

**UNIVERSIDAD
DE ANTIOQUIA**
1803

Introducción a Android

Ing. Edwin Andrés Cubillos Vega Msc.





Introduction

History

The Software Stack

The Android Development Environment

First APP: Hello Android

Android Project Structure

Testing the app



Introduction

History

The Software Stack

The Android Development Environment

First APP: Hello Android

Android Project Structure

Testing the app

- ❖ 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 mas 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.





Introduction

History

The Software Stack

The Android Development Environment

First APP: Hello Android

Android Project Structure

Testing the app



Historia

- ❖ 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

- ❖ 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)** en Febrero

- Actualización para solucionar muchos de los bugs y errores de la primera versión

❖ **Cupcake (1,5)** 30 de abril

- 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:

- 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) en octubre:

- 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) 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....

❖ Gingerbread (2,3) en 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) en 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) 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

- 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

- ❖ En octubre 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 borrar su contenido
 - Nuevo diseño de la aplicación Descargas
 - Cambia de pantalla de inicio fácilmente
 - Actualización de la aplicación Correo

- ❖ El 3 de noviembre 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

- Optimización de Google Now, con un simple toque, podrás ver tarjetas con aplicaciones y toda la información que necesitas.
- Batería mas 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

- 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
- Funcion ahorro de datos, apps en segundo plano no acceden
- Actualización en segundo plano

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



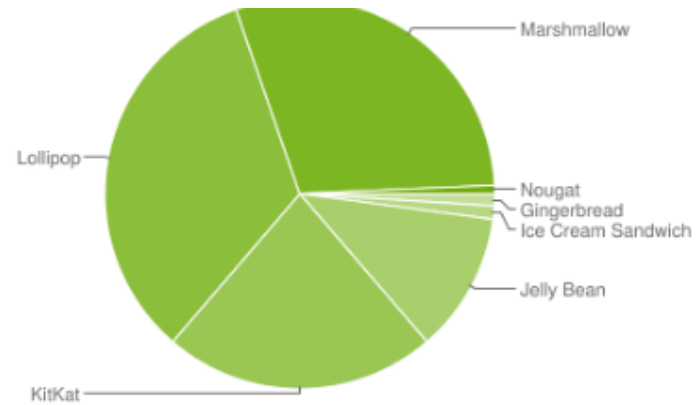
Android 6.0 Marshmallow



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	1.0%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	1.1%
4.1.x	Jelly Bean	16	4.0%
4.2.x		17	5.9%
4.3		18	1.7%
4.4	KitKat	19	22.6%
5.0	Lollipop	21	10.1%
5.1		22	23.3%
6.0	Marshmallow	23	29.6%
7.0	Nougat	24	0.5%
7.1		25	0.2%



Data collected during a 7-day period ending on January 9, 2017.

Any versions with less than 0.1% distribution are not shown.

Historia

 Android Platform/API Version Distribution

ANDROID PLATFORM VERSION	API LEVEL	CUMULATIVE DISTRIBUTION
2.3 Gingerbread	10	97,4%
4.0 Ice Cream Sandwich	15	95,2%
4.1 Jelly Bean	16	87,4%
4.2 Jelly Bean	17	76,9%
4.3 Jelly Bean	18	73,9%
4.4 KitKat	19	40,5%
5.0 Lollipop	21	24,1%
5.1 Lollipop	22	4,7%
6.0 Marshmallow	23	



Introduction

History

The Software Stack

The Android Development Environment

First APP: Hello Android

Android Project Structure

Testing the app



The Software Stack

- ❖ El sistema operativo Android esta formado por varias capas
- ❖ Cada capa tiene sus propias características y propósitos
- ❖ Pero estas no están separadas entre si.
- ❖ Android esta 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 mas portabilidad.



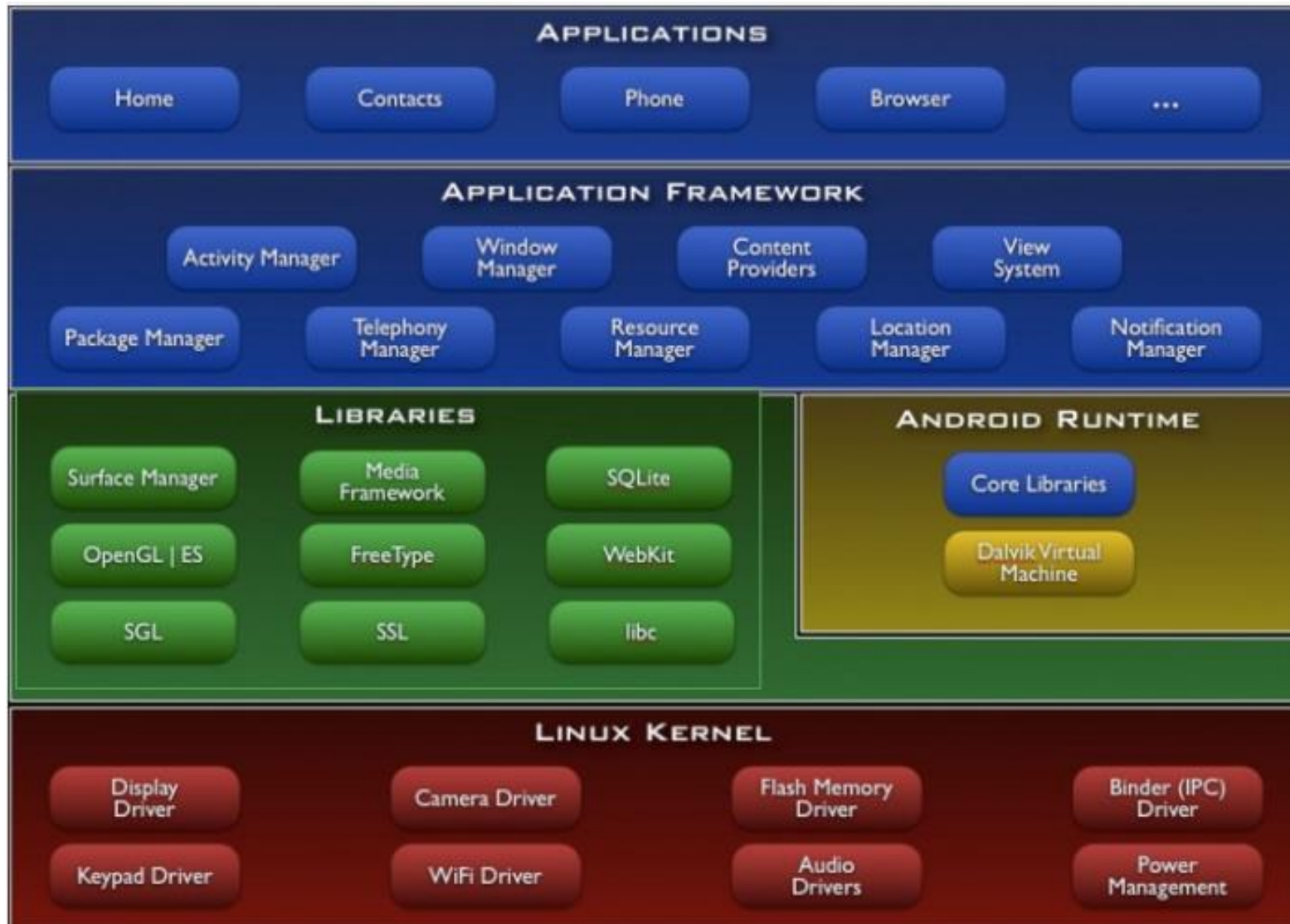
Seguridad:

- ❖ 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 poderosos 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





Linux Kernel:

- ❖ Es la capa mas 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



Linux Kernel:

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



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 Mas

The Software Stack

LIBRARIES

Surface Manager

Media
Framework

SQLite

OpenGL | ES

FreeType

WebKit

SGL

SSL

libc

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.

The Software Stack

LIBRARIES

Surface Manager

Media
Framework

SQLite

OpenGL | ES

FreeType

WebKit

SGL

SSL

libc

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



ck Software

- ❖ 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**



ck Software

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



ck Software

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

APPLICATION FRAMEWORK

Activity Manager

Window Manager

Content Providers

View System

Package Manager

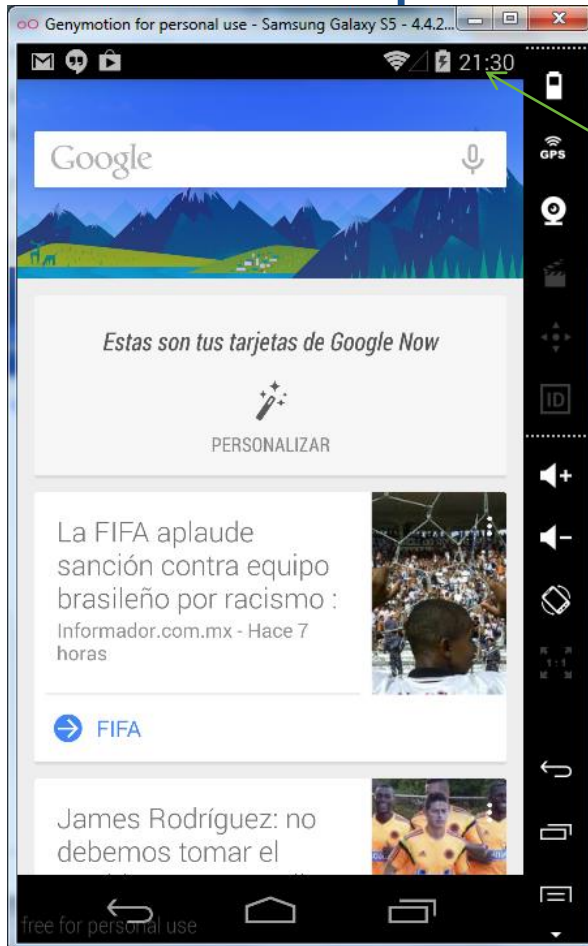
Telephony Manager

Resource Manager

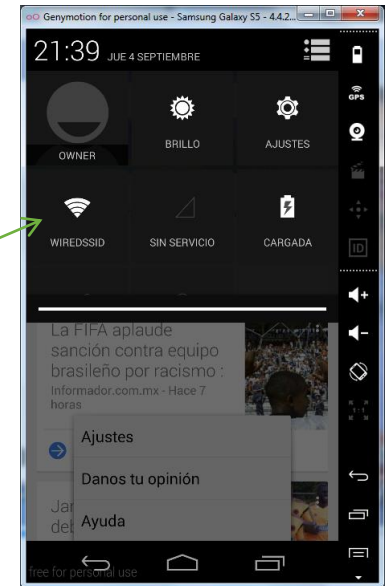
Location Manager

Notification Manager

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



Barra de notificaciones



Ventana Principal (Browser)

APPLICATION FRAMEWORK

Activity Manager

Window Manager

Content Providers

View System

Package Manager

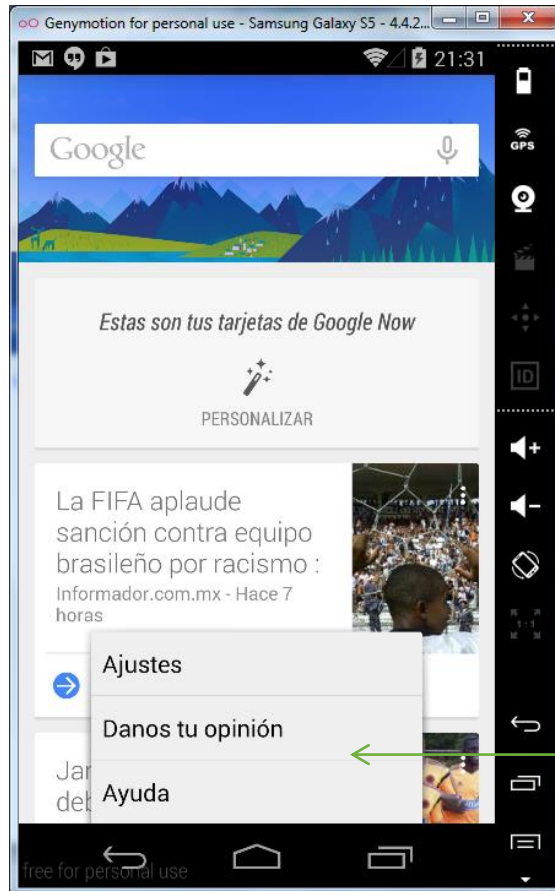
Telephony Manager

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

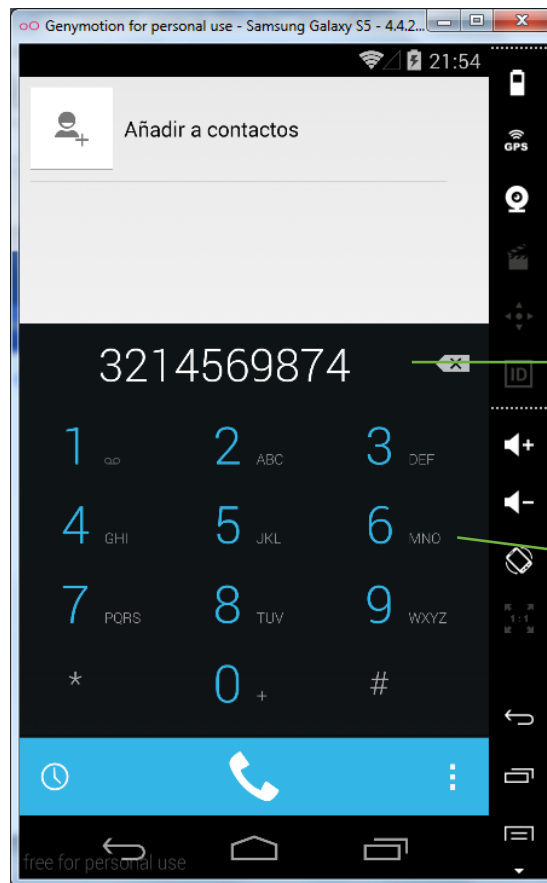
Telephony Manager

Resource Manager

Location Manager

Notification Manager

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



TextView

Buttons

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

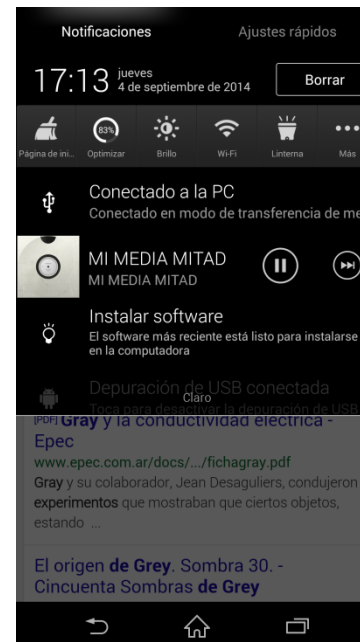
Telephony Manager

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



- ❖ *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

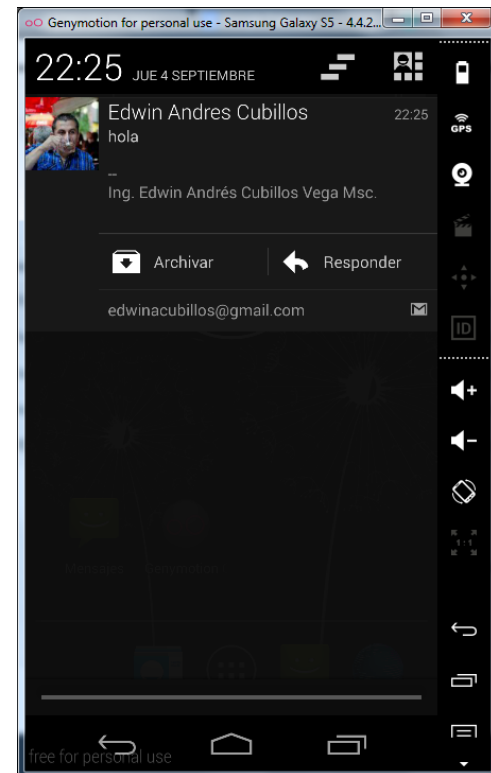
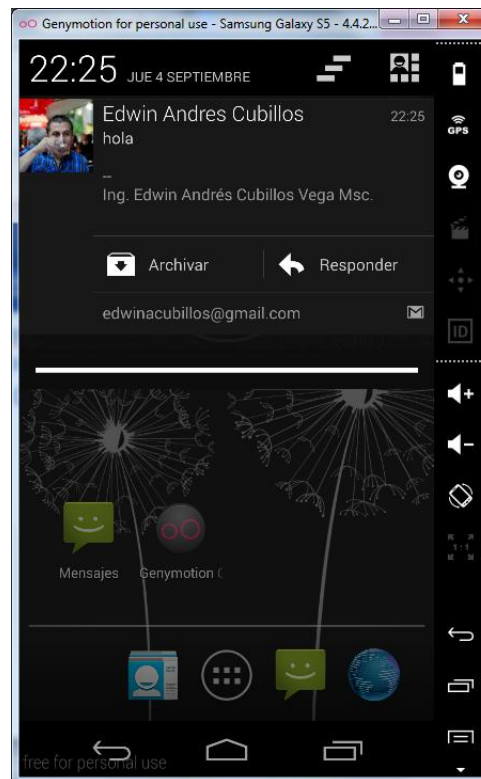
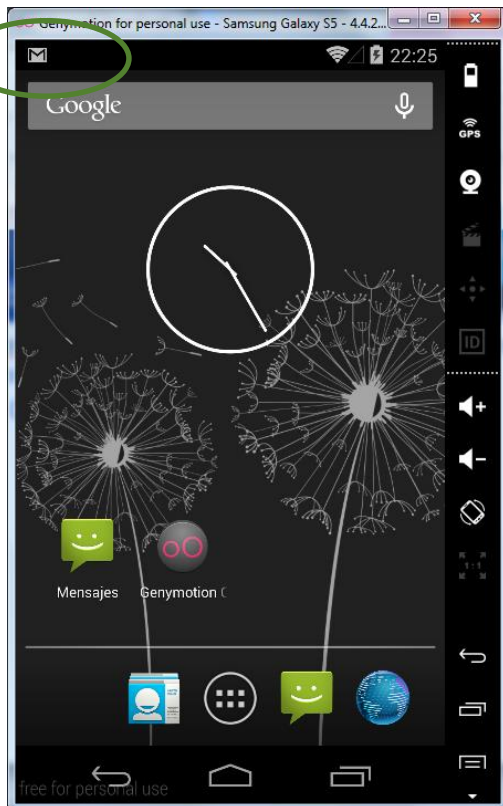
Telephony Manager

Resource Manager

Location Manager

Notification Manager

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





Applications

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



Introduction

History

The Software Stack

The Android Development Environment

First APP: Hello Android

Android Project Structure

Testing the app



Prerequisites


- ❖ Microsoft® Windows® 8/7/Vista/2003 (32 or 64-bit)
 - 2 GB RAM minimum, 4 GB RAM recommended
 - 400 MB hard disk space
 - At least 1 GB for Android SDK, emulator system images, and caches
 - 1280 x 800 minimum screen resolution
 - Java Development Kit (JDK) 7
 - Optional for accelerated emulator: Intel® processor with support for Intel® VT-x, Intel® EM64T (Intel® 64), and Execute Disable (XD) Bit functionality

❖ Mac OS X 10.5.8 or later (x86 only)

- Mac® OS X® 10.8.5 or higher, up to 10.9 (Mavericks)
- 2 GB RAM minimum, 4 GB RAM recommended
- 400 MB hard disk space
- At least 1 GB for Android SDK, emulator system images, and caches
- 1280 x 800 minimum screen resolution
- Java Runtime Environment (JRE) 6
- Java Development Kit (JDK) 7
- Optional for accelerated emulator: Intel® processor with support for Intel® VT-x, Intel® EM64T (Intel® 64), and Execute Disable (XD) Bit functionality
- On Mac OS, run Android Studio with Java Runtime Environment (JRE) 6 for optimized font rendering. You can then configure your project to use Java Development Kit (JDK) 6 or JDK 7.

- ❖ Linux (tested on Ubuntu Linux, Lucid Lynx)
 - GNOME or KDE desktop
 - GNU C Library (glibc) 2.15 or later
 - 2 GB RAM minimum, 4 GB RAM recommended
 - 400 MB hard disk space
 - At least 1 GB for Android SDK, emulator system images, and caches
 - 1280 x 800 minimum screen resolution
 - Oracle® Java Development Kit (JDK) 7
 - Tested on Ubuntu® 14.04, Trusty Tahr (64-bit distribution capable of running 32-bit applications).

❖ Android Studio Descargar e Instalar



The official Android IDE

Android Studio

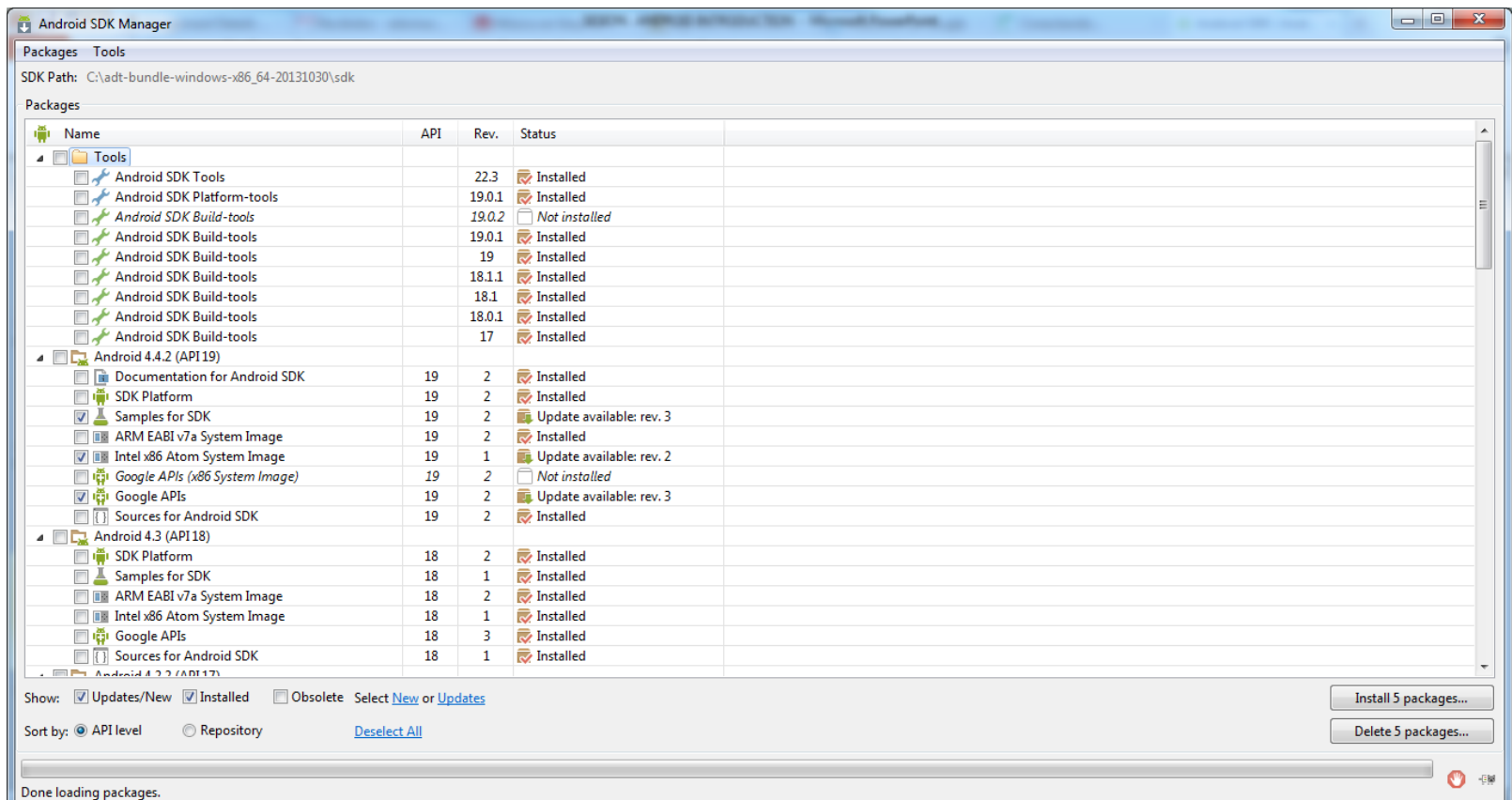
- Android Studio IDE
- Android SDK tools
- Android 6.0 (Marshmallow) Platform
- Android 6.0 emulator system image with Google APIs

DOWNLOAD ANDROID STUDIO FOR WINDOWS

The image shows a laptop displaying the Android Studio IDE. The interface includes a project explorer on the left, a central code editor with Java code, and a preview window on the right showing a mobile app interface with a login form. The text 'The official Android IDE' is displayed above the laptop screen. To the left of the laptop, the Android Studio logo and name are shown, along with a list of components included in the IDE and a green button that says 'DOWNLOAD ANDROID STUDIO FOR WINDOWS'.

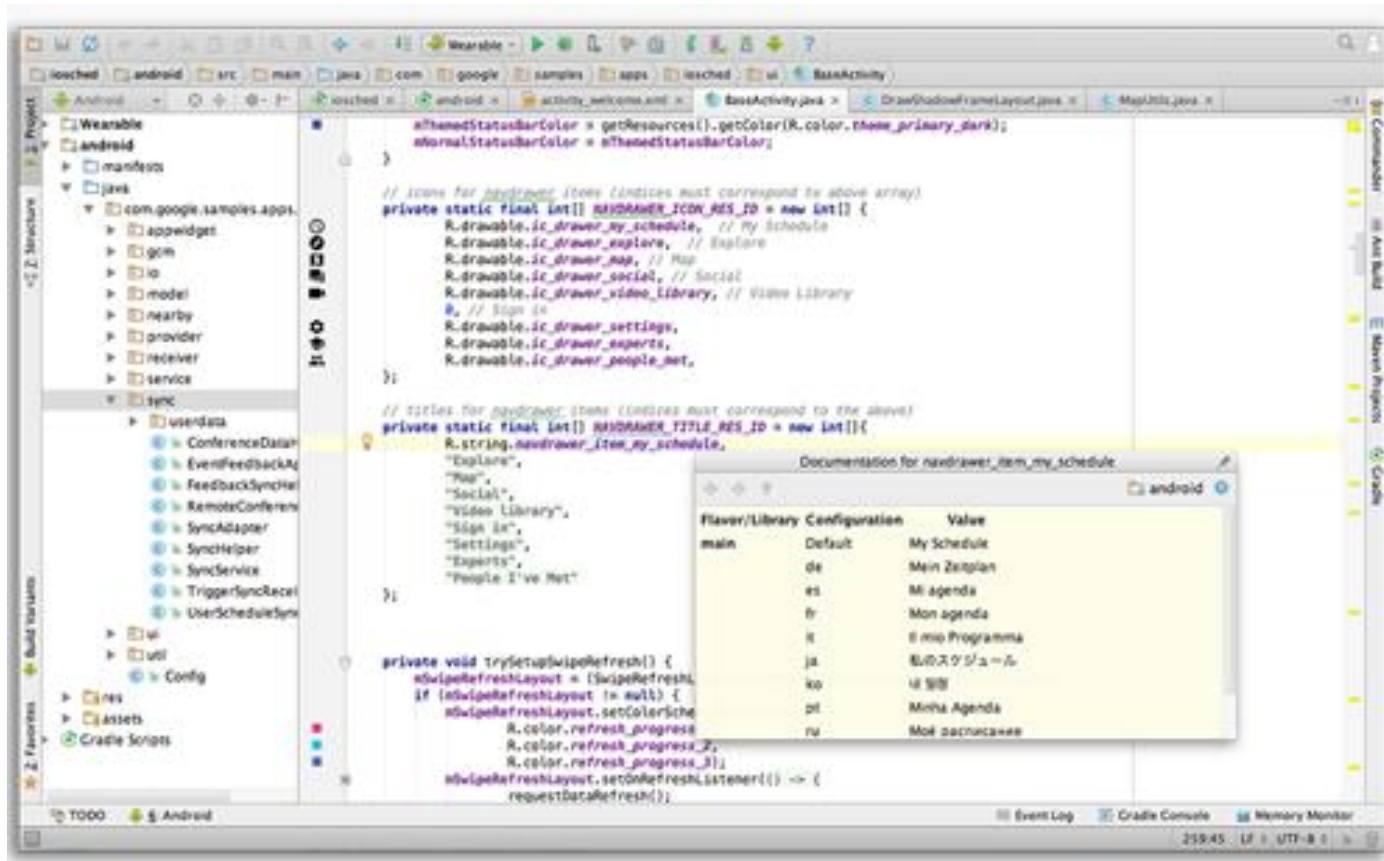
Instalación y configuración

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



Android Studio

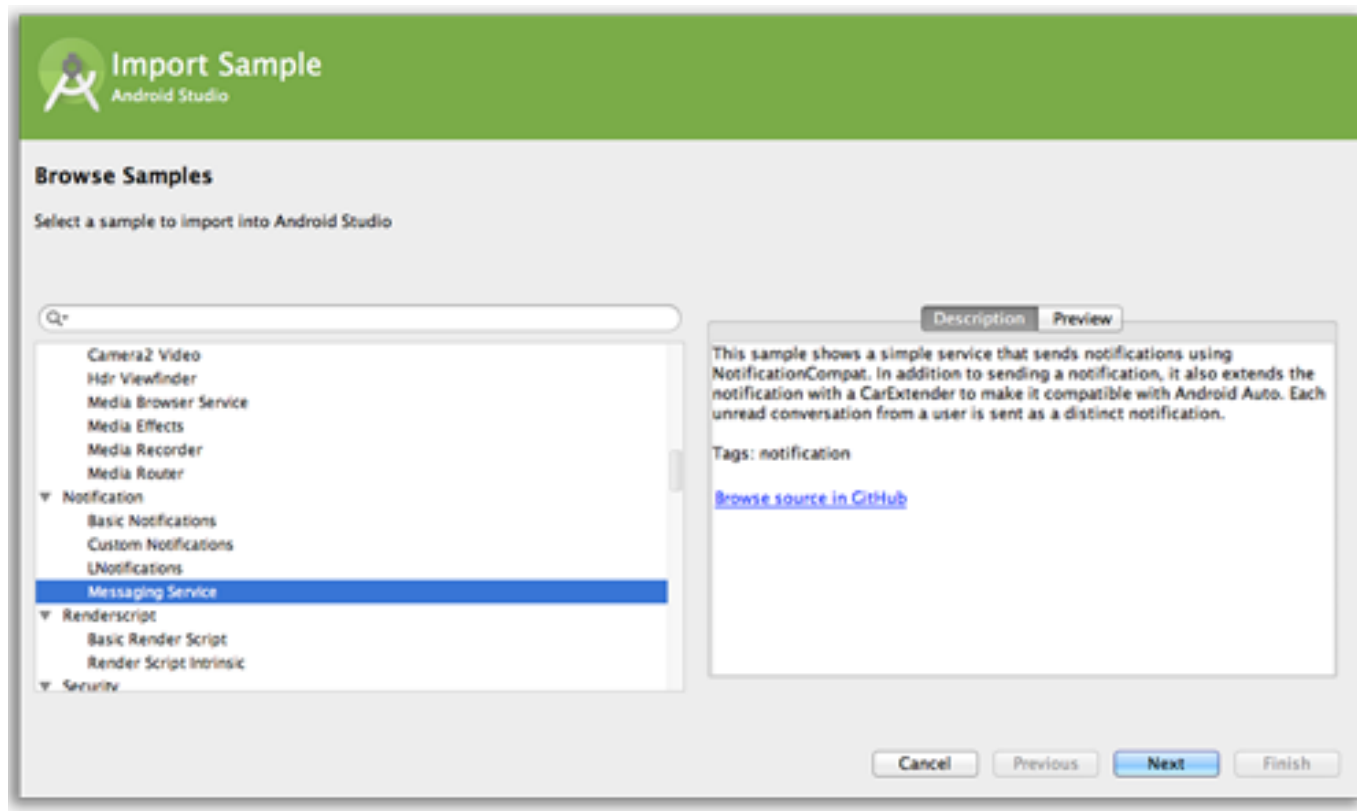
❖ Editor Intelligente



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.

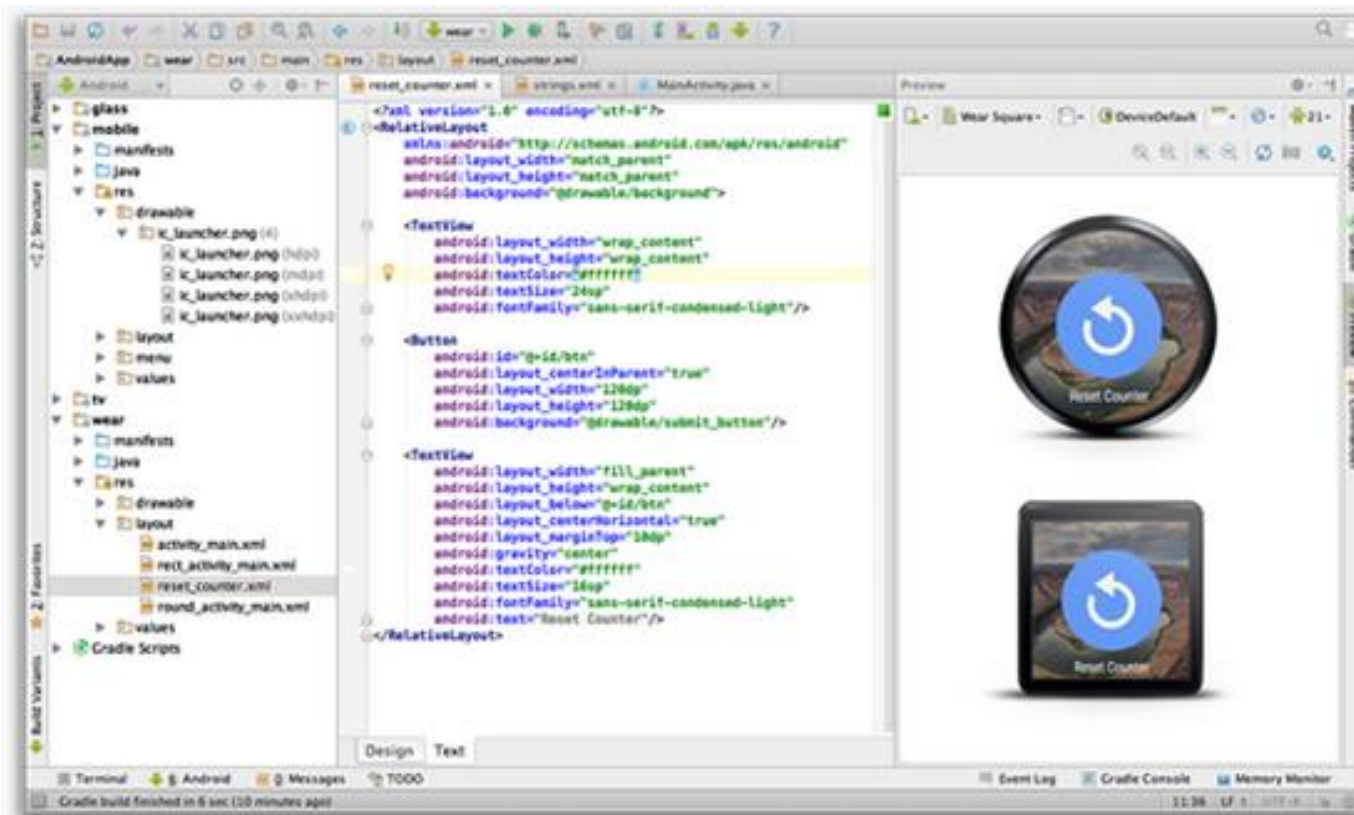
❖ Plantillas con código e Integración GitHub



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

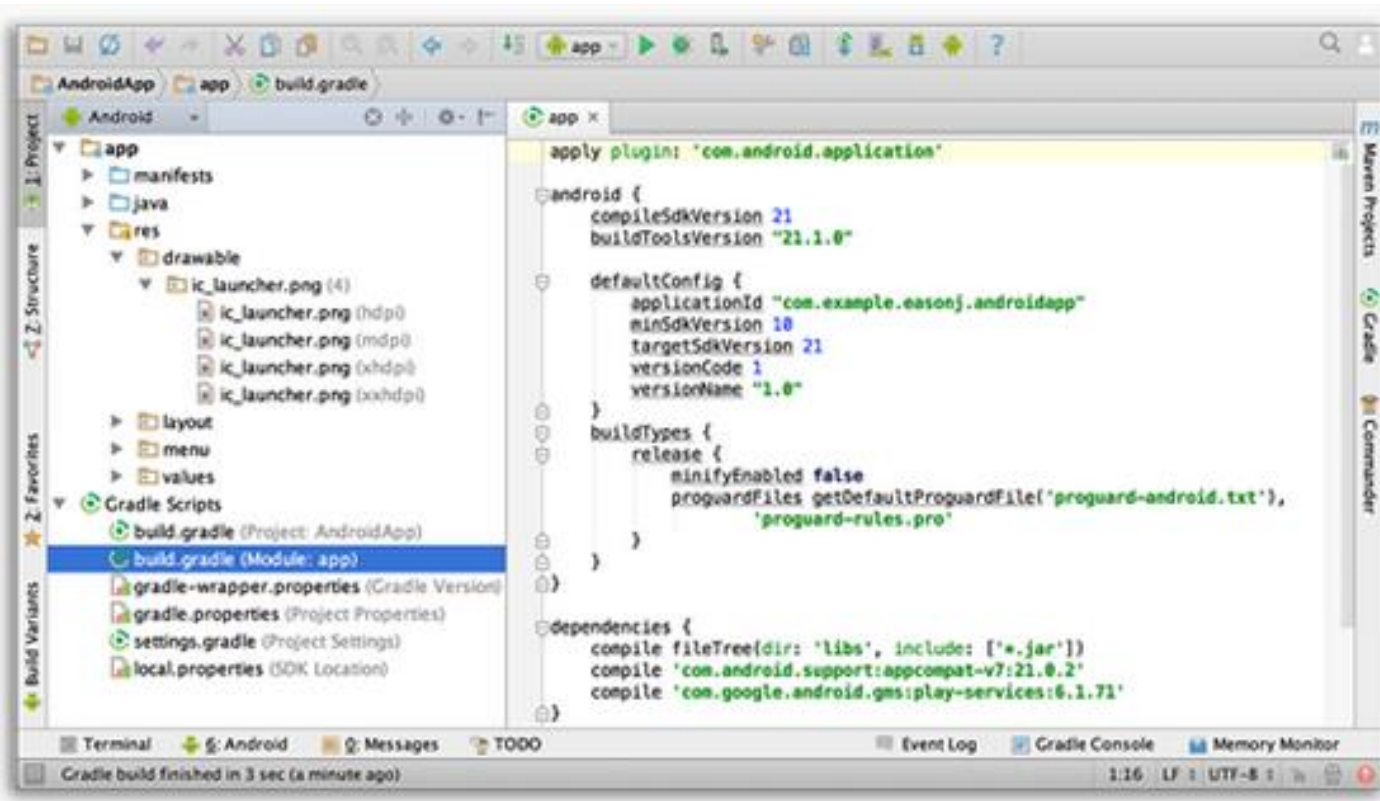
Install and run your apps faster than with a physical device and test your app on virtually any Android device configuration: Android phones, Android tablets, Android Wear, and Android TV devices.

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.



Android Studio

❖ Uso de Gradle y Maven



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 usada por grandes proyectos "**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 las 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.



Introduction

History

The Software Stack

The Android Development Environment

First APP: Hello Android

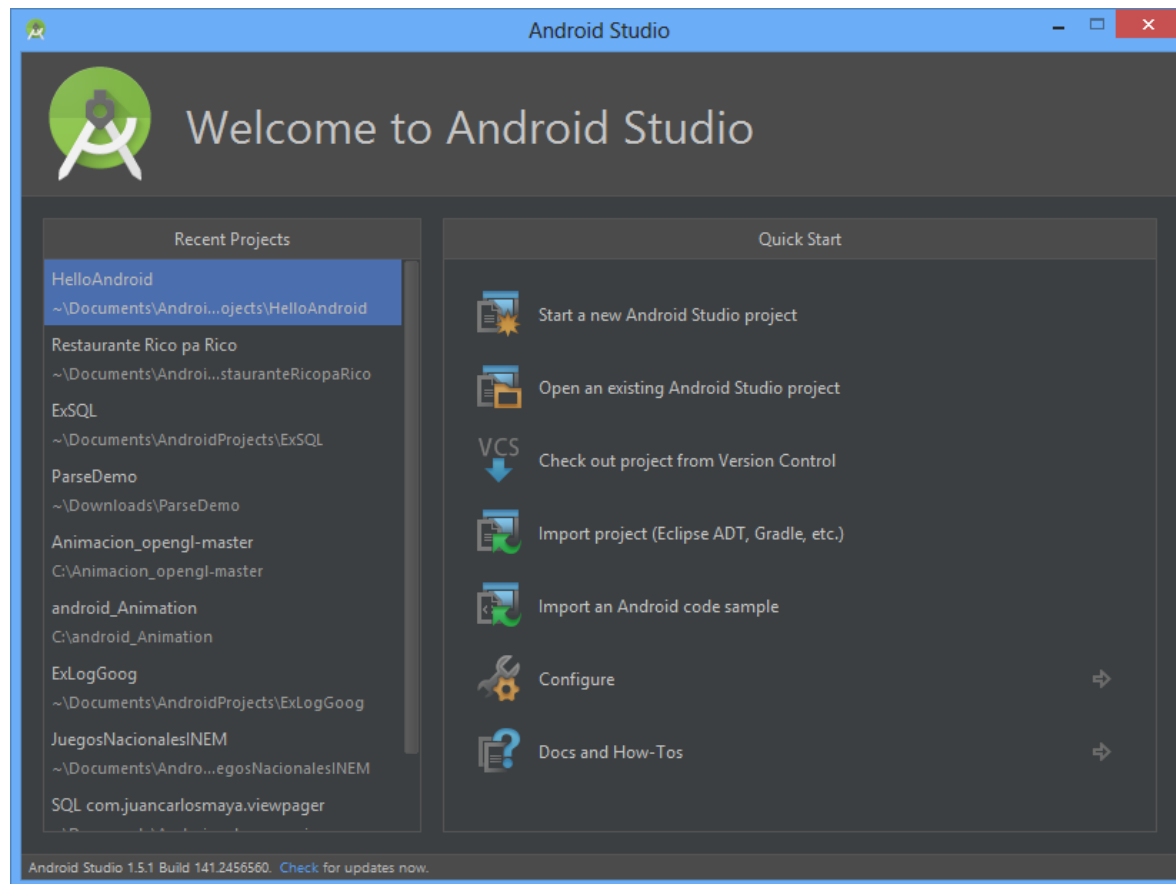
Android Project Structure

Testing the app

Primer App: Hello Android

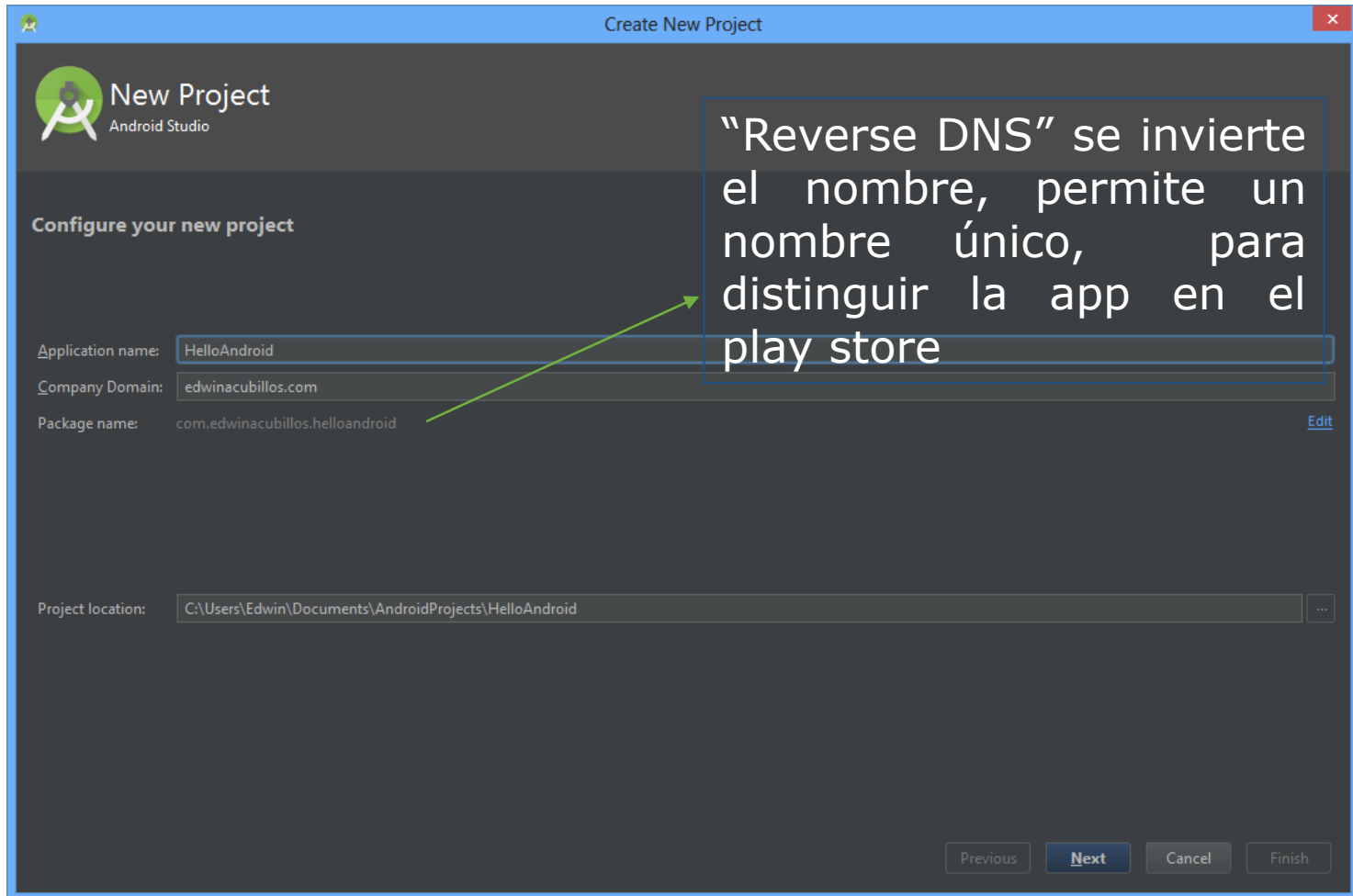
Creación del proyecto

❖ Start a new Android Studio Project



Primer App: Hello Android

❖ Colocar nombre de la Aplicación



Create New Project

New Project
Android Studio

Configure your new project

Application name: HelloAndroid

Company Domain: edwinacubillos.com

Package name: com.edwinacubillos.helloandroid [Edit](#)

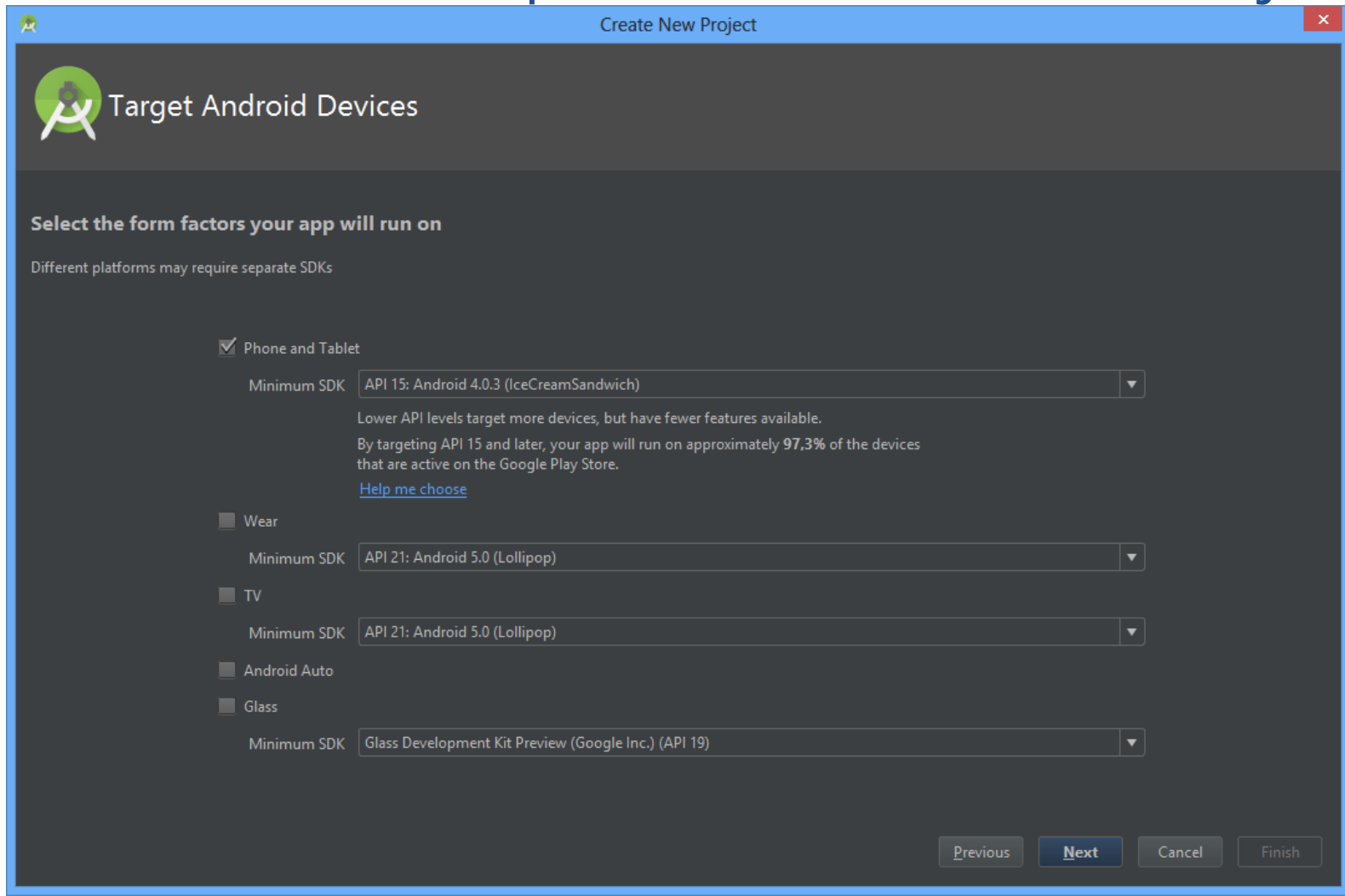
Project location: C:\Users\Edwin\Documents\AndroidProjects\HelloAndroid ...

Previous Next Cancel Finish

“Reverse DNS” se invierte el nombre, permite un nombre único, para distinguir la app en el play store

Primer App: Hello Android

❖ Seleccionar la API para la cual vamos a trabajar



The screenshot shows the 'Create New Project' dialog in Android Studio. The title bar is blue and contains the text 'Create New Project' and a close button. The main content area has a dark gray background. At the top left, there is an Android logo and the text 'Target Android Devices'. Below this, the heading 'Select the form factors your app will run on' is displayed, followed by the subtitle 'Different platforms may require separate SDKs'. There are five form factors listed, each with a checkbox and a 'Minimum SDK' dropdown menu. The 'Phone and Tablet' option is selected with a checkmark. The 'Wear', 'TV', 'Android Auto', and 'Glass' options are not selected. The 'Minimum SDK' for 'Phone and Tablet' is set to 'API 15: Android 4.0.3 (IceCreamSandwich)'. Below this dropdown, there is a note: 'Lower API levels target more devices, but have fewer features available. By targeting API 15 and later, your app will run on approximately 97,3% of the devices that are active on the Google Play Store.' and a link 'Help me choose'. The 'Minimum SDK' for 'Wear', 'TV', and 'Glass' are all set to 'API 21: Android 5.0 (Lollipop)'. The 'Minimum SDK' for 'Android Auto' is set to 'Glass Development Kit Preview (Google Inc.) (API 19)'. At the bottom right, there are four buttons: 'Previous', 'Next', 'Cancel', and 'Finish'.

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: API 15: Android 4.0.3 (IceCreamSandwich)

Lower API levels target more devices, but have fewer features available.
By targeting API 15 and later, your app will run on approximately 97,3% of the devices that are active on the Google Play Store.
[Help me choose](#)

☐ Wear

Minimum SDK: API 21: Android 5.0 (Lollipop)

☐ TV

Minimum SDK: API 21: Android 5.0 (Lollipop)

☐ Android Auto

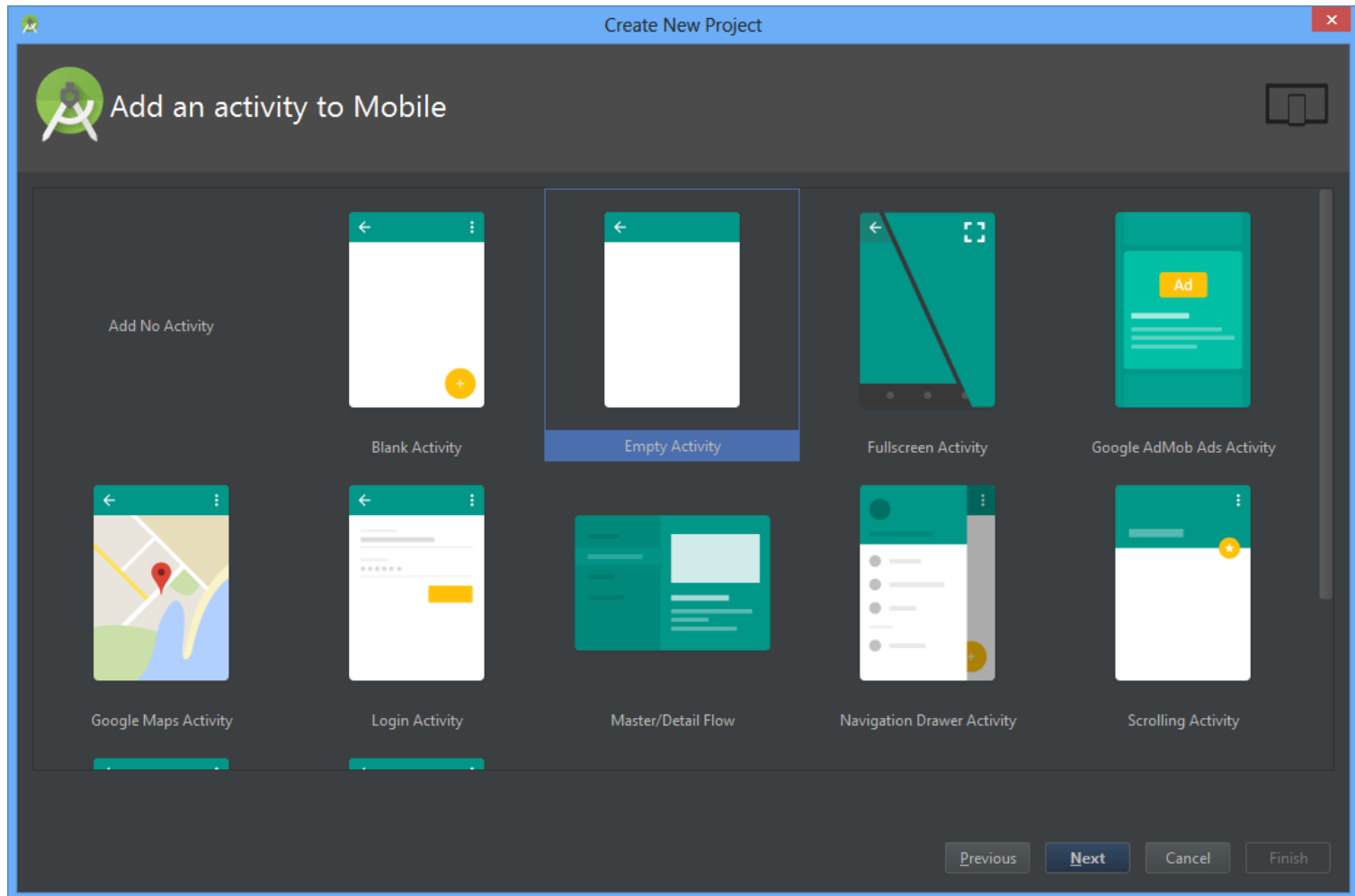
☐ Glass

Minimum SDK: Glass Development Kit Preview (Google Inc.) (API 19)

Previous Next Cancel Finish

Primer App: Hello Android

❖ Seleccionar Actividad en Blanco



Primer App: Hello Android

❖ Configurar nombres de la Actividad y Layout

Create New Project

Customize the Activity

Creates a new empty activity

Activity Name: MainActivity

☒ Generate Layout File

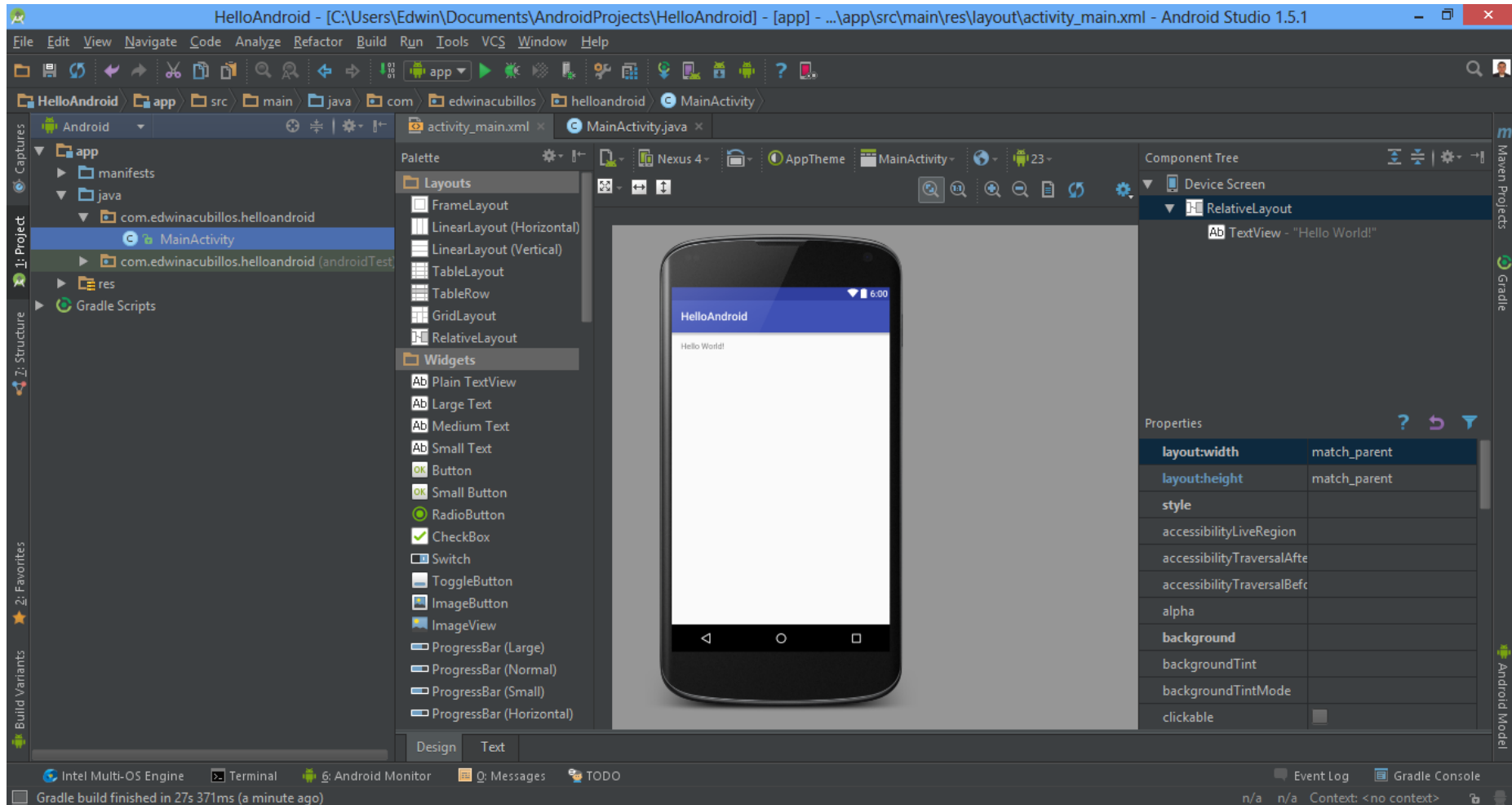
Layout Name: activity_main

Empty Activity

The name of the activity class to create

Previous Next Cancel Finish

Primer App: Hello Android





Introduction

History

The Software Stack

The Android Development Environment

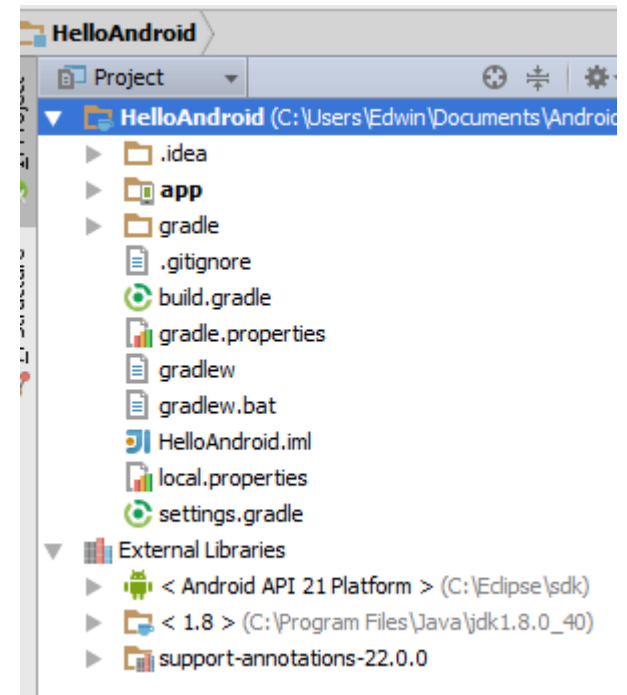
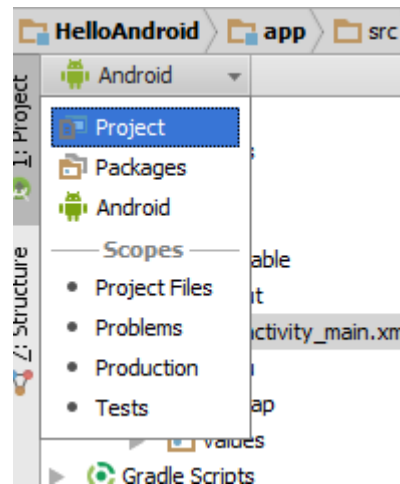
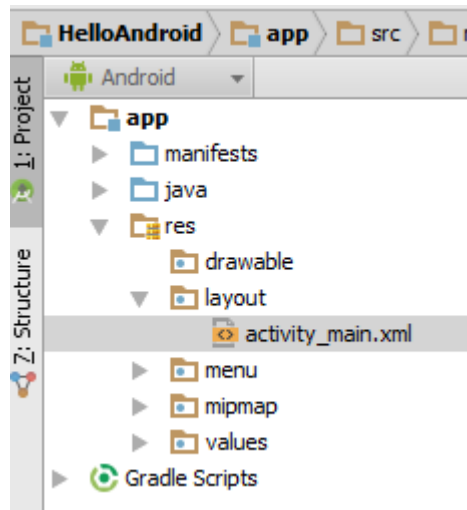
First APP: Hello Android

Android Project Structure

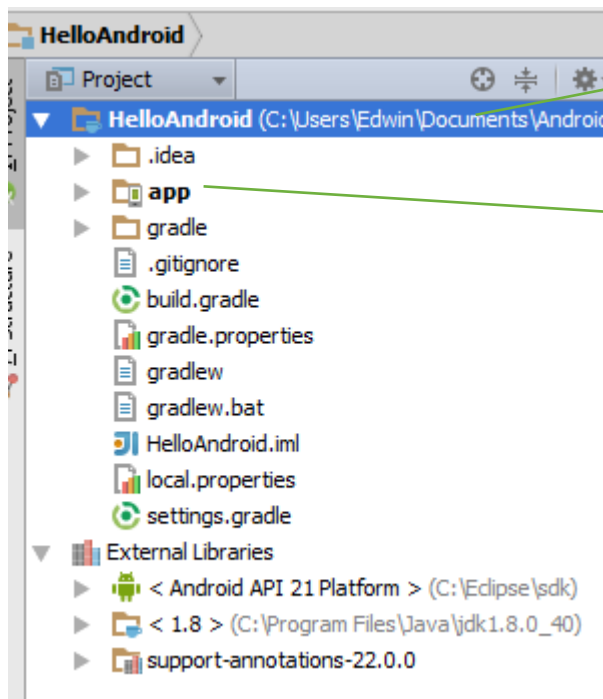
Testing the app

Estructura de un proyecto Android

❖ Cambiar la vista del proyecto de Android a Project



Estructura de un proyecto Android



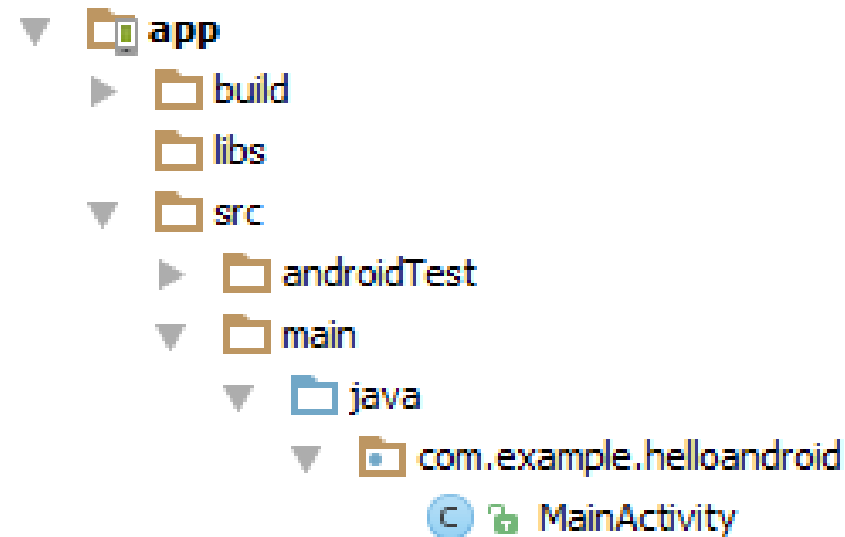
Proyecto

Módulo

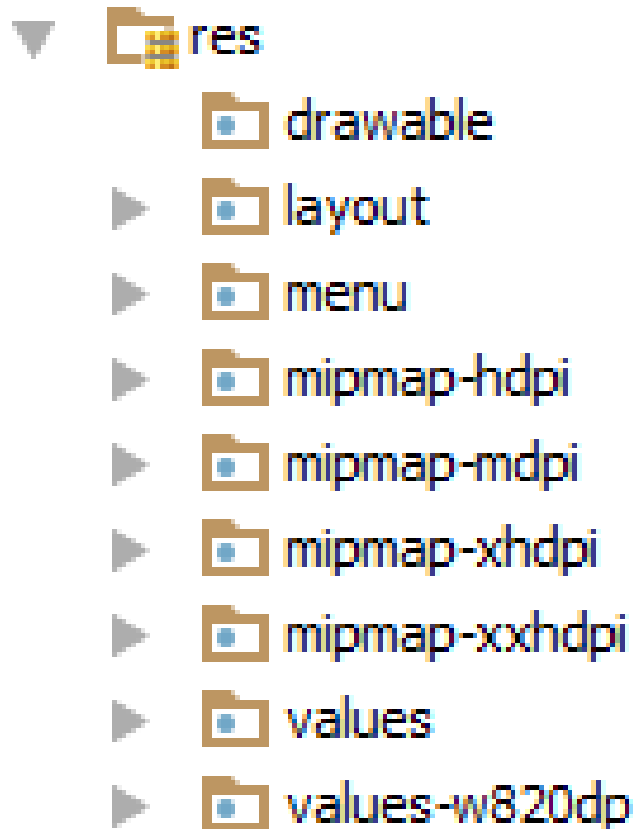
Estructura de un proyecto Android

/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



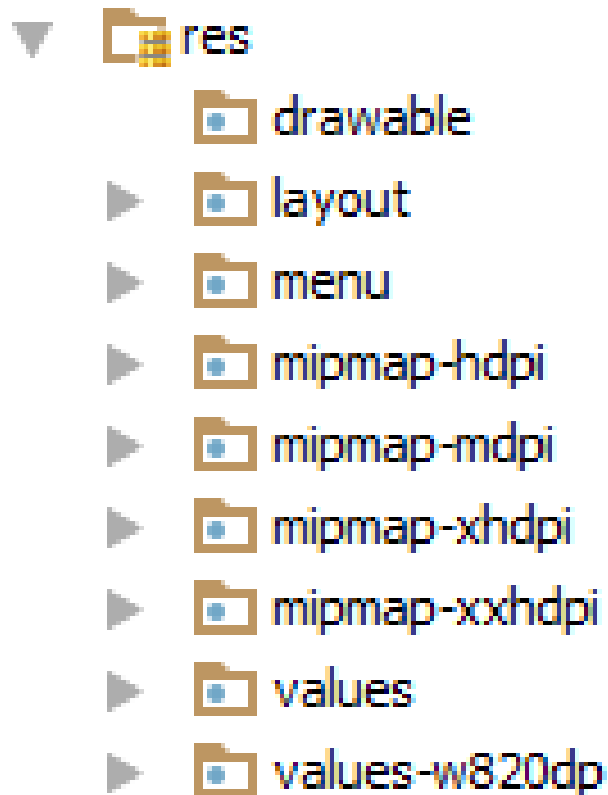
/res

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



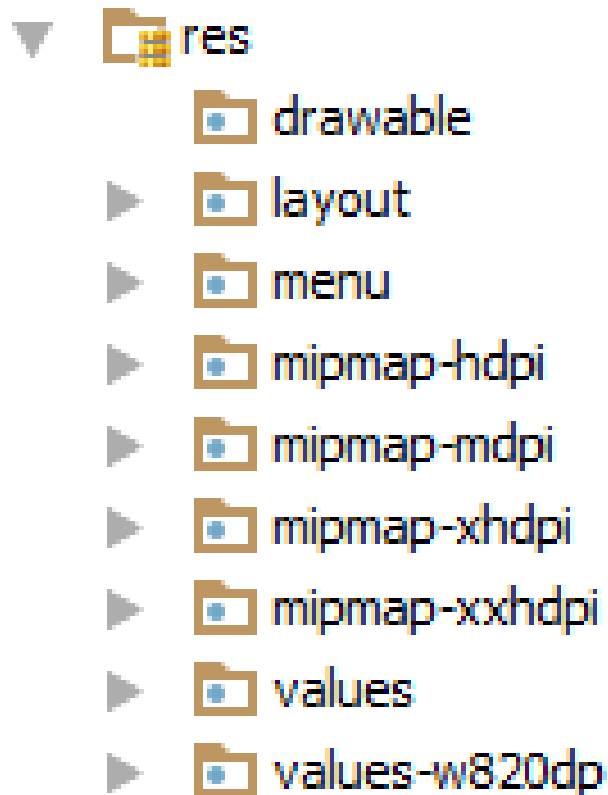
/res/layout/

- ❖ 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

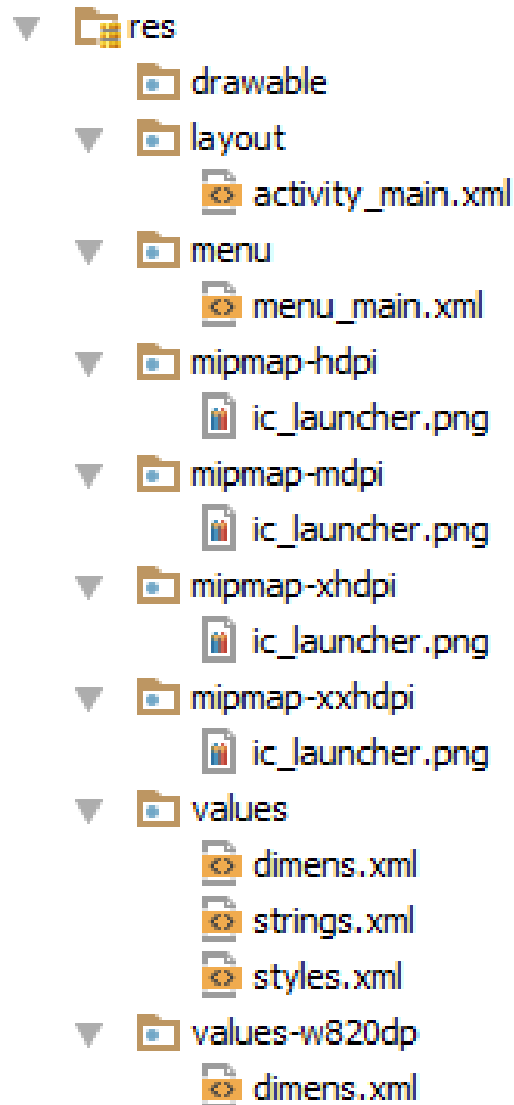
/res/color/

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



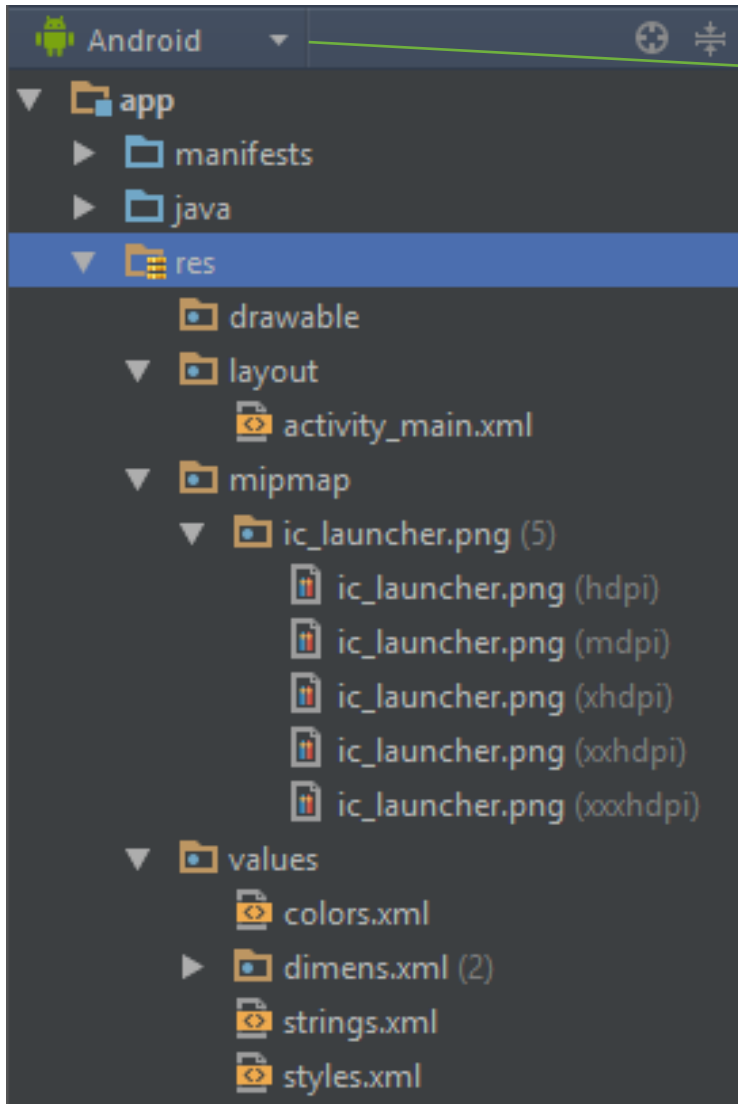
❖ Por defecto se crean estos recursos en la aplicación

❖ Las carpetas con sufijo adicional se refiere a recursos especiales para determinadas versiones Ej:

- values-w820dp: aplica a dispositivos mínimo con pantalla de 820 dp

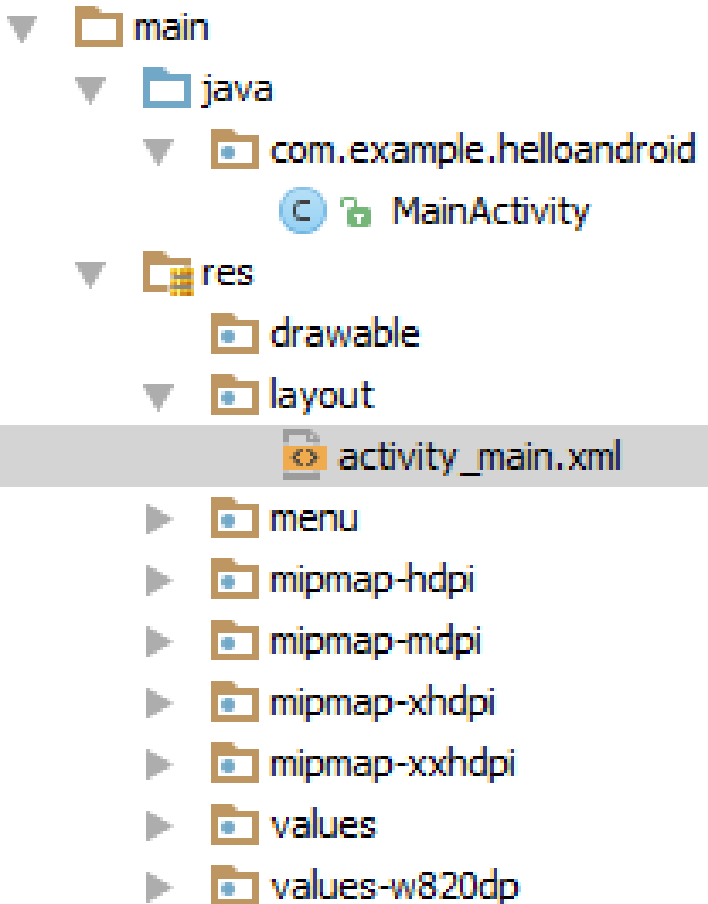
❖ [Mas Información](#)

Estructura de un proyecto Android



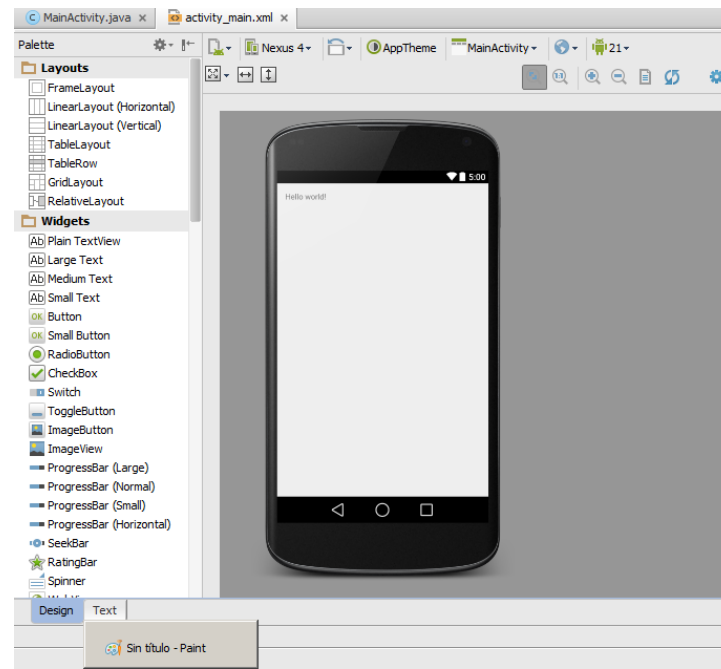
Cambia la vista

Estructura de un proyecto Android

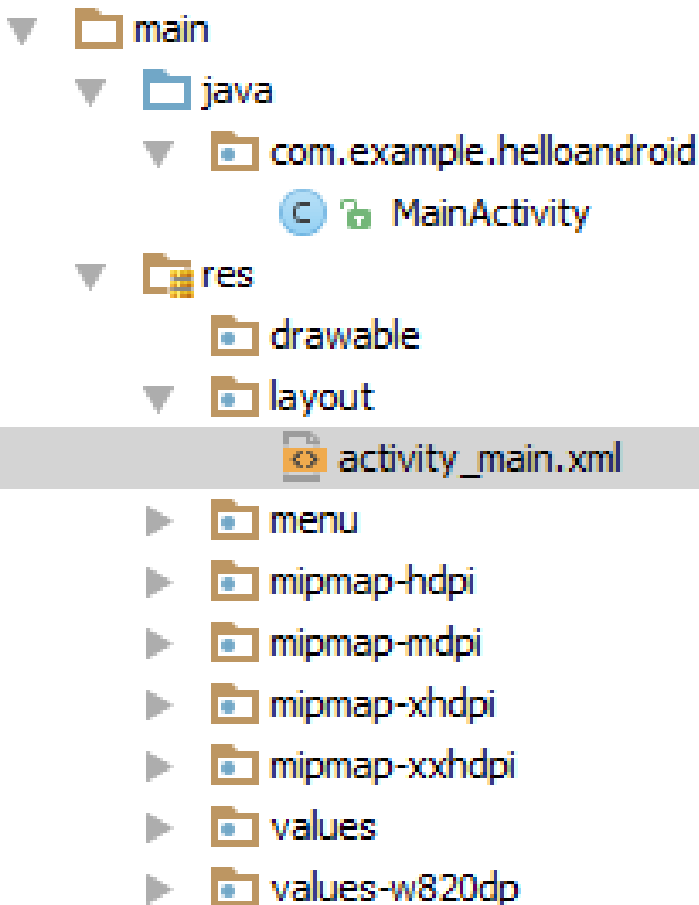


Activity_main.xml

- ❖ 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

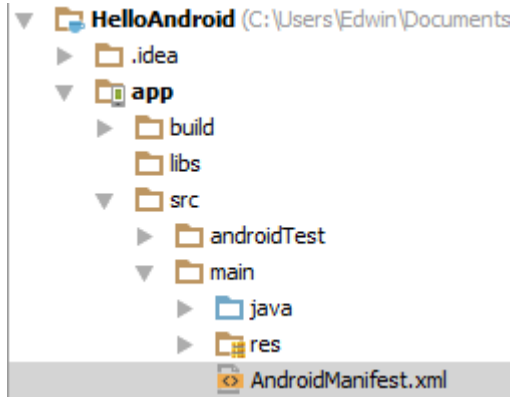


Activity_main.xml

La interfaz tiene asociado un archivo .XML



Estructura de un proyecto Android



AndroidManifest.xml

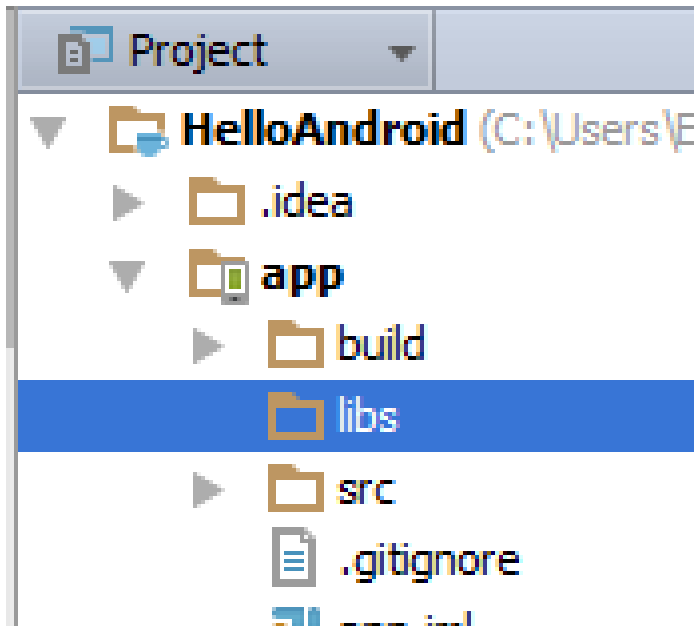
- ❖ Contiene la definición en XML de los aspectos principales de la aplicación:
- ❖ Su identificación:
 - nombre,
 - versión,
 - Icono, etc
- ❖ Componentes
 - pantallas,
 - mensajes, etc
- ❖ las librerías auxiliares utilizadas, o
- ❖ los permisos necesarios para su ejecución



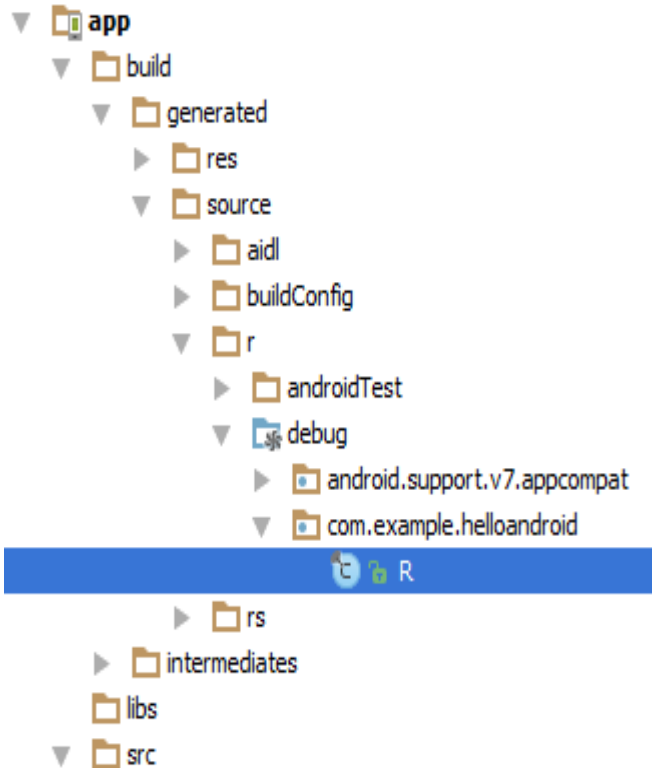
Estructura de un proyecto Android

/libs/

- ❖ Contendrá las librerías auxiliares, normalmente en formato “.jar” que utilizemos 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

- ❖ 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



```
MainActivity.java x activity_main.xml x R.java x
Files under the build folder are generated and should not be edited.

    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.AppCompat.DropDownUp=0x7f0c0000;
```



Introduction

History

The Software Stack


The Android Development Environment

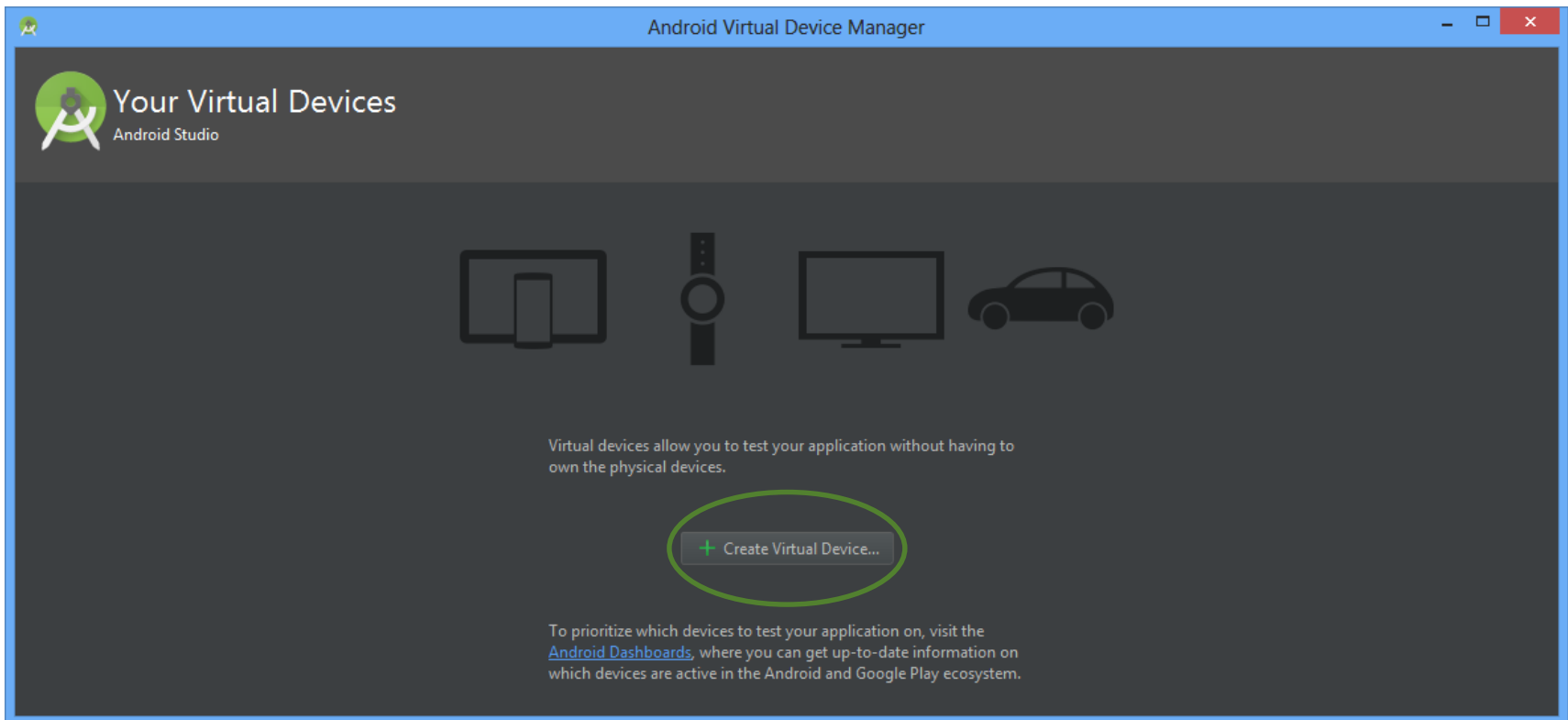
First APP: Hello Android

Android Project Structure

Testing the app

Probando la App

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



❖ Seleccionar el Hardware

Virtual Device Configuration

Select Hardware
Choose a device definition

Search

Category	Name	Size	Resolution	Density
TV	Nexus S	4,0"	480x800	hdpi
Wear	Nexus One	3,7"	480x800	hdpi
Phone	Nexus 6P	5,7"	1440x2560	560dpi
Tablet	Nexus 6	5,96"	1440x2560	560dpi
	Nexus 5X	5,2"	1080x1920	420dpi
	Nexus 5	4,95"	1080x1920	xxhdpi
	Nexus 4	4,7"	768x1280	xhdpi
	Galaxy Nexus	4,65"	720x1280	xhdpi
	Android Wear Square	1,65"	280x280	hdpi
	Android Wear Round	1,65"	320x320	hdpi
	5.4" FWVGA	5,4"	480x854	mdpi
	5.1" WVGA	5,1"	480x800	mdpi

Nexus 5

1080px
4,95"
1920px

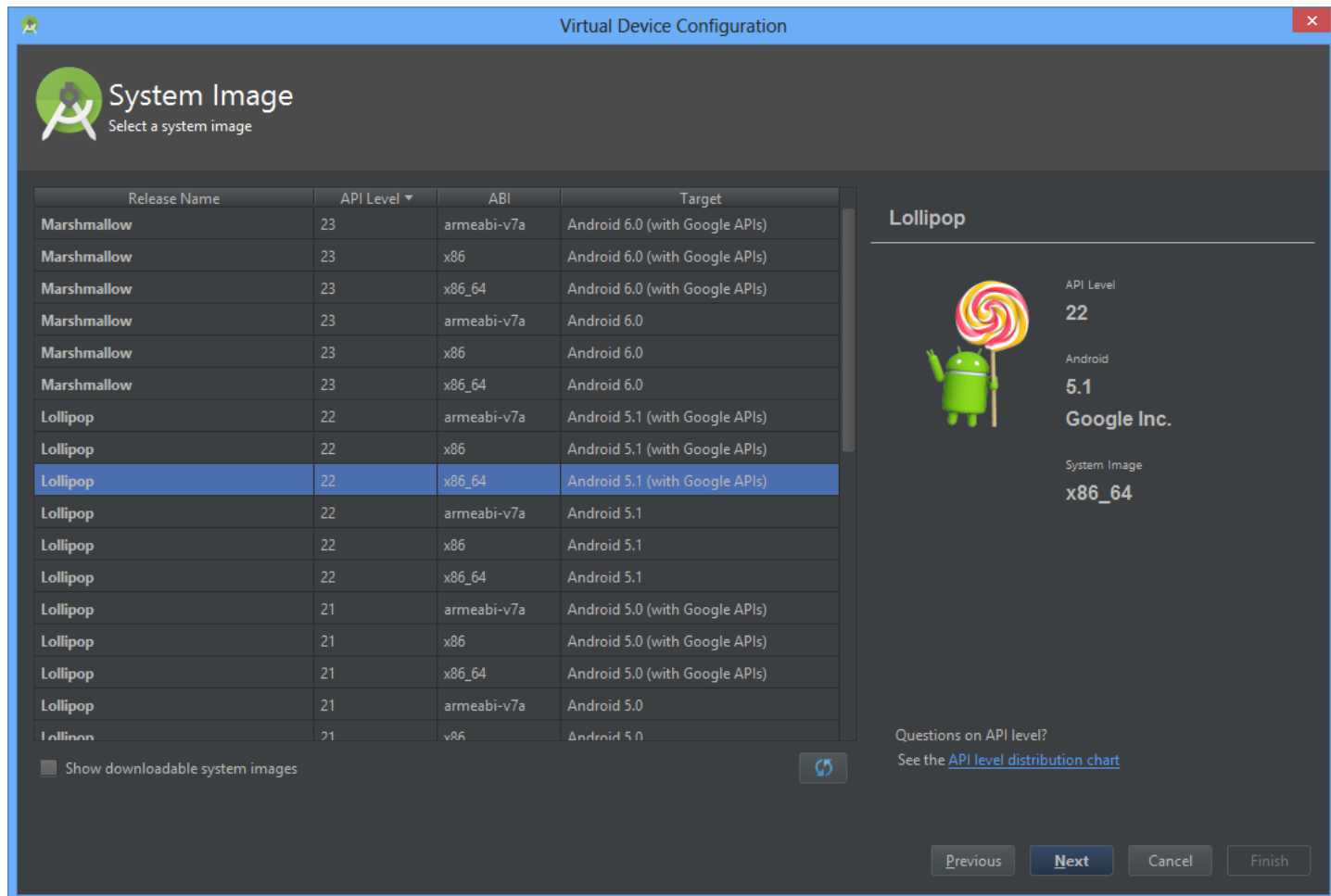
Size: normal
Ratio: notlong
Density: xxhdpi

New Hardware Profile Import Hardware Profiles Clone Device...

Previous Next Cancel Finish

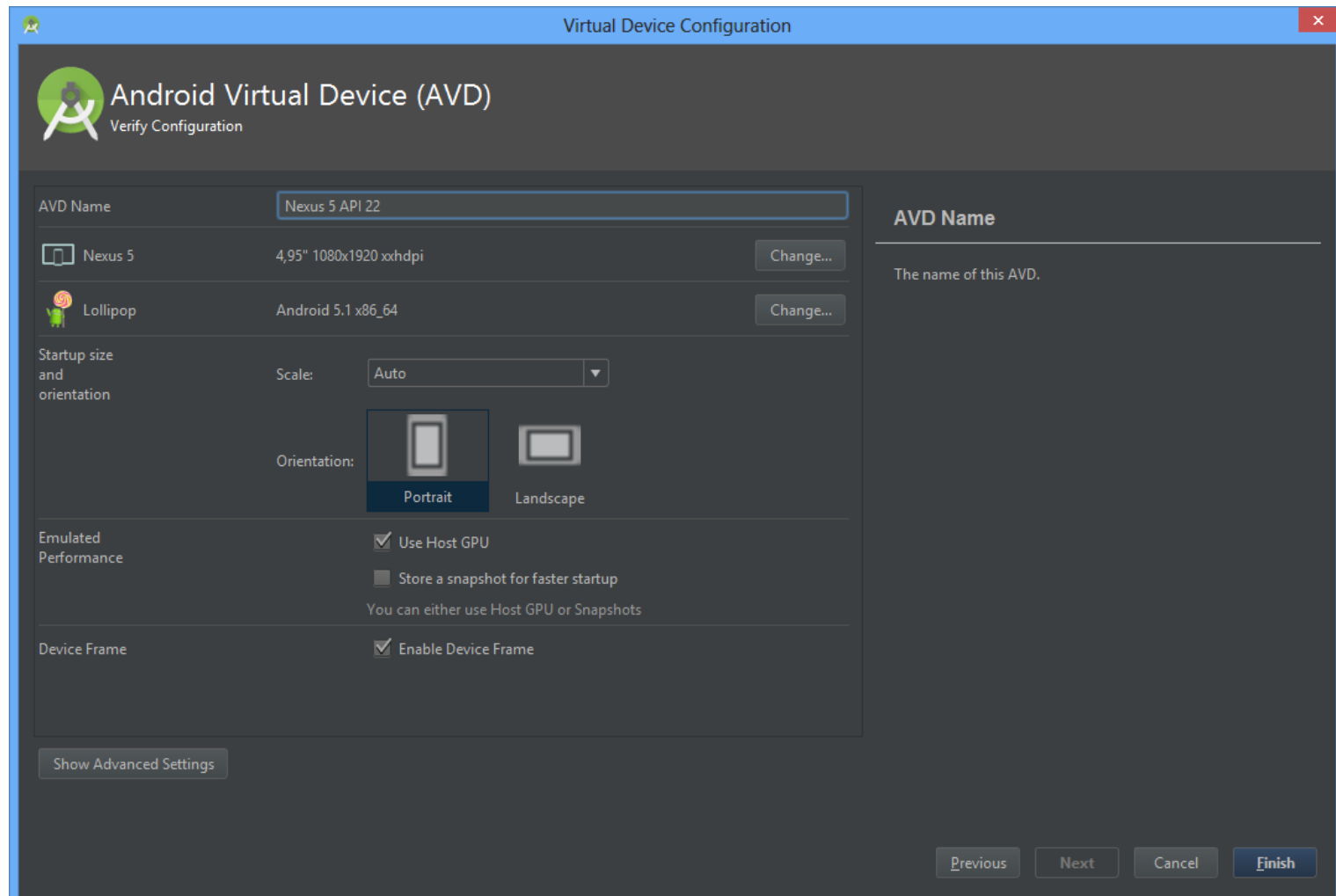
Probando la App

❖ Seleccionar la Imagen del SO



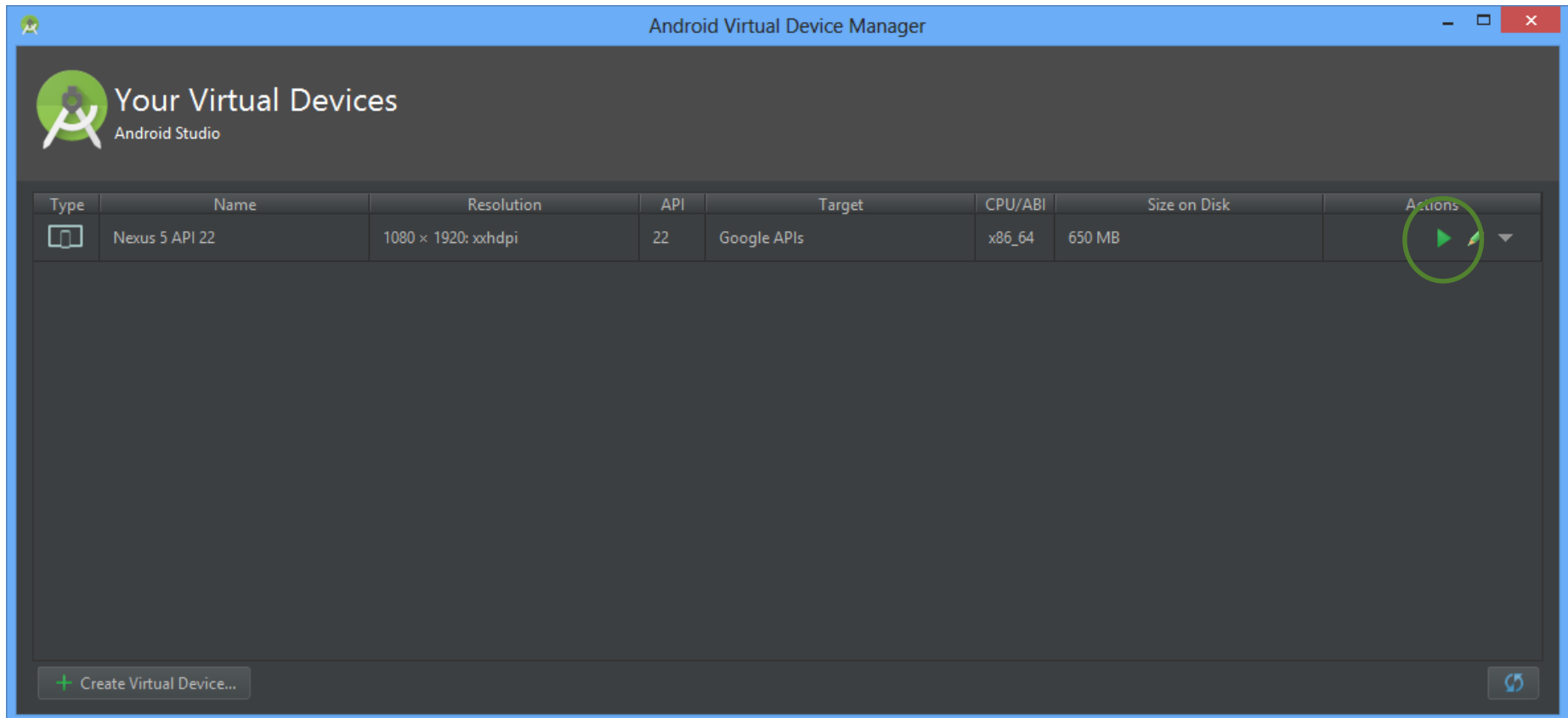
Probando la App

❖ 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/>

GENYMOTION^{OO} by  Features Docs Pricing Contact More 

Sign in

Build. Test. Flaunt. Support.
with the world's leading Android Emulator

Choose a plan

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](#).

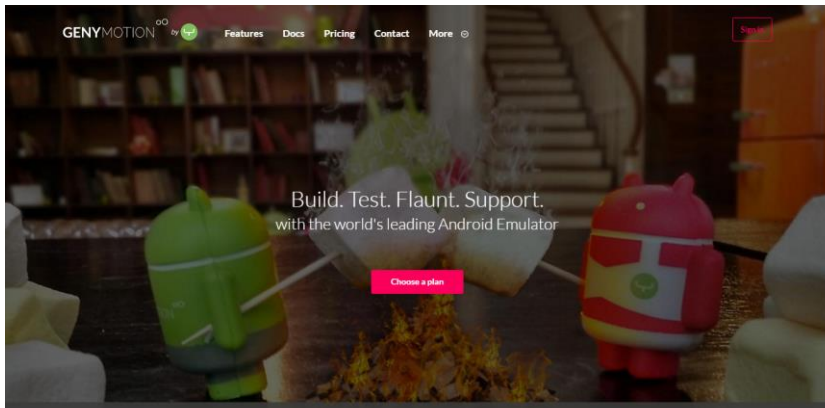
twitter yeip Spotify SAMSUNG EVERNOTE

I Agree


Probando la App

❖ Click en Sign In, luego en crear una cuenta

» Seleccionar Plan Individual -> GetStarted



Download Genymotion




with VirtualBox:

Download for Windows - 143.44 MB

without VirtualBox:

Download for Windows - 34.86 MB

[How to register my license](#)



System 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
2GB RAM

Individual	Enterprise
BASIC Great for start Free for personal use only Get started	INDIE Independent developers and small businesses from \$136 /year per business < 2 employees Get started

Probando la App

❖ También existe un plugin para Android Studio


GENYMOTION^{OO} by 

Features

Docs

Pricing

Contact

More 

Plugins

FAQ

Release Notes



Android Studio / IntelliJ IDEA

To install the Genymotion plugin for Android Studio

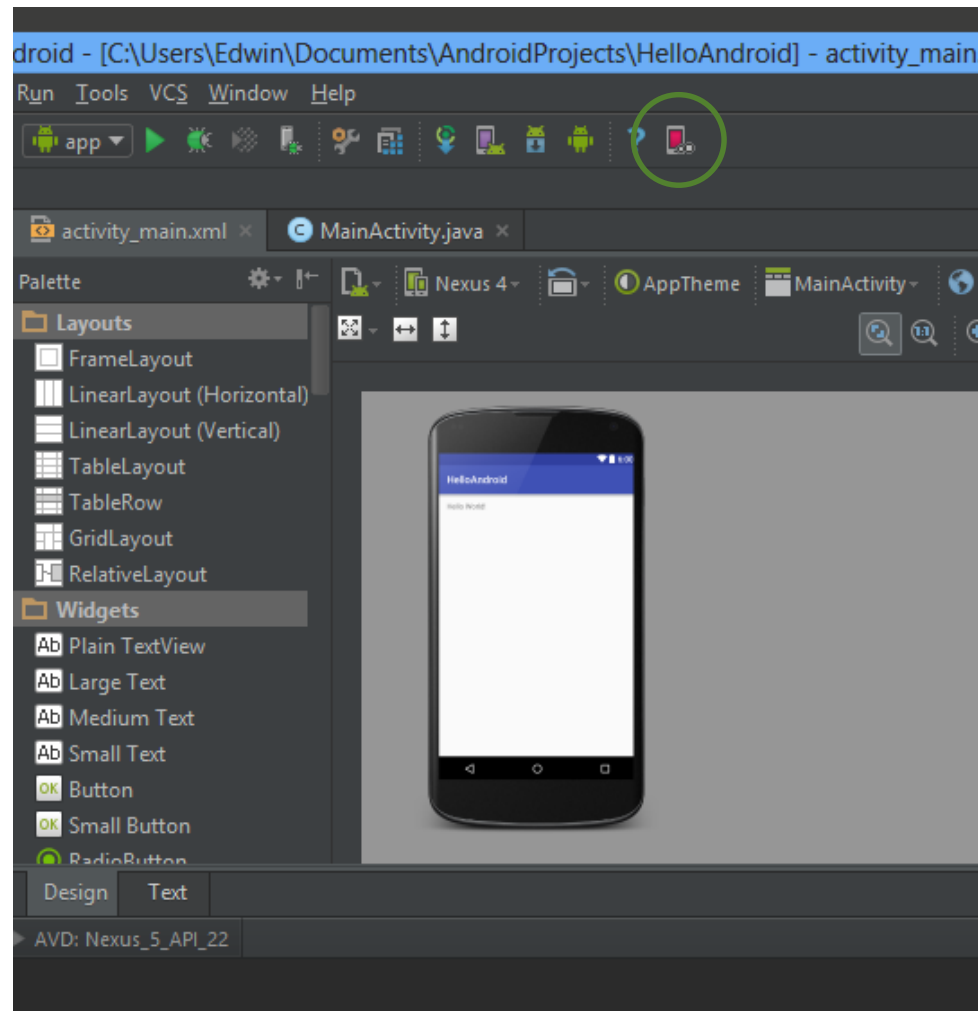
1. In Android Studio, go to **File/Settings** (for Windows and Linux) or to **Android Studio/Preferences** (for Mac OS X)
2. Select **Plugins** and click **Browse Repositories**.
3. Right-click on **Genymotion** and click **Download and install**. To see Genymotion plugin icon, display the toolbar by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

❗ To use this plugin, Genymotion must be installed on your system.

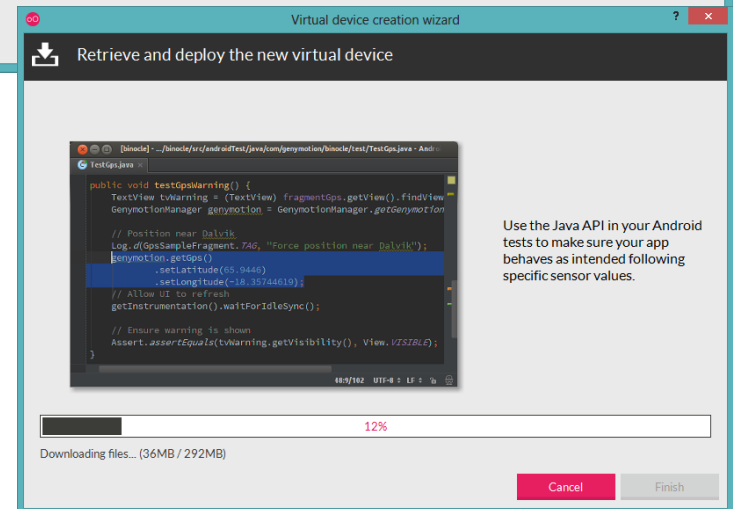
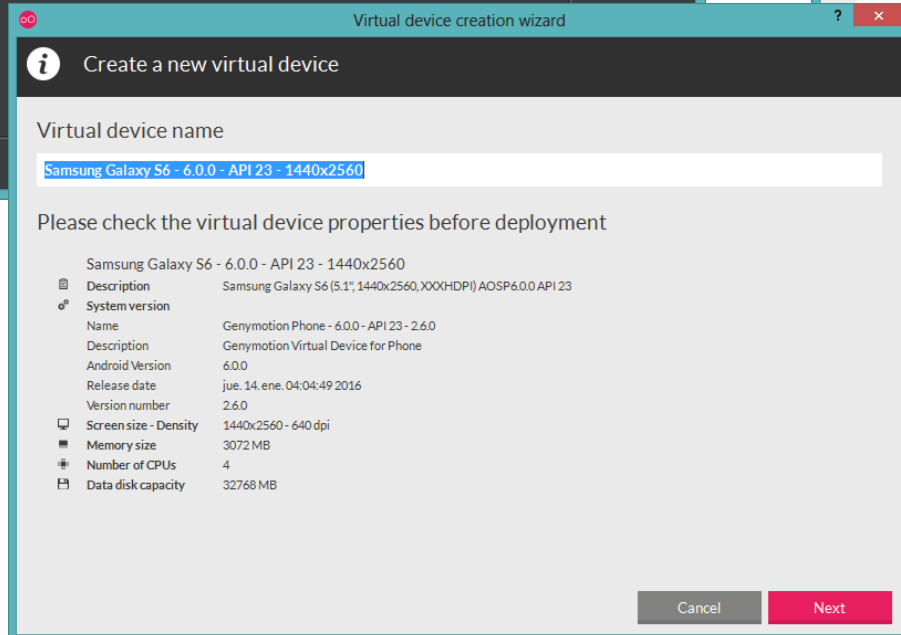
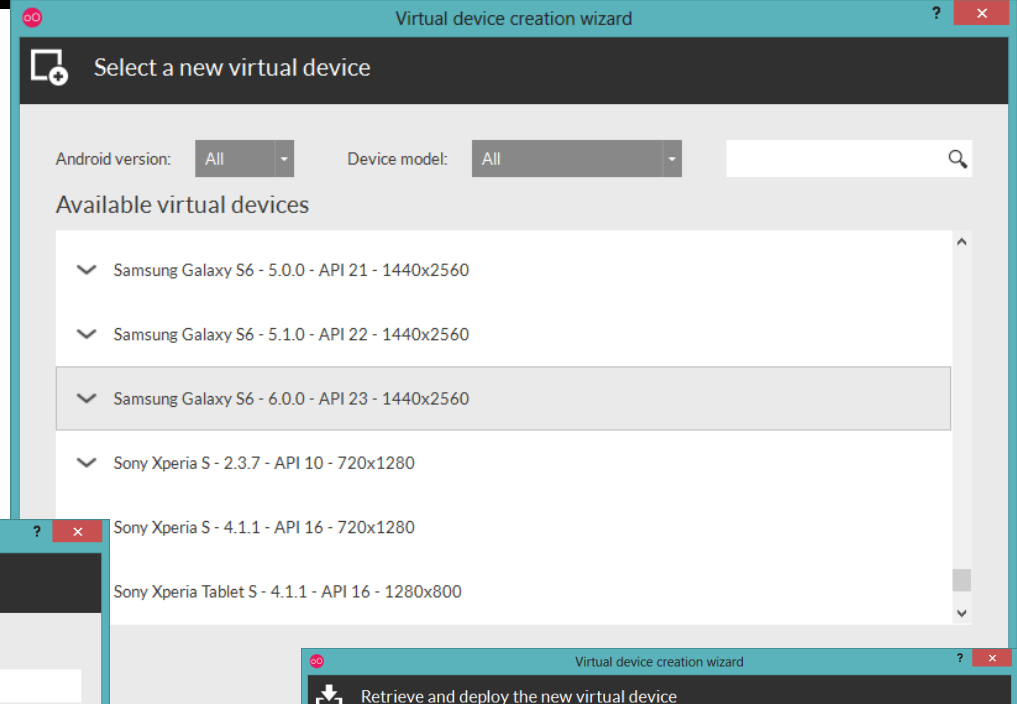
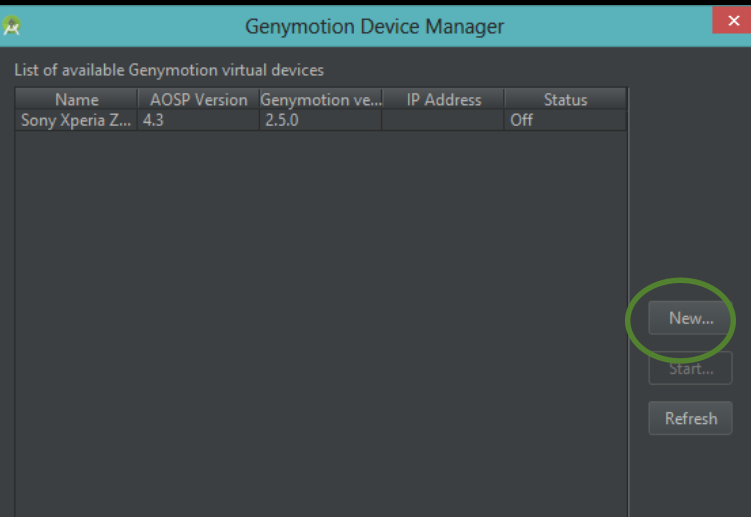
Or download and install manually the plugin [genymotion-idea-plugin.jar](#) (not recommended)

Probando la App

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

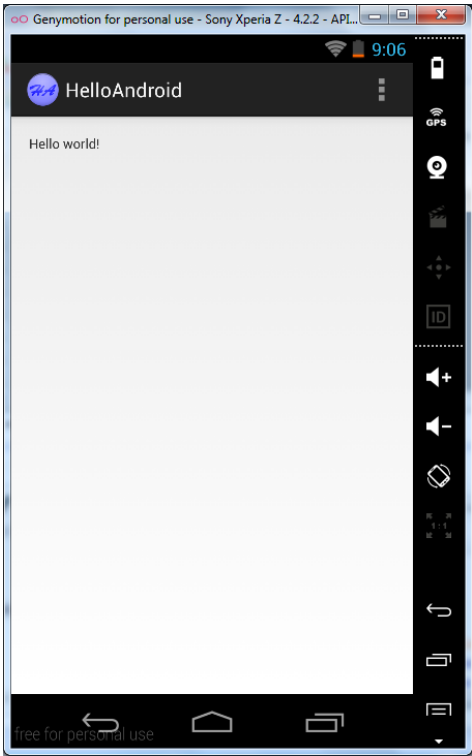
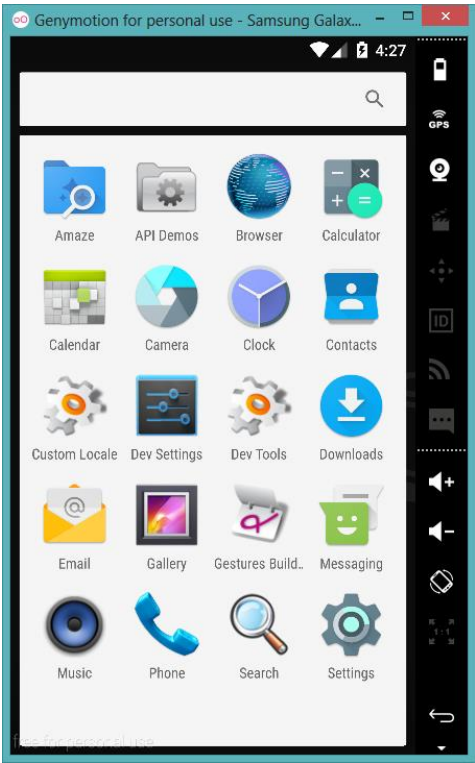
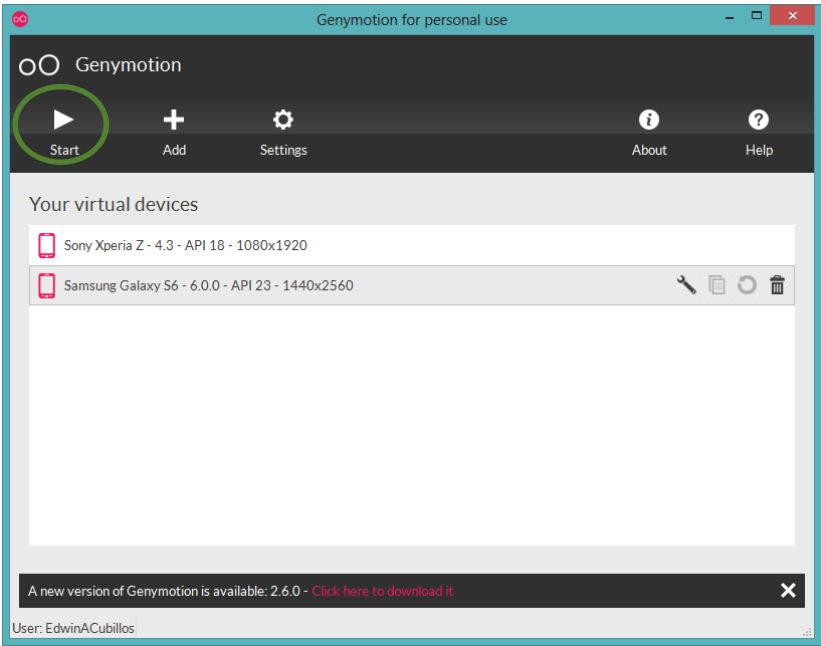


Probando la App



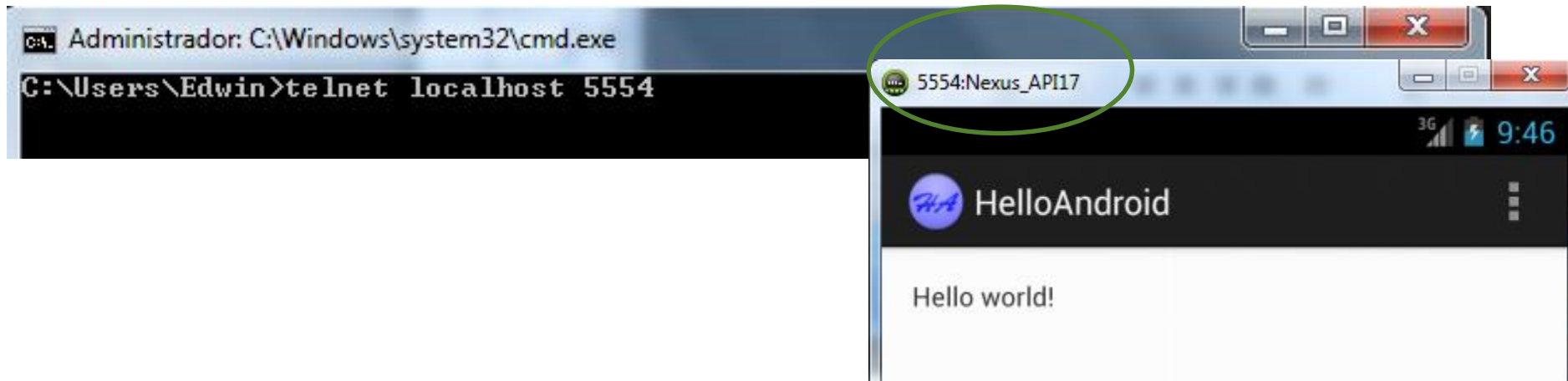


Probando la App



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



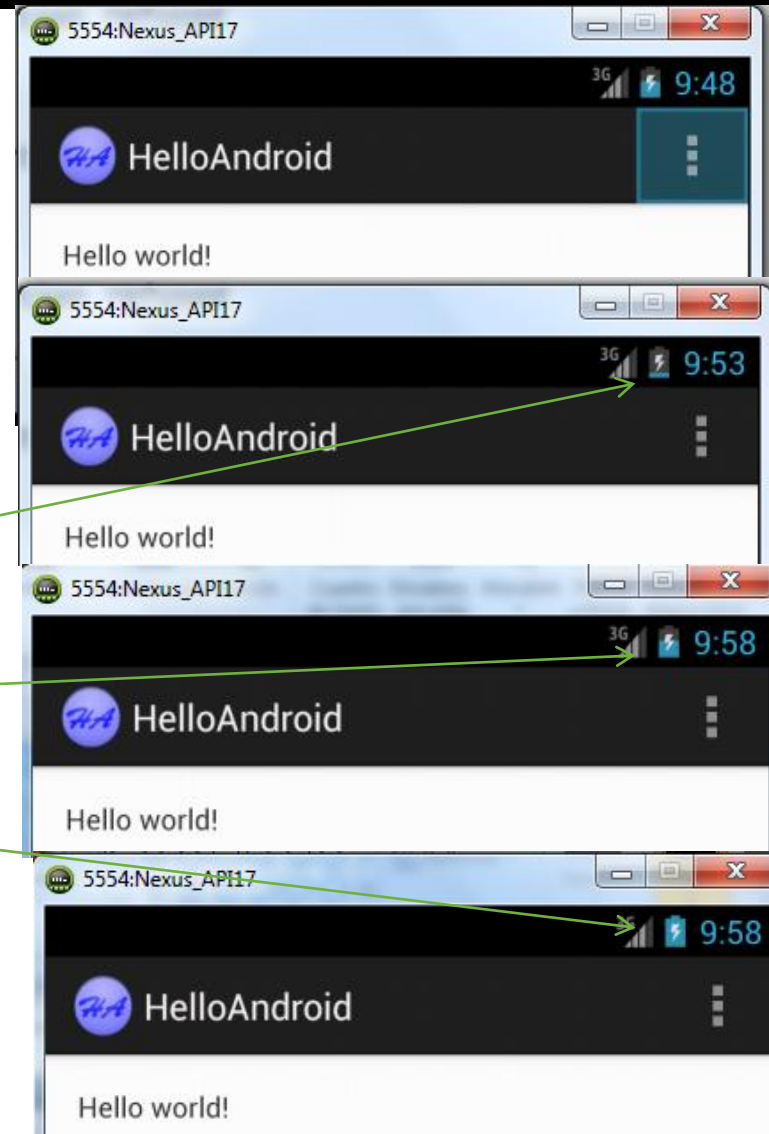


Probando la App

❖ Control de la batería

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>