estructura será común a cualquier aplicación, independientemente de su tamaño y complejidad.

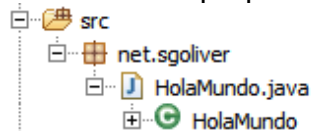
En la siguiente imagen vemos los elementos creados inicialmente para un nuevo proyecto Android:



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

### **2.1.1 Carpeta /src/**

Contiene todo el código fuente de la aplicación, código de la interfaz gráfica, clases auxiliares, etc. Inicialmente, Eclipse creará por nosotros el código básico de la pantalla (Activity) principal de la aplicación, siempre bajo la estructura del paquete java definido.



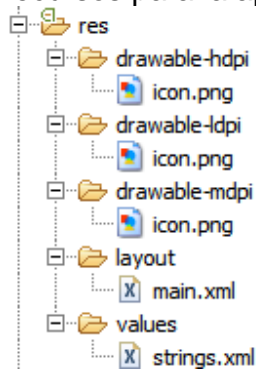
### **2.1.2 Carpeta /res/**

Contiene todos los ficheros de recursos necesarios para el proyecto: imágenes, vídeos, cadenas de texto, etc. Los diferentes tipos de recursos de deberán distribuir entre las siguientes carpetas:

Carpeta	Descripción
/res/drawable/	Contienen las imágenes de la aplicación. Para utilizar diferentes recursos dependiendo de la resolución del dispositivo se suele dividir en varias subcarpetas: <ul style="list-style-type: none"><li>➤ /drawable-ldpi</li></ul>

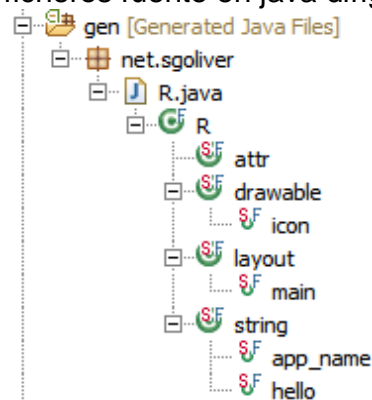
	<ul style="list-style-type: none"> <li>➤ /drawable-mdpi</li> <li>➤ /drawable-hdpi</li> </ul>
/res/layout/	<p>Contienen los ficheros de definición de las diferentes pantallas de la interfaz gráfica. Para definir distintos layouts dependiendo de la orientación del dispositivo se puede dividir en dos subcarpetas:</p> <ul style="list-style-type: none"> <li>➤ /layout</li> <li>➤ /layout-land</li> </ul>
/res/anim/	Contiene la definición de las animaciones utilizadas por la aplicación.
/res/menu/	Contiene la definición de los menús de la aplicación.
/res/values/	Contiene otros recursos de la aplicación como por ejemplo cadenas de texto ( <i>strings.xml</i> ), estilos ( <i>styles.xml</i> ), colores ( <i>colors.xml</i> ), etc.
/res/xml/	Contiene los ficheros XML utilizados por la aplicación.
/res/raw/	Contiene recursos adicionales, normalmente en formato distinto a XML, que no se incluyan en el resto de carpetas de recursos.

Como ejemplo, para un proyecto nuevo Android, se crean los siguientes recursos para la aplicación:



### 2.1.3 Carpeta /gen/

Contiene una serie de elementos de código generados automáticamente al compilar el proyecto. Cada vez que generamos nuestro proyecto, la maquinaria de compilación de Android genera por nosotros una serie de ficheros fuente en java dirigidos al control de los recursos de la aplicación.



El más importante es el que se puede observar en la imagen, el fichero [R.java](#), y la clase [R](#).

Esta clase **R** contendrá en todo momento una serie de constantes con los ID de todos los recursos de la aplicación incluidos en la carpeta **/res/**, de forma que podamos acceder fácilmente a estos recursos desde nuestro código a través de este dato. Así, por ejemplo, la constante **R.drawable.icon** contendrá el ID de la imagen **"icon.png"** contenida en la carpeta **/res/drawable/**. Veamos como ejemplo la clase R creada por defecto para un proyecto nuevo:

```
package net.sgoliver;
public final class R {
    public static final class attr {
    }
    public static final class drawable {
        public static final int icon=0x7f020000;
    }
    public static final class layout {
        public static final int main=0x7f030000;
    }
    public static final class string {
        public static final int app_name=0x7f040001;
        public static final int hello=0x7f040000;
    }
}
```

#### 2.1.4 Carpeta **/assets/**

Contiene todos los demás ficheros auxiliares necesarios para la aplicación (y que se incluirán en su propio paquete), como por ejemplo ficheros de configuración, de datos, etc.

La diferencia entre los recursos incluidos en la carpeta **/res/raw/** y los incluidos en la carpeta **/assets/** es que para los primeros se generará un ID en la clase **R** y se deberá acceder a ellos con los diferentes métodos de acceso a recursos. Para los segundos sin embargo no se generarán ID y se podrá acceder a ellos por su ruta como a cualquier otro fichero del sistema. Usaremos uno u otro según las necesidades de nuestra aplicación.

#### 2.1.5 Fichero **AndroidManifest.xml**

Contiene la definición en XML de los aspectos principales de la aplicación, como por ejemplo su identificación (nombre, versión, icono, ...), sus componentes (pantallas, mensajes, ...), o los permisos necesarios para su ejecución. Veremos más adelante más detalles de este fichero.

En el siguiente apartado veremos los componentes software principales con los que podemos construir una aplicación Android.

**Guarda el documento en tu procesador de textos (Microsoft Word, Writer, etc) en tu unidad con el nombre **"Ejercicio2android"**, respondiendo a las siguientes preguntas dentro del documento:**

1. ¿Qué contiene la carpeta **scr**?

2. ¿Qué contiene la carpeta **res**?
3. ¿Qué contiene la carpeta **gen**?
4. ¿Qué contiene la carpeta **assets**?
5. ¿Qué contiene el fichero **AndroidManifest.xml**?

***Guarda el documento en tu carpeta de alumno con el nombre "Ejercicio2android".***