

Tema 2 - Usando un proyecto plantilla

Descargar estos apuntes [pdf](#) o [html](#)

Índice

- [Introducción](#)
- [Renombrando un proyecto base](#)

Introducción

Renombrando un proyecto base

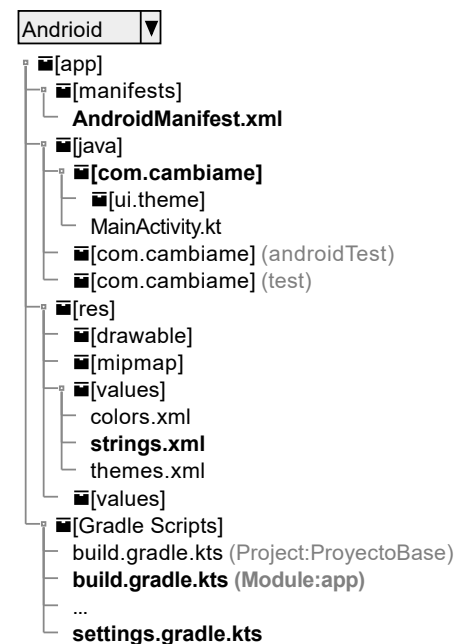
Una vez hemos descargado el proyecto base y lo hemos descomprimido.

Los pasos a realizar para renombrar correctamente el proyecto son los siguientes:

1. Copiamos el proyecto base en el lugar que deseemos.
2. Cambiamos el **nombre a la carpeta** o 'workspace' del proyecto base, si lo consideramos necesario. Por ejemplo, `MiPrimerPrograma`.
3. Abrimos Android Studio y seleccionamos la carpeta que contiene nuestro proyecto base.

4. Cuando Gradle finalice todo seleccionaremos la vista **Android**. En esta más que carpeta lo que contendrá una serie de unidades organizativas jerárquicas de los elementos del proyecto similar al esquema de la derecha. renombraremos el paquete de la aplicación, para ello haremos **click derecho** sobre el paquete `com.cambiamo` y seleccionaremos **Refactor -> Rename** o **Ctrl + R, R** si usamos el Keymap de Visual Studio.

- Seleccionaremos **All Directories**
- Cambiaremos `cambiamo` por el nombre de nuestra app en minúsculas, sin espacios y sin caracteres especiales. Por ejemplo, `miprimerprograma` y pulsaremos **Refactor**. Si nos ofrece un preview de los cambios pulsaremos **Do Refactor**.



5. Aunque los paquetes se hayan renombrado, todavía quedaría por realizar algún paso más. Si abrimos el fichero `app → manifest → AndroidManifest.xml` veremos que se produce un error al no encontrar la actividad de entrada, además de que todavía aparece el nombre `ProyectoBase` en la etiqueta y en el `Theme`. Para resolver estos problemas tendremos que hacer lo siguiente:

```
<application ...>
    ...
    <activity
4         // Añadiremos la ruta con el nuevo nombre del paquete
5         // para encontrar la clase MainActivity
        android:name="com.holamundo.MainActivity"

        android:exported="true"
        android:label="@string/app_name"

11         // Renombraremos el ProyectoBase a HolaMundo con
12         // la refactorización del menú contextual o Ctrl + R, R
        android:theme="@style/Theme.HolaMundo">
        ...
    </activity>
```

6. Cambiaremos el nombre de la APP en `app → res → values → strings.xml`. Por ejemplo, por `HolaMundo`.

```
<resources>
2     <string name="app_name">HolaMundo</string>
</resources>
```

7. Abriremos al archivo de configuración de Gradle `app → Gradle Scripts → settings.gradle.kts` y cambiaremos el nombre del proyecto `ProyectoBase` por el nuevo nombre que le queramos dar. Por ejemplo, `HolaMundo`.

```
...
2 rootProject.name = "HolaMundo"
include(":app")
```

🔴 **Nota:** Este será el nombre del proyecto que aparecerá al abrir el proyecto con Android Studio independientemente del nombre que le hayamos puesto a la carpeta del *'Workspace'*.

8. Abriremos el archivo gradle del módulo de la aplicación que se encuentra en

app → Gradle Scripts → build.gradle.kts (Module:app)

Cambiaremos el nombre del **paquete** de nuestro programa principal **com.cambiame** por el que hayamos puesto en el paso 4. En nuestro ejemplo **com.holamundo**. Fíjate en el siguiente script de ejemplo.

```
plugins {  
    id ("com.android.application")  
    id ("org.jetbrains.kotlin.android")  
}  
  
android {  
    // Cambiamos com.cambiame por com.holamundo  
    namespace = "com.holamundo"  
    compileSdk = 34  
  
    defaultConfig {  
        // Cambiamos com.cambiame por com.holamundo  
        applicationId = "com.holamundo"  
        minSdk = 28  
        targetSdk = 34  
        versionCode = 1  
        versionName = "1.0"  
  
        ...  
    }  
}
```

9. Por último, sincronizaremos el proyecto con los cambios en los scripts de Gradle con la opción de menú **File → Sync Project with Gradle Files** o **Ctrl + Mayús + 0**. También podemos pulsar el botón de sincronización de Gradle que aparece en la barra de herramientas arriba a la derecha con el icono del elefante de Gradle.

Volveremos a abrir el archivo del **AndroidManifest.xml** y veremos que el error ha desaparecido al volver a sincronizar y ya podremos ejecutar la aplicación para ver si todo ha ido bien