

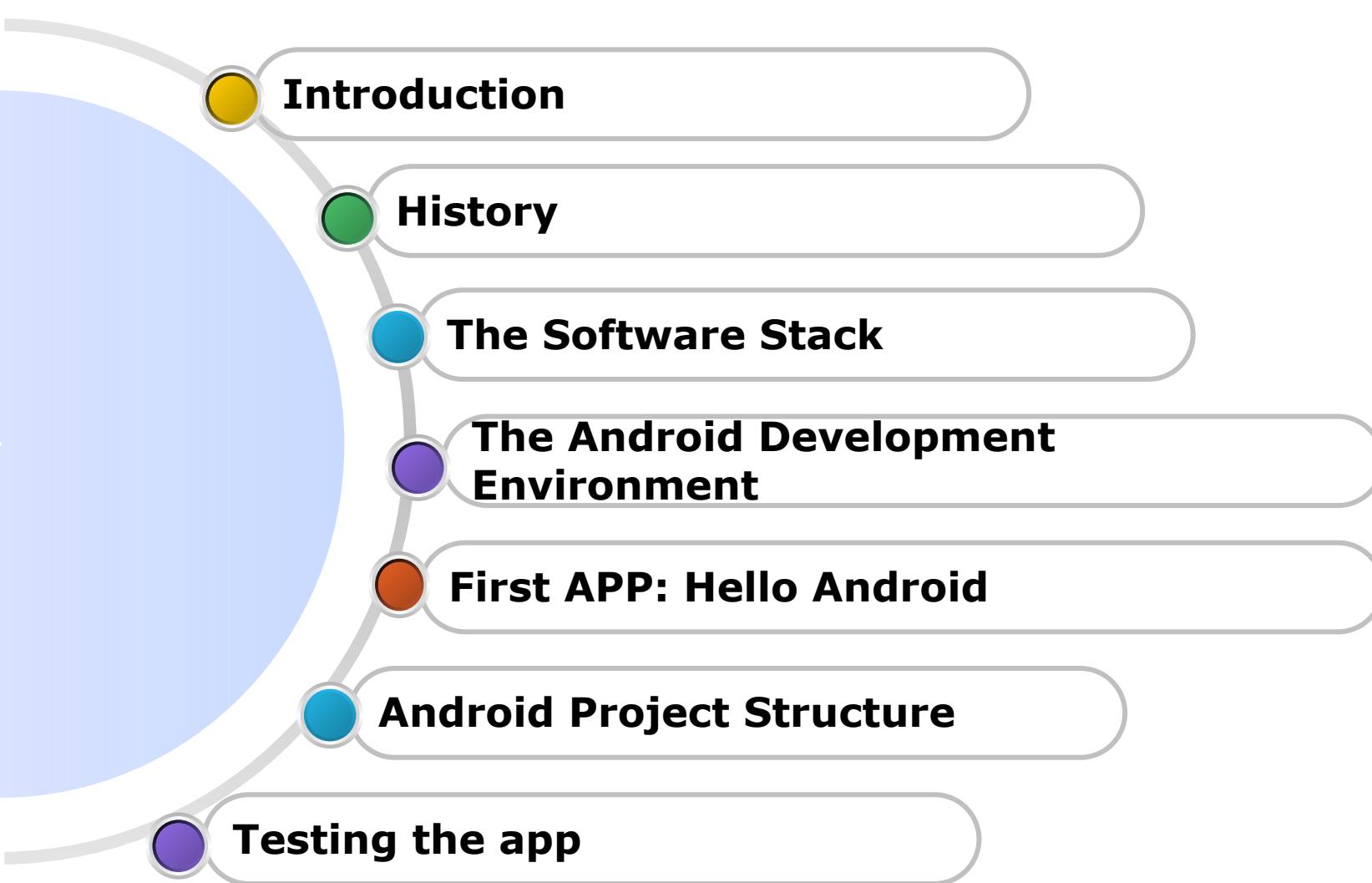
UNIVERSIDAD  
DE ANTIOQUIA

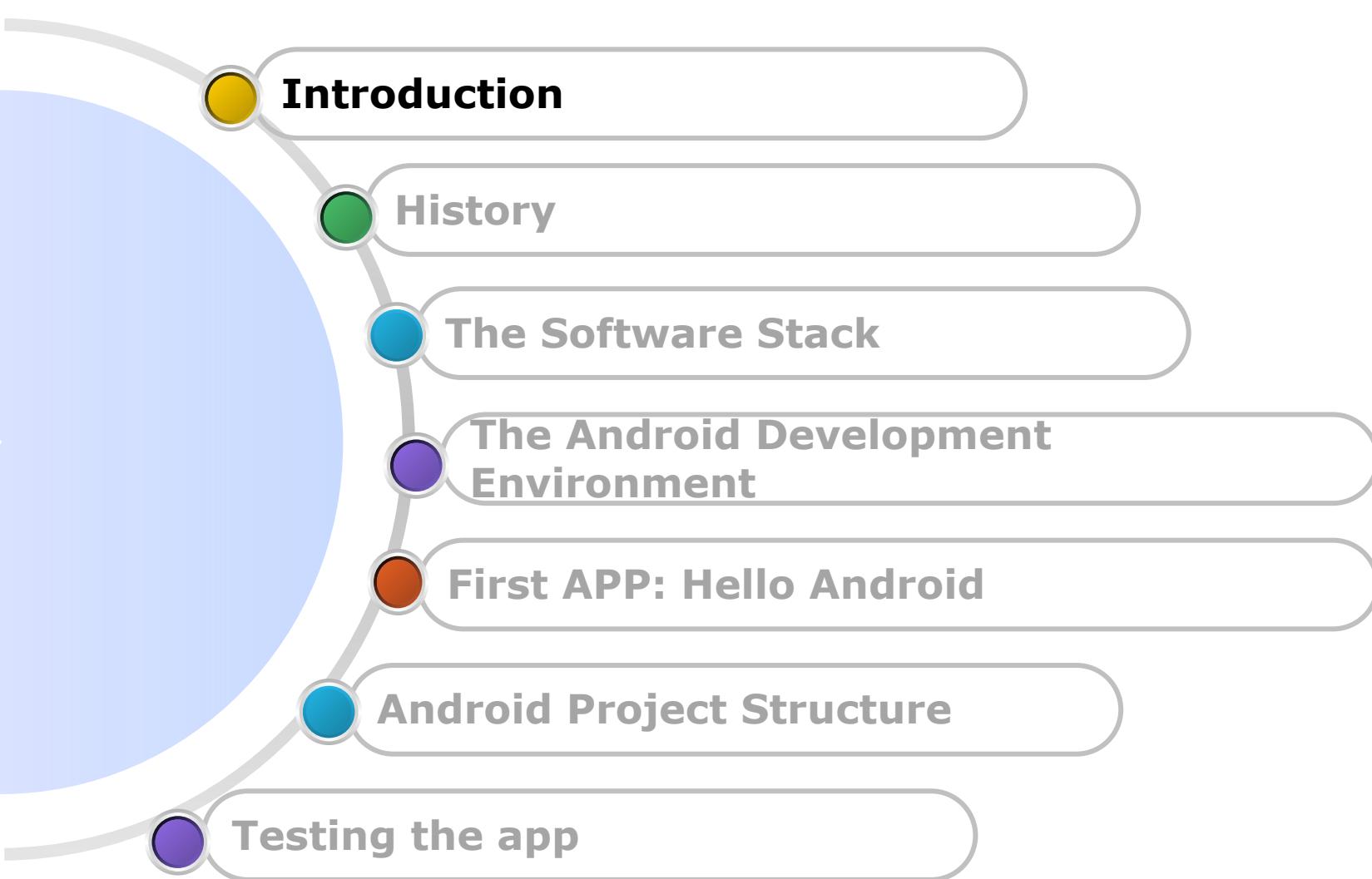
1 8 0 3

# Introducción a Android

Ing. Edwin Andrés Cubillos Vega Msc.







# Introducción

- ❖ Android es una plataforma de código abierto diseñada para dispositivos móviles
- ❖ Desarrollado por Android Inc, comprada por Google en 2005 y presentada junto con la Open Handset Alliance en 2007.
- ❖ Open Handset Alliance: Objetivo Acelerar la innovación en móviles y ofrecer al consumidor un enriquecido, económico y mejor experiencia móvil.
- ❖ La gran innovación se dió porque es una plataforma que separa el hardware del software.
- ❖ Esto permite que un gran número de dispositivos puedan ejecutar las mismas aplicaciones.
- ❖ Mejora el ecosistema para consumidores y desarrolladores.

Android es una plataforma completa: pila completa de software para un dispositivo móvil

## Para Desarrolladores:

- ❖ provee todas las herramientas y frameworks para desarrollar aplicaciones móviles rápido y fácil
- ❖ Esta herramienta se llama SDK de Android.
- ❖ No se necesita un dispositivo físico



## Para Usuarios:

Trabaja con solo sacarlo de la caja

Puede personalizar su teléfono a su gusto

Según estudios es el más utilizado en el Mundo

## Para fabricantes:

- ❖ Es la más completa solución para sus dispositivos.
- ❖ Por encima de sistemas para hardware específico.
- ❖ Proporciona todo lo demás para que sus dispositivos funcionen.
- ❖ Les da la opción de innovar en hardware en todas sus gamas.

 acer

 ARM®

 ASUS

 BROADCOM.  
Connecting everything.

 FOSSIL  
GROUP

 htc

 HUAWEI

 intel

 LG  
Life's Good

 MEDIATEK

 MIPS  
by Imagination

 motorola

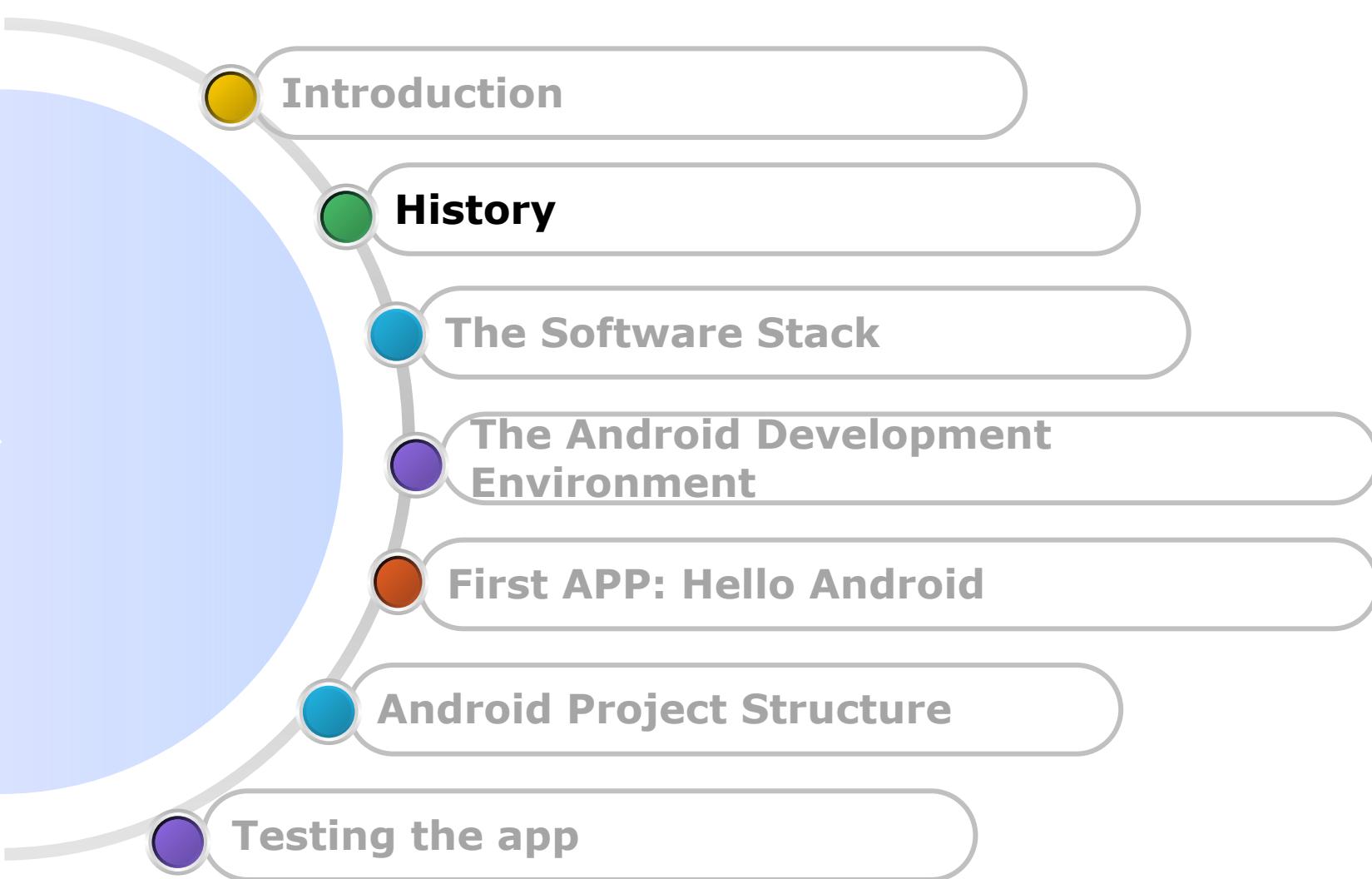
 QUALCOMM

 SAMSUNG

 SONY

 TAG Heuer  
SWISS AVANT-GARDE SINCE 1860

# Content



- ❖ En 2003 4 socios crean Android Inc en Palo Alto California con la idea de crear un SO abierto basado en Linux.
- ❖ Sus fundadores, decidieron que fuese Android Inc por la novela de Philip K. Dick y su título ¿Sueñan los androides con ovejas eléctricas?
- ❖ En 2005 Google compro Android, Inc, (US\$ 50 M)
- ❖ En 2007 se firma convenio entre LG, Intel, T-Mobile y 23 empresas mas.
- ❖ En 2007, la Open Handset Alliance anuncia Android oficialmente el Android SDK 1.0 (Apple Pie)

## Android Apple Pie 1.0 (23 sep 2008)

- ❖ Notificaciones en un menú desplegable
- ❖ Ya se incluían widgets de escritorio
- ❖ Traía de serie la que a día de hoy es la Google Play, en aquel entonces el Android Market, que por cierto no contaba con ningún sistema de pago, sino que ofrecía totalmente gratis el catálogo completo de la tienda de apps.
- ❖ Integración con las aplicaciones de Google; Google Mail, Contacts y Calendar
- ❖ Otras funciones relacionadas con productos Google como eran por ejemplo el navegador, los mapas, Google Talk, el reproductor de YouTube.
- ❖ Ya se ofrecía soporte para cámaras

- ❖ El primer dispositivo fue HTC Dream vendido por T-Mobile USA en 2008.



- ❖ Pantalla táctil, acceso a conexión de internet y navegación fácil, cámara de 3,2 MP, pantalla inicial configurable por el usuario, y personalización de varias opciones. Conexión 3G y WiFi, así como un acceso directo a las apps de Google como Gmail y Google Maps
- ❖ También se le llamó Google Phone o GPhone, y en 6 meses se vendieron un millón de unidades en los EE.UU y 100.000 en Inglaterra. Su precio era de 179\$.

2009 salen al mercado varios dispositivos basados con Android (mas de 20), igual nuevas versiones:

- ❖ **Banana Bread (1,1) (9-Feb 2009)**
  - Actualización para solucionar muchos de los bugs y errores de la primera versión
- ❖ **Cupcake (1,5) (30-abril 2009)**
  - Grabación y reproducción de vídeos con camcorder
  - Subida de vídeos desde el terminal a Youtube y Picassa
  - Nuevo teclado con predicción textual
  - Soporte Bluetooth A2DP y AVRCP
  - Conexión automática de Bluetooth en determinada distancia
  - Nuevos widgets y carpetas que ahora pueden formar parte de la pantalla de inicio
  - Transiciones de pantalla animadas

## ❖ Donut (1,6) 15 de septiembre 2009:

- Mejor experiencia en el Android Market
- Cámara, grabación y galería integrados entre sí.
- Selección múltiple de fotos en la galería para poder eliminarlas.
- Búsqueda por voz actualizada
- Mejora de la experiencia de búsqueda que ahora incluye marcadores, historiales, contactos y páginas web.
- Soporte CDMA/EVDO, 802.1x, VPN y text-to-speech
- Introducción de soporte de pantallas WVGA
- Mejoras de rendimiento en búsqueda y cámara
- GestureBuilder
- Navegación gratuita turn-by-turn

## ❖ Eclair (2,0 y 2,1) 26 de octubre 2009:

- Mejora de la velocidad de hardware
- Soporte de varios tamaños de pantalla y resoluciones
- Interfaz de usuario renovada
- Soporte para HTML5
- Introducción de novedades en las listas de contactos
- Actualización de Google Maps 3.1.2
- Soporte para Microsoft Exchange
- Soporte de flash integrado en la cámara
- Zoom digital
- Posibilidad de captura multitáctil con MotionEvent mejorada.
- Mejoras en el teclado virtual
- Bluetooth 2.1
- Fondos de pantalla animados

- ❖ 2010 es el segundo SO Móvil más vendido después de BlackBerry.
- ❖ Froyo (2,2) 20 de Mayo de 2010:
  - Se optimiza de forma general el rendimiento del SO. Se consigue una mejora de la velocidad, de la memoria y de las aplicaciones.
  - Se integra JavaScript V8 del Chrome en la aplicación navegador
  - Se mejora el lanzador de apps y se crean accesos directos para el teléfono y el navegador
  - Aparece la función de hotspot con WiFi y la de tethering por USB
  - Se añade la función de deshabilitación del tráfico de datos del operador
  - Llegan las actualizaciones automáticas a las apps descargadas desde la tienda oficial (Android Market)
  - Cambio de idiomas rápido en el teclado
  - Marcación por voz y posibilidad de compartir contactos en Bluetooth
  - Se añade soporte para contraseñas con números y letras y otras....

## ❖ Gingerbread (2,3) 6 de diciembre de 2010

- Se rediseña la interfaz de usuario general de Android
- Se añade soporte para las pantallas de mayores dimensiones y resolución WXGA y más grandes.
- Se introduce el soporte nativo en Android para llamadas VoIP SIP
- Se añade reproducción de vídeos WebM/VP8 y decodificación de audio AAC
- Se añade el soporte para Near Field Communication
- Se introducen las clásicas funciones de cortar, copiar y pegar en el sistema
- Se añade soporte nativo para sensores como giroscopios y barómetros
- Se introduce el administrador de descargas para archivos grandes
- Se mejora el apartado de administrador de energía, y el control de las apps con el administrador de tareas.
- Se puede utilizar de manera nativa más cámaras

- ❖ 2011 Android es el #1, sale al mercado Honeycomb (3,0) el 22 de febrero de 2011
  - Escritorio 3D con rediseño de widgets
  - Se mejora el sistema multitarea
  - Se añaden múltiples avances en el navegador web preestablecido. Entre ellos: navegación por pestañas, relleno en automático de formularios, sincronización de los favoritos con Google Chrome y pestañas de navegación privada.
  - Se añade el soporte a videochat con Talk
  - Se añade compatibilidad con periféricos que usan conexión USB que se ejecutan en automático desde la aplicación que les da la orden.
  - **Se permite la redimensión de aplicaciones creadas para móviles, para su adaptación a las mayores pantallas de las tabletas.**

## ❖ Ice Cream Sandwich (4,0) 18 octubre de 2011:

- Se unifica el uso de Android en cualquier dispositivo, ya sean tabletas, smartphones o netbooks
- Se añade la aceleración por hardware lo que mejora considerablemente la rapidez y la experiencia de usuario
- Multitarea mejorada. Se da la posibilidad de terminar tareas sacándolas de la lista
- Se añade gestor de consumo de datos de tráfico móvil dando el control de éste al usuario
- Se añaden teclas de acceso rápido para hacer capturas de pantalla
- Se mejora la aplicación cámara añadiendo la posibilidad de las panorámicas
- Reconocimiento de voz del usuario
- Reconocimiento facial
- Se añade soporte nativo para lápices táctiles

## ❖ Jelly Bean (4,1) y Jelly Bean (Gummy Bear) 4,2 y 4,3 - 9 julio 2012

- Mejoras de rendimiento del sistema y de los gráficos con Project Butter
- Se mejora la entrada de datos táctiles
- Aparece Google Now, el servicio-asistente de voz inteligente de Google
- Se añade el navegador propio Google Chrome
- Búsqueda mediante voz mejorada
- Las notificaciones interactivas de escritorio añaden novedades de diseño
- Se añaden mejoras relacionadas con los ajustes de tamaño de los widgets
- Se mejora la corrección ortográfica y la predicción en el teclado
- Dictado de voz offline

- ❖ El 31 - octubre 2013 se lanza la versión KitKat (4,4), API #19
  - Diseño más perfeccionado, rendimiento mejorado y nuevas funciones.
  - Multitarea más rápida
  - Mensajes en una sola aplicación Hangouts
  - Emoji por todas partes (Emoticons)
  - Vista web de Chrome
  - Subtítulos ocultos
  - Administración de dispositivos integrada: Encontrar el dispositivo o barrar su contenido
  - Nuevo diseño de la aplicación Descargas
  - Cambia de pantalla de inicio fácilmente
  - Actualización de la aplicación Correo

- ❖ El 12 de noviembre 2014 sale la API 21 Lollipop 5,0 y en Abril la 5.1 (Api 22)
  - Teléfono, Tablets, Relojes y TV
  - Material Design
  - Aumento de seguridad
  - Dispositivo compartidos
  - Nuevos ajustes rápidos
  - Mejor conexión a Internet en cualquier lugar y funciones de Bluetooth de baja energía más potentes
  - Gráficos más atractivos y mejores funciones de audio, vídeo y cámara
  - Ok Google Fácil acceso a información y realización de tareas
  - Ahora disponible en más de 68 idiomas
  - Enciende y utiliza el dispositivo en solo unos segundos

## ❖ Android 6,0 Marshmallow (5 oct 2015)

- Optimización de Google Now, con un simple toque, podrás ver tarjetas con aplicaciones y toda la información que necesitas.
- Batería más eficiente e inteligente
- Función de descanso
- Aplicaciones inactivas limitadas en consumo de batería
- Configuración de permisos y compartir aplicaciones
- Identificación con huella dactilar

## ❖ Android 7.0 Nougat (15-jun 2016)

- Dos idiomas la mismo tiempo
- Vista multiventana
- Cambio rápido de aplicación
- API de VULKAN: Gráficos 3D de alto rendimiento
- Modo de realidad virtual requiere compatibilidad con Daydream
- Descanso en movimiento
- Respuesta directa a las notificaciones
- Función ahorro de datos, apps en segundo plano no acceden
- Actualización en segundo plano

- ❖ Android 8.0 Oreo (21-agosto 2017)
  - API para redes neuronales
  - Notificaciones con iconos
  - Texto autoescalable
  - Tipos de letra (fonts) descargables
  - Fonts en xml
  - Accesos directos para tareas específicas de la app

# Historia



**Angel Cake**

Android 1.0



**Battenberg**

Android 1.1



**Cupcake**

Android 1.5



**Donut**

Android 1.6



**Eclair**

Android 2.0/2.1



**Froyo**

Android 2.2



**Gingerbread**

Android 2.3



**Honeycomb**

Android 3.0



**Ice Cream Sandwich**

Android 4.0



**Jelly Bean**

Android 4.1



**KitKat**

Android 4.4



**Lollipop**

Android 5.0



Android 6.0 Marshmallow

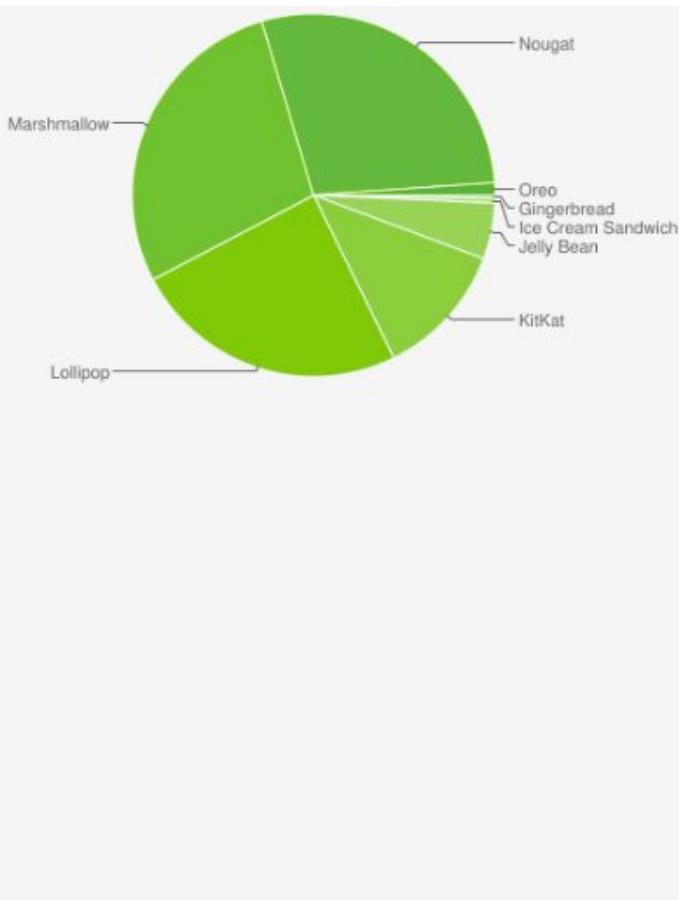


7.0 (Nougat)



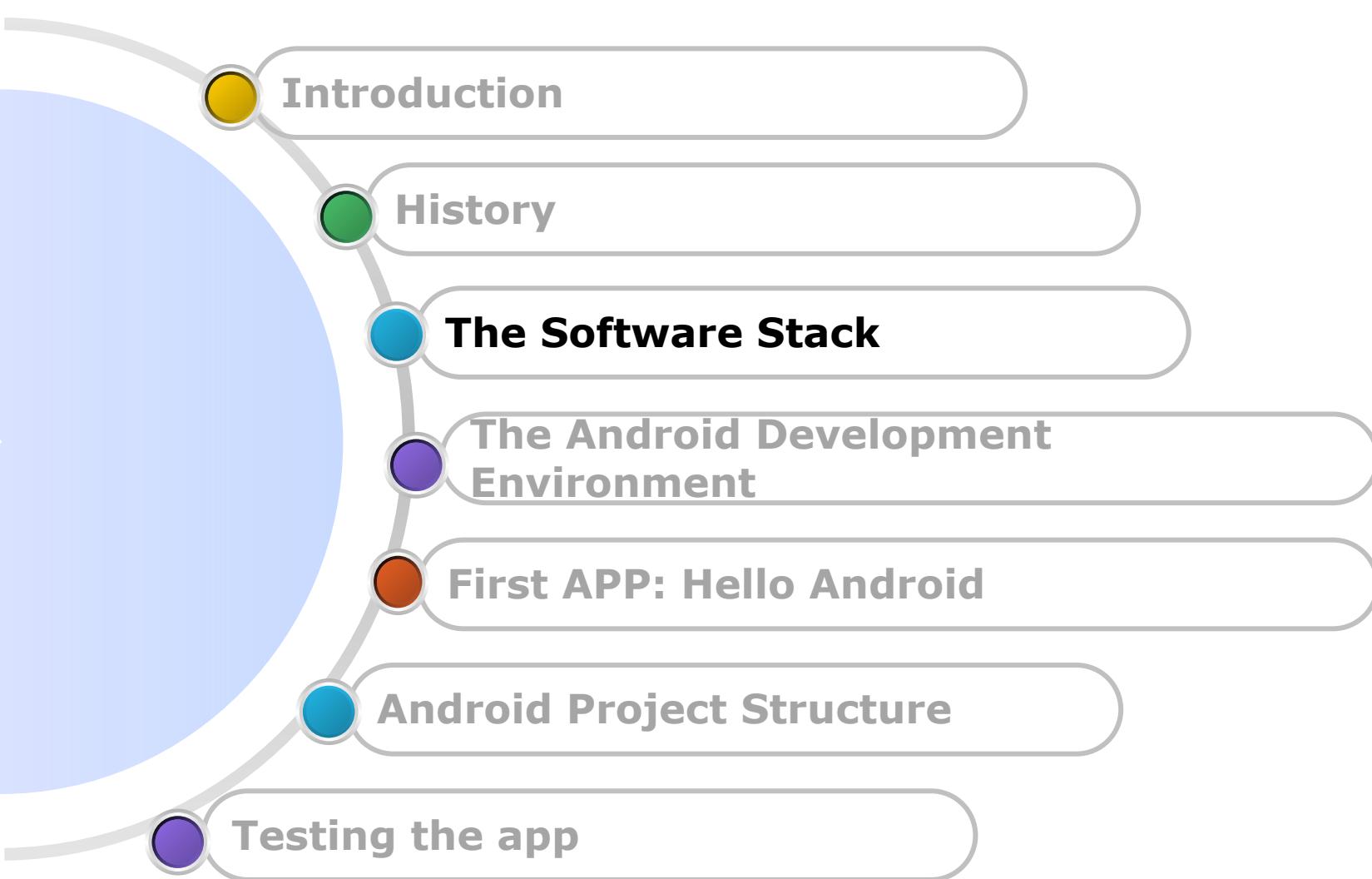
- ❖ Como desarrolladores debemos tener en cuenta para que versión de Android se va a desarrollar

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	0.3%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.4%
4.1.x	Jelly Bean	16	1.7%
4.2.x		17	2.6%
4.3		18	0.7%
4.4	KitKat	19	12.0%
5.0	Lollipop	21	5.4%
5.1		22	19.2%
6.0	Marshmallow	23	28.1%
7.0	Nougat	24	22.3%
7.1		25	6.2%
8.0	Oreo	26	0.8%
8.1		27	0.3%



ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
4.0 Ice Cream Sandwich	15	
4.1 Jelly Bean	16	99.2%
4.2 Jelly Bean	17	96.0%
4.3 Jelly Bean	18	91.4%
4.4 KitKat	19	90.1%
5.0 Lollipop	21	71.3%
5.1 Lollipop	22	62.6%
6.0 Marshmallow	23	39.3%
7.0 Nougat	24	8.1%
7.1 Nougat	25	1.5%

# Content



# The Software Stack

- ❖ El sistema operativo Android está formado por varias capas
- ❖ Cada capa tiene sus propias características y propósitos
- ❖ Pero estas no están separadas entre sí.
- ❖ Android está construido en la parte superior del kernel de Linux y aprovecha sus fortalezas
  - Portabilidad
  - Seguridad
  - Características

## Portabilidad:

- ❖ Linux es de plataforma flexible, se puede portar a cualquier arquitectura HW.
- ❖ Característica que es heredada por Android y no nos tenemos que preocupar por las características de hardware.
- ❖ La mayoría de las partes de bajo nivel se han escrito en lenguaje C, lo que da más portabilidad.



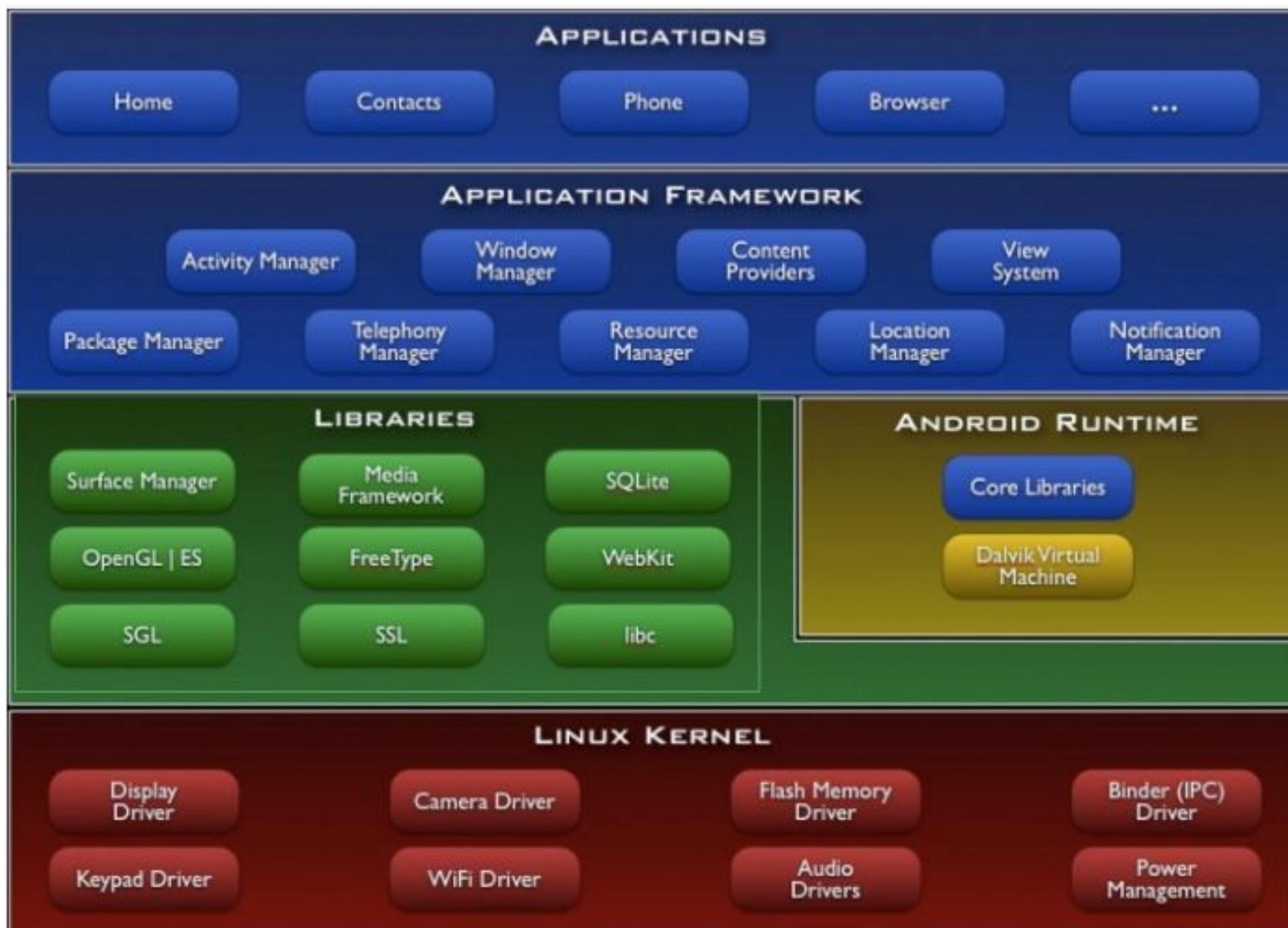
## Seguridad:

- ❖ Linux es un sistema altamente seguro, esto ha sido probado durante décadas.
- ❖ Android se basa principalmente en Linux para la seguridad.
- ❖ Todas las aplicaciones de Android se ejecutan en procesos separadas con los permisos definidos por el sistema Linux.
- ❖ A diferencia de otras plataformas móviles basadas en java, el kernel de Android es el único ejecutor de permisos de Android.
- ❖ Lo que permite un simple y poderoso mecanismo de seguridad.
- ❖ Esto también permite a las app de Android acceder a código nativo escritos en C a través de varias librerías utilizando la interfaz de java.

## Features

- ❖ El kernel de Linux tiene muchas características que son heredadas por Android tales como:
  - Administración de memoria
  - Administración de energía
  - Creación de redes
  - Funcionalidad de radio

# The Software Stack



Display  
Driver

Camera Driver

Flash Memory  
DriverBinder (IPC)  
Driver

Keypad Driver

WiFi Driver

Audio  
DriversPower  
Management

## Linux Kernel:

- ❖ Es la capa más baja del software de la plataforma Android
- ❖ Provee los servicios básicos que cualquier dispositivo Android deba realizar.
- ❖ El Kernel de Android y el Kernel de Linux proporciona servicios de sistemas operativos genéricos como:
  - Una arquitectura de permisos: Restringir el acceso a datos y/o recursos a determinados procesos. (Security)
  - Administración de memoria y energía para que se puedan ejecutar múltiples procesos sin interferencia entre ellos.
  - Se encarga de detalles de bajo nivel de archivos y redes I/O

Display  
Driver

Camera Driver

Flash Memory  
DriverBinder (IPC)  
Driver

Keypad Driver

WiFi Driver

Audio  
DriversPower  
Management

## Linux Kernel:

- Permitir la comunicación con un amplio rango de dispositivos que a menudo son acoplados con dispositivos de cómputo
- Permitir la comunicación con un amplio rango de dispositivos que a menudo son acoplados con dispositivos de cómputo
  - Memorias
  - Radio: Bluetooth
  - Cámaras

Display  
Driver

Camera Driver

Flash Memory  
DriverBinder (IPC)  
Driver

Keypad Driver

WiFi Driver

Audio  
DriversPower  
Management

## Linux Kernel:

- ❖ Incluye varios componentes específicos de Android:
  - Administración de Energía
  - Características de administración de memoria - Memoria Compartida (memory sharing)
  - Mecanismos de comunicación entre procesos llamado “**binder**”: permite a varios procesos compartir datos y servicios.
  - Muchos Más

Surface Manager

Media Framework

SQLite

OpenGL | ES

FreeType

WebKit

SGL

SSL

libc

# The Software Stack

## Libraries

- ❖ Están escritas normalmente en C y C++.
- ❖ Denominadas librerías nativas
- ❖ Manejan el core, actividades de rendimiento del dispositivo, como abrir rápidamente páginas web:
- ❖ Media libraries: Permite la reproducción y grabación de formato de audio, video: mpeg4, h.264, mp3, AAC, AMR, JPG y PNG
- ❖ Surface Manager: Administra el acceso a los subsistemas de pantalla y utilización de gráficos 2D y 3D desde varias aplicaciones.

Surface Manager

Media Framework

SQLite

OpenGL | ES

FreeType

WebKit

SGL

SSL

libc

# The Software Stack

## Libraries

- ❖ libWebCore: es un web browser, potencia tanto el “Android Browser” como las Web View.
- ❖ OpenGL para el rendimiento gráfico
- ❖ 3D Libraries: pueden usar tanto el hardware de aceleración 3D
- ❖ SQLite: poderoso y liviano motor de bases de datos relacionales disponible para todas las aplicaciones



- ❖ Adicional la capa cuenta con Android Runtime
- ❖ Apoyan la escritura y funcionamiento de apps.

## Core Java Libraries

- Las Apps para Android son escritas en Java
- Android provee bloques de código reusable de java: Java, JavaX o paquetes de extensión java (JAVA.\* , JAVAX.\* )
- Utilizado para tareas comunes como estructuras de datos, mecanismos de concurrencia y I/O archivos



ck Software

- Android Packages: (ANDROID.\* ) - Estos paquetes son un software específico que tiene un ciclo de vida para app móviles.
- Org Packages: operaciones de internet y web (ORG.\* )
- Junit Packages: utilizado para el testeo o prueba de aplicaciones



## Dalvik Virtual Machine

- Es el software que actualmente ejecuta las aplicaciones Android
- Como las aplicaciones son escritas en Java el DVM se pensaría que es una máquina virtual de JAVA pero no
- Flujo:
  1. Las App se escriben en JAVA
  2. El compilador de java convierte en múltiples archivos bytecode de java



3. Una herramienta llamada DX transforma los bytecode de Java en un archivo con formato dex, usualmente llamado classes.dex
  4. Luego el archivo .dex es empaquetado con otro recurso de la app e instalado en el dispositivo
  5. Finalmente cuando el usuario abre la aplicación el Dalvik Virtual Machine ejecuta el archivo classes.dex
- ❖ EL DVM es diseñado para dispositivos con restricciones como los móviles:
    - CPU lenta, Menor RAM, Batería limitada
  - ❖ Video recomendado: [Link](#)

## APPLICATION FRAMEWORK

Activity Manager

Window  
Manager

Content  
Providers

View  
System

Package Manager

Telephony  
Manager

Resource  
Manager

Location  
Manager

Notification  
Manager

# Application Framework

- ❖ Contiene software reusable que muchas aplicaciones pueden necesitar
- ❖ *Package Manager*: es una base de datos que mantiene todas las aplicaciones instaladas en el dispositivo

Activity Manager

Window Manager

Content Providers

View System

Package Manager

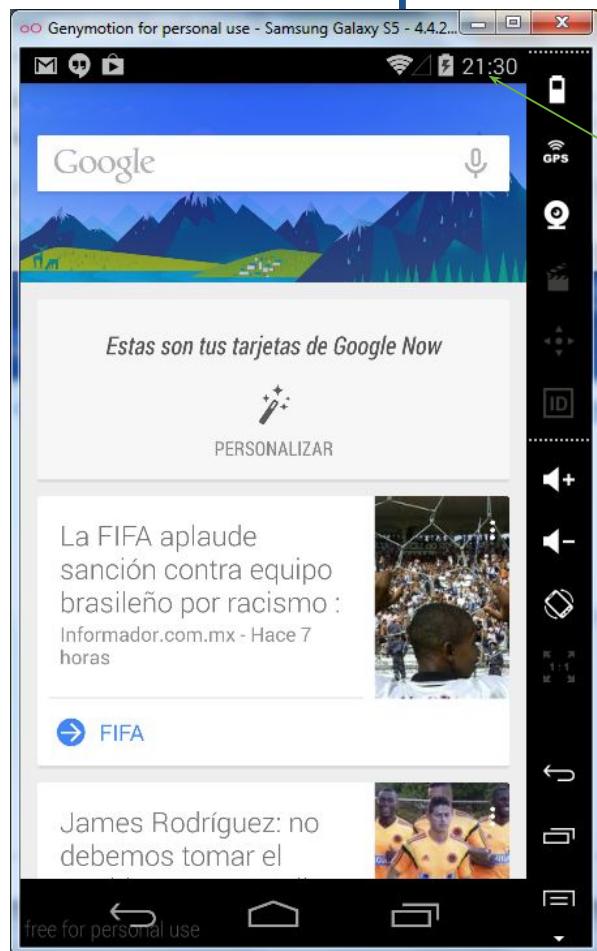
Telephony Manager

Resource Manager

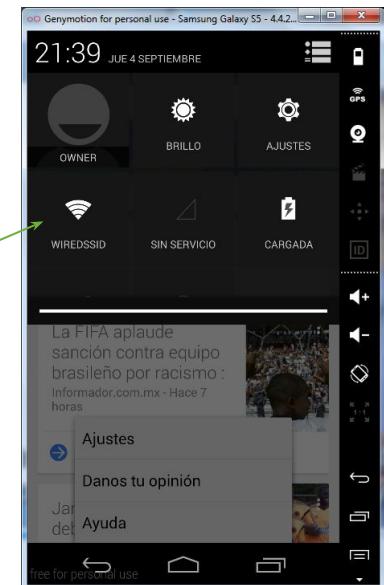
Location Manager

Notification Manager

## ❖ *Window Manager: Gestiona las ventanas de las aplicaciones*



Barra de notificaciones



Ventana Principal (Browser)

Activity Manager

Window Manager

Content Providers

View System

Package Manager

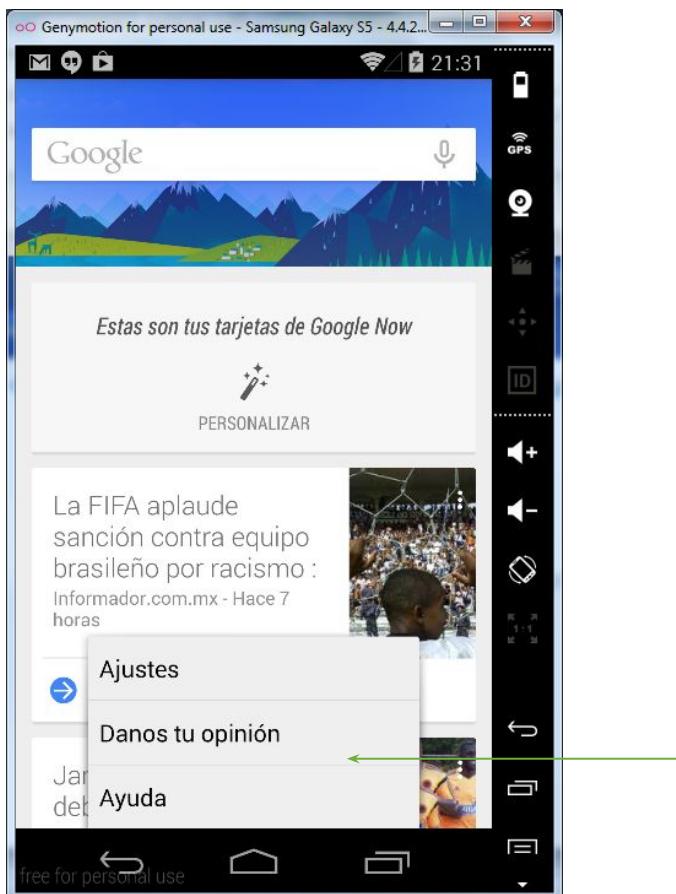
Telephony Manager

Resource Manager

Location Manager

Notification Manager

❖ *Window Manager: Gestiona las ventanas de las aplicaciones*



Subventanas como menus

## APPLICATION FRAMEWORK

Activity Manager

Window Manager

Content Providers

View System

Package Manager

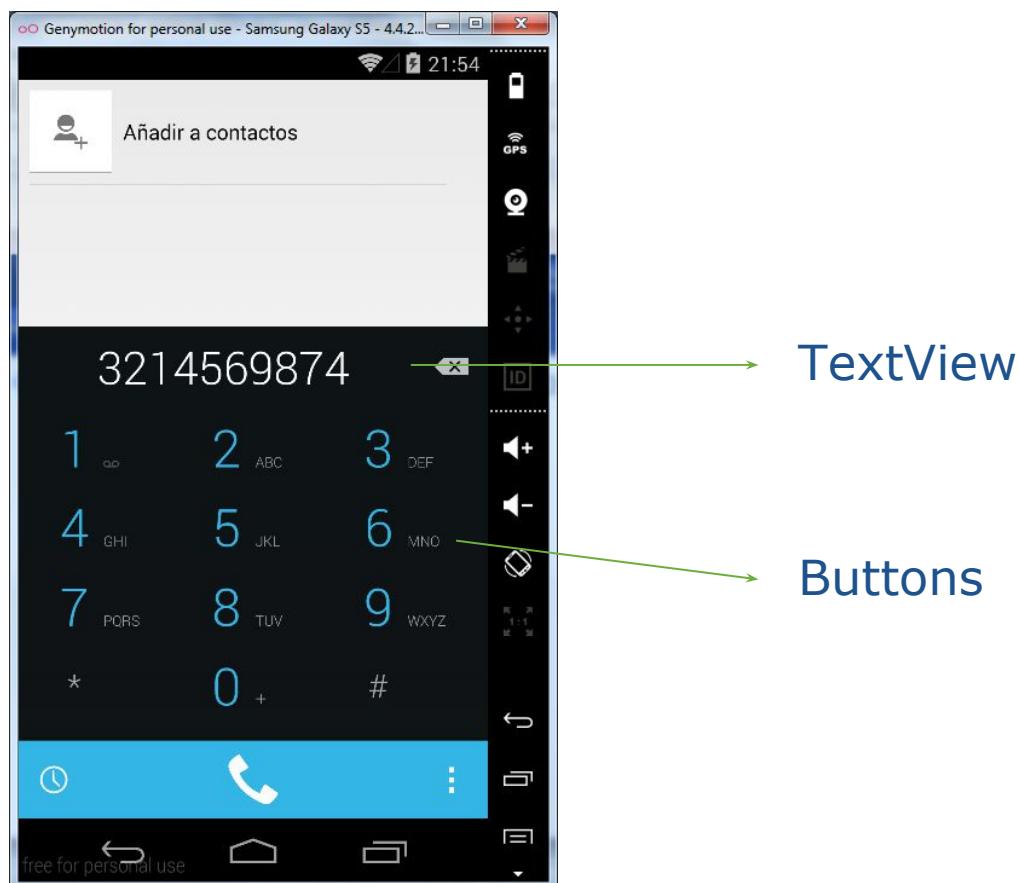
Telephony Manager

Resource Manager

Location Manager

Notification Manager

- ❖ *View System*: contiene elementos gráficos comunes: botones etc.



## APPLICATION FRAMEWORK

Activity Manager

Window Manager

Content Providers

View System

Package Manager

Telephony Manager

Resource Manager

Location Manager

Notification Manager

- ❖ *Resource Manager:* Administran los recursos no compilados que conforman una aplicación
- ❖ Archivos Strings, gráficos, y layouts
- ❖ Un ejemplo específico es la configuración multilenguaje de una aplicación.
- ❖ Esto se hace con un archivo llamado Strings.xml y se crea uno para cada lenguaje

## APPLICATION FRAMEWORK

Activity Manager

Window Manager

Content Providers

View System

Package Manager

Telephony Manager

Resource Manager

Location Manager

Notification Manager

- ❖ *Activity Manager:* Cada actividad corresponde a una interfaz de usuario.
- ❖ Una actividad puede activar varias actividades
- ❖ El activity Manager ayuda a coordinar y apoyar el tipo de navegación



## APPLICATION FRAMEWORK

Activity Manager

Window Manager

Content Providers

View System

Package Manager

Telephony Manager

Resource Manager

Location Manager

Notification Manager

- ❖ *Content providers:* Son Bases de datos que permiten a las apps almacenar y compartir información estructurada.
- ❖ Ejemplo: Acceder a los contactos guardados en el smartphone
- ❖ *Location Manager:* Permite recibir información de localización y de movimiento como el generado por el sistema gps.
- ❖ Este permite realizar tareas específicas con estos datos como ubicar direcciones desde la dirección actual.

## APPLICATION FRAMEWORK

Activity Manager

Window Manager

Content Providers

View System

Package Manager

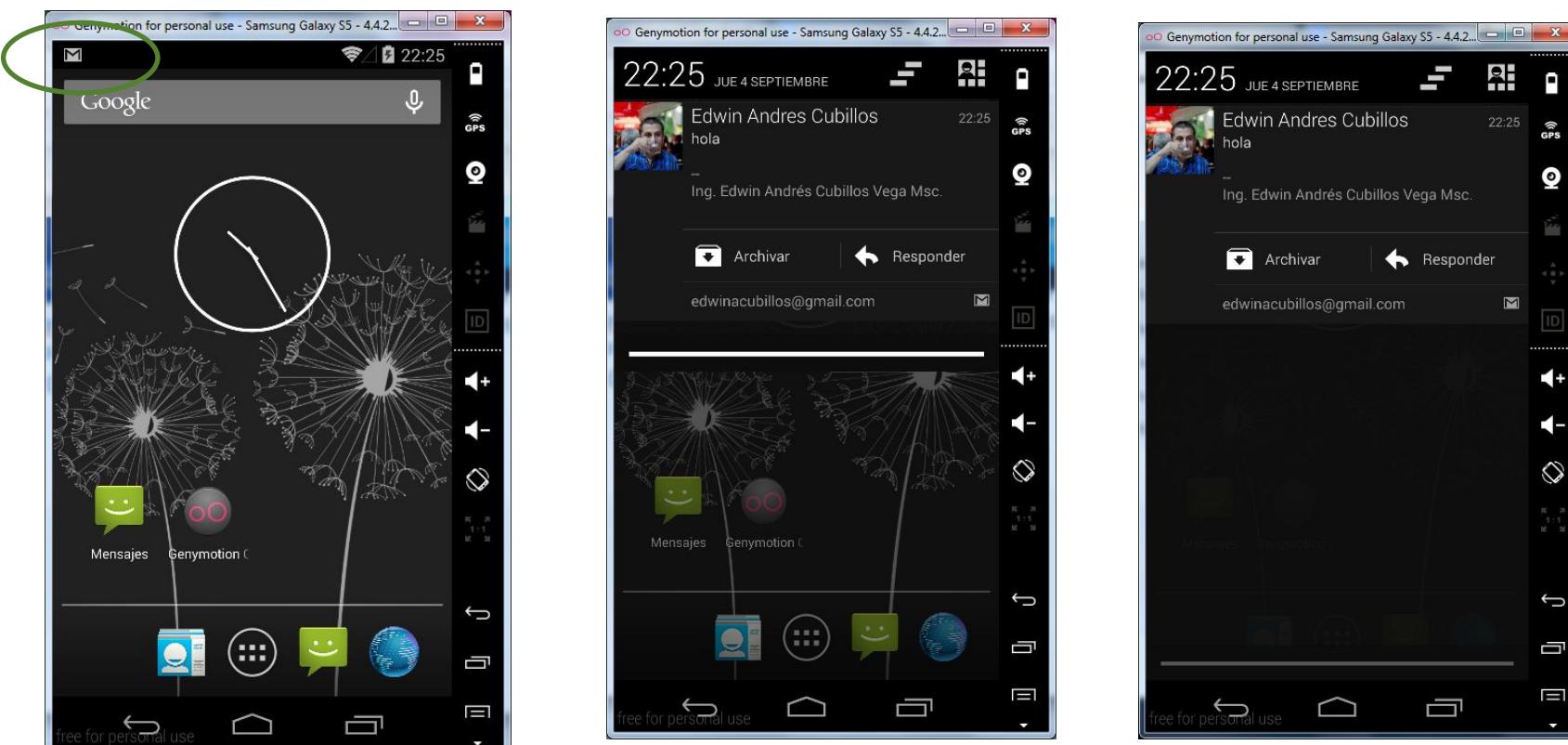
Telephony Manager

Resource Manager

Location Manager

Notification Manager

- ❖ *Notification Manager:* Permite a las aplicaciones poner información en la barra de notificaciones
- ❖ Ejemplo: Cuando llegan correos electrónicos



Home

Contacts

Phone

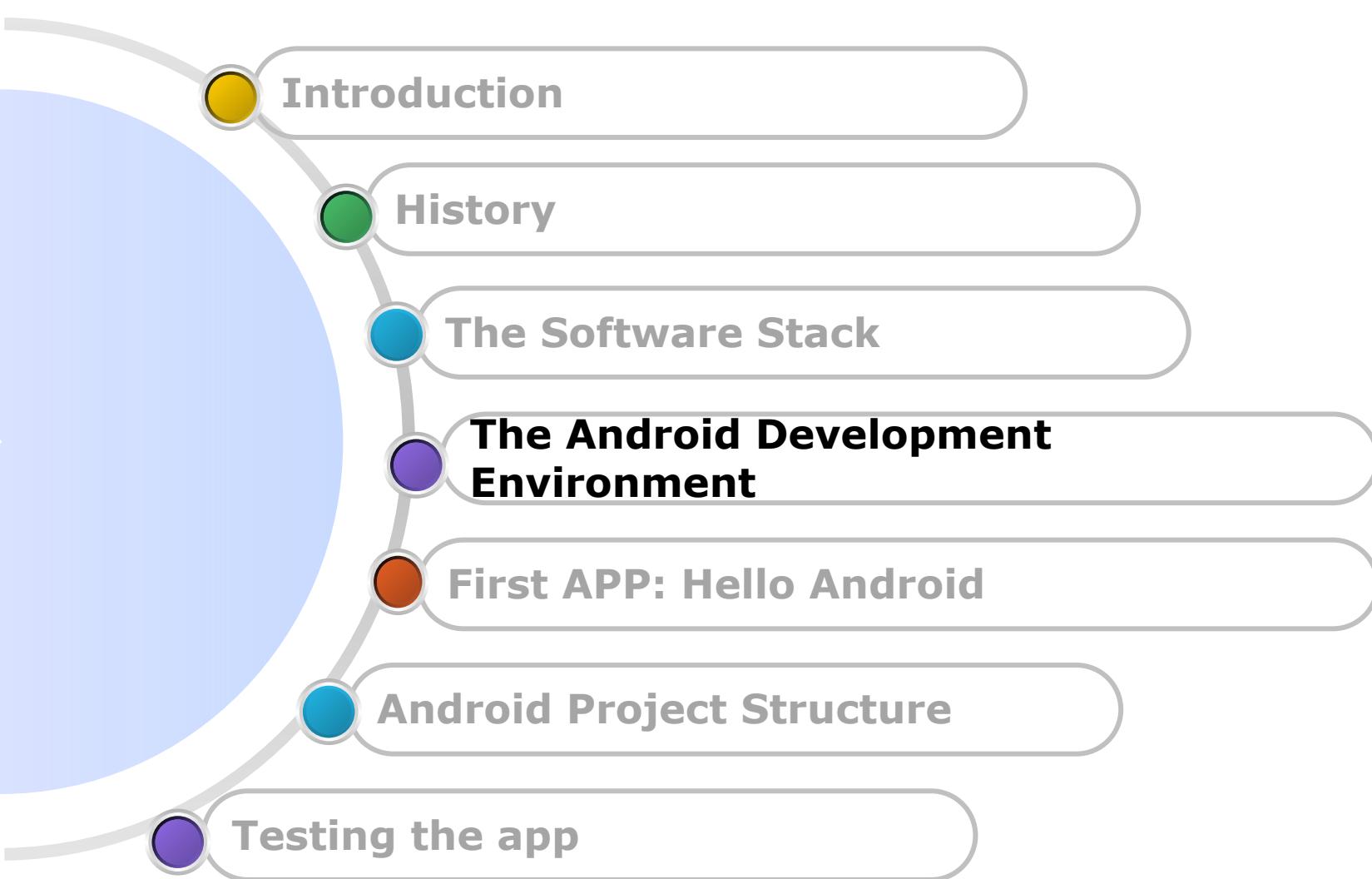
Browser

...

## Applications

- ❖ Son las aplicaciones estándar que traen los dispositivos:
  - Home – Main Screen
  - Phone Dialer
  - Web Browser
  - Email Reader

# Content



# Prerrequisitos

## System Requirements

### Windows

- Microsoft® Windows® 7/8/10 (32- or 64-bit)
- 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator
- 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

### Mac

- Mac® OS X® 10.10 (Yosemite) or higher, up to 10.13 (macOS High Sierra)
- 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator
- 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

### Linux

- GNOME or KDE desktop  
*Tested on Ubuntu® 14.04 LTS, Trusty Tahr (64-bit distribution capable of running 32-bit applications)*
- 64-bit distribution capable of running 32-bit applications
- GNU C Library (glibc) 2.19 or later
- 3 GB RAM minimum, 8 GB RAM recommended; plus 1 GB for the Android Emulator
- 2 GB of available disk space minimum, 4 GB Recommended (500 MB for IDE + 1.5 GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

# Instalación y configuración

## ❖ Android Studio Descargar e Instalar

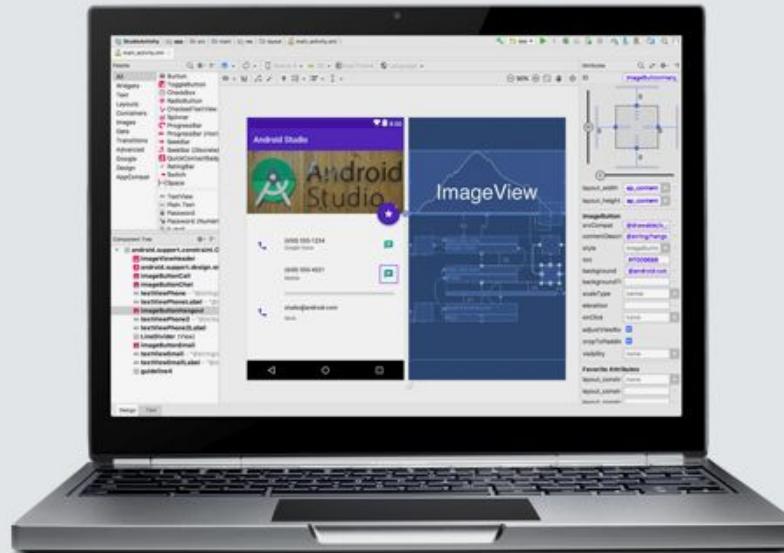
### Android Studio

The Official IDE for Android

Android Studio provides the fastest tools for building apps on every type of Android device.

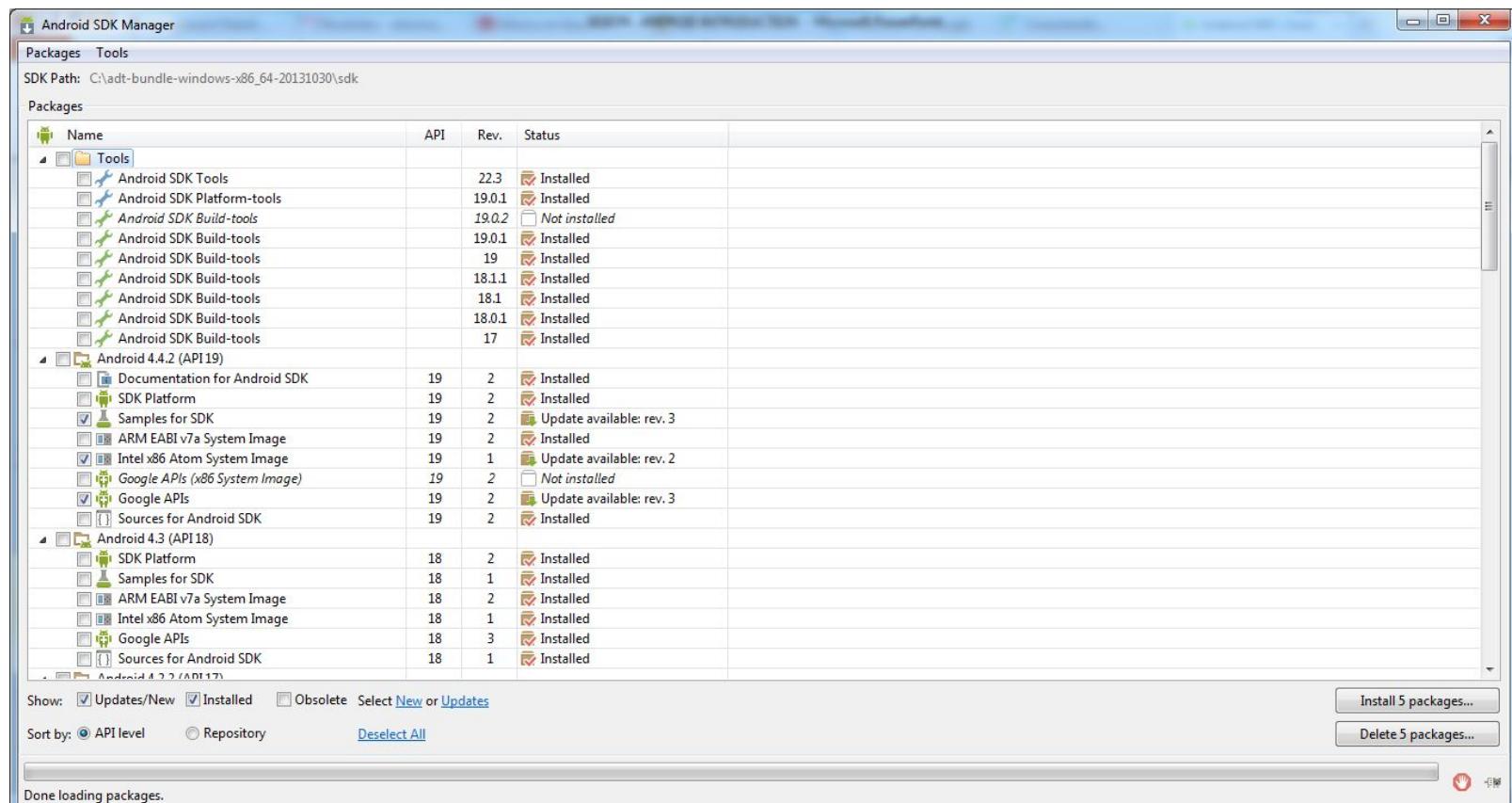
World-class code editing, debugging, performance tooling, a flexible build system, and an instant build/deploy system all allow you to focus on building unique and high quality apps.

**DOWNLOAD ANDROID STUDIO**  
3.0.1 FOR LINUX (737 MB)



# Instalación y configuración

La instalación trae la última plataforma Android, si se necesita una versión anterior se deben descargar *Android SDK Manager*



# Android Studio

## ❖ Editor Inteligente

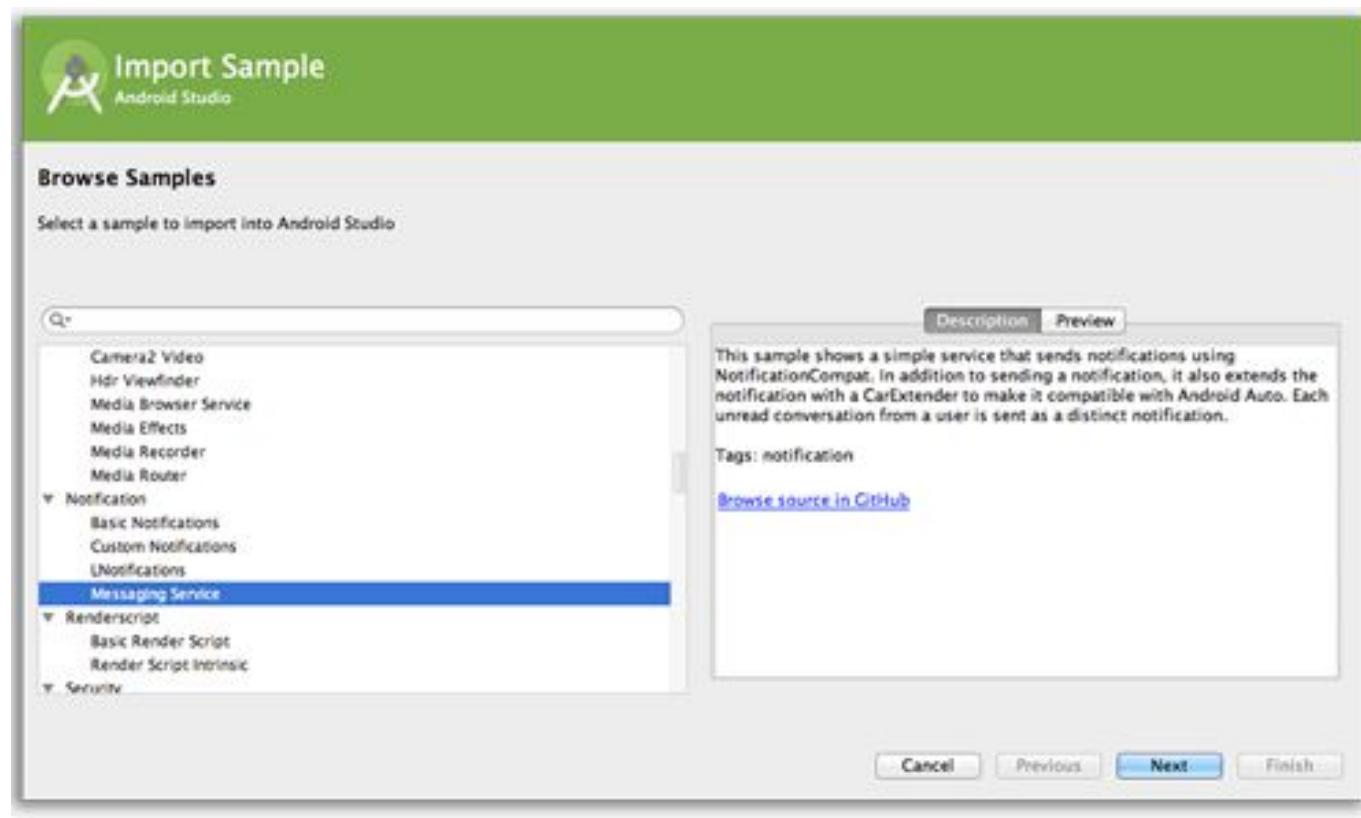
The screenshot shows the Android Studio interface with the code editor open. The code is for a Wearable application, specifically a BaseActivity.java file. A tooltip is displayed over the code, providing documentation for a variable named 'navDrawerItem\_my\_schedule'. The tooltip shows the variable name, its type (String), and its value, which is 'navDrawer\_item\_my\_schedule'. Below the tooltip, a table lists the value for various languages: main (Default) is 'My Schedule', de is 'Mein Zeitplan', es is 'Mi agenda', fr is 'Mon agenda', it is 'Il mio Programma', ja is '私のスケジュール', ko is '내 일정', pt is 'Minha Agenda', and ru is 'Мой расписае'.

Flavor/Library Configuration	Value	
main	Default	My Schedule
	de	Mein Zeitplan
	es	Mi agenda
	fr	Mon agenda
	it	Il mio Programma
	ja	私のスケジュール
	ko	내 일정
	pt	Minha Agenda
	ru	Мой расписае

At the core of Android Studio is an intelligent code editor capable of advanced code completion, refactoring, and code analysis.

The powerful code editor helps you be a more productive Android app developer.

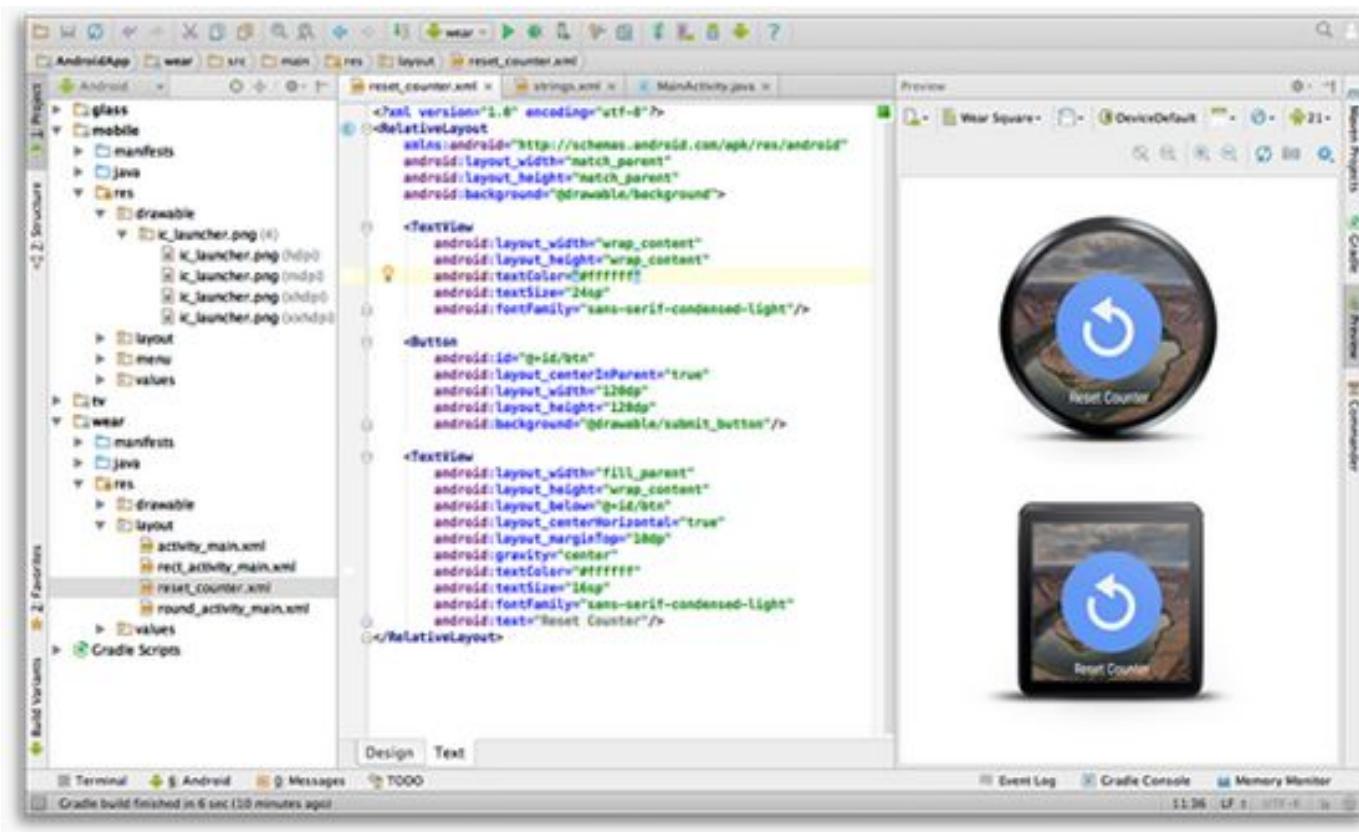
## ❖ Plantillas con código e Integración GitHub



New project wizards make it easier than ever to start a new project.

Start projects using template code for patterns such as navigation drawer and view pagers, and even import Google code samples from GitHub.

## ❖ Desarrollo multiscreen de apps



Build apps for Android phones, tablets, Android Wear, Android TV, Android Auto and Google Glass.

With the new Android Project View and module support in Android Studio, it's easier to manage app projects and resources.

- ❖ Dispositivo Virtual para todas las formas y tamaños

## Fast and feature-rich emulator

Install and run your apps faster than with a physical device and test your app on virtually any Android device configuration: Android phones, Android tablets, Android Wear, and Android TV devices.

The new Android Emulator 2.0 is faster than ever and allows you to dynamically resize the emulator and access a suite of sensor controls.



## ❖ Uso de Gradle y Maven

The screenshot shows the Android Studio interface with the following details:

- Project Structure:** The left sidebar shows the project structure with modules like app, manifests, java, and res.
- Gradle Scripts:** Under the app module, the build.gradle file is selected.
- Code Editor:** The main area displays the build.gradle code:

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 21
    buildToolsVersion "21.1.0"

    defaultConfig {
        applicationId "com.example.easonj.androidapp"
        minSdkVersion 10
        targetSdkVersion 21
        versionCode 1
        versionName "1.0"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'),
                'proguard-rules.pro'
        }
    }
}

dependencies {
    compile fileTree(dir: 'libs', include: ['*.jar'])
    compile 'com.android.support:appcompat-v7:21.0.2'
    compile 'com.google.android.gms:play-services:6.1.71'
}
```
- Toolbars and Bottom Bar:** Standard Android Studio toolbars and a bottom bar showing 'Gradle build finished in 3 sec (a minute ago)'.

Create multiple APKs for your Android app with different features using the same project.

Manage app dependencies with Maven.

Build APKs from Android Studio or the command line.

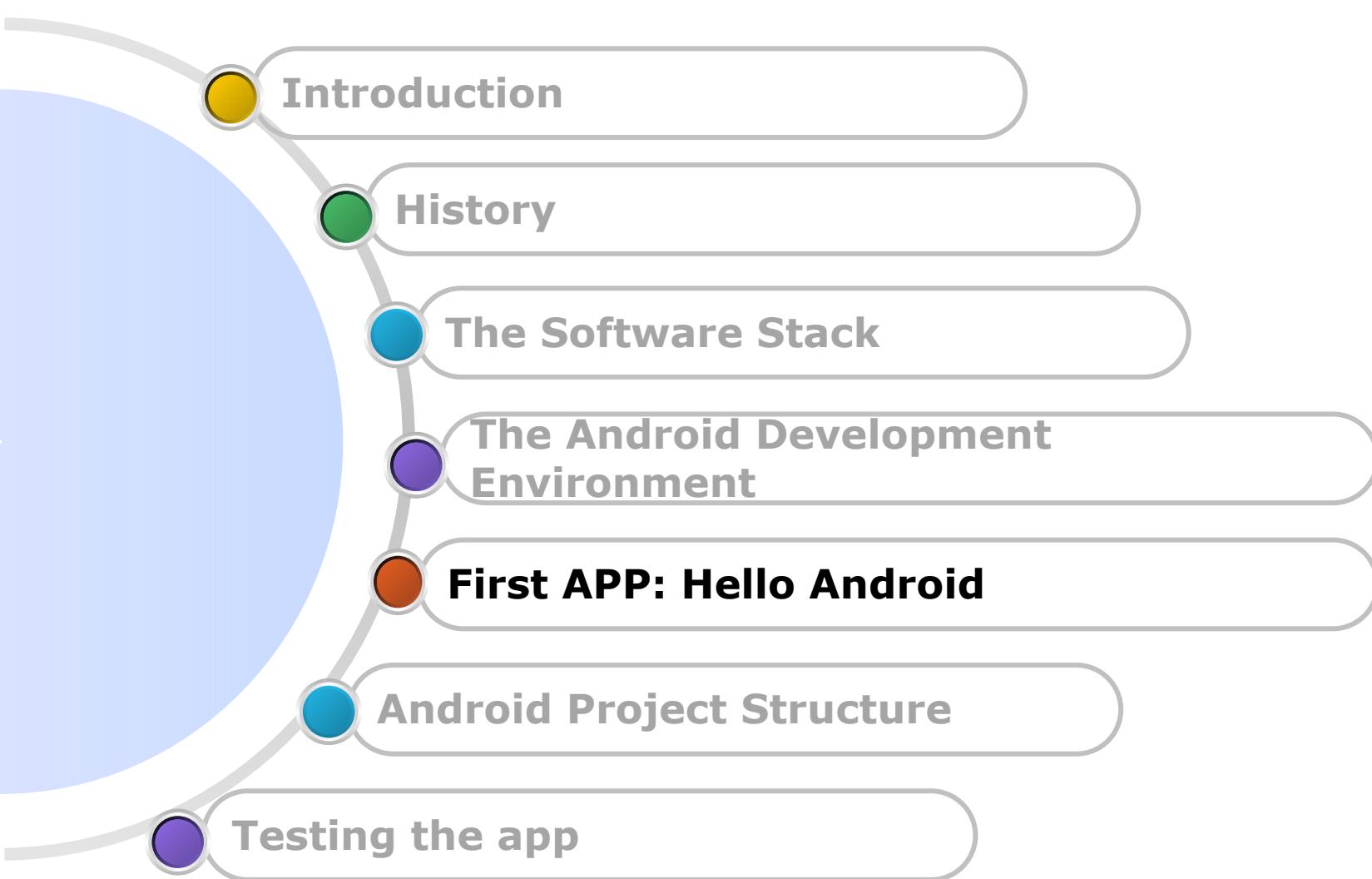
Maven es un gestor de dependencias para Java

## Que es Gradle?



- ❖ Herramienta para automatizar la construcción de nuestros proyectos: tareas de compilación, testing, empaquetado y el despliegue de los mismos.
- ❖ Es muy flexible para la configuración, pero además ya tiene armadas las tareas para la mayoría de los proyectos por default.
- ❖ Esta herramienta es usado por grandes proyecto “**Open Source**” como “**Spring**”, “**Hibernate**”, y “**Grails**”. (También lo usan empresas como “**LinkedIn**” para sus proyectos)
- ❖ **Gradle** usa un “**Domain Specific Language**” (**DSL**) basado en “**Groovy**” para declarar la formas de construir el proyecto.
- ❖ Para configurarlo simplemente podemos escribir código en **Groovy**, el cual en su sintaxis es muy similar a **Java**, lo que nos va a hacer sencillo expresar, las tareas que queremos que realice.

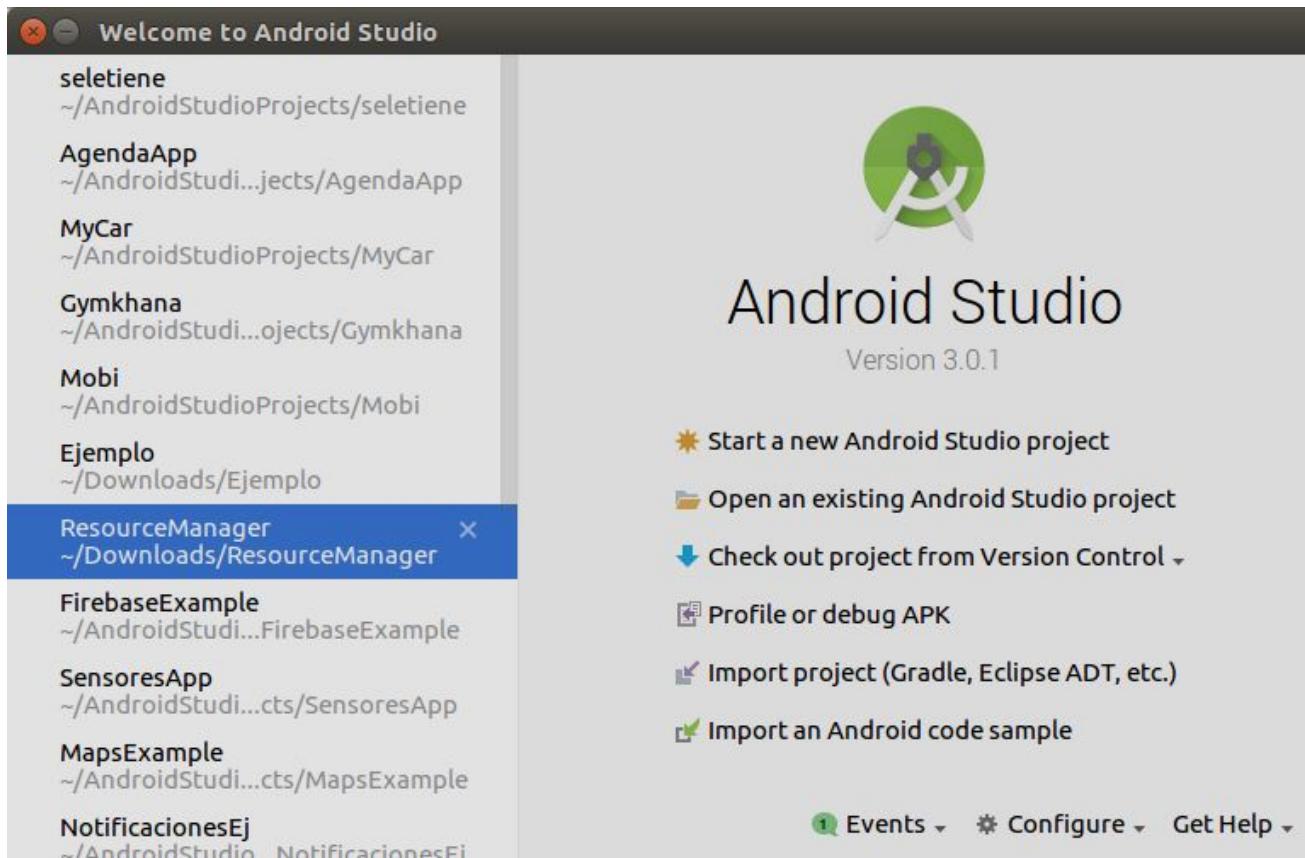
# Content



# Primer App: Hello Android

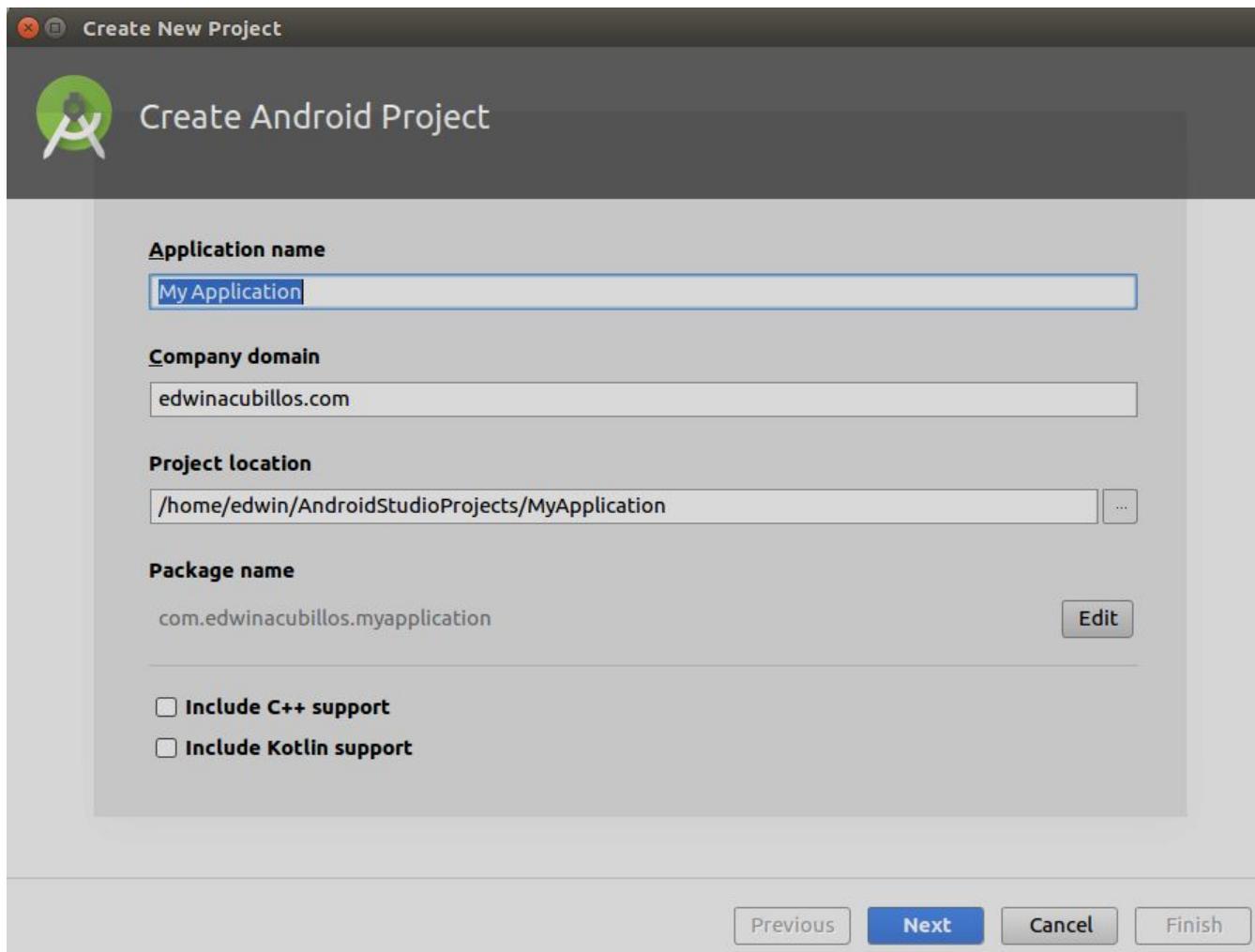
## Creación del proyecto

- ❖ Start a new Android Studio Project



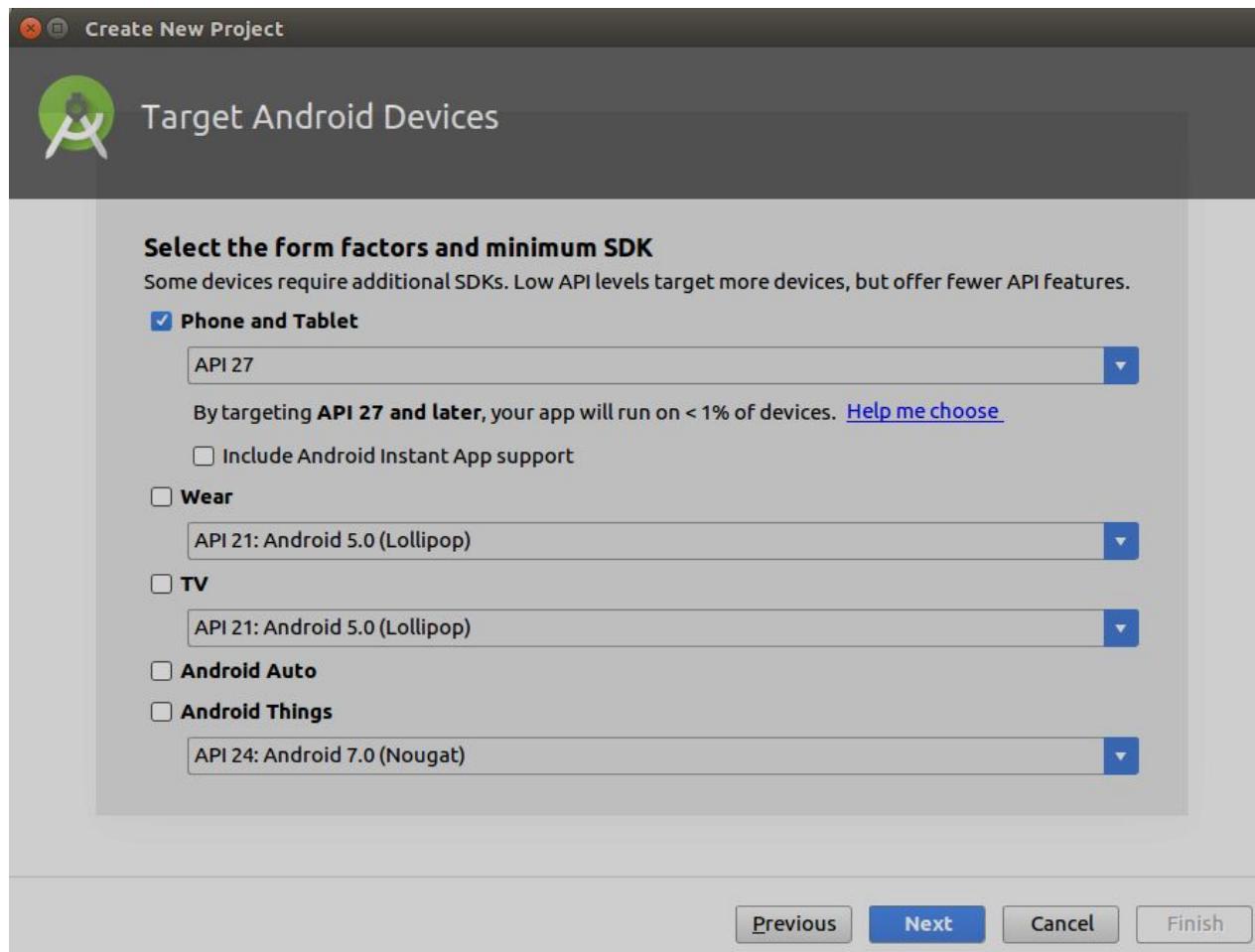
# Primer App: Hello Android

## ❖ Colocar nombre de la Aplicación



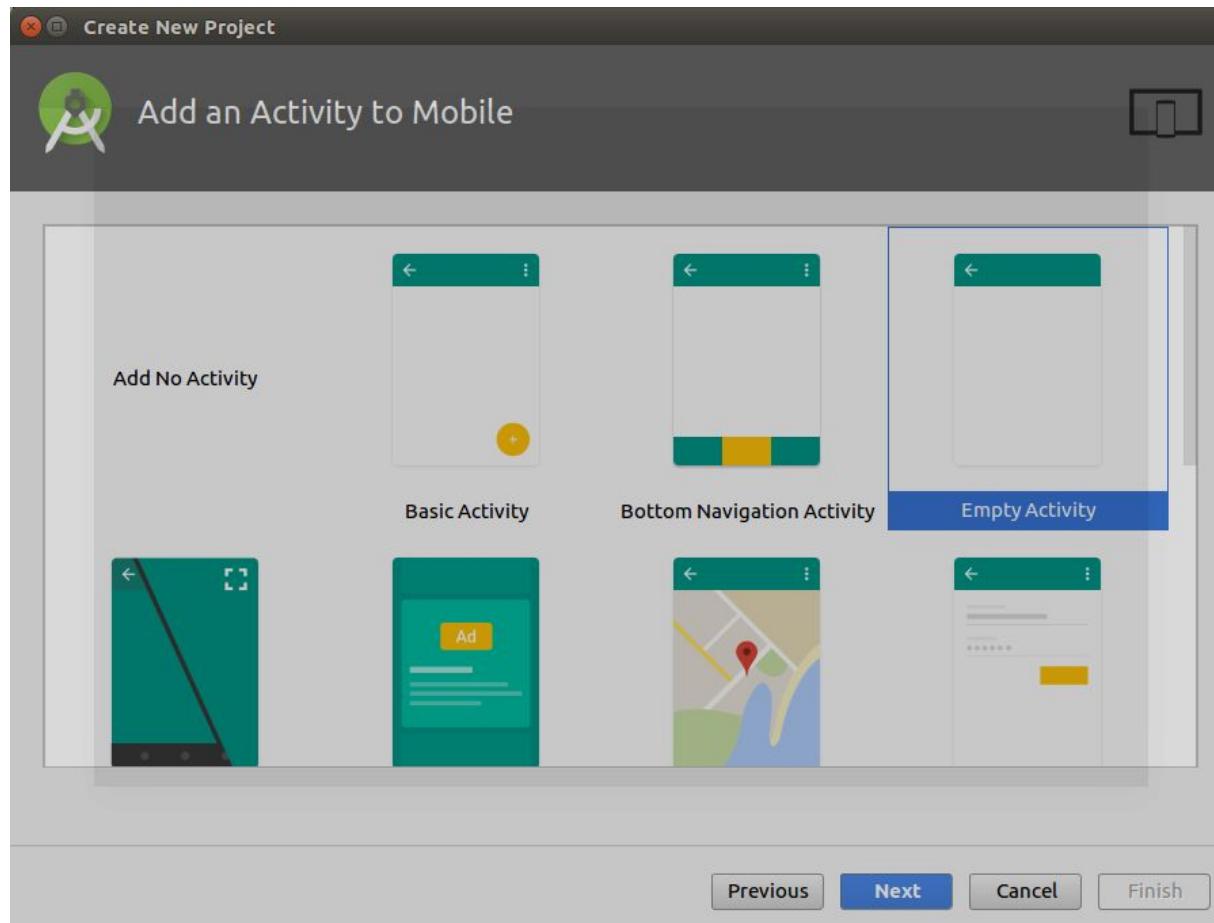
# Primer App: Hello Android

- ❖ Seleccionar la API para la cual vamos a trabajar



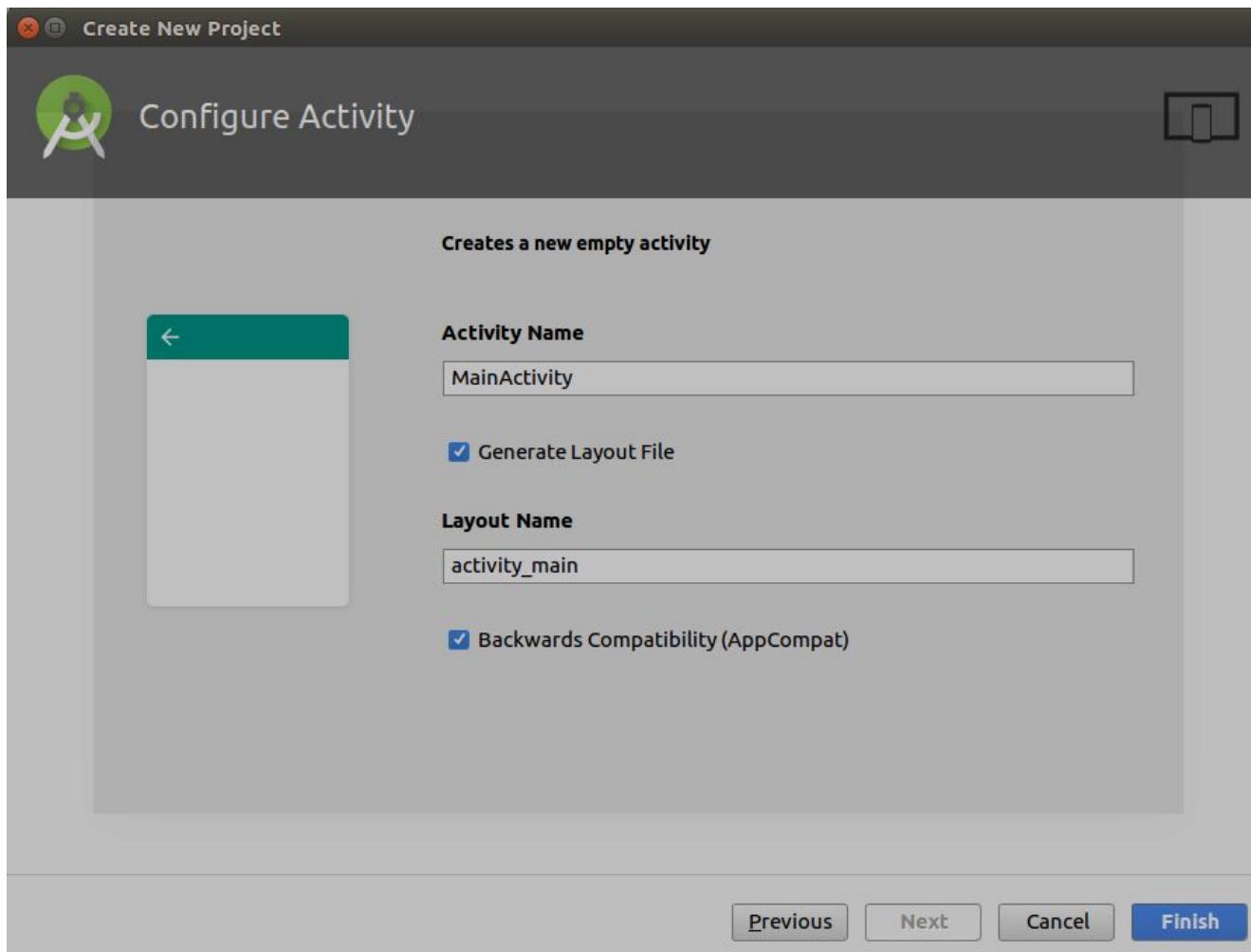
# Primer App: Hello Android

## ❖ Seleccionar Actividad en Blanco

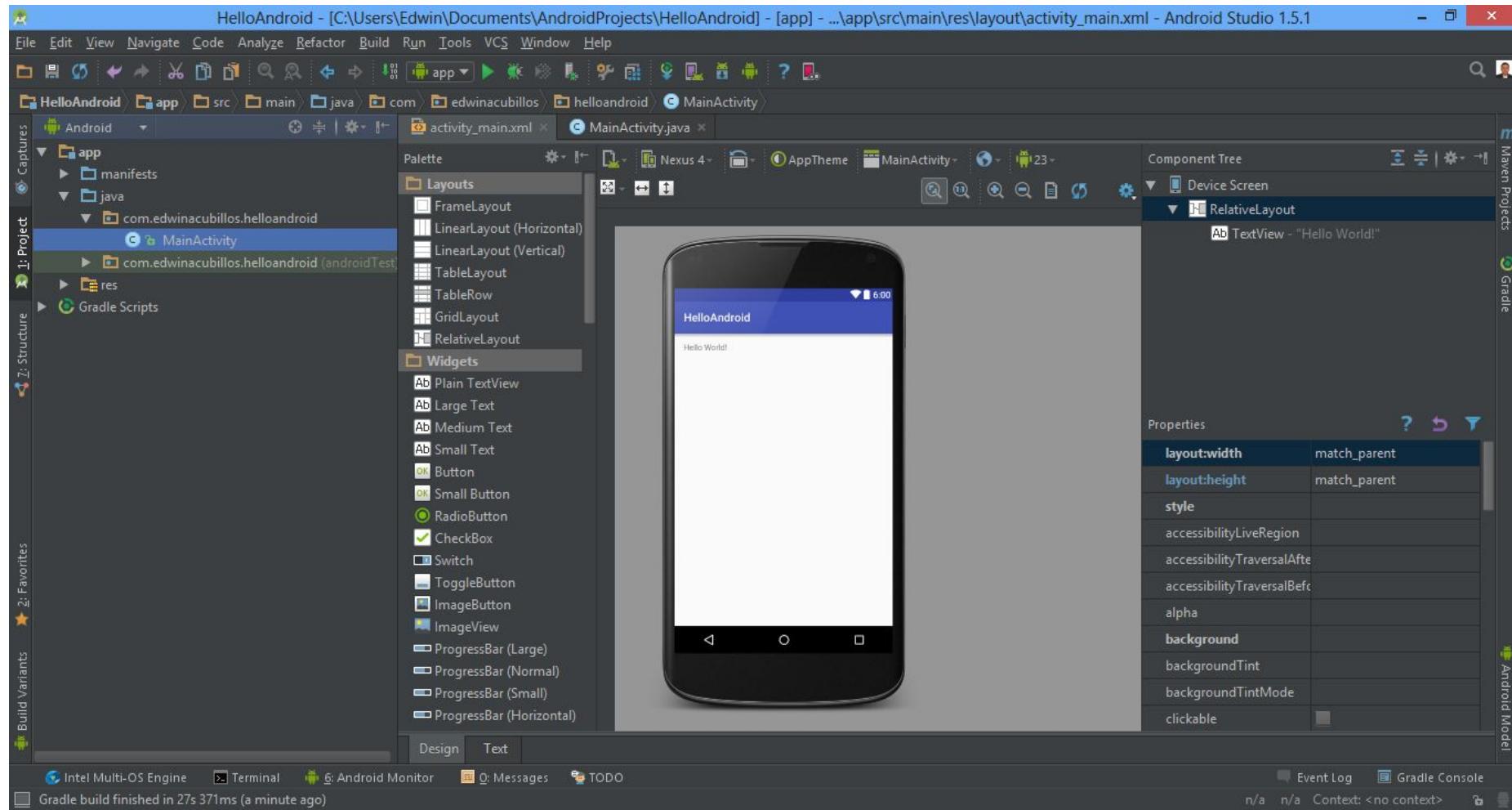


# Primer App: Hello Android

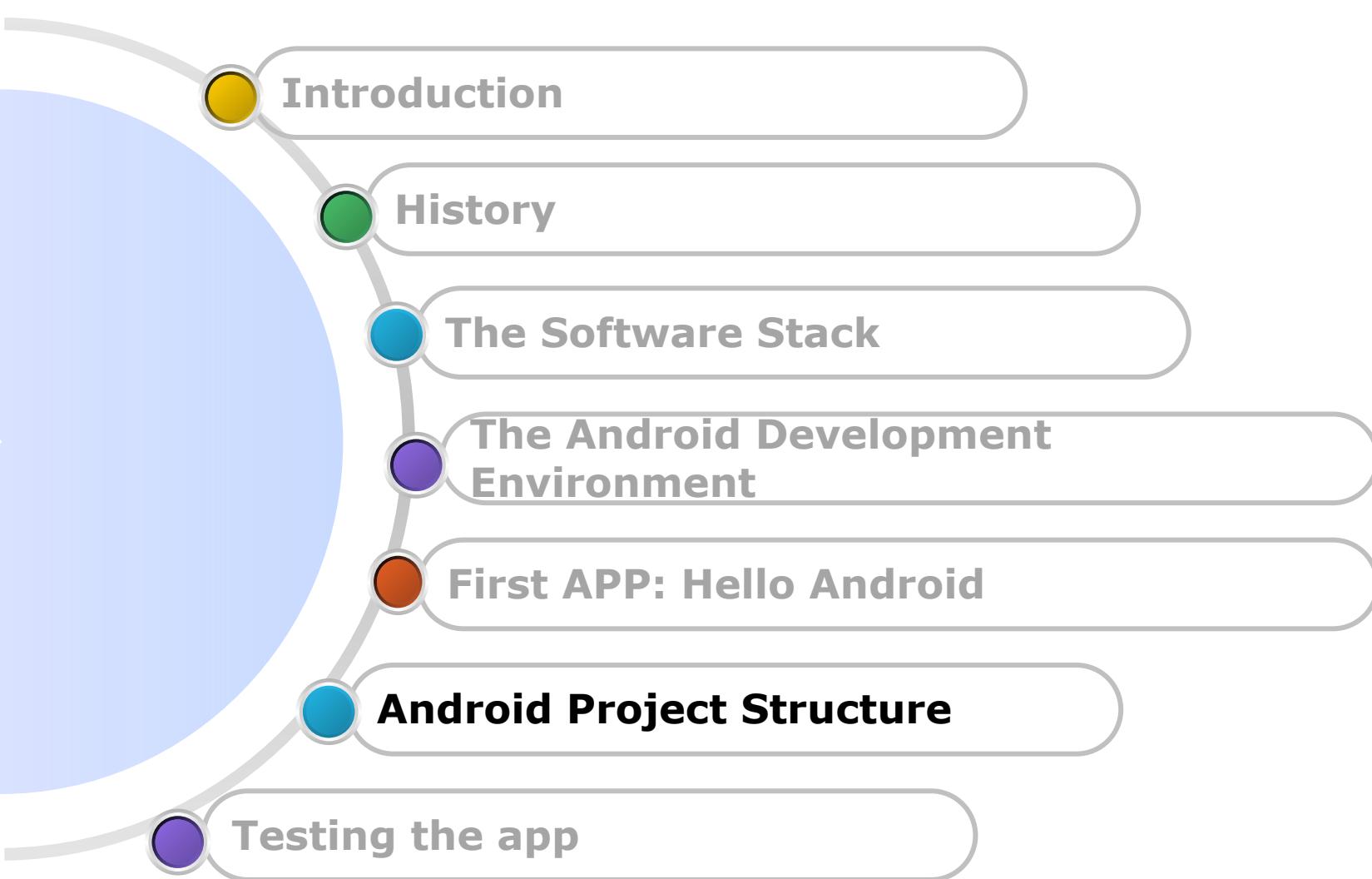
## ❖ Configurar nombres de la Actividad y Layout



# Primer App: Hello Android

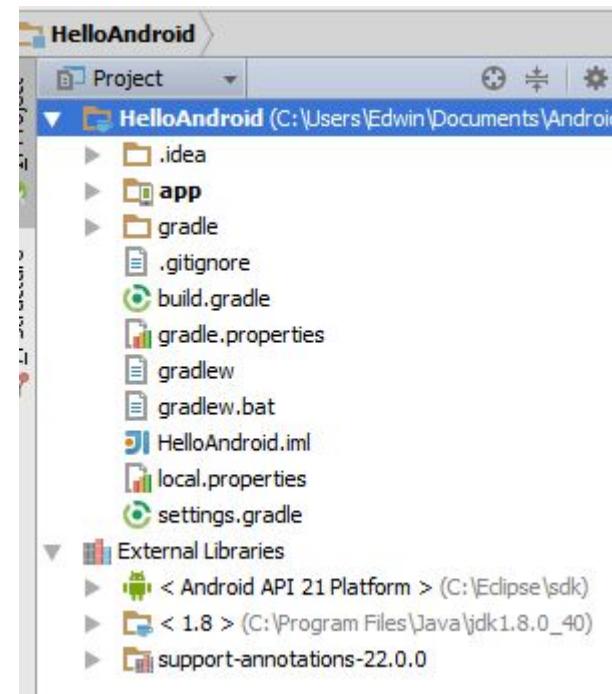
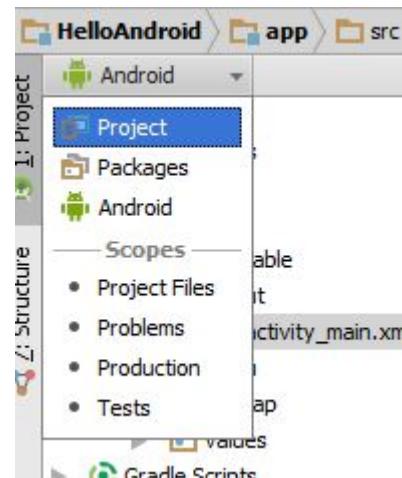
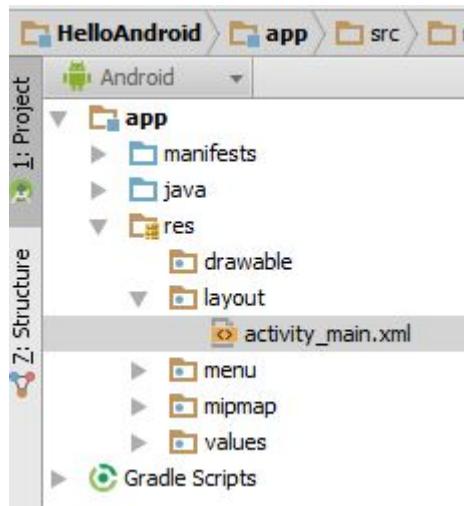


# Content

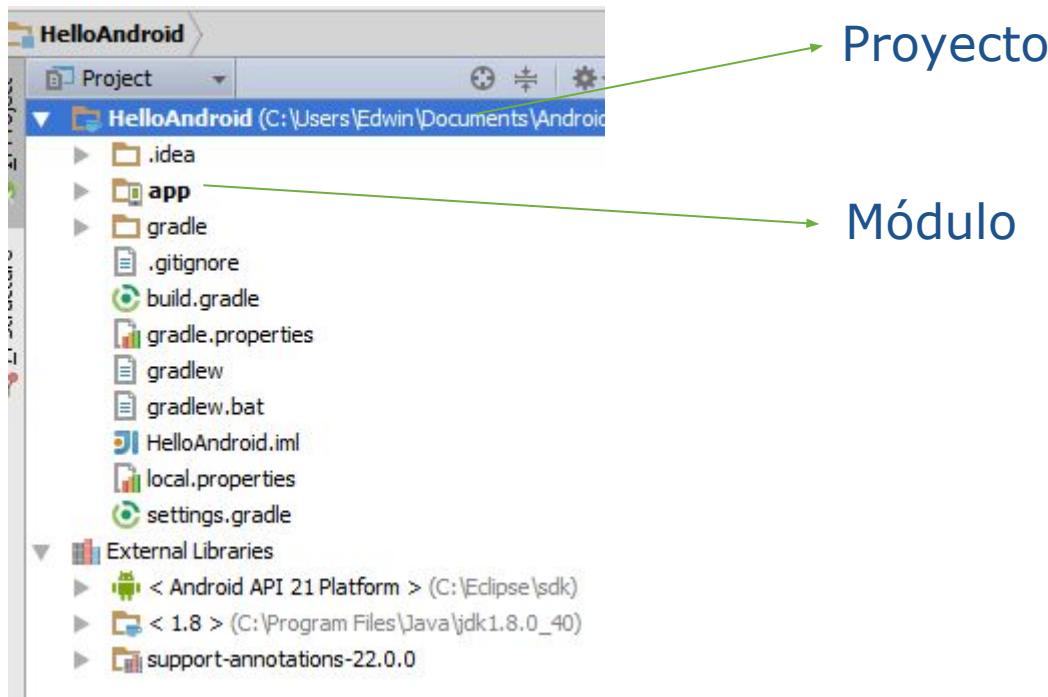


# Estructura de un proyecto Android

## ❖ Cambiar la vista del proyecto de Android a Project



# Estructura de un proyecto Android



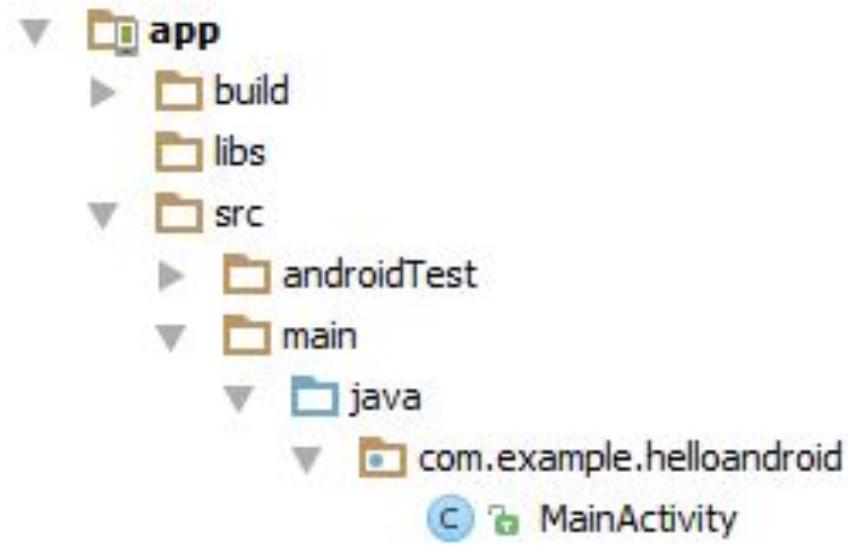
Proyecto

Módulo

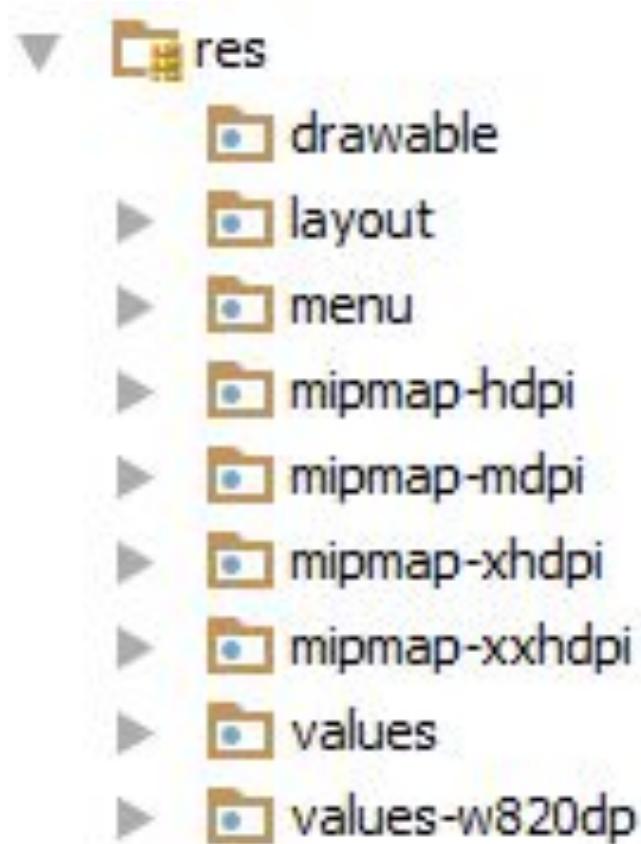
# Estructura de un proyecto Android

## /src/

- ❖ Contiene todo el código fuente de la aplicación, código de la interfaz gráfica, clases auxiliares, etc.
- ❖ Inicialmente, Android Studio creará por nosotros el código básico de la pantalla (Activity) principal de la aplicación.
- ❖ En este caso es MainActivity, y siempre bajo la estructura del paquete java definido.



# Estructura de un proyecto Android



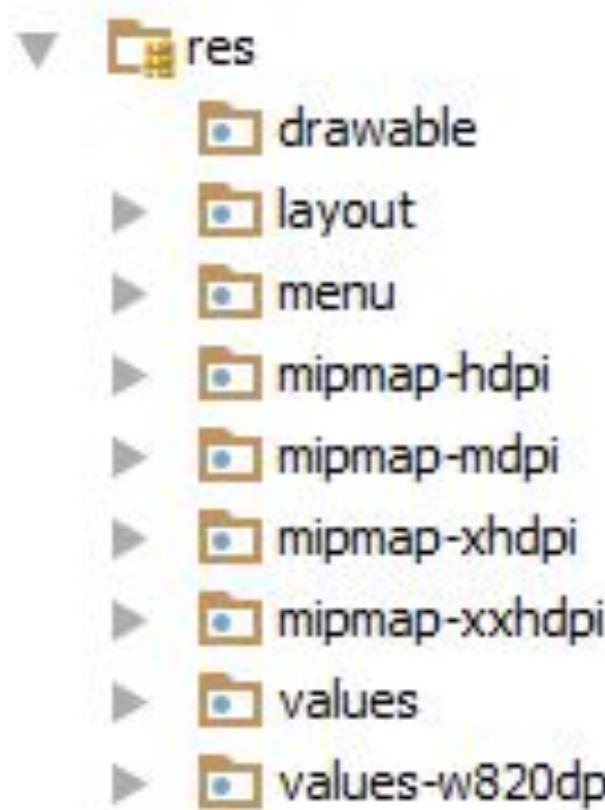
## /res

Contiene todos los ficheros de recursos necesarios para el proyecto: imágenes, vídeos, cadenas de texto, etc. Se divide en:

### /res/drawable/

- ❖ Contiene las imágenes (y otros elementos gráficos) usados en por la aplicación.
- ❖ Para definir diferentes recursos dependiendo de la resolución y densidad de la pantalla del dispositivo se suele dividir en varias subcarpetas:
  1. /drawable-hdpi (densidad alta)
  2. /drawable-ldpi (densidad baja)
  3. /drawable-mdpi (densidad media)
  4. /drawable-xhdpi (densidad muy alta)

# Estructura de un proyecto Android



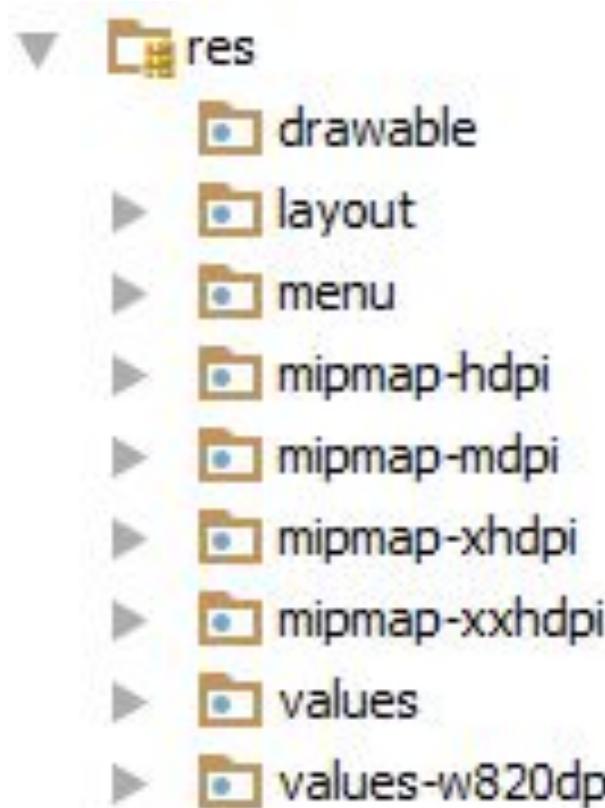
## /res/layout/

- ❖ Contiene los ficheros de definición XML de las diferentes pantallas de la interfaz gráfica.
- ❖ Para definir distintos *layouts* dependiendo de la orientación del dispositivo se puede dividir en dos subcarpetas:
  1. /layout (vertical)
  2. /layout-land (horizontal)

## /res/menu/

Contiene la definición XML de los menús de la aplicación.

# Estructura de un proyecto Android



## /res/values/

- ❖ Contiene otros ficheros XML de recursos de la aplicación, como por ejemplo cadenas de texto (*strings.xml*), estilos (*styles.xml*), colores (*colors.xml*), arrays de valores (*arrays.xml*), etc.

Se pueden definir nuevas carpetas como:

## /res/anim/

Contienen la definición de las animaciones utilizadas por la aplicación

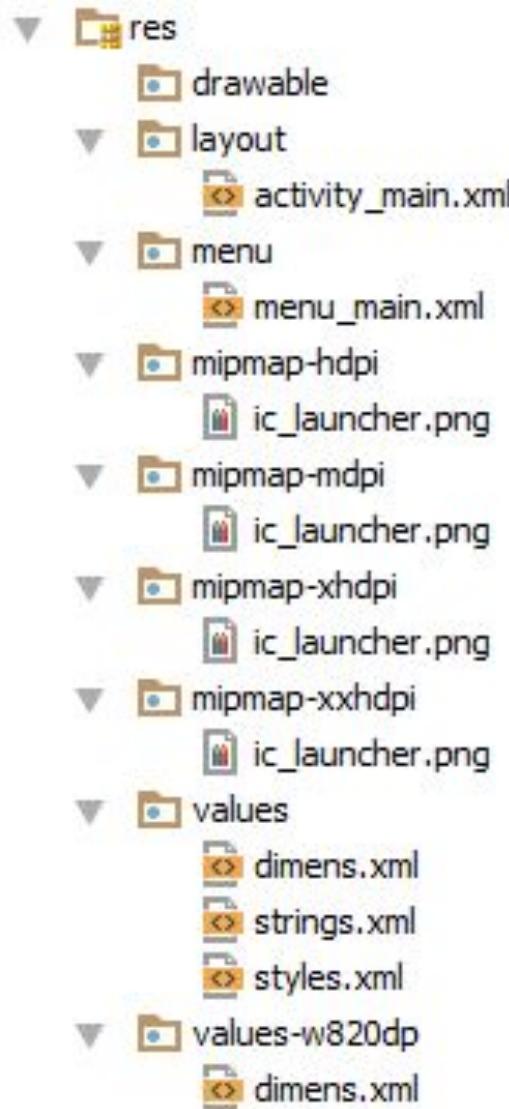
## /res/color/

Contiene ficheros XML de definición de colores según estado.

## /res/xml/

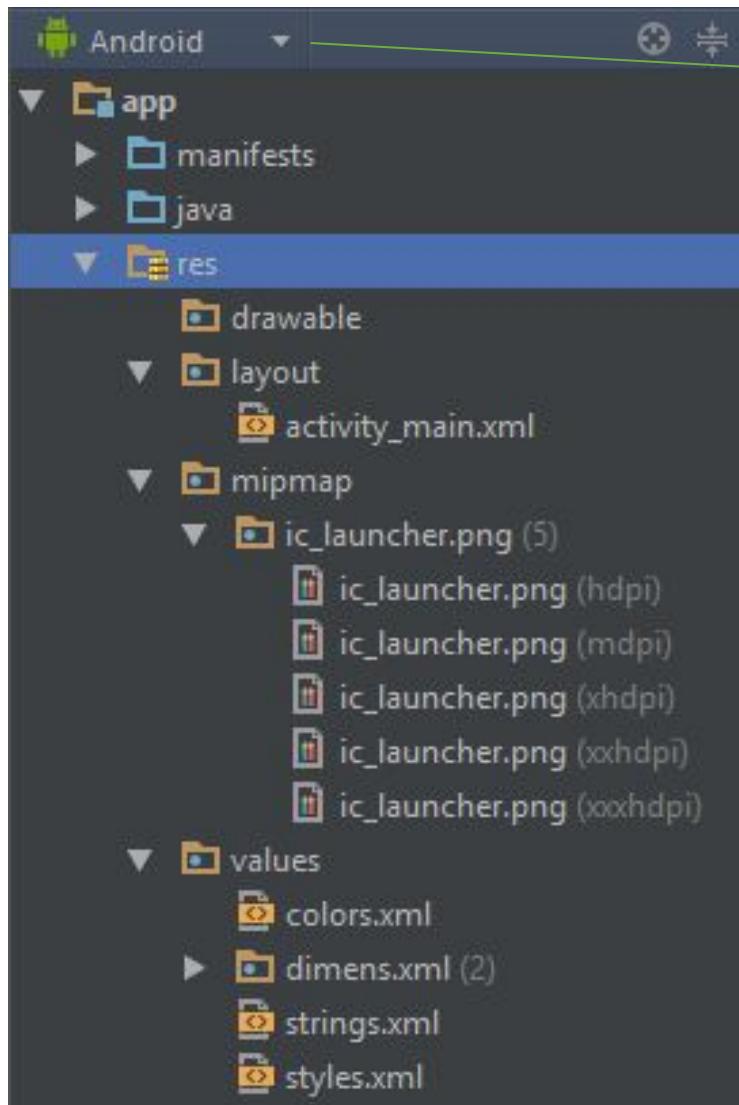
Contiene otros ficheros XML de datos utilizados por la aplicación

# Estructura de un proyecto Android



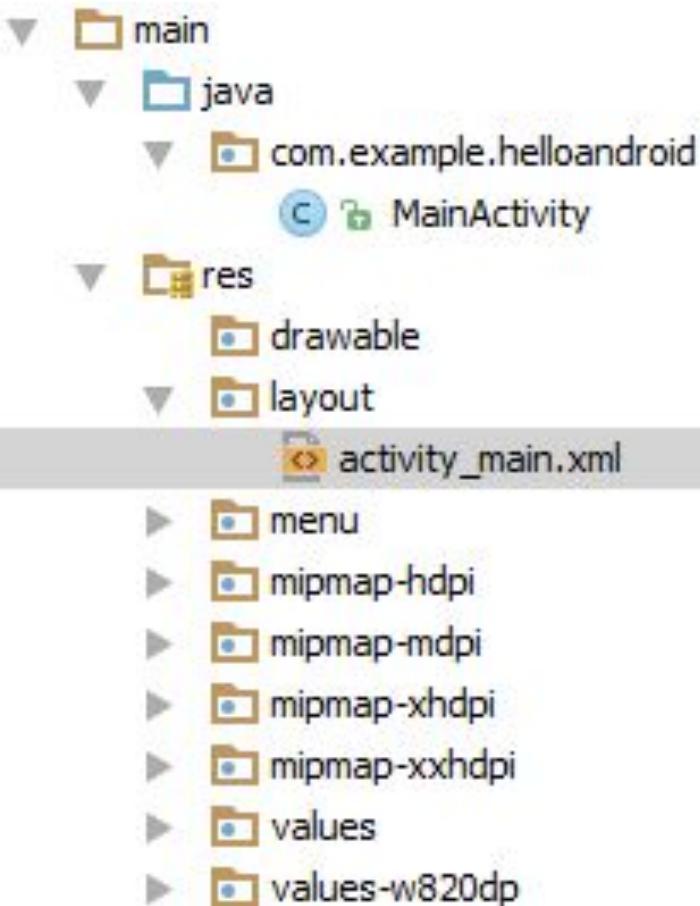
- ❖ Por defecto se crean estos recursos en la aplicación
- ❖ Las carpetas con sufijo adicional se refiere a recursos especiales para determinas versiones Ej:
  - values-w820dp: aplica a dispositivos mínimo con pantalla de 820 dp
- ❖ Mas Información

# Estructura de un proyecto Android



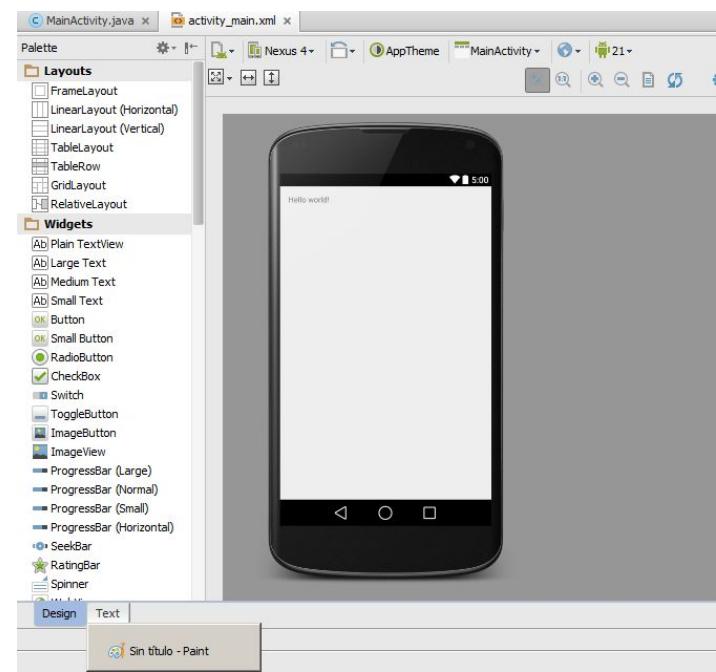
Cambia la vista

# Estructura de un proyecto Android

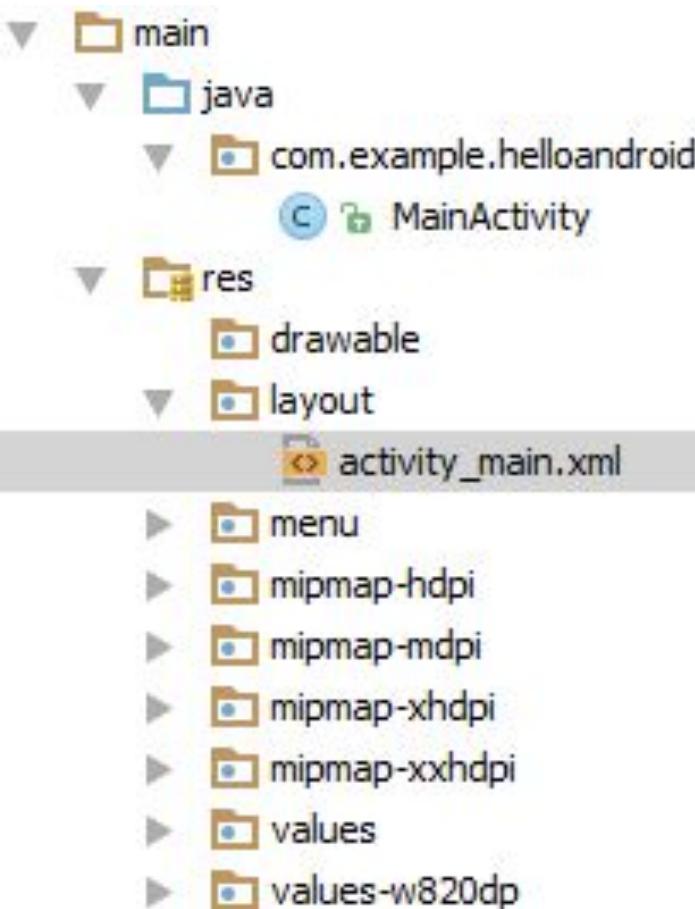


## ***Activity\_main.xml***

- ❖ Contiene la definición de la interfaz gráfica de la pantalla principal de la aplicación.
- ❖ Si hacemos doble clic sobre el fichero nos mostrará esta interfaz en su editor gráfico (tipo arrastrar y soltar)



# Estructura de un proyecto Android



## ***Activity\_main.xml***

La interfaz tiene asociado un archivo .XML

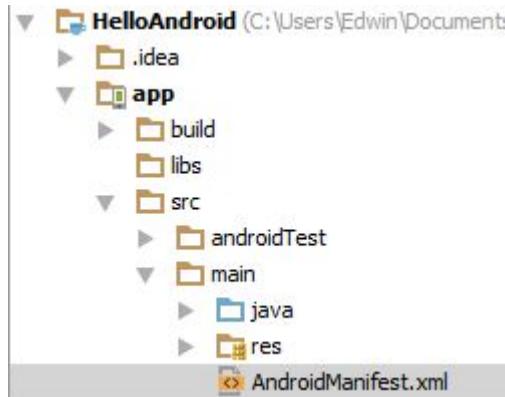
The screenshot shows an IDE interface with two tabs open:

- MainActivity.java**: The Java code for the main activity.
- activity\_main.xml**: The XML layout file for the main activity, which is circled in green.

The XML code in **activity\_main.xml** is:

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
    xmlns:tools="http://schemas.android.com/tools" android:layout_width="match_parent"  
    android:layout_height="match_parent" android:paddingLeft="16dp"  
    android:paddingRight="16dp"  
    android:paddingTop="16dp"  
    android:paddingBottom="16dp" tools:context=".MainActivity">  
  
    <TextView android:text="Hello world!" android:layout_width="wrap_content"  
        android:layout_height="wrap_content" />  
  
</RelativeLayout>
```

# Estructura de un proyecto Android



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.helloandroid" >

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="Hello Android"
        android:theme="@style/AppTheme" >
        <activity
            android:name=".MainActivity"
            android:label="Hello Android" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

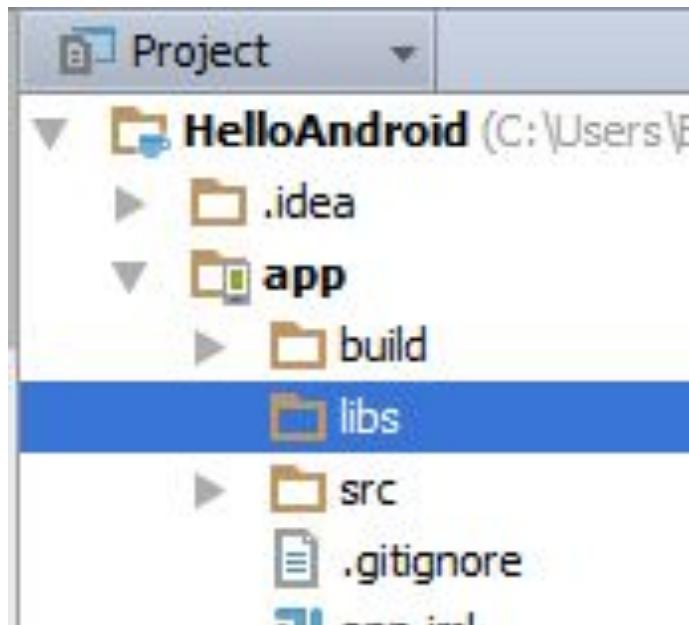
## AndroidManifest.xml

- ❖ Contiene la definición en XML de los aspectos principales de la aplicación:
- ❖ Su identificación:
  - nombre,
  - versión,
  - Icono, etc
- ❖ Componentes
  - pantallas,
  - mensajes, etc
- ❖ las librerías auxiliares utilizadas, o
- ❖ los permisos necesarios para su ejecución

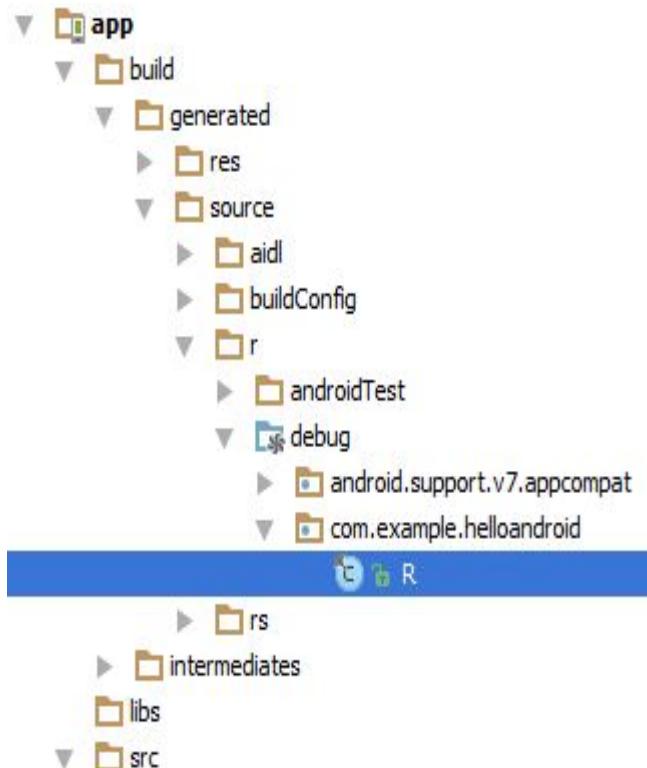
# Estructura de un proyecto Android

## /libs/

- ❖ Contendrá las librerías auxiliares, normalmente en formato “.jar” que utilicemos en nuestra aplicación Android.



# Estructura de un proyecto Android



- ❖ Contiene una serie de elementos de código **generados de forma automática** al compilar el proyecto.
- ❖ Cada vez que generamos nuestro proyecto, la maquinaria de compilación de Android genera por nosotros una serie de ficheros fuente java dirigidos al control de los recursos de la aplicación.
- ❖ Importante: dado que estos ficheros se generan automáticamente tras cada compilación del proyecto es importante que no se modifiquen manualmente bajo ninguna circunstancia.

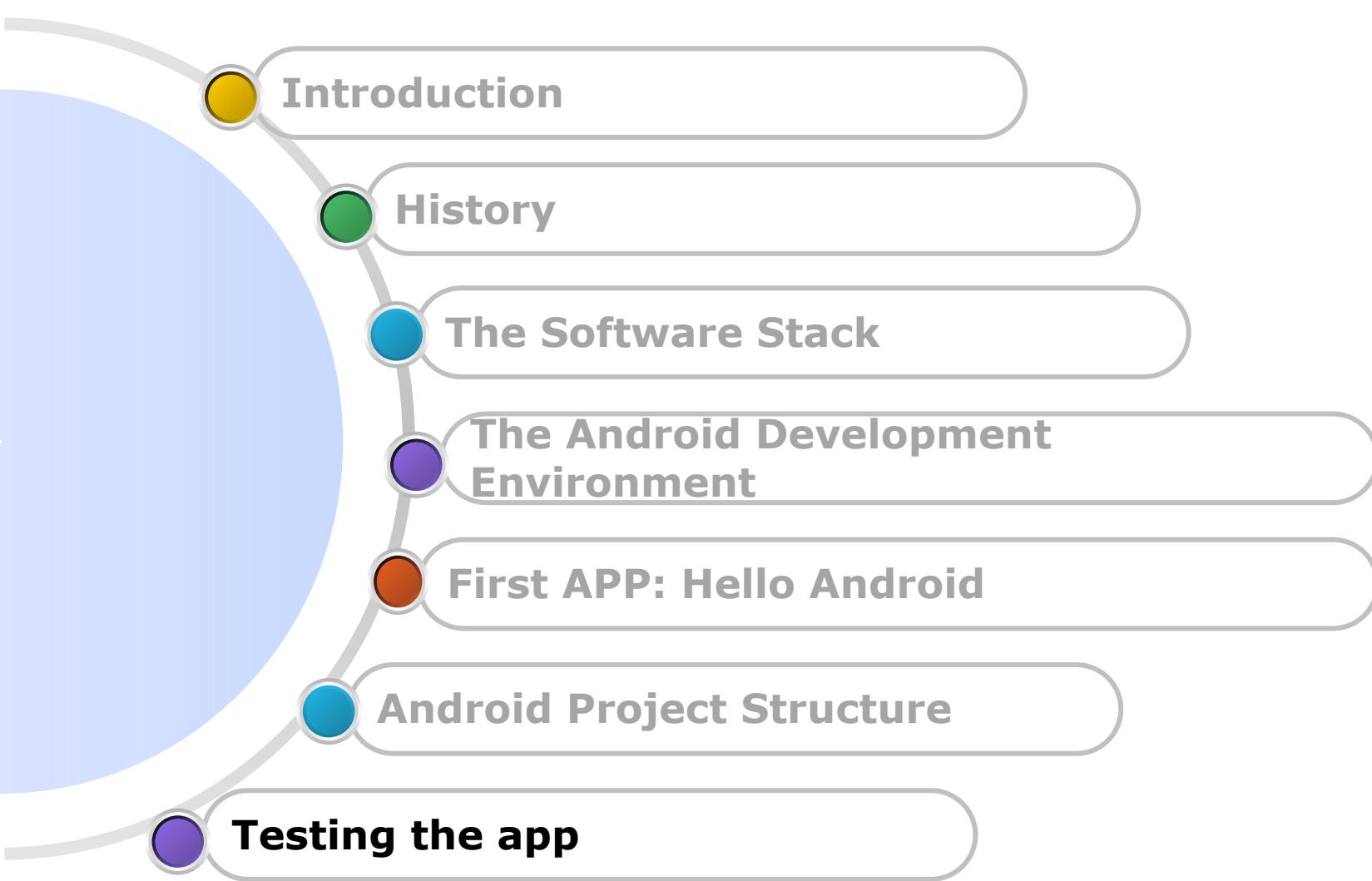
# Estructura de un proyecto Android

The screenshot shows a Java code editor with three tabs: `MainActivity.java`, `activity_main.xml`, and `R.java`. The `R.java` tab is currently selected. The code in `R.java` is a generated class containing static final int variables representing resource IDs. A yellow highlighter is used to mark several sections of the code, including the entire class definition and specific resource ID declarations.

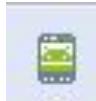
```
public static final int support_simple_spinner_dropdown_item=0x7f040017;
}
public static final class menu {
    public static final int menu_main=0x7f0d0000;
}
public static final class mipmap {
    public static final int ic_launcher=0x7f030000;
}
public static final class string {
    public static final int abc_action_bar_home_description=0x7f0b0000;
    public static final int abc_action_bar_home_description_format=0x7f0b0001;
    public static final int abc_action_bar_home_subtitle_description_format=0x7f0b0002;
    public static final int abc_action_bar_up_description=0x7f0b0003;
    public static final int abc_action_menu_overflow_description=0x7f0b0004;
    public static final int abc_action_mode_done=0x7f0b0005;
    public static final int abc_activity_chooser_view_see_all=0x7f0b0006;
    public static final int abc_activitychooserview_choose_application=0x7f0b0007;
    public static final int abc_searchview_description_clear=0x7f0b0008;
    public static final int abc_searchview_description_query=0x7f0b0009;
    public static final int abc_searchview_description_search=0x7f0b000a;
    public static final int abc_searchview_description_submit=0x7f0b000b;
    public static final int abc_searchview_description_voice=0x7f0b000c;
    public static final int abc_shareactionprovider_share_with=0x7f0b000d;
    public static final int abc_shareactionprovider_share_with_application=0x7f0b000e;
    public static final int abc_toolbar_collapse_description=0x7f0b000f;
    public static final int action_settings=0x7f0b0010;
    public static final int app_name=0x7f0b0011;
    public static final int hello_world=0x7f0b0012;
}
public static final class style {
    public static final int Animation_DropCompat_DropDownItem=0x7f0c0000+
```

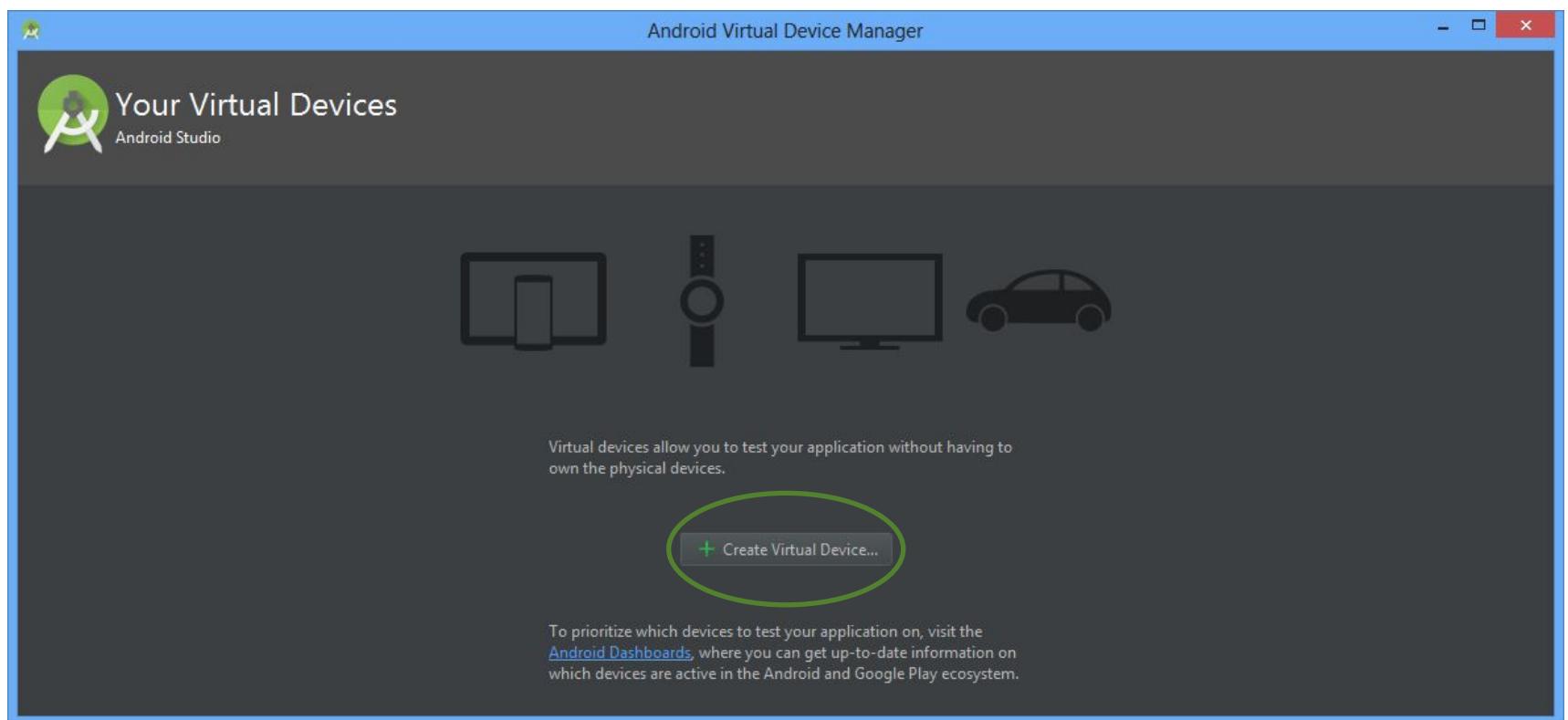
- ❖ `R.java`: se define la clase `R`.
- ❖ Contendrá una serie de constantes con los ID de todos los recursos de la aplicación
- ❖ carpeta `/res/`
- ❖ Permite acceder fácilmente a estos recursos desde nuestro código a través de este dato

# Content



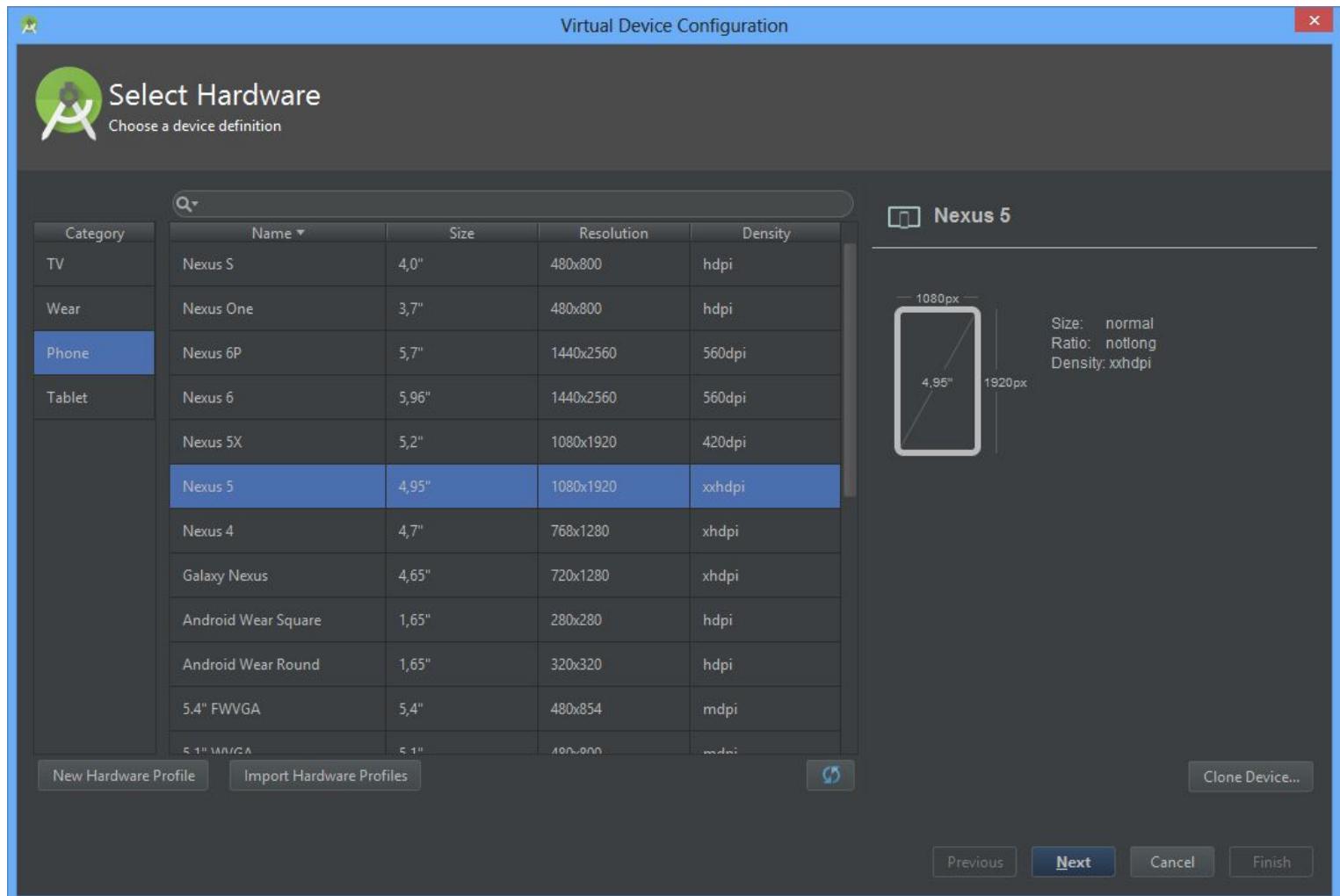
# Probando la App

- ❖ Para crear un AVD (Android Virtual Device)
- ❖ *Tools -> Android -> AVD Manager*
- ❖ O Click en el icono 



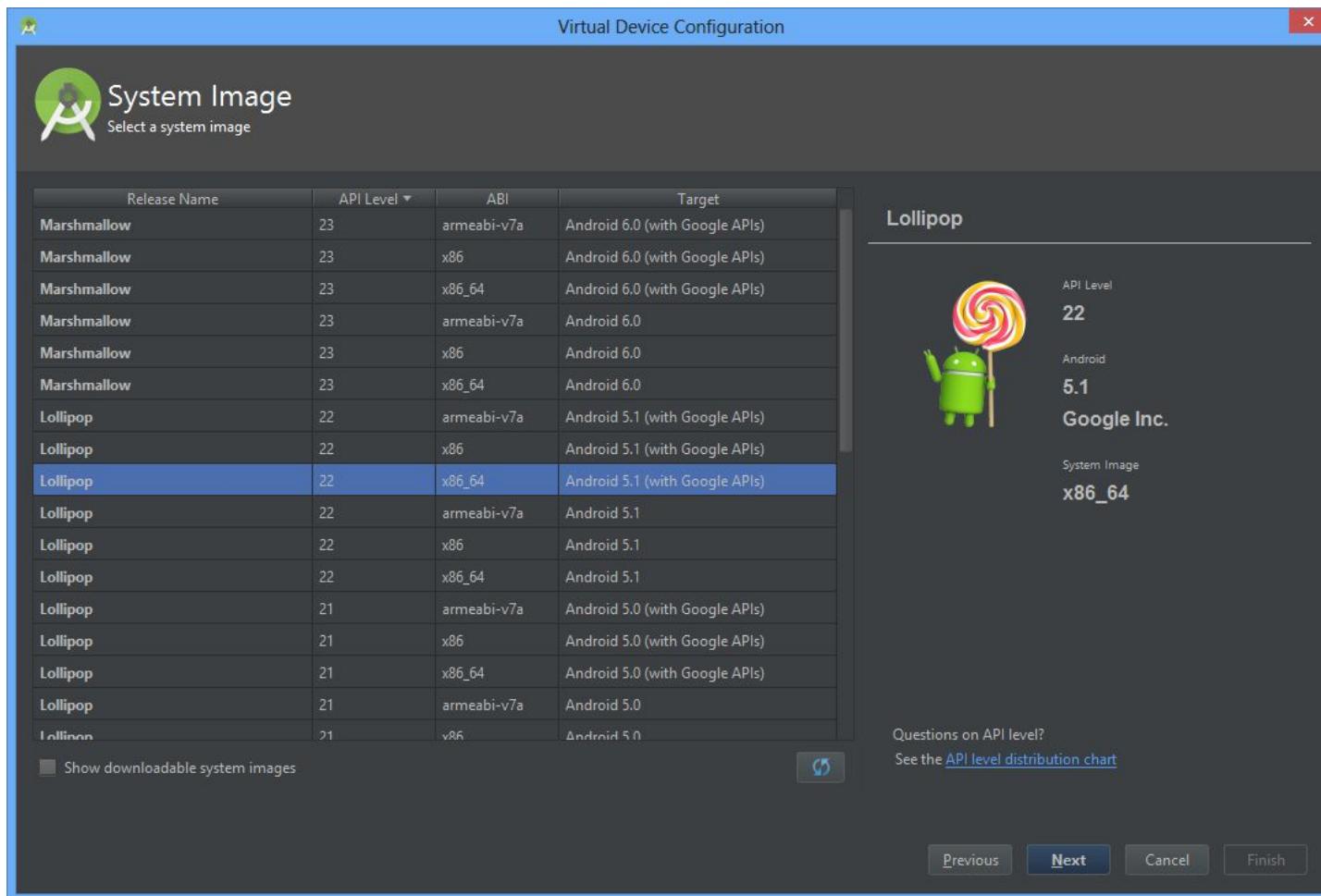
# Probando la App

## ❖ Seleccionar el Hardware



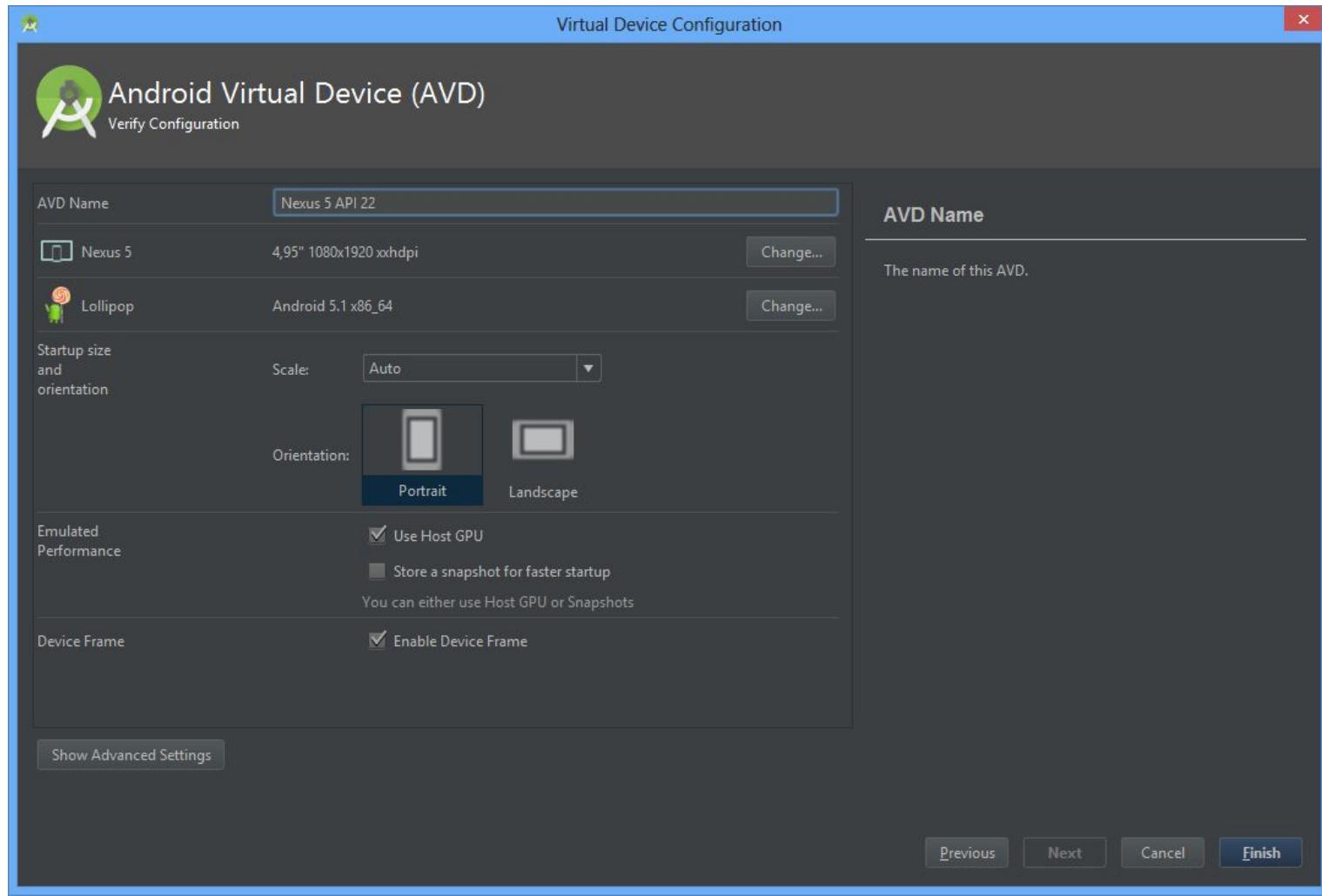
# Probando la App

## ❖ Seleccionar la Imagen del SO



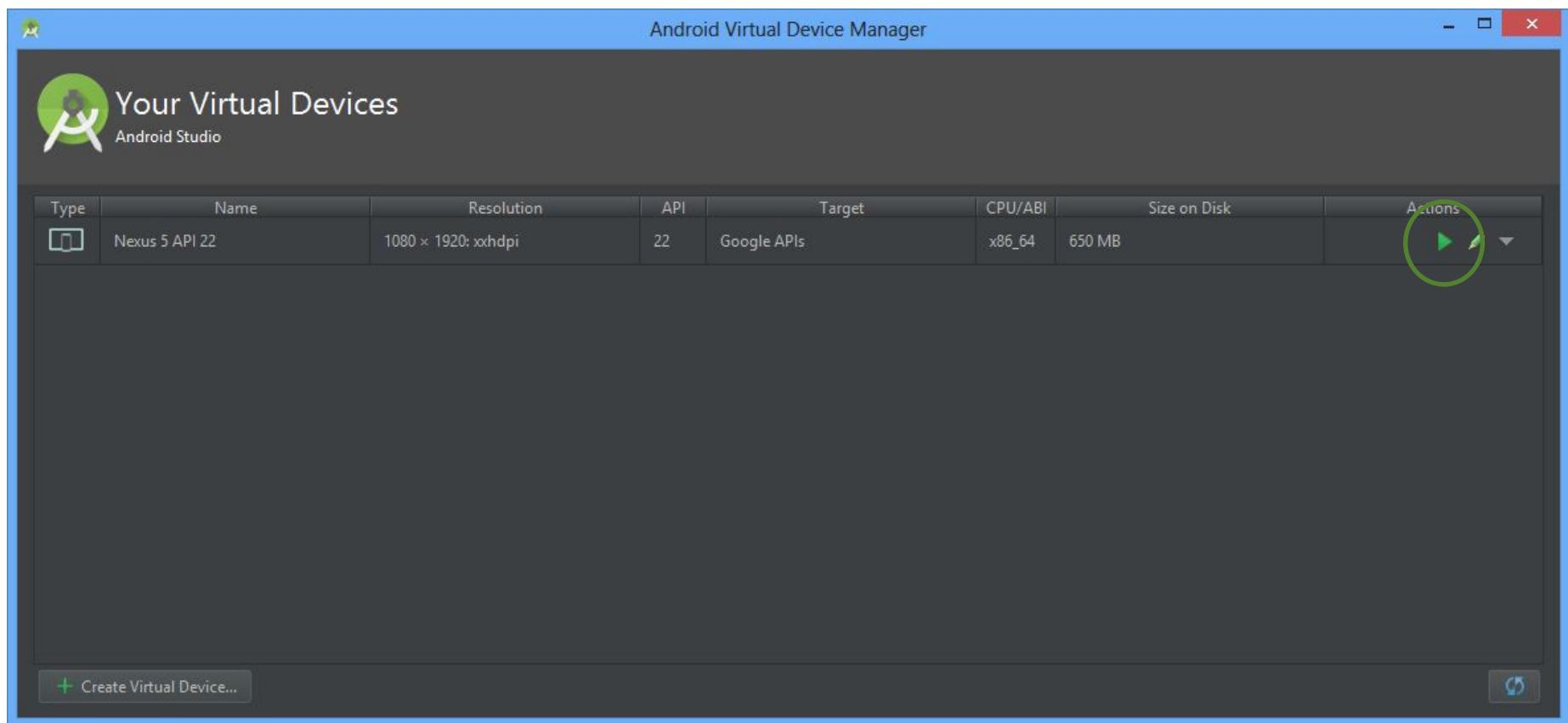
# Probando la App

## ❖ Seleccionar nombre y configurar modo



# Probando la App

## ❖ Iniciar la máquina



# Probando la App

- ❖ El emulador de Eclipse es muy lento
- ❖ Una alternativa es Genymotion version Free
- ❖ <http://www.genymotion.com/>

The screenshot shows the homepage of the Genymotion website. At the top, there's a navigation bar with links for 'Features', 'Docs', 'Pricing', 'Contact', and 'More'. A 'Sign in' button is located in the top right corner. The main visual is a photograph of several colorful Android robot figurines (known as Genymon) standing around a small fire. Overlaid on the image is the text 'Build. Test. Flaunt. Support.' followed by 'with the world's leading Android Emulator'. A pink 'Choose a plan' button is positioned in the lower-left area of the image. At the bottom of the page, there's a footer with a cookie consent message, social media links (Twitter, Yelp, Spotify), and logos for Samsung and Evernote, along with an 'I Agree' button.

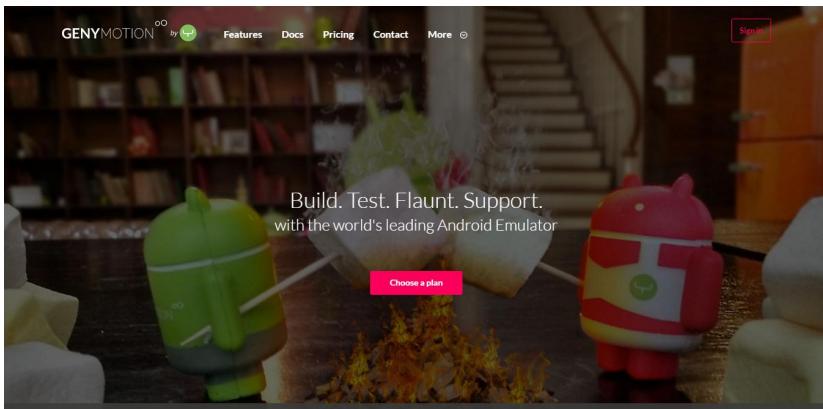
We are using cookies to provide statistics that help us give you the best experience of our site. By continuing to use the site you are agreeing to our use of cookies. You can find our [Privacy Statement](#).

twitter yelp Spotify SAMSUNG EVERNOTE

I Agree

# Probando la App

❖ Click en Sign In, luego en crear una cuenta



Download Genymotion

The download section of the Genymotion website. It offers two options: "with VirtualBox:" which is 143.44 MB and "without VirtualBox:" which is 34.86 MB. Both options are highlighted with a green oval. Below these are links for "How to register my license".

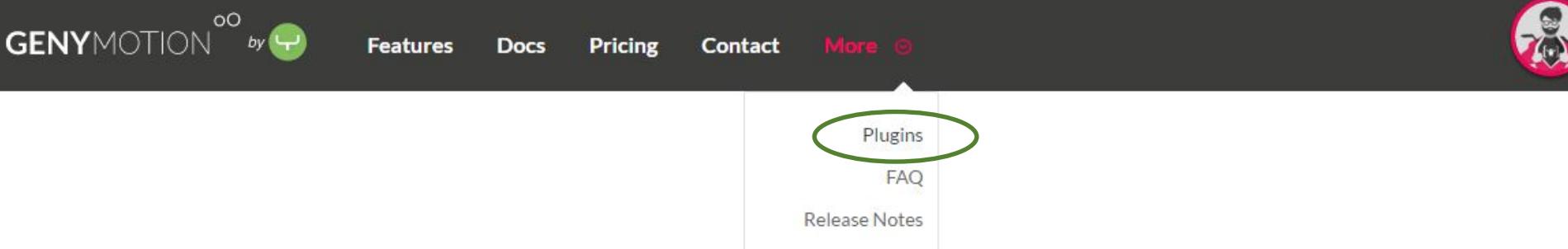
The system requirements section of the Genymotion website. It starts with a Windows logo and the heading "System Requirements". It lists the following requirements: Microsoft Windows 7, 8/8.1, 10 (32/64 bit), 64 bit CPU, with VT-x or AMD-V capability, enabled in BIOS settings, Recent and dedicated GPU, 400 MB disk space, and 2GB RAM.

» Seleccionar Plan Individual -> GetStarted

The pricing section of the Genymotion website. It shows two tabs: "Individual" (selected) and "Enterprise". Below are two plans: "BASIC" and "INDIE". The "BASIC" plan is described as "Great for start" and "Free for personal use only", with a "Get started" button. The "INDIE" plan is described as "Independent developers and small businesses" and has a price of "from \$136/year per business < 2 employees", with a "Get started" button.

# Probando la App

## ❖ También existe un plugin para Android Studio



The screenshot shows the official Genymotion website. At the top, there's a navigation bar with links for Features, Docs, Pricing, Contact, and More. Below this, a dropdown menu from the 'More' link is open, showing options for Plugins, FAQ, and Release Notes. The 'Plugins' option is highlighted with a red oval. On the left side of the page, there's a screenshot of the Android Studio interface. A magnifying glass highlights the Genymotion toolbar icon, which is a red circle with a white 'G' and a play button symbol. The Android Studio interface shows Java code for testing battery levels.



Android Studio / IntelliJ IDEA

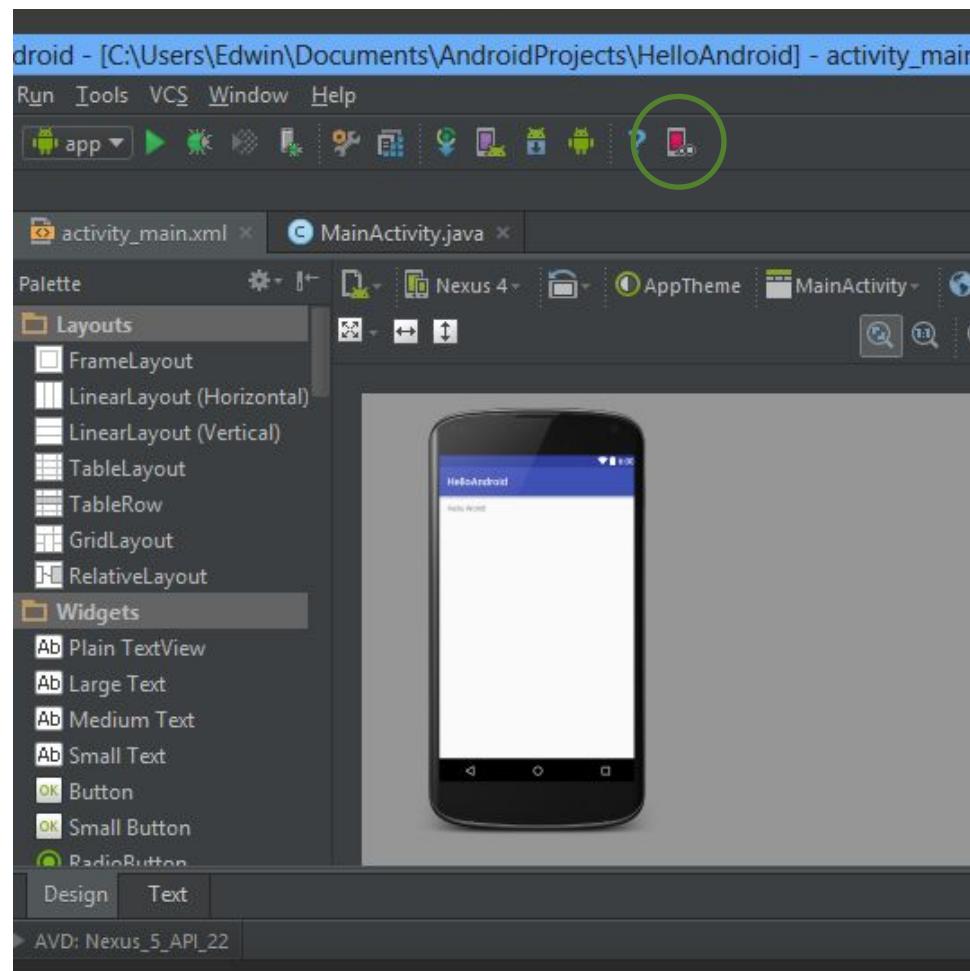
To install the Genymotion plugin for Android Studio

1. In Android Studio, go to **File/Settings** (for Windows and Linux) or to **Android Studio/Preferences** (for Mac OS X)
2. Select **Plugins** and click **Browse Repositories**.
3. Right-click on **Genymotion** and click **Download and install**. To see Genymotion plugin icon, display the toolbar by clicking **View > Toolbar**.

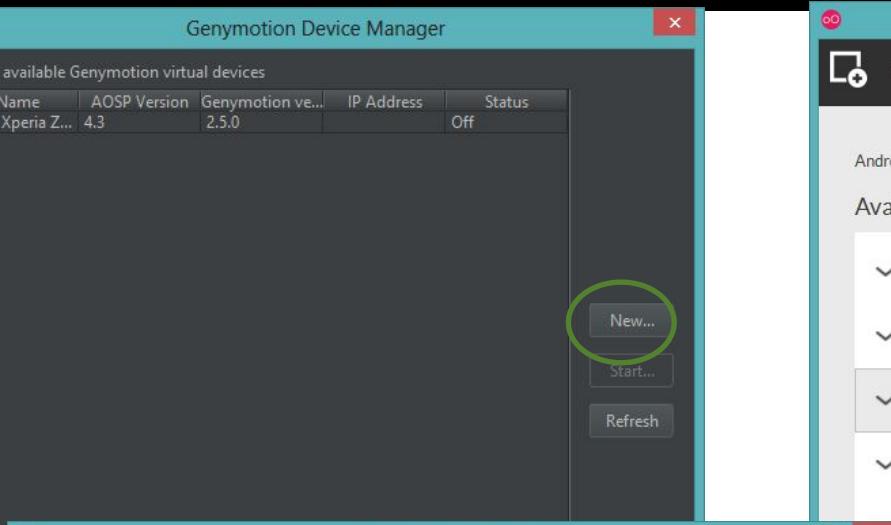
**!** To use this plugin, Genymotion must be installed on your system.  
Or download and install manually the plugin [genymotion-idea-plugin.jar](#) (not recommended)

# Probando la App

- ❖ Click en el Plugin y seguir los pasos para crear la máquina



# Probando la App



Virtual device creation wizard

Create a new virtual device

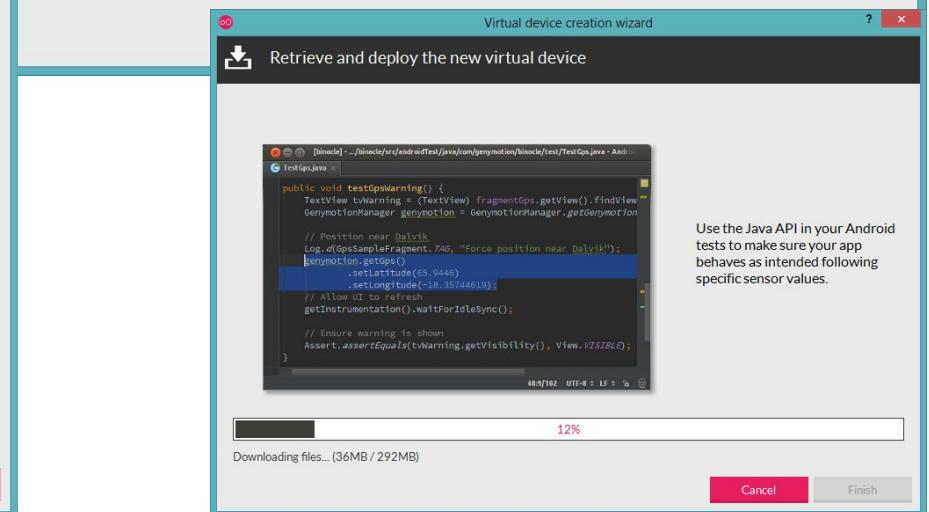
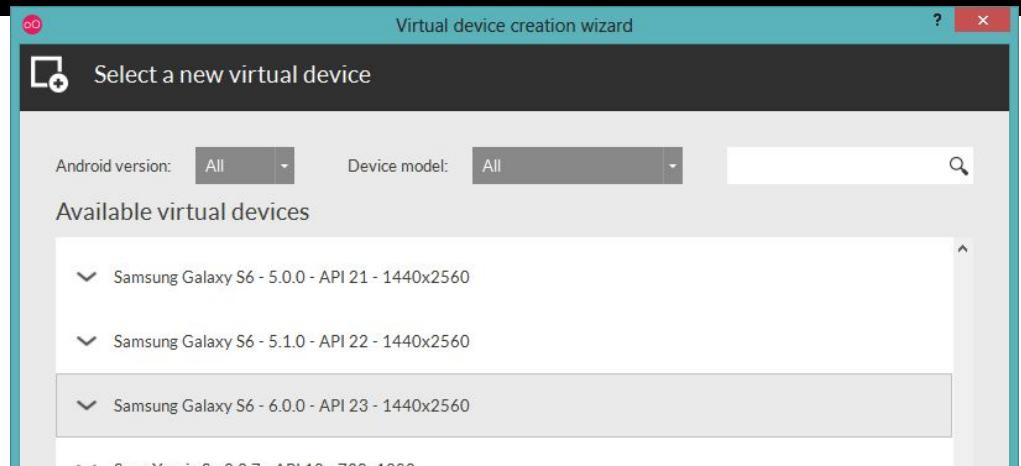
Virtual device name  
Samsung Galaxy S6 - 6.0.0 - API 23 - 1440x2560

Please check the virtual device properties before deployment

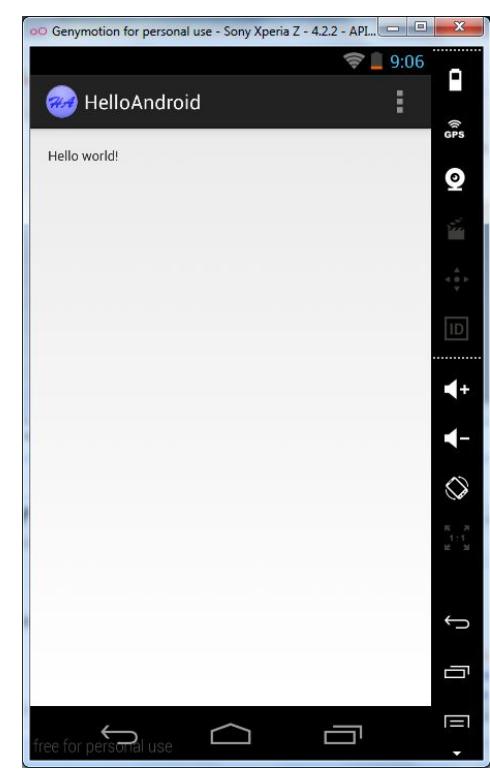
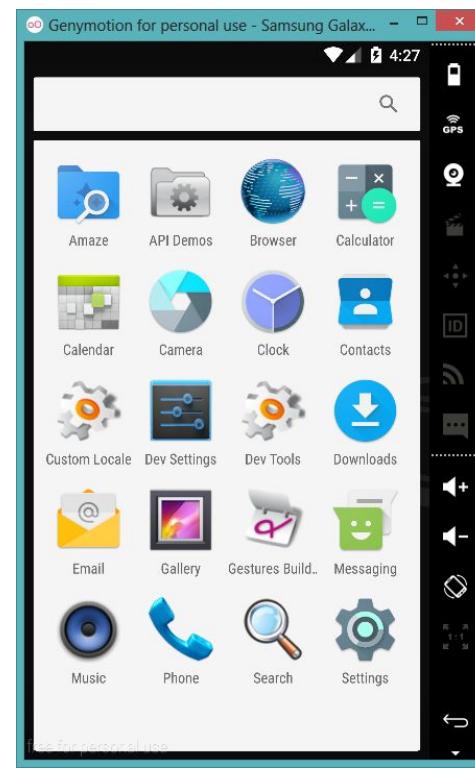
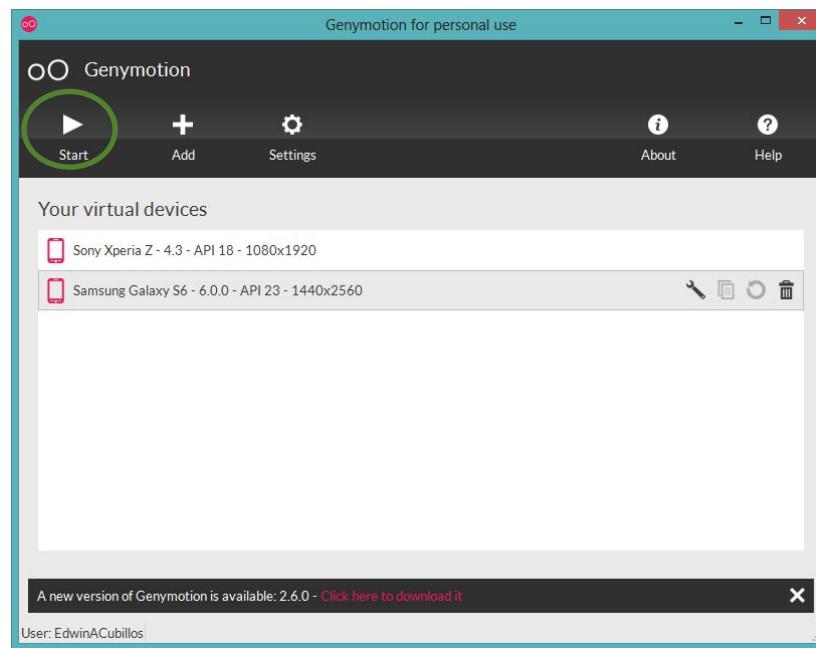
Samsung Galaxy S6 - 6.0.0 - API 23 - 1440x2560

Description	Samsung Galaxy S6 (5.1", 1440x2560, XXXHDPI) AOSP 6.0 API 23
System version	Genymotion Phone - 6.0.0 - API 23 - 2.6.0
Name	Genymotion Virtual Device for Phone
Description	Genymotion Virtual Device for Phone
Android Version	6.0.0
Release date	jue. 14.ene.04:04:49 2016
Version number	2.6.0
Screen size - Density	1440x2560 - 640 dpi
Memory size	3072 MB
Number of CPUs	4
Data disk capacity	32768 MB

Cancel Next

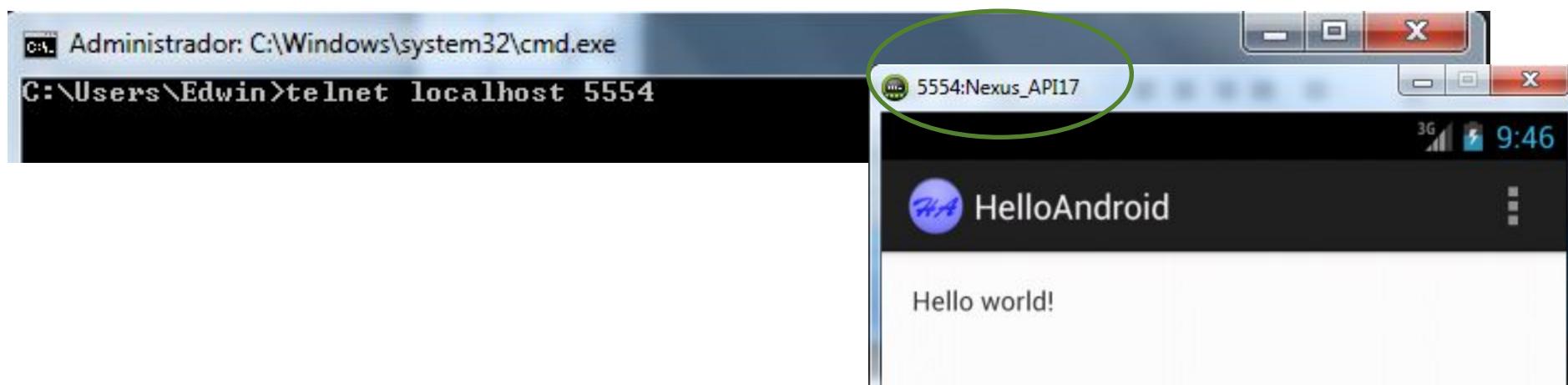


# Probando la App



# Probando la App

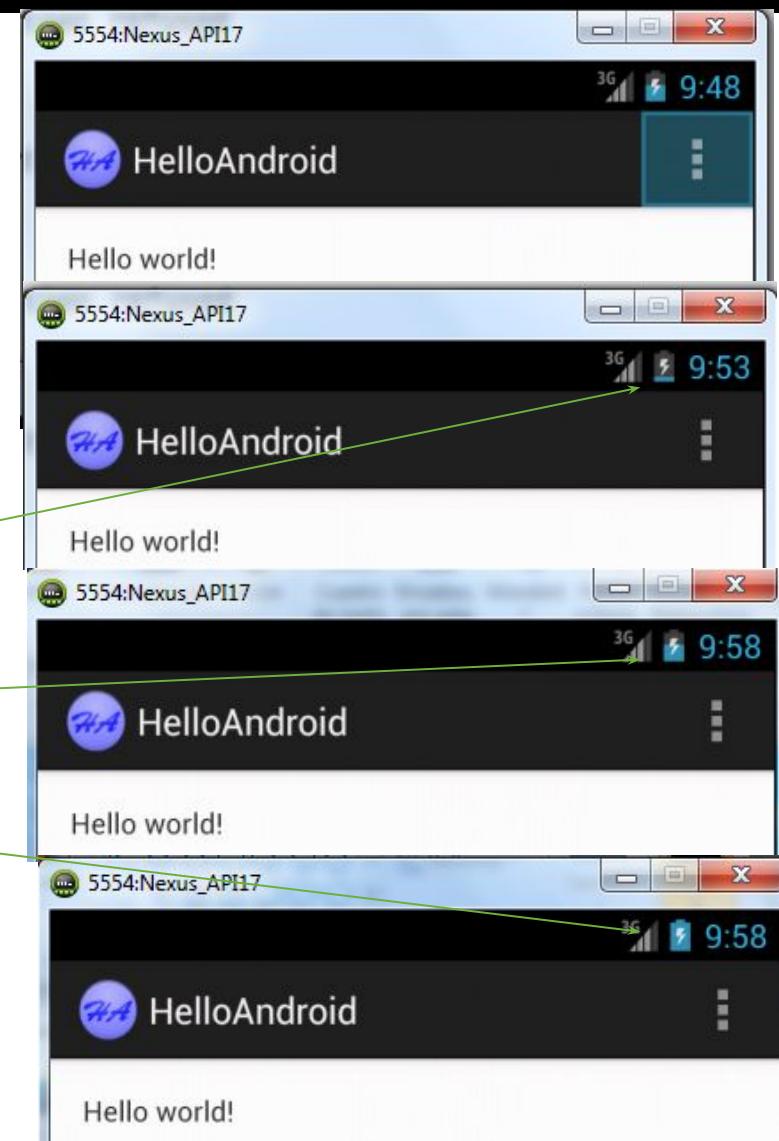
- ❖ Una de las ventajas que ofrece el emulador de Eclipse es que se pueden emular diferentes parámetros como:
  - Velocidad de red y latencia de redes celulares
  - Nivel de Batería
  - Coordenadas de localización
- ❖ Todo esto lo podemos hacer por consola conectándonos al emulador por telnet



# Probando la App

## ❖ Control de la batería

```
c:\> Telnet localhost  
power capacity 5%  
OK  
power capacity 50%  
OK  
power capacity 99%  
OK
```



# Probando la App

- ❖ Mas información sobre el shell

<https://cloud.genymotion.com/page/doc/#collapse9>