

# Desarrollo de aplicaciones Android (161811FP07)

---

Marzo-Abril de 2016. CEP de Granada

---

José Antonio Vacas @javacasm

---

[javacasm@gmail.com](mailto:javacasm@gmail.com)



## Creando la primera aplicación

---

Siguiendo el asistente vamos a ir viendo algunos de los principales componentes de Android

(Ejemplo Hello World)

1. Pulsamos la opción de crear una nueva aplicación



# Welcome to Android Studio

## Recent Projects

### Prueba1

~/AndroidStudioProjects/Prueba1

### Agenda

~/AndroidStudioProjects/Agenda

### Aplicacion Test

~/AndroidStudioProjects/AplicacionTest

### aagdxgame

~/AndroidStudioProjects/aagdxgame

### gradle

~/AndroidStudioProjects/MaterializeYourApp

### android

~/AndroidStudioProjects/aagdxgame/android

### TestGitHub

~/AndroidStudioProjects/Test Git Hub

### Calculadora2.3.3

~/AndroidStudioProjects/Calculadora2.3.3

### App2Pantallas

~/AndroidStudioProjects/App2Pantallas

### Calculadora

~/AndroidStudioProjects/Calculadora

## Quick Start



Start a new Android Studio project



Open an existing Android Studio project



Check out project from Version Control



Import project (Eclipse ADT, Gradle, etc.)



Import an Android code sample



Configure




Docs and How-Tos



1. Seleccionamos la arquitectura a la que irá destinada nuestra aplicación, que pueden ser móviles, tablets, TVs, relojes, ...

Create New Project

 Target Android Devices

**Select the form factors your app will run on**

Different platforms may require separate SDKs

☒ Phone and Tablet

Minimum SDK

Lower API levels target more devices, but have fewer features available.  
By targeting API 23 and later, your app will run on < 1% of the devices that are active on the Google Play Store.  
[Help me choose](#)

☐ Wear

Minimum SDK

☐ TV

Minimum SDK

☐ Android Auto

☐ Glass (Not Available)

Minimum SDK

Cada plataforma tendrá una aplicación distinta, aunque comparten código y recursos


En este momento vamos a seleccionar sólo a opción de hacer crear una aplicación para móviles y tablets.

1. Vamos a dar nombre a la aplicación para lo que tendremos que dar un título que será el que vea el usuario que luego la instale, **Application Name** y un **Package Name** que es el identificador único de la aplicación que la identifica entre los millones de aplicaciones que existen.

Es importante elegir bien este identificar puesto que es el que se ha de mantener a lo largo de toda la vida de la aplicación.

El **Application Name** puede modificarse pero el **Package Name** no.

Create New Project

 **New Project**  
Android Studio

Configure your new project

Application name:

Company Domain:

Package name:  [Edit](#)

Project location:  ...

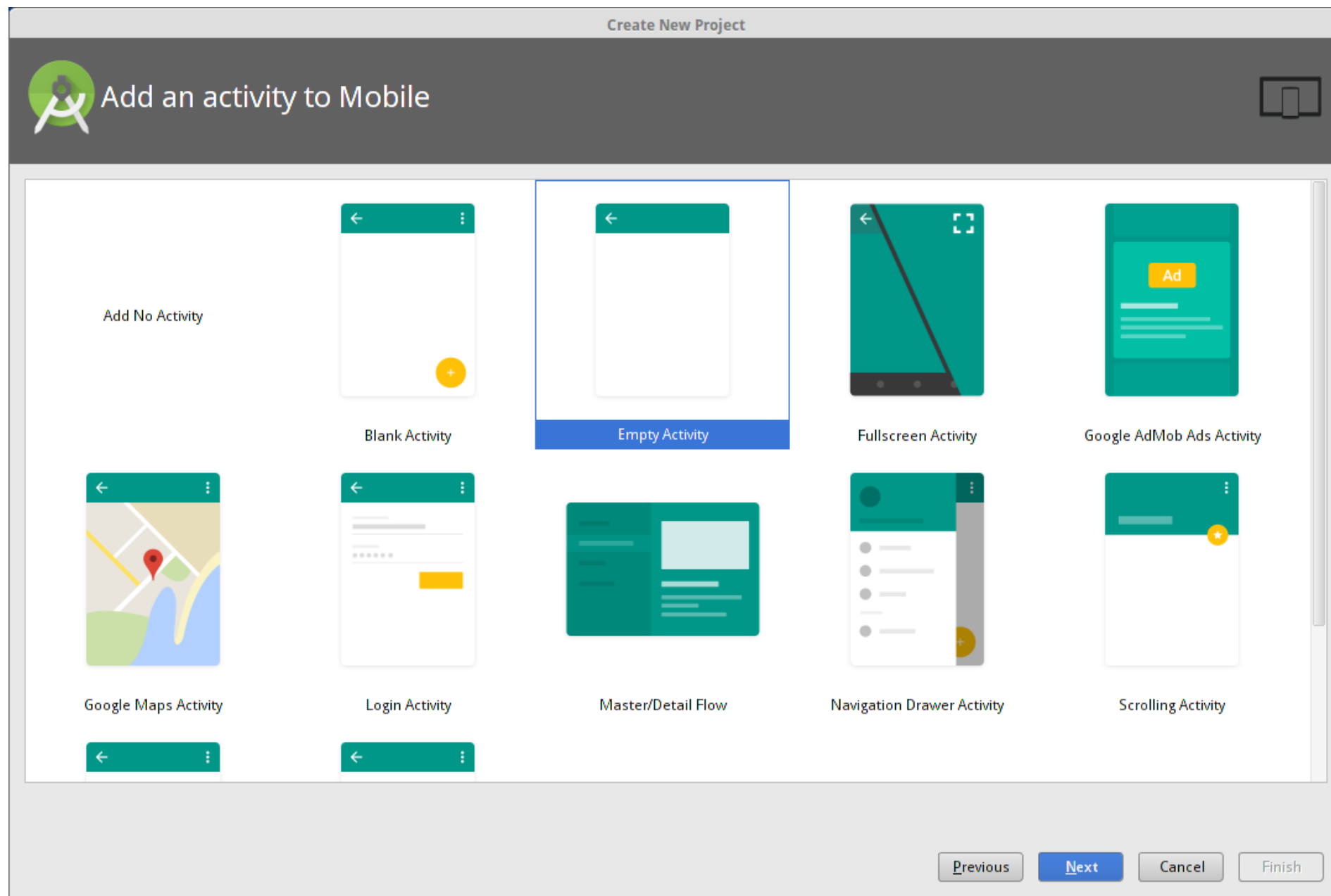
Previous **Next** Cancel Finish

Normalmente el **Package Name** se crea a partir de la url de nuestra web, aunque podemos poner cualquier que se no ocurra.

Podemos pensar que el **Package Name** es una especie de path que permite identificar de manera única a una aplicación dada entre todas las existentes.

1. Ahora vamos a elegir el tipo de pantalla que va a tener nuestra aplicación entre las plantillas que nos enseñan

En Android las pantallas se llaman **Activity**. Podemos pensar en ellas como una pantalla (en el pasado donde las pantallas de los móviles eran pequeñas, pantalla equivalía a funcionalidad, la actividad que podía hacer el usuario).



Vemos que existen distintas plantillas:

- Algunas tienen ya un diseño y están pensadas para hacer algún tipo de tarea concreta, como la de login, o la de acceso a la configuración de la aplicación, por ejemplo.
- Otras ya implementan un tipo de navegación: por medio de un menú lateral o al deslizar el elemento a la derecha o a la izquierda como en la aplicación Galería de fotos

Para empezar vamos a elegir la más simple que la de nombre Empty (la plantilla Blank, que aparece por defecto es sencilla pero tiene algo más de complejidad). Usaremos este tipo sencillo para que la restante funcionalidad no nos produzca más complejidad en nuestro primer proyecto.

Este asistente para crear Activities lo podremos llamar en cualquier momento para poder añadir más pantallas a nuestra aplicación.



Algo que puede llamar nuestra atención es que podemos elegir no crear ningún Activity: efectivamente una aplicación puede funcionar sin tener ninguna pantalla. Se trata de aplicaciones que funcionan en segundo plano, sin la interacción directa del usuario.

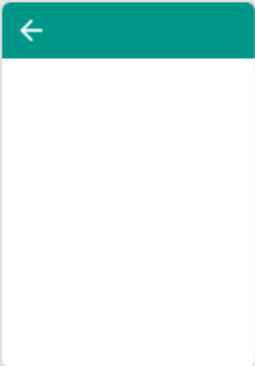
Un ejemplo de este tipo de aplicaciones pueden ser las que nos permiten sincronizar nuestros datos con servicios web o las que nos avisan de que hay cambios como por ejemplo pueden ser las aplicaciones de las redes sociales que nos avisan de la actividad de otros miembros de la red.

1. Una vez elegido el tipo hemos de darle nombre:



Create New Project

Customize the Activity



Empty Activity

Creates a new empty activity

Activity Name:

☒ Generate Layout File

Layout Name:

The name of the activity class to create

Previous

Next

Cancel

Finish

Por defecto se crearán 2 ficheros:

- Un fichero Java, donde escribiremos nuestro código
- Un fichero de recursos de tipo Layout con formato xml. Un layout es la definición del interface de usuario de una pantalla

Estos dos ficheros están conectados en el proyecto. En concreto el fichero java es el que realiza esa conexión. En concreto en la línea

```
setContentView(R.layout.activity_main)
```

1. Con todo esto ya tenemos la aplicación definida, y se generarán todos sus contenidos:

- El código
- Los ficheros XML
- Los ficheros de configuración
- Los recursos



Android Studio



# Welcome to Android Studio

## Recent Projects

Prueba1

~/AndroidStudioProjects/Prueba1

Agenda

~/AndroidStudioProjects/Agenda

Aplicacion Test

~/AndroidStudioProjects/Aplica

aagdxgame

~/AndroidStudioProjects/aagd

gradle

~/AndroidStudioProjects/Materialize Your App

android

~/AndroidStudioProjects/aagdxgame/android

TestGitHub

~/AndroidStudioProjects/Test GitHub

Calculadora2.3.3

~/AndroidStudioProjects/Calculadora2.3.3

App2Pantallas

~/AndroidStudioProjects/App2Pantallas

Calculadora

~/AndroidStudioProjects/Calculadora

## Quick Start



Start a new Android Studio project



Open an existing Android Studio project

Creating project...

Initializing Phone and Tablet



Import an Android code sample



Configure



Docs and How-Tos



Cuando tengamos todo el proyecto se construirá todo el proyecto, compilando el código, empaquetando los recursos y generando e fichero APK de la aplicación.



Android Studio



# Welcome to Android Studio

## Recent Projects

Prueba1

~/AndroidStudioProjects/Prueba1

Agenda

~/AndroidStudioProjects/Agenda

Aplicacion Test

~/AndroidStudioProjects/AplicacionTest

aagdxgame

~/AndroidStudioProjects/aagdxgame

gradle

~/AndroidStudioProjects/gradle

android

~/AndroidStudioProjects/aagdxgame/android

TestGitHub

~/AndroidStudioProjects/Test GitHub

Calculadora2.3.3

~/AndroidStudioProjects/Calculadora2.3.3

App2Pantallas

~/AndroidStudioProjects/App2Pantallas

Calculadora

~/AndroidStudioProjects/Calculadora

## Quick Start



Start a new Android Studio project



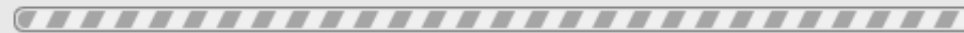
Open an existing Android Studio project



Check out project from Version Control

### Building 'PrimeraAplicacion' Gradle project info

Gradle: Build



Cancel



Configure



Docs and How-Tos



El encargado de hacer todo esto es Gradle, que se dedicará a mantener actualizado nuestro proyecto como un proceso de segundo plano.

## Estructura de una aplicación

---

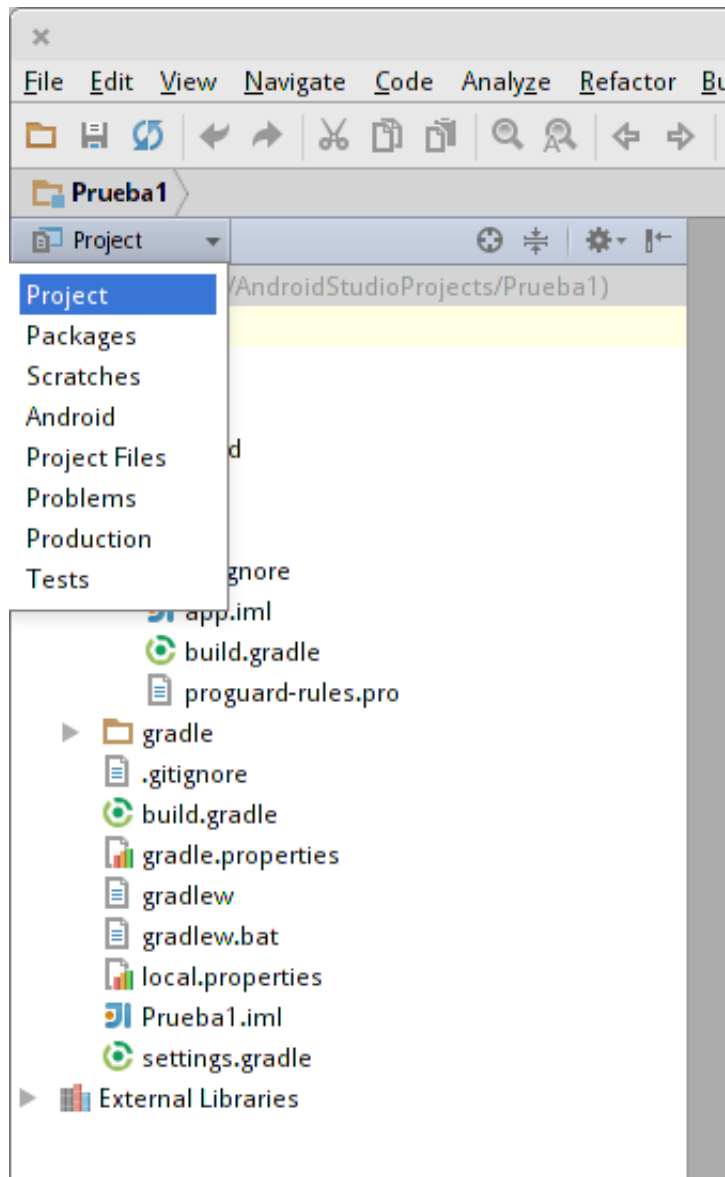
Los proyectos de Android Studio son muy complejos en cuanto a que incorporan un gran número de ficheros. La filosofía de Google a la hora de usar el entorno es intentar que los desarrolladores hagamos un trabajo todo lo profesional posible.

Por eso han dotado al entorno de todo lo necesario para mejorar la calidad de sistema:

- Se han incorporado herramientas de control de versiones como Git o SVN y conectores con repositorios como github.com
- Al crear una aplicación se crean casos de test para garantizar la calidad del código
- Se "obliga" a aplicar buenas prácticas por medio de avisos que se muestran al ir creando las aplicaciones. Un ejemplo de esto puede ser la "obligación" hacer toda aplicación fácilmente traducible.

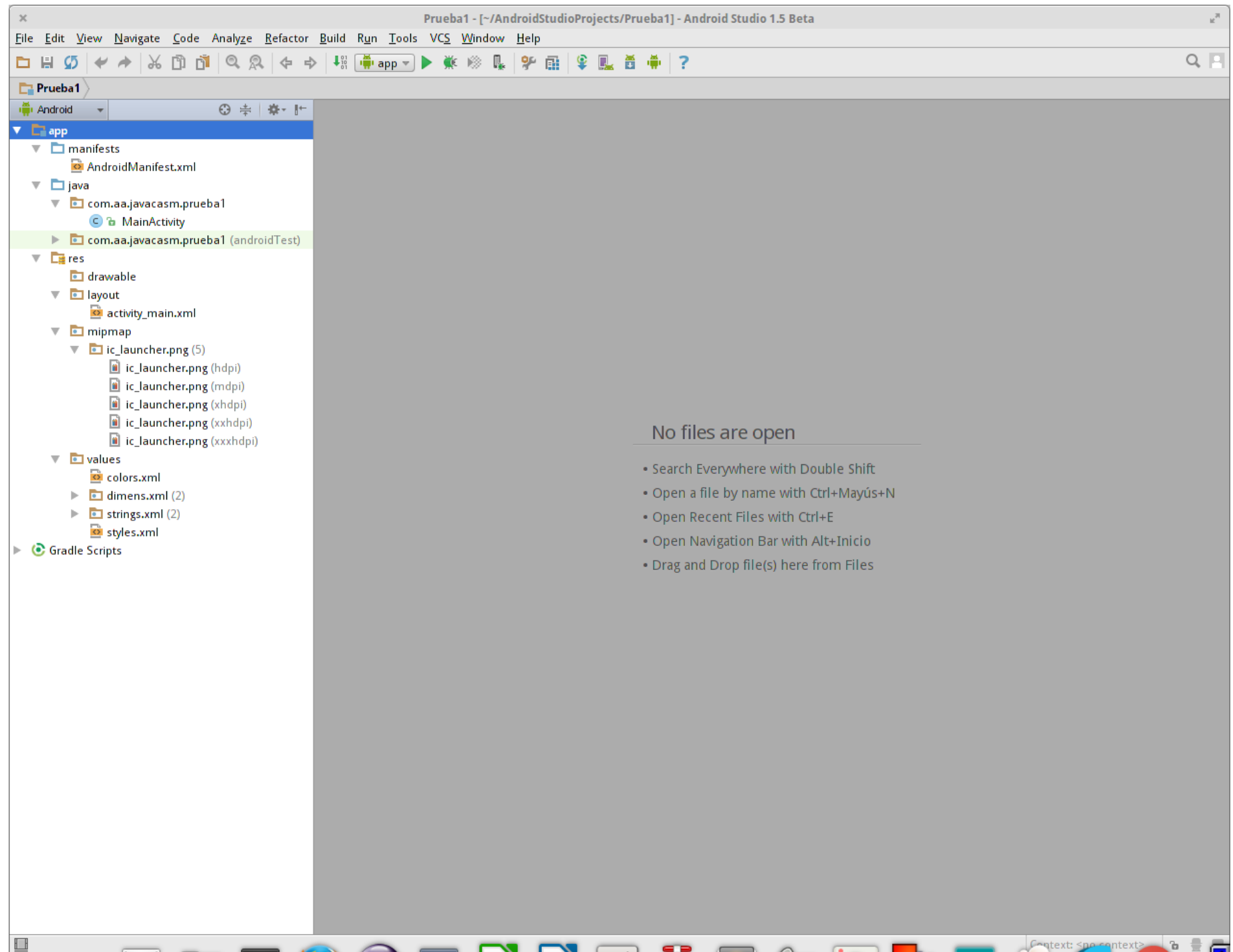
Esto da lugar a una simple aplicación está compuesta por docenas de ficheros.

Para simplificar la vida existen diversas vistas de un mismo proyecto



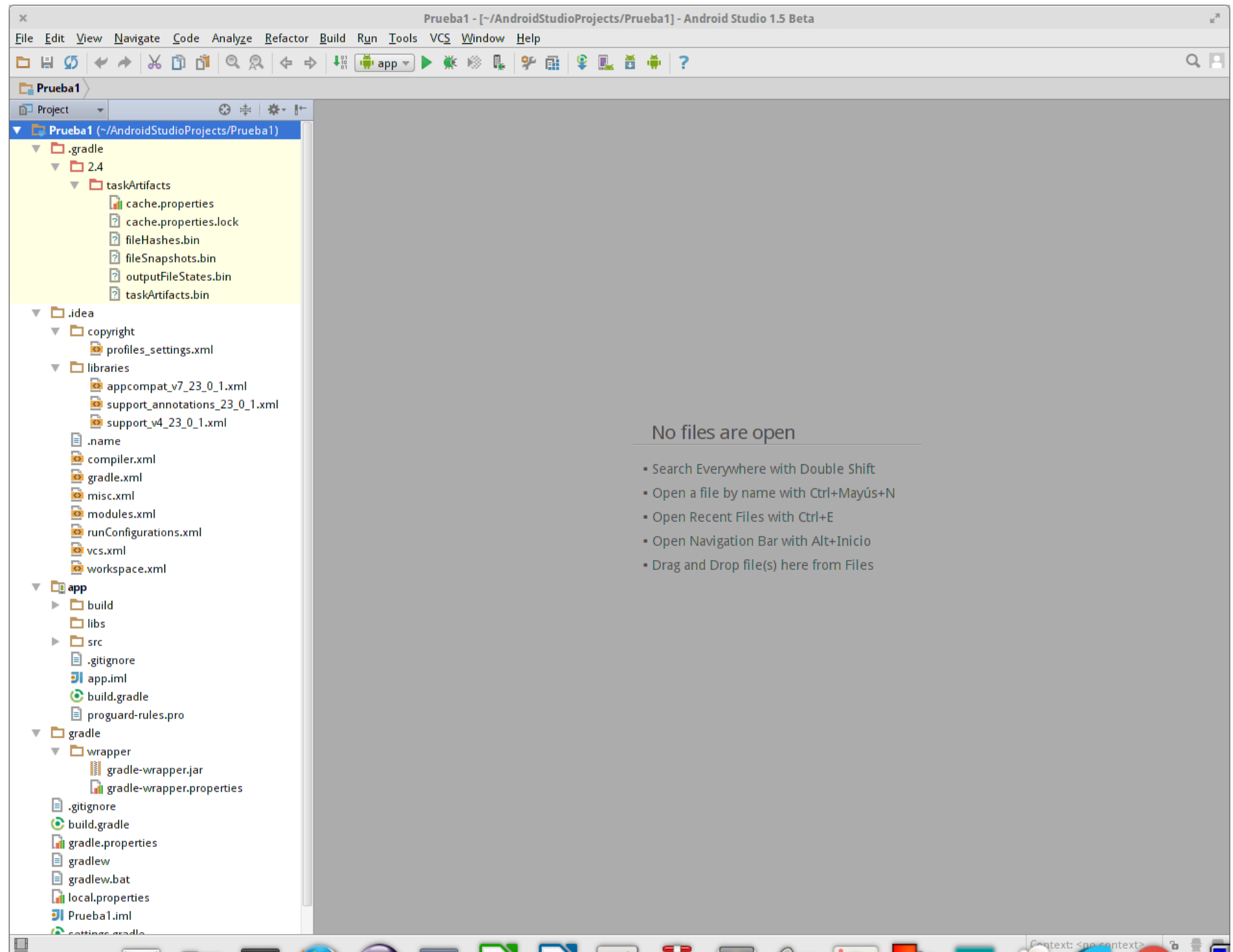
De todas ellas, las más interesantes de momento son:

- Android: Una vista simplificada que nos muestra los ficheros que se usan y modifican el 90% del tiempo





- Project: vemos todo el detalle del proyecto



Mi recomendación es que inicialmente usemos la vista Android.

Veamos lo que contiene cada carpeta de esta vista Android:

- Código Java
- Recursos
  - diseños:layout
  - imágenes
  - traducciones
  - estilos