
INTRODUCCIÓN

El desarrollo de Aplicaciones Móviles para el Sistema Operativo Android se realizaba frecuentemente en Eclipse diseñada por IBM, creado como un IDE agnóstico adaptable a través de Plugins con la desventaja de consumo de muchos recursos, para ello Google desarrollo su IDE Android Studio.

Este Libro te permite iniciarte en el lenguaje Java 8 y Desarrollo de Aplicaciones para dispositivos móviles en Android utilizando su nuevo IDE Android Studio basado en IntelliJ IDEA y el Nuevo SDK (Revisión 26.1.1) Android.



Android actualmente ha sufrido algunos cambios fundamentales que a continuación detallaremos:

- ▀ Utilización del Compilador GRADLE en reemplazo de ANT, Gradle es una herramienta de automatización y gestión de construcción basada en Java. Gradle reemplaza a “Ant”, que se utilizaba en Eclipse. Los antiguos

proyectos creados con eclipse deben convertirse de Ant a Gradle antes de poder importarse a Android Studio.



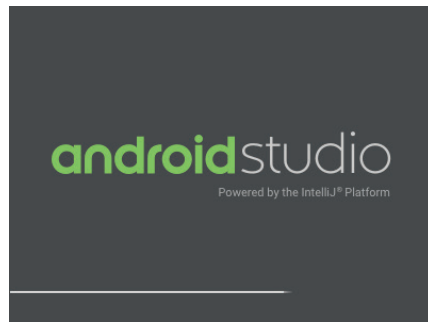
- Reemplazo de la Máquina Virtual DALVIK por ART. ART es la nueva máquina virtual de ejecución Android Introducido de forma experimental en la versión 4.4. Con Android 4.4, estamos empezando a lanzar una nueva forma de ejecución de Android ART. Este tiempo de ejecución ofrece una serie de nuevas características que mejoran el rendimiento de las Apps de Android. Actualmente, ART está disponible en una serie de Android 4.4 dispositivos como el Nexus 4, Nexus 5, Nexus 7 y Pixel.

Además, tocamos temas complementarios como son el uso de redes sociales, mapas y análisis forense, en este libro vendrá con las siguientes aplicaciones:

- Lector Código Barras.
- Suma.
- Alerta Ciudad.
- Pago.
- Rss.
- Rutas.
- Service.
- ControlLLamadas.
- Facebook.
- Ejemplo_Service.
- Mi Mapa.
- WebApp.
- Calculadora.
- Notificaciones.

1

INSTALACIÓN DE ANDROID STUDIO



1.1 INSTALACIÓN DE ANDROID STUDIO EN WINDOWS

1.1.1 Pre Requisitos





Las Aplicaciones Móviles de este Texto se han creado con Android Studio. Android Studio incluye Android SDK 5. El requisito para Android Studio es Java JDK para lo cual seguimos lo siguientes Pasos:

1. Ingrese a la siguiente URL para descargar el JAVA JDK

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>

Presione clic en JDK.

The screenshot shows the Oracle Java SE Downloads page. At the top, there are navigation tabs: Overview, Downloads, Documentation, Community, Technologies, and Training. The main heading is "Java SE Downloads". Below this is the Java logo and a "DOWNLOAD" button. Underneath, it says "Java Platform (JDK) 11". A section titled "Java Platform, Standard Edition" contains the text "Java SE 11.0.2(LTS)" and "Java SE 11.0.2 is the latest release for the Java SE 11 Platforms". There is a "Learn more" link. A list of links includes: Installation Instructions, Release Notes, Oracle JDK License, Java SE Licensing Information User Manual (with a sub-link for Includes Third Party Licenses), Certified System Configurations, and Readme. On the right side, there is a red-bordered box containing the text "Oracle JDK" and a "DOWNLOAD" button.

Aceptamos el acuerdo de Licencia.

Java SE Development Kit 11.0.2

You must accept the [Oracle Technology Network License Agreement for Oracle Java SE](#) to download this software.

Accept License Agreement
 Decline License Agreement

Product / File Description	File Size	Download
Linux	147.28 MB	jdk-11.0.2_linux-x64_bin.deb
Linux	154.01 MB	jdk-11.0.2_linux-x64_bin.rpm
Linux	171.32 MB	jdk-11.0.2_linux-x64_bin.tar.gz
macOS	166.13 MB	jdk-11.0.2_osx-x64_bin.dmg
macOS	166.49 MB	jdk-11.0.2_osx-x64_bin.tar.gz
Solaris SPARC	186.78 MB	jdk-11.0.2_solaris-sparcv9_bin.tar.gz
Windows	150.94 MB	jdk-11.0.2_windows-x64_bin.exe
Windows	170.96 MB	jdk-11.0.2_windows-x64_bin.zip

Descargamos JDK para 64 bits de Windows.

Java SE Development Kit 11.0.2

You must accept the [Oracle Technology Network License Agreement for Oracle Java SE](#) to download this software.

Thank you for accepting the Oracle Technology Network License Agreement for Oracle Java SE; you may now download this software.

Product / File Description	File Size	Download
Linux	147.28 MB	jdk-11.0.2_linux-x64_bin.deb
Linux	154.01 MB	jdk-11.0.2_linux-x64_bin.rpm
Linux	171.32 MB	jdk-11.0.2_linux-x64_bin.tar.gz
macOS	166.13 MB	jdk-11.0.2_osx-x64_bin.dmg
macOS	166.49 MB	jdk-11.0.2_osx-x64_bin.tar.gz
Solaris SPARC	186.78 MB	jdk-11.0.2_solaris-sparcv9_bin.tar.gz
Windows	150.94 MB	jdk-11.0.2_windows-x64_bin.exe
Windows	170.96 MB	jdk-11.0.2_windows-x64_bin.zip

i NOTA

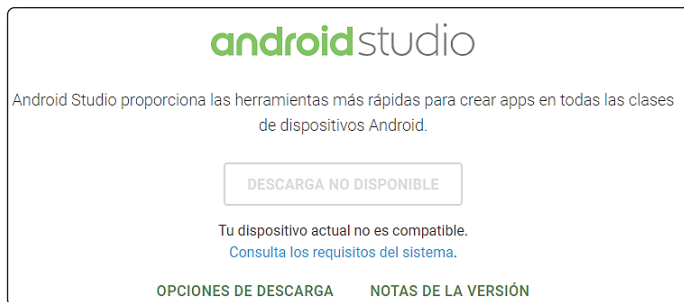
Luego de descargarlo lo Instalamos

1.1.2 Inicio de la Instalación de Android Studio

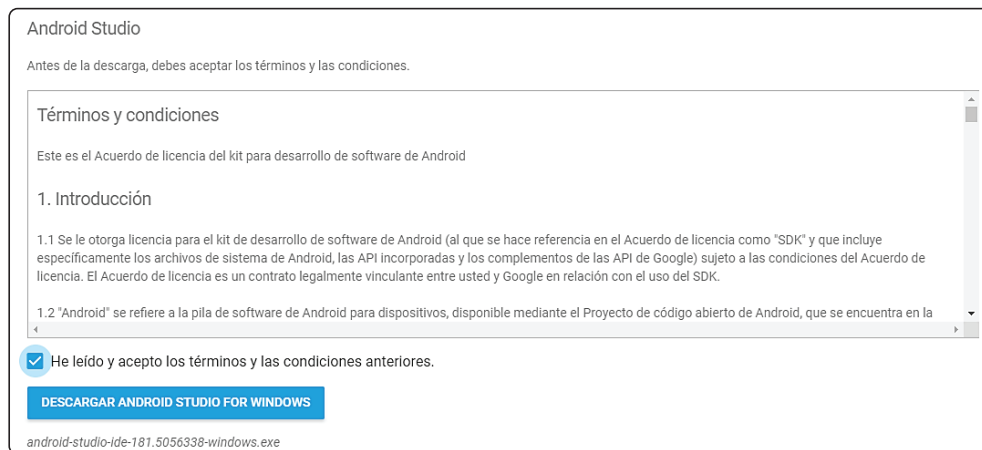
Para descargar la última versión para Windows de Android Studio puede descargar de la siguiente URL:

<http://developer.android.com/sdk/installing/studio.html>

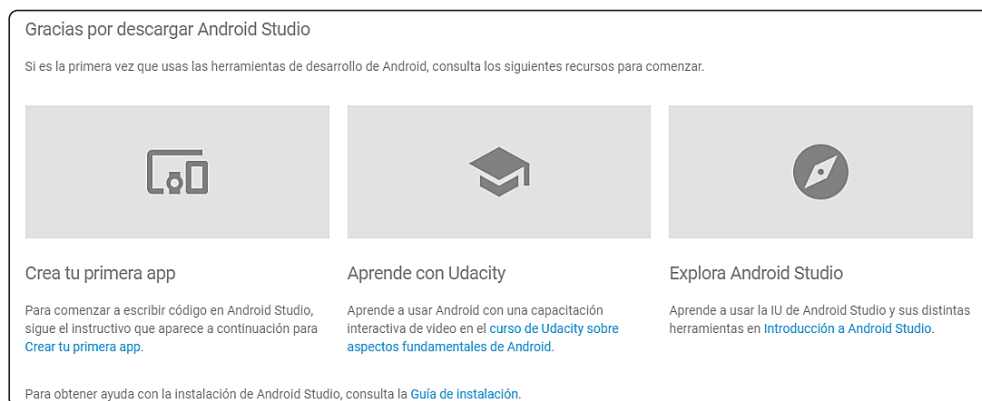
A continuación, ingresara a la siguiente pantalla y presionara clic en el botón Download Android Studio for Windows.

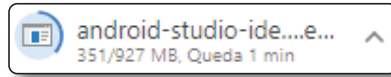


Acepte las condiciones y comience la descarga.

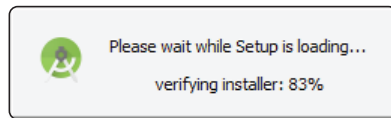


Espere que concluya la descarga:





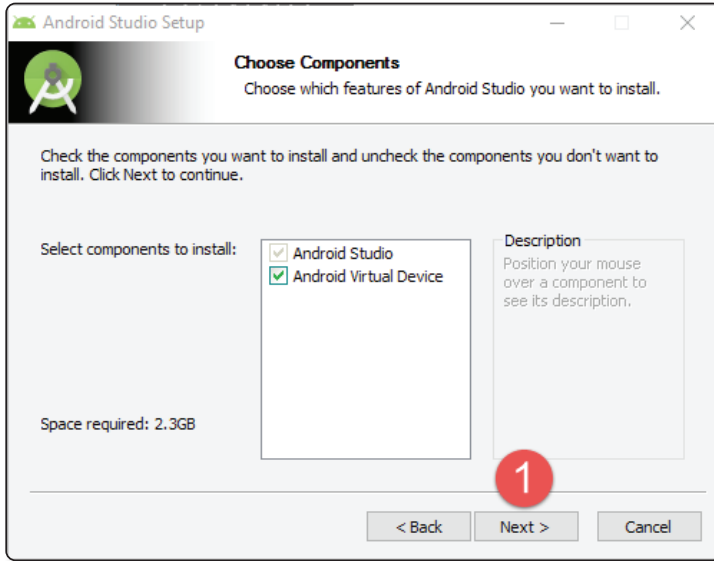
Una vez realizada la descarga, proceder a instalar, En Windows ejecute el archivo EXE:



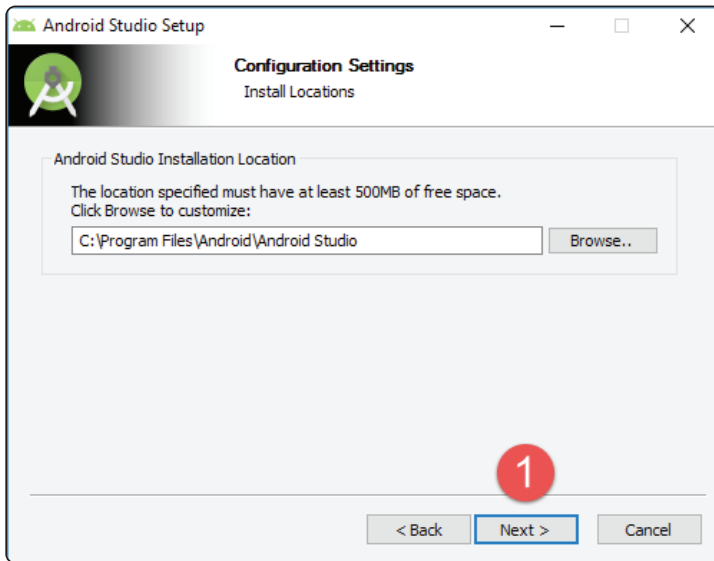
Presionar el Botón Next (Siguiete):



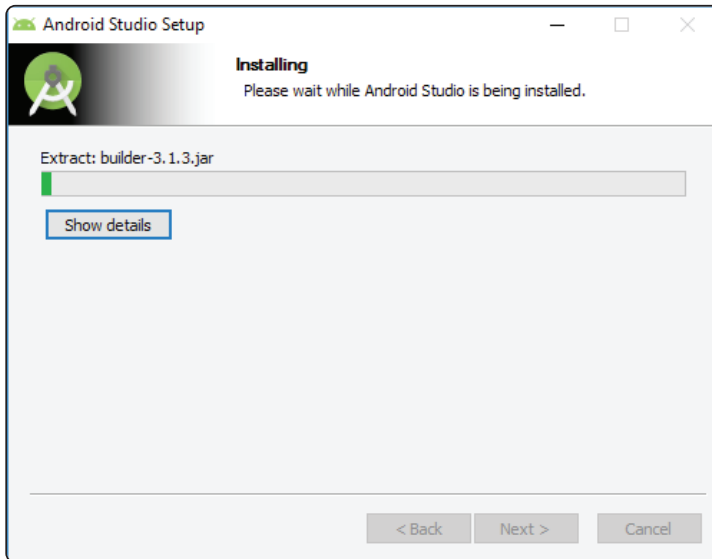
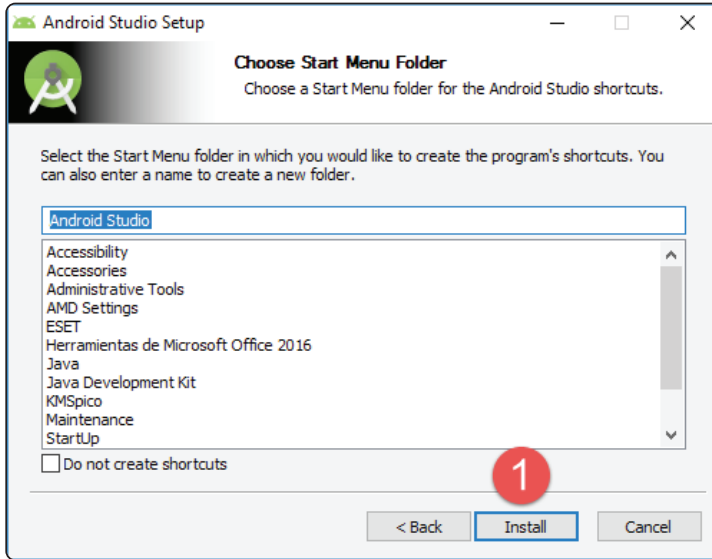
Presionar el Botón Next (Siguiete):



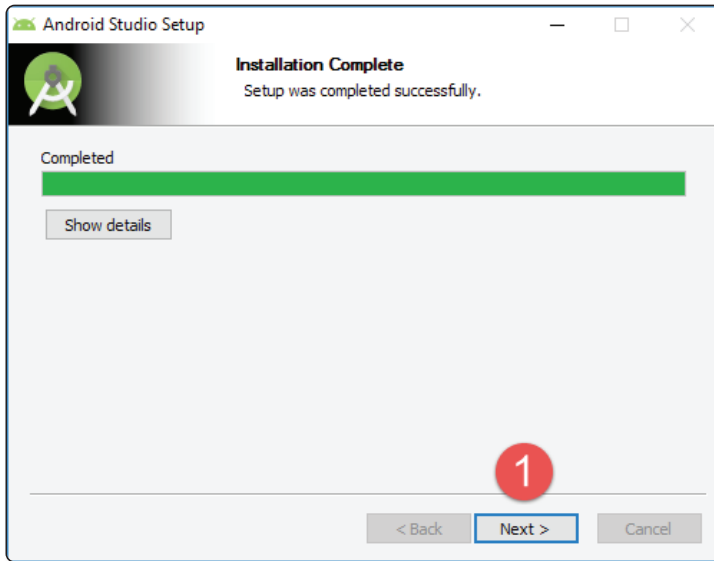
Presionar el Botón Next (Siguiete):



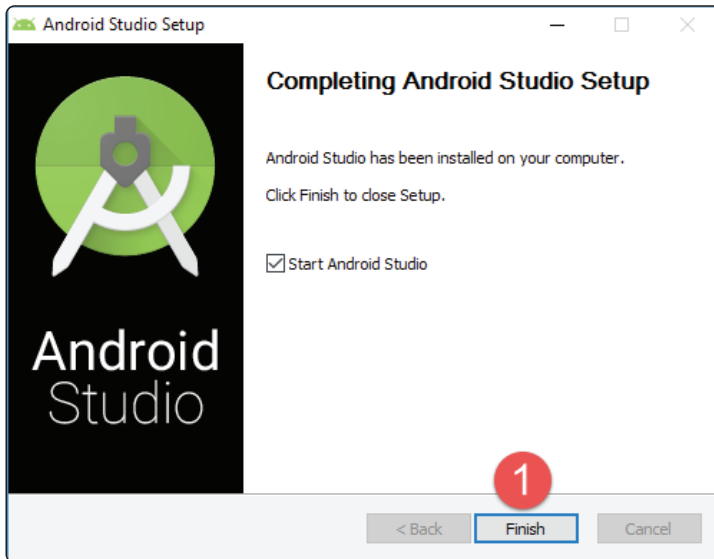
Presionar el Botón Install (Instalar):



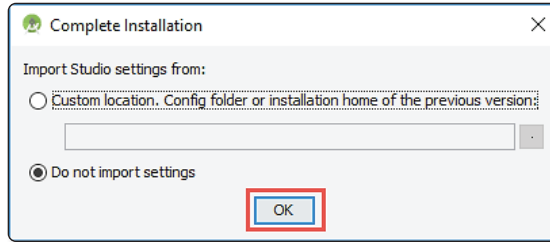
Presionar el Botón Next (Siguiente):



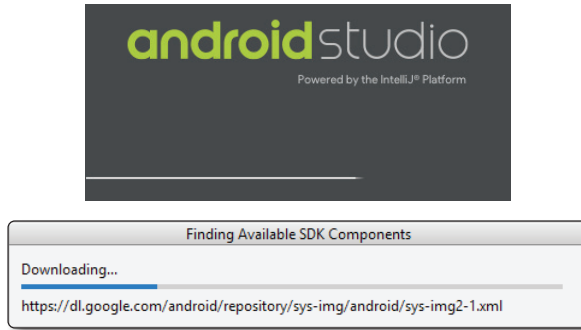
Presionar el Botón Finish (Finalizar):



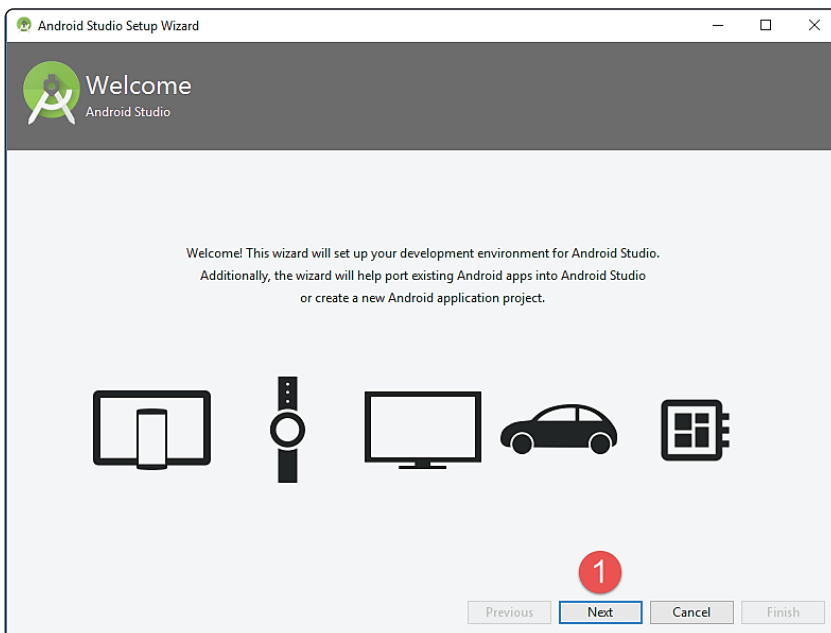
Presionar el Botón Ok:



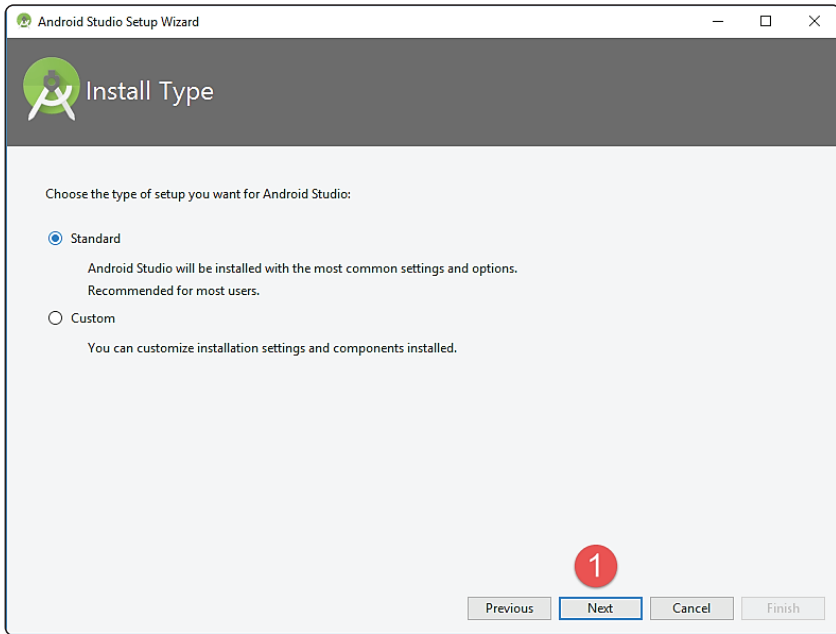
Espere que cargue:



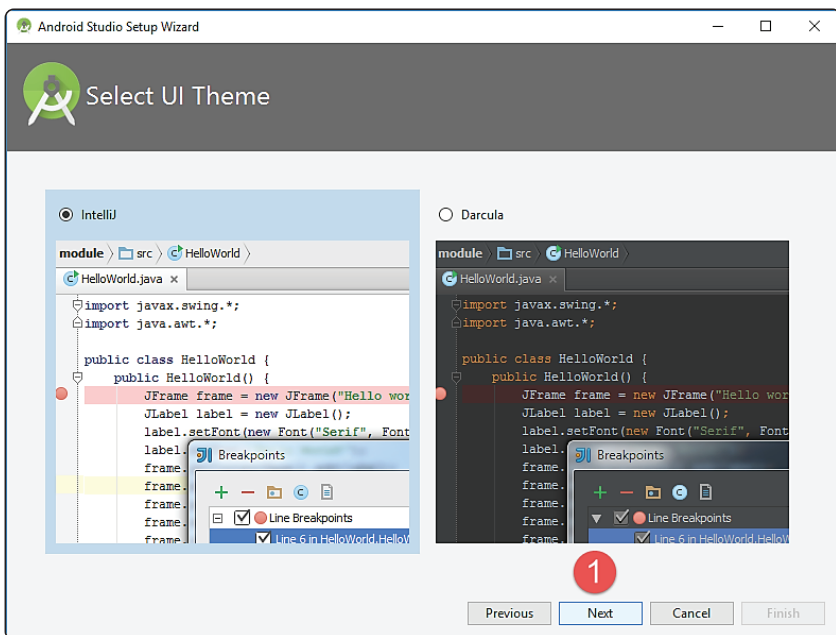
Presionar el Botón Next (Siguiente):



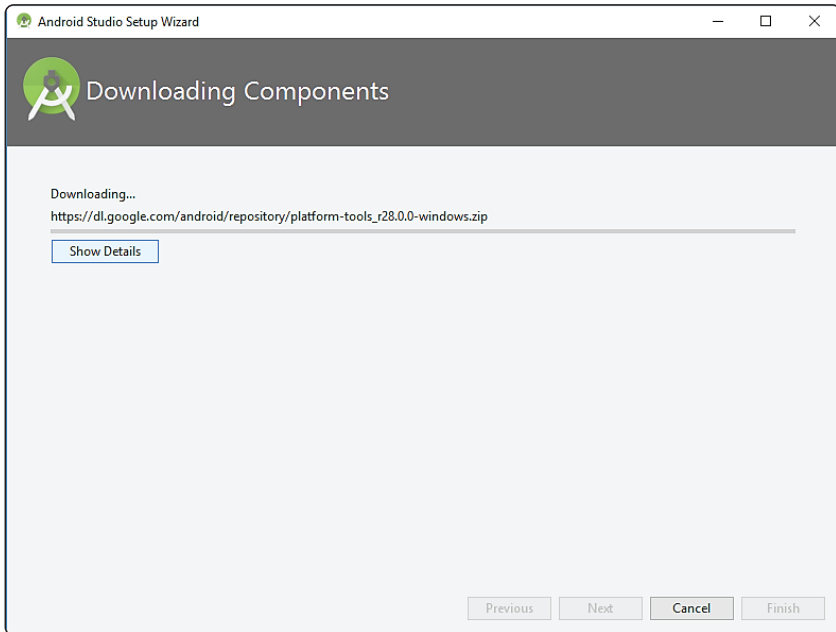
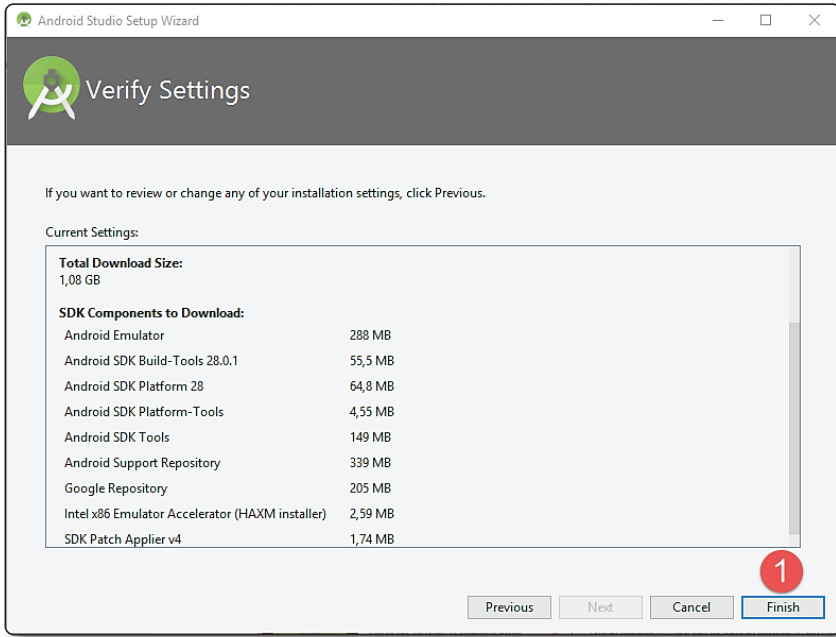
Presionar el Botón Next (Siguiente):



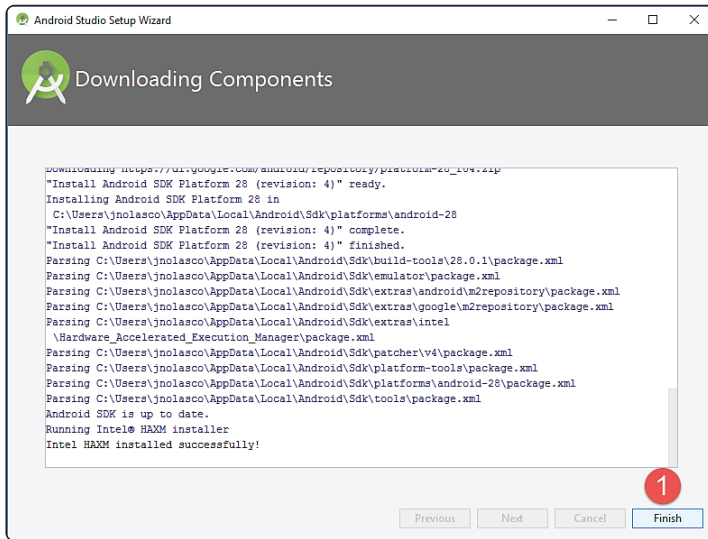
Presionar el Botón Next (Siguiente):



Presionar el Botón Finish (Finalizar):

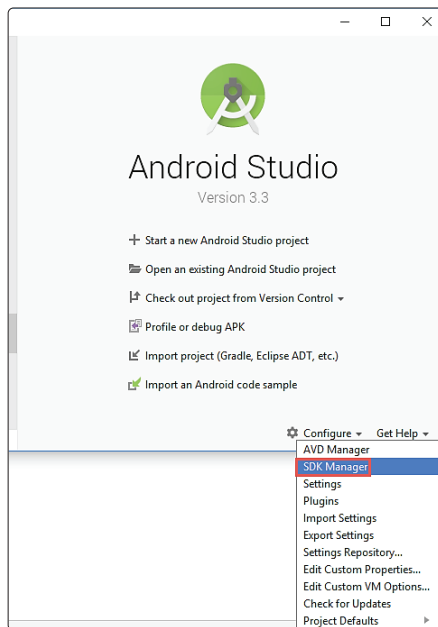


Presionar el Botón Finish (Finalizar):

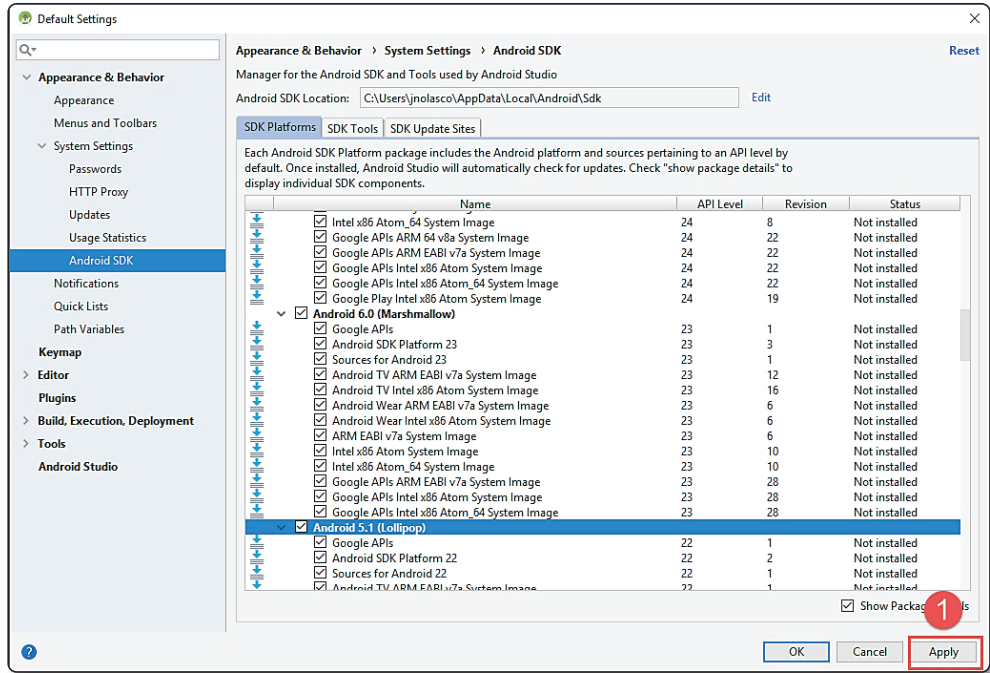


1.2 CONFIGURACIÓN DEL ANDROID STUDIO

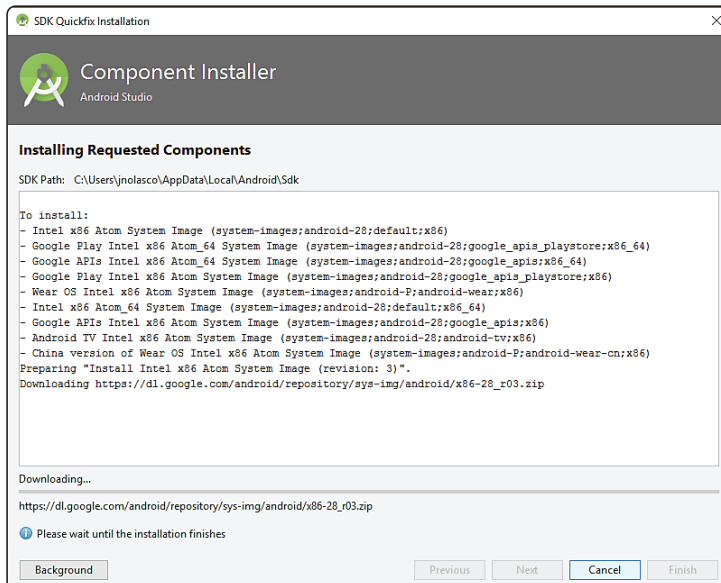
El paquete de Android Studio descargado incluye el último Android SDK. Antes de empezar con tu primer proyecto Android, se debe configurar Android Studio. Para ello debemos presionar clic en Configure:



Luego debemos seleccionar en que versiones del sistema operativo se dará soporte por ejemplo deseo dar soporte a la versión 4.X y luego presiones el botón Install:



Espera que finalice:


















NOTA

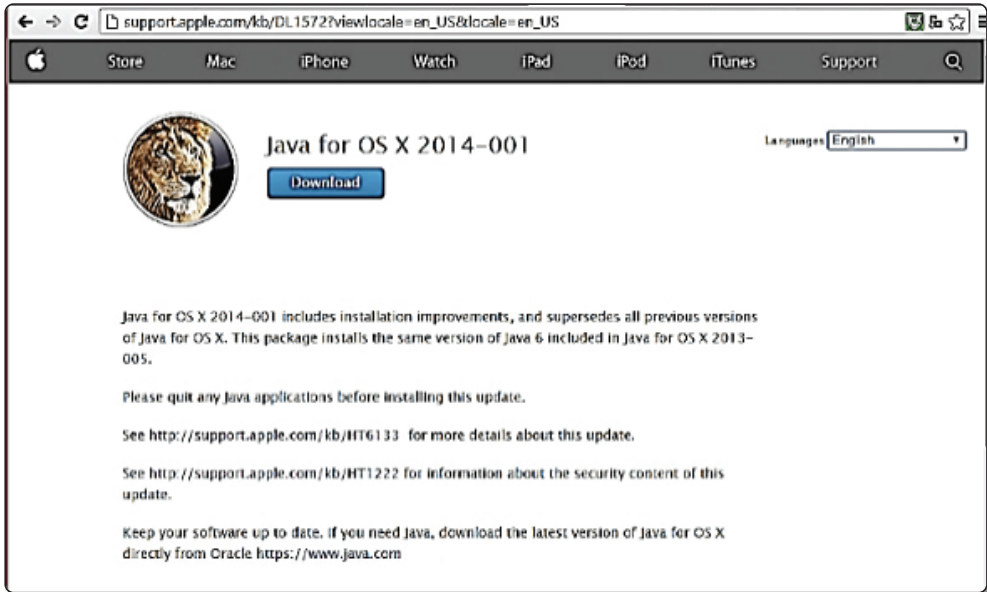
Recomendamos dar soporte a versiones antiguas según la estadística de versiones.

1.3 INSTALACIÓN DE ANDROID STUDIO EN OS X

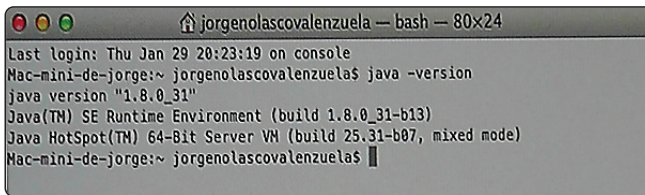
- Mac® OS X® 10.8.5 o superior 10.9 (Mavericks), Recomendable 10.9.4:
- 2 GB RAM minimum, 4 GB RAM Recomendable.
- 400 MB Espacio de Disco Duro.
- 1280 x 800 mínimo de resolución.
- Descargar Java Runtime Environment (JRE) 6, del siguiente Link:
<http://www.oracle.com/technetwork/java/javase/downloads/jre7-downloads-1880261.html>.

Java SE Runtime Environment 7u75		
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.		
Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.		
Product / File Description	File Size	Download
Linux x86	31.58 MB	 jre-7u75-linux-i586.rpm
Linux x86	46.24 MB	 jre-7u75-linux-i586.tar.gz
Linux x64	32.1 MB	 jre-7u75-linux-x64.rpm
Linux x64	44.87 MB	 jre-7u75-linux-x64.tar.gz
Mac OS X x64	48.58 MB	 jre-7u75-macosx-x64.dmg
Mac OS X x64	44.53 MB	 jre-7u75-macosx-x64.tar.gz
Solaris x86	52.25 MB	 jre-7u75-solaris-i586.tar.gz
Solaris x64	16.15 MB	 jre-7u75-solaris-x64.tar.gz
Solaris SPARC	54.89 MB	 jre-7u75-solaris-sparc.tar.gz
Solaris SPARC 64-bit	18.12 MB	 jre-7u75-solaris-sparcv9.tar.gz
Windows x86 Online	0.89 MB	 jre-7u75-windows-i586-iftw.exe
Windows x86 Offline	28.1 MB	 jre-7u75-windows-i586.exe
Windows x86	40.02 MB	 jre-7u75-windows-i586.tar.gz
Windows x64	29.6 MB	 jre-7u75-windows-x64.exe
Windows x64	41.73 MB	 jre-7u75-windows-x64.tar.gz

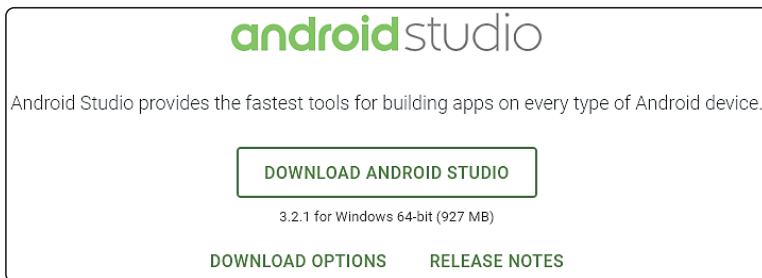
Java Runtime Environment (JRE) 6, del siguiente Link:



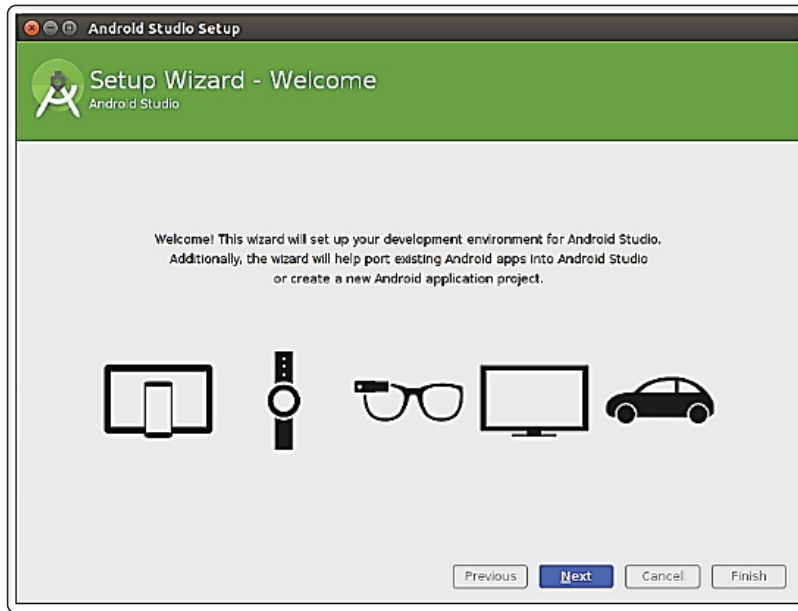
Opcionalmente acelerador emulador: Intel® VT-x, Intel® EM64T (Intel® 64)
Es necesario verificar la instalación del JDK, mediante el terminal.



Inicio de la Instalación de Android Studio, Descargarnos del siguiente Link
Android Studio para Mac:<http://developer.android.com/sdk/index.html>



Lo arrastramos a la carpeta de aplicaciones:



Inicio de la Instalación de Android Studio, Descargarnos del siguiente Link
Android Studio para Mac:<http://developer.android.com/sdk/index.html>.

1.4 OTROS SISTEMAS OPERATIVOS – 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.

1.5 OTRAS OPCIONES DE DESCARGA

Android Studio downloads

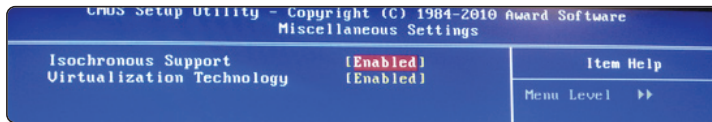
Platform	Android Studio package	Size	SHA-256 checksum
Windows (64-bit)	android-studio-ide-173.4819257-windows.exe Recommended	758 MB	2d11cd16ffefc7f4aca82bd95b0d0ca849a854a07ba1a353adf65dfc102aee9b
	android-studio-ide-173.4819257-windows.zip No .exe installer	855 MB	6941761a9324998d9cdf5d7548ff16fcb65c9e71ea70e1bc75d066b51b77c7dd
Windows (32-bit)	android-studio-ide-173.4819257-windows32.zip No .exe installer	854 MB	85c9fb33c94183abcb70caac34ea034214a079276fb1b2e10bc7b4ed71c05cf0
Mac	android-studio-ide-173.4819257-mac.dmg	849 MB	d4a8502c5aabfc5477f30dffe296bf705bd7e62650a76796b646a8f28b5e5c
Linux	android-studio-ide-173.4819257-linux.zip	853 MB	d86748e44d658fd39581b40f7b706fb397fc1eca5dd6f8066a56c0beb856dea4

1.6 ALGUNAS RECOMENDACIONES AL UTILIZAR EL EMULADOR

```
emulator: Failed to open the HAX device!
HAX is not working and emulator runs in emulation mode
emulator: Open HAX device failed
```

En algunas oportunidades existe inconvenientes en la ejecución del emulador para ello deberá verificar lo siguiente:

1.6.1 Activar en la Bios Tecnología de Virtualización



1.6.2 Instalación IntelHaxm.exe

Ubicado en la siguiente Carpeta:
 [Android SDK Root] \ extras \ intel \ Hardware_Accelerated_Execution_Manager.



2

CREACIÓN DE MI PRIMERA APP

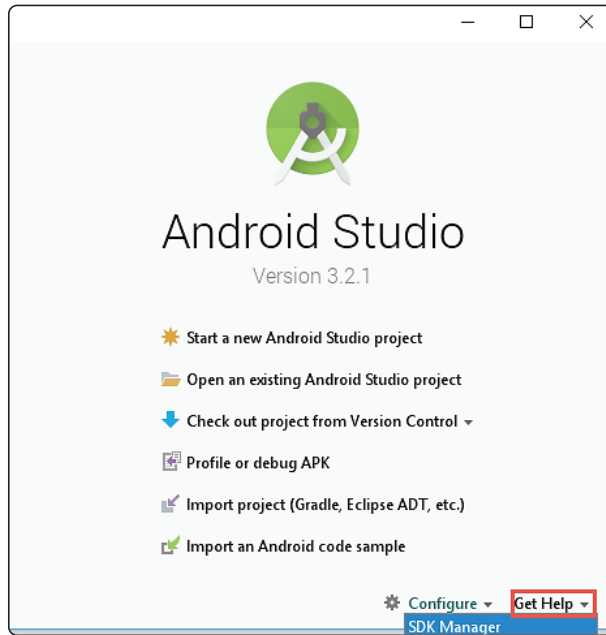


2.1 CREANDO LA PRIMERA APLICACIÓN

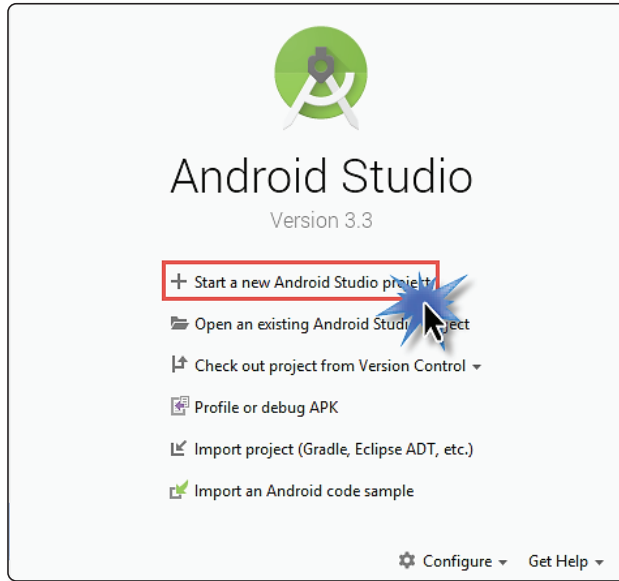
Ante de crear la primera App describiremos las distintas opciones:

The image shows the start screen of Android Studio Version 3.3. The screen has a light grey background with the Android Studio logo at the top left. Below the logo, the text "Android Studio" and "Version 3.3" are displayed. There are six main options listed with icons: "Start a new Android Studio project", "Open an existing Android Studio project", "Check out project from Version Control", "Profile or debug APK", "Import project (Gradle, Eclipse ADT, etc.)", and "Import an Android code sample". At the bottom, there are links for "Configure" and "Get Help". To the right of the screen, there are six red callout boxes with white text, each connected to one of the options by a red line. The callouts are: "INICIO O CREACION DE UN NUEVO PROYECTO" (pointing to "Start a new..."), "ABRIR UN PROYECTO EXISTENTE" (pointing to "Open an existing..."), "CONTROL DE VERSIONES" (pointing to "Check out project..."), "DEPURAR APK" (pointing to "Profile or debug APK"), "IMPORTAR PROYECTO CREADO EN VERSIONES ANTIGUAS O EN ECLIPSE" (pointing to "Import project..."), and "IMPORTAR PROYECTO EXISTENTE" (pointing to "Import an Android code sample").

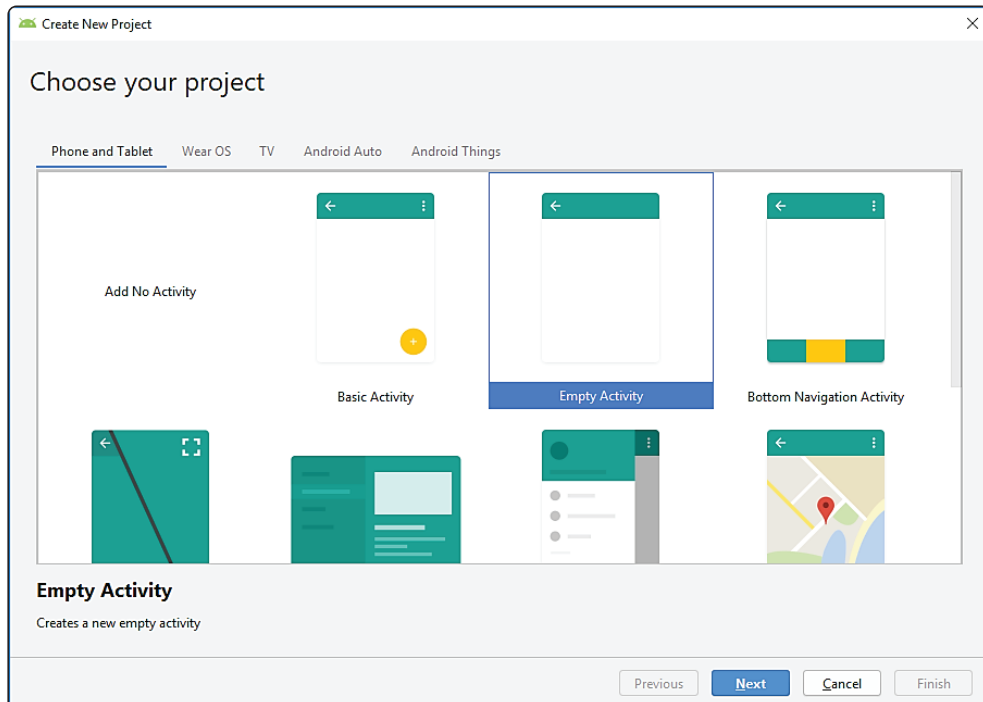
Menú de Ayuda:



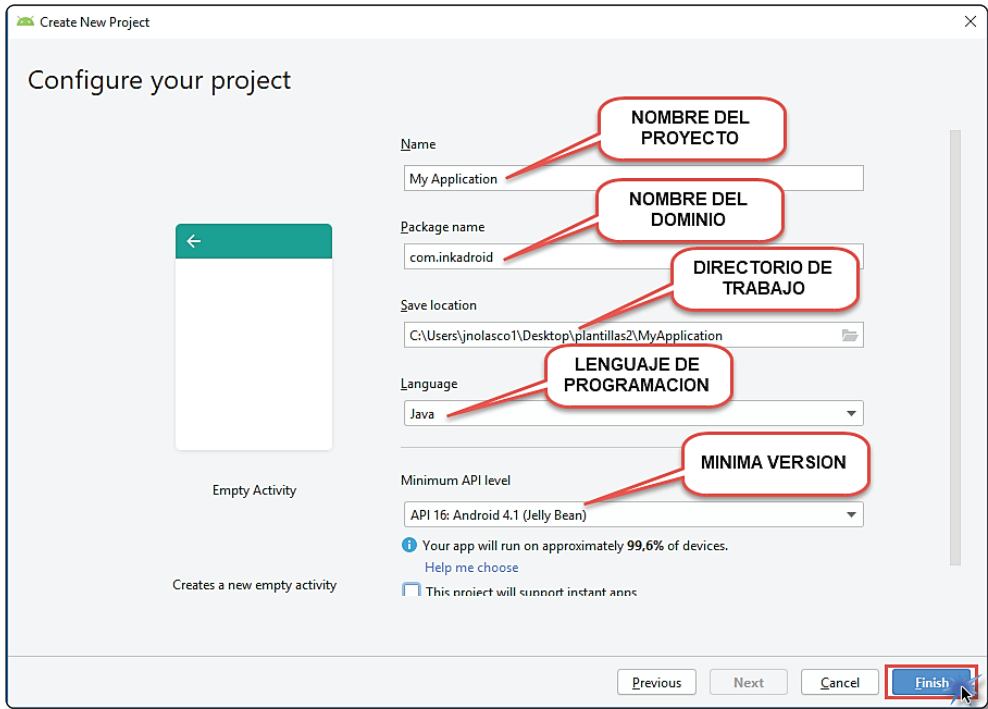
Para comenzar a crear un nuevo proyecto haga clic en Start a new Android Studio Project (Inicie un nuevo proyecto Android Studio):



Ahora elegimos usar actividad en Blanco y luego presionamos el botón Next (Siguiete):





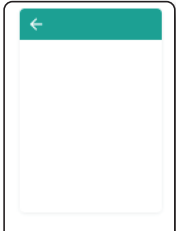
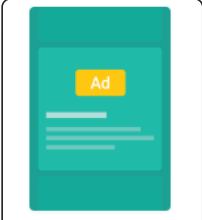
Ahora comenzamos a crear el proyecto indicando los siguientes datos y luego presionamos el botón Finish (Finalizar):



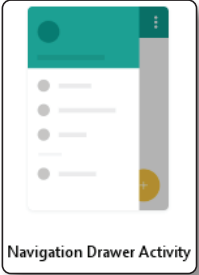
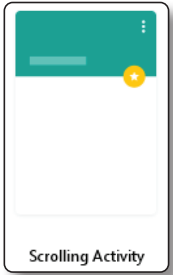
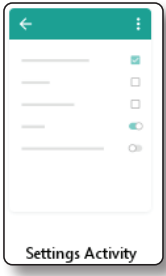

Más sobre elección de su tipo de actividad

Este paso le permite crear la actividad principal de su aplicación, entre los diferentes tipos de actividad tenemos los siguientes:

<p>Basic Activity: Esto crea una actividad en blanco con un botón Flotantes.</p>	
---	--

<p>Bottom Navigation Activity: Esto crea una actividad con una barra de navegación inferiores facilitan la exploración y el cambio entre las vistas de nivel superior con un solo toque.</p>	 <p>Bottom Navigation Activity</p>
<p>Empty Activity: Esto crea una actividad en blanco con una barra de acción. La barra de acción incluye un título y un menú de opciones.</p>	 <p>Empty Activity</p>
<p>Fragment + ViewModel: La ViewModelclase está diseñada para almacenar y administrar datos relacionados con la IU en una forma consciente del ciclo de vida. La ViewModelclase permite que los datos sobrevivan a cambios de configuración, como rotaciones de pantalla.</p>	 <p>Fragment + ViewModel</p>
<p>Google AdMob Ads Activity : esta plantilla utiliza AdMob como un servicio de Google para integrar anuncios en tus aplicaciones Android, con el fin de monetizar la actividad de tu audiencia a través de redes de publicidad (<i>ad networks</i>).</p>	 <p>Google AdMob Ads Activity</p>

<p>Fullscreen Activity: Esta plantilla oculta la interfaz de usuario del sistema (tales como la barra notificación) en una vista de pantalla completa. El modo de pantalla completa se alterna con una barra de acción que aparece cuando el usuario toca la pantalla del dispositivo.</p>	 <p>Fullscreen Activity</p>
<p>Google map activity: Esta plantilla Activity de Google Maps te ayuda a reducir el proceso para incluir en nuestra aplicación la API de Google Maps Android.</p>	 <p>Google Maps Activity</p>
<p>Login activity: Esta plantilla crea su vista como una pantalla de conexión o validación que permite a los usuarios iniciar sesión o registrarse con una dirección de correo y contraseña.</p>	 <p>Login Activity</p>
<p>Master/Detail Flow: Esta plantilla divide la pantalla en dos partes: a la izquierda menú y los detalles del elemento seleccionado a la derecha. En una pantalla más pequeña, Se muestra sólo una sección, pero en una pantalla más grande, ambas secciones son visualizados al mismo tiempo.</p>	 <p>Master/Detail Flow</p>

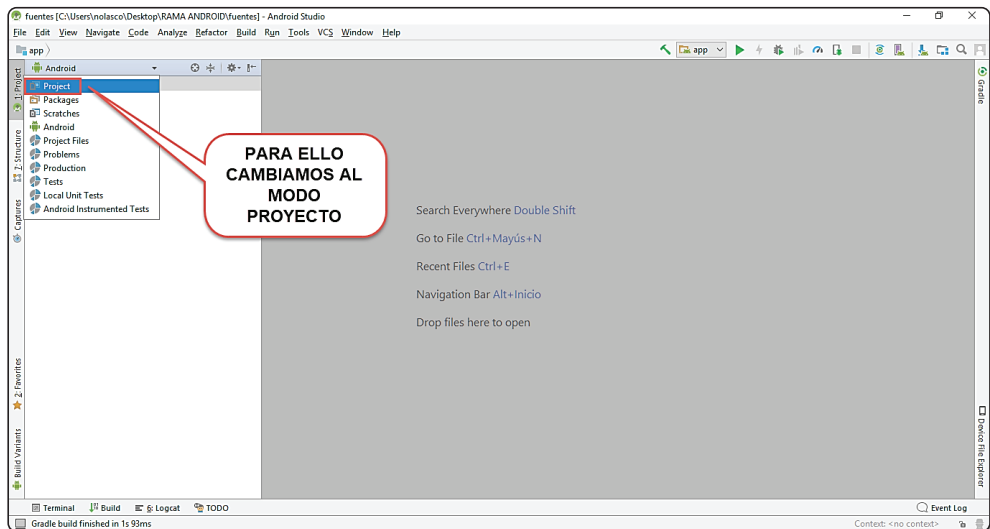
<p>Navigation Drawer Activity : Esto crear una actividad con un panel lateral de navegación es un panel en el que se muestran las principales opciones de navegación de la app en el borde izquierdo de la pantalla. La mayor parte del tiempo está oculto, pero aparece cuando el usuario desliza un dedo desde el borde izquierdo de la pantalla o, mientras está en el nivel superior de la app, el usuario toca el ícono de la app en la barra de acciones.</p>	 <p>Navigation Drawer Activity</p>
<p>El Scrolling Activity : Esto crea una actividad con una interfaz donde se expone un título llamativo con una imagen y seguido a este un contenido desplazable.</p>	 <p>Scrolling Activity</p>
<p>Settings Activity: Esto crea una actividad de preferencia con una lista de ajustes.</p>	 <p>Settings Activity</p>
<p>Tabbed Activity : Esto crea una actividad con una serie de pestañas.</p>	 <p>Tabbed Activity</p>

2.1.1 Conociendo el Entorno de Trabajo

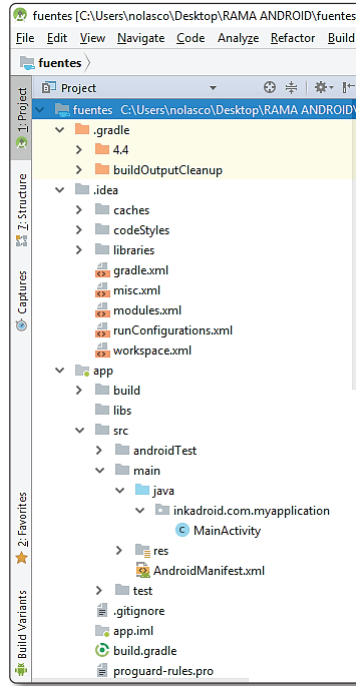


2.1.2 Conociendo la Estructura del Proyecto

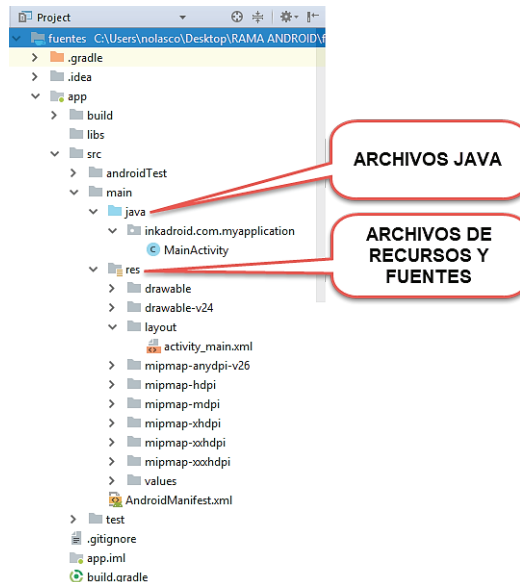
Al crear un proyecto en Android Studio la estructura del proyecto será diferente a la creada con eclipse. Esta nueva estructura es por el cambio de Ant a Gradle es más dinámica y flexible, a continuación, explicaremos la misma pero previamente cambiamos a modo proyecto:



Primero comenzamos explicando algunas carpetas del proyecto que a continuación observamos:



Expandimos la carpeta src (source o código fuente):



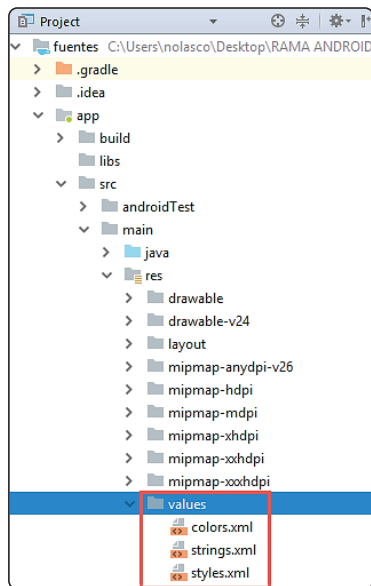
NOTA

Se recomienda la utilización de Android Asset Studio que te permite la generación de bitmaps para las interfaces: <http://romannurik.github.io/AndroidAssetStudio/>

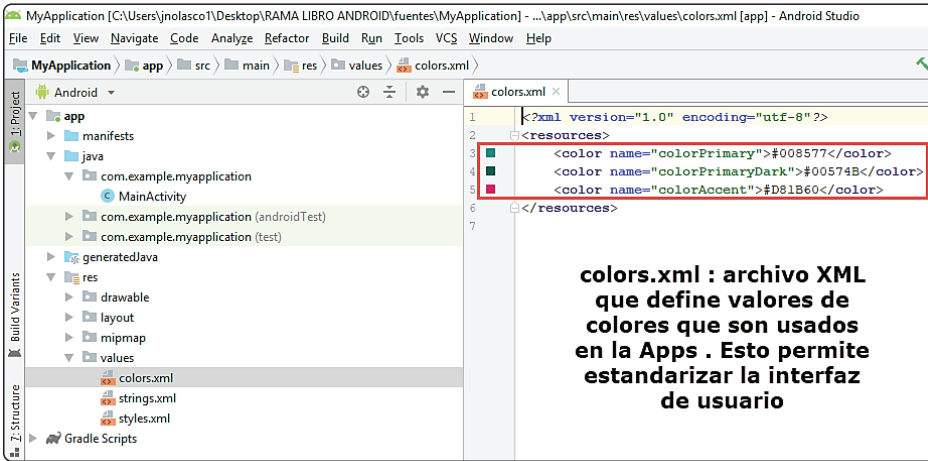
Ahora examinamos la carpeta res:



Ahora examinamos la carpeta res – value:



2.1.3 Archivo colors.xml

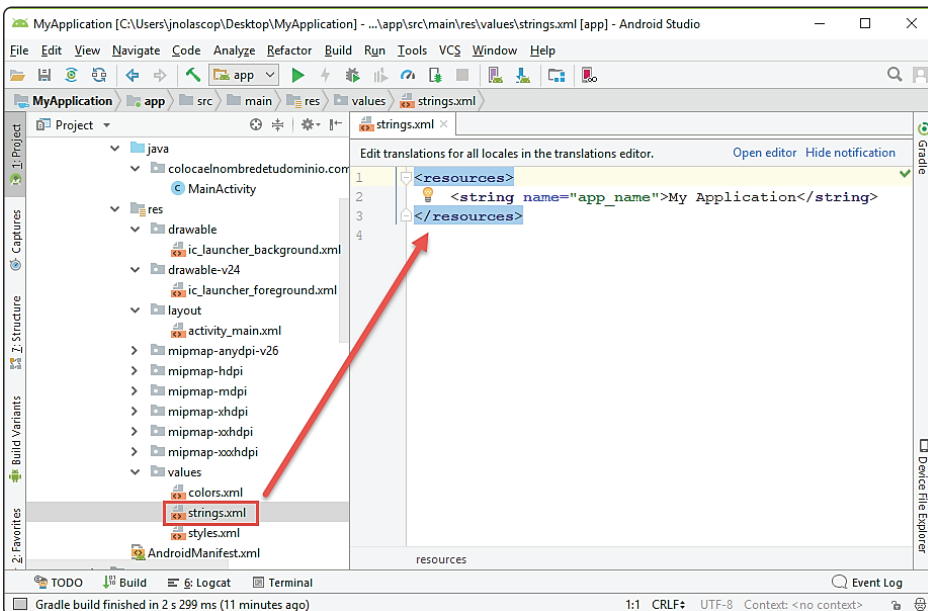


NOTA

En este recurso XML almacenaremos todos los colores en su valor hexadecimal que usaremos en nuestra aplicación. Deben ser almacenados en el fichero "res/values/colors.xml".

2.1.4 Archivo strings.xml

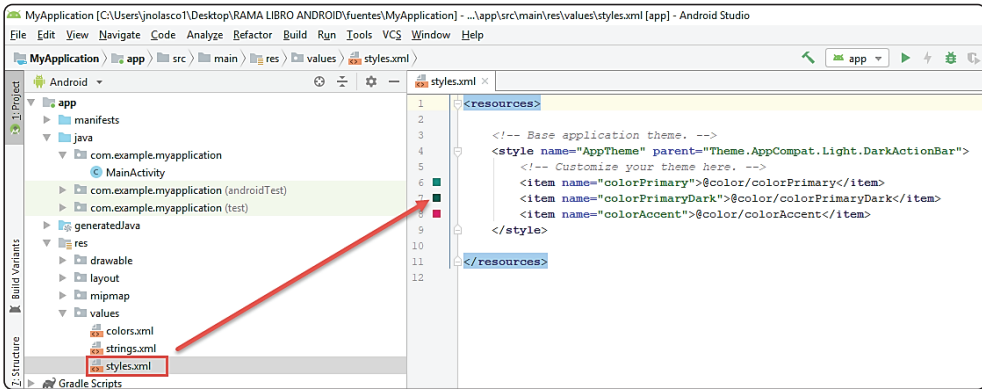
En este archivo se almacenará las cadenas constantes.



NOTA

Archivo XML que define cadenas de texto usadas en la aplicación. Por ejemplo, para colocar los títulos de las ventanas o el nombre de la aplicación y referenciarlas cuando lo necesites en el código.

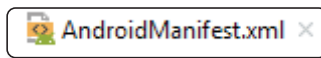
2.1.5 Archivo styles.xml



NOTA

Archivo XML que define los estilos usados en la Apps. Estos estilos pueden ser aplicados a los elementos de la interfaz de usuario, de modo que separamos la plantilla de las funcionalidades. Lo cual hace nuestra aplicación más fácil de mantener.

2.1.6 Archivo de Manifiesto o Configuración de Nuestra App



Todas las aplicaciones deben tener un archivo AndroidManifest.xml (con ese nombre exacto) en el directorio raíz. El archivo de manifiesto proporciona información esencial sobre tu aplicación al sistema Android, información que el sistema debe tener para poder ejecutar el código de la app.

Entre otras cosas, el archivo de manifiesto hace lo siguiente:

- ▀ Nombra el paquete de Java para la aplicación. El nombre del paquete sirve como un identificador único para la aplicación.

- Describe los componentes de la aplicación, como las actividades, los servicios, los receptores de mensajes y los proveedores de contenido que la integran. También nombra las clases que implementa cada uno de los componentes y publica sus capacidades, como los mensajes Intent con los que pueden funcionar. Estas declaraciones notifican al sistema Android los componentes y las condiciones para el lanzamiento.
- Determina los procesos que alojan a los componentes de la aplicación.
- Declara los permisos que debe tener la aplicación para acceder a las partes protegidas de una API e interactuar con otras aplicaciones. También declara los permisos que otros deben tener para interactuar con los componentes de la aplicación.
- Enumera las clases Instrumentación que proporcionan un perfil y otra información mientras la aplicación se ejecuta. Estas declaraciones están en el manifiesto solo mientras la aplicación se desarrolla y se quitan antes de la publicación de esta.
- Declara el nivel mínimo de Android API que requiere la aplicación.
- Enumera las bibliotecas con las que debe estar vinculada la aplicación.

Teoría Sobre el Manifiesto

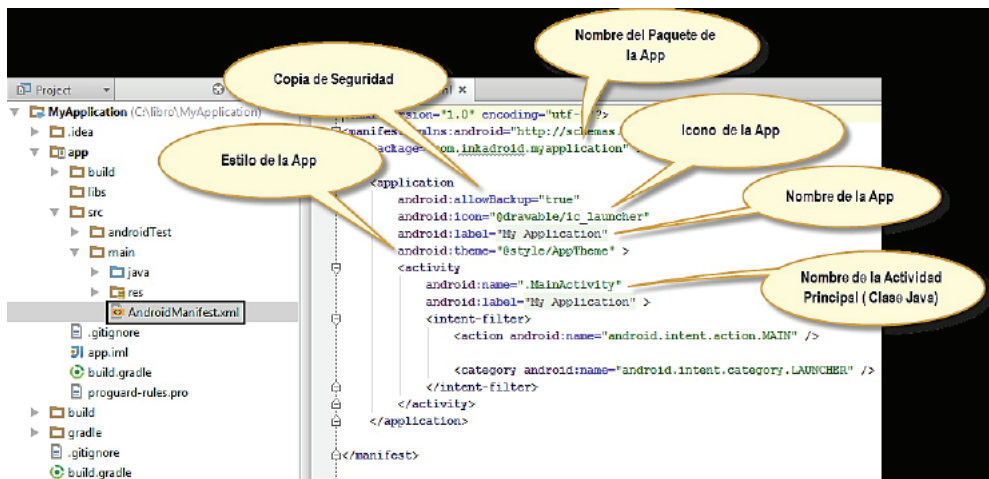
1. Nombra el paquete de Java para la aplicación. El nombre del paquete sirve como un identificador único para la aplicación.
2. Describe los componentes de la aplicación, como las actividades, los servicios, los receptores de mensajes y los proveedores de contenido que la integran. También nombra las clases que implementa cada uno de los componentes y publica sus capacidades, como los mensajes Intent con los que pueden funcionar. Estas declaraciones notifican al sistema Android los componentes y las condiciones para el lanzamiento.
3. Determina los procesos que alojan a los componentes de la aplicación.
4. Declara los permisos que debe tener la aplicación para acceder a las partes protegidas de una API e interactuar con otras aplicaciones. También declara los permisos que otros deben tener para interactuar con los componentes de la aplicación.
5. Enumera las clases Instrumentación que proporcionan un perfil y otra información mientras la aplicación se ejecuta. Estas declaraciones están en el manifiesto solo mientras la aplicación se desarrolla y se quitan antes de la publicación de esta.

6. Declara el nivel mínimo de Android API que requiere la aplicación.
7. Enumera las bibliotecas con las que debe estar vinculada la aplicación.

NOTA

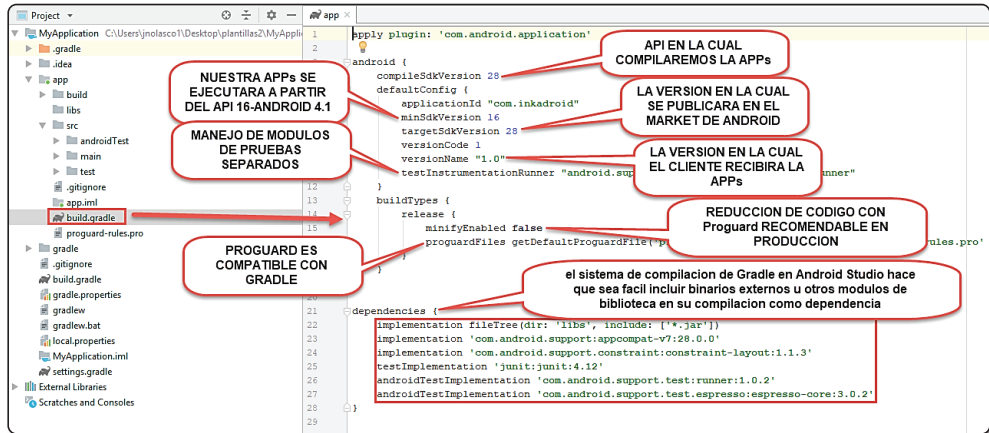
Este archivo se genera automáticamente y se declara información básica como: Nombre Paquete, Actividades, Estilos, iconos, permisos, etc.

AndroidManifest.xml



2.1.7 Archivo Gradle

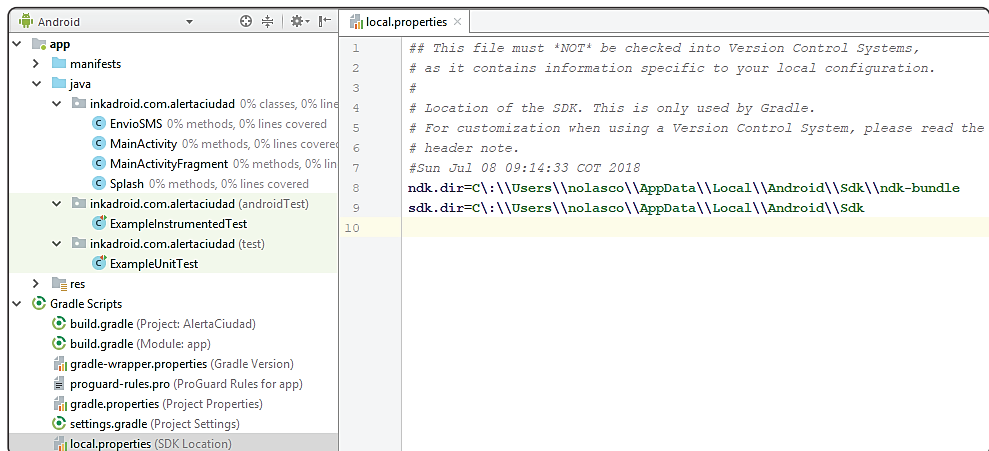
Gradle es un sistema de construcción de App a través de plugins, a continuación, mostramos su estructura:



2.1.8 En Otros Archivos Importantes

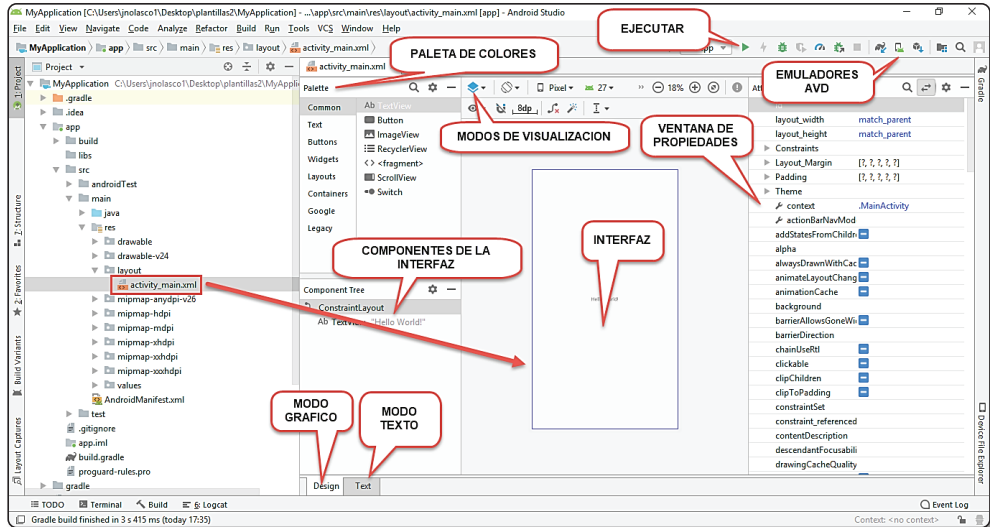
local.properties.

Archivo que establece la ubicación del SDK.



2.1.9 Conociendo la Interfaz

Ahora comenzaremos a conocer más sobre la Interfaz:



2.1.10 Ejecutando la Aplicación

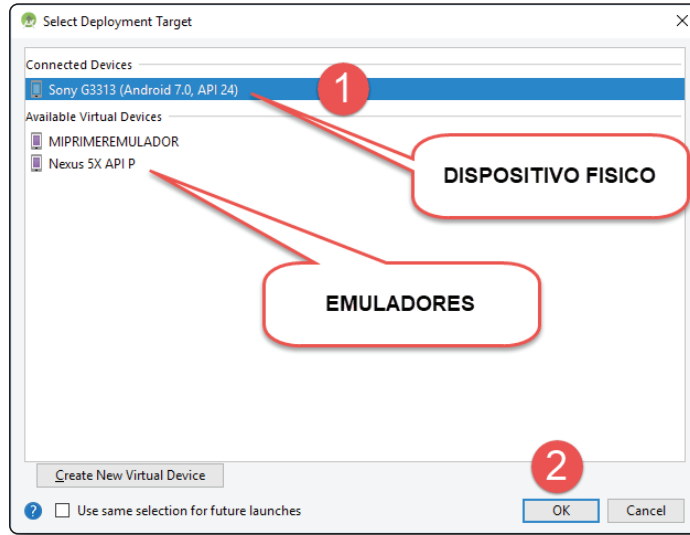
Ahora para ejecutar la App podemos hacerlo utilizando el emulador o un dispositivo físico recomendando usar un dispositivo físico para ello procedemos a conectar nuestro dispositivo o celular.

Hecho esto seguimos los siguientes pasos:

Presione el botón Run.

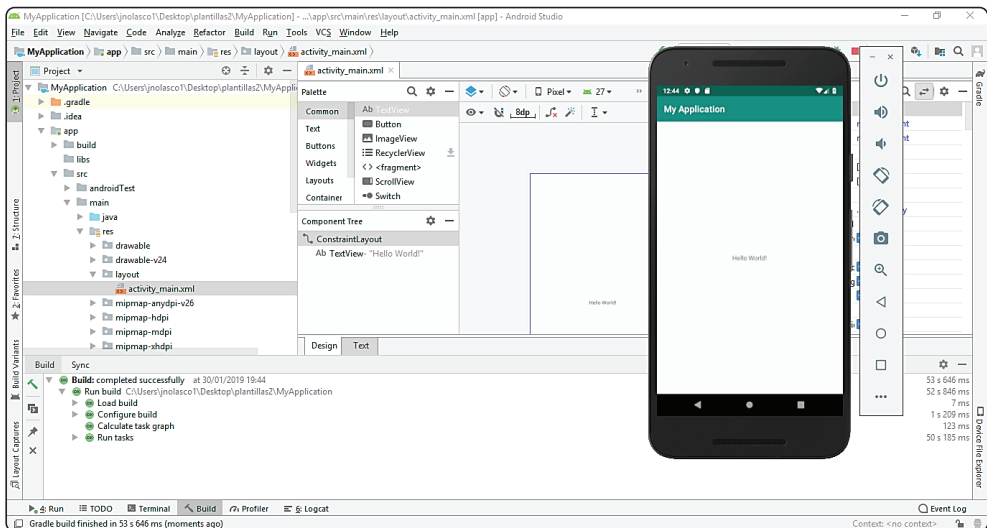


Ahora indicamos si utilizamos un dispositivo físico o un emulador:



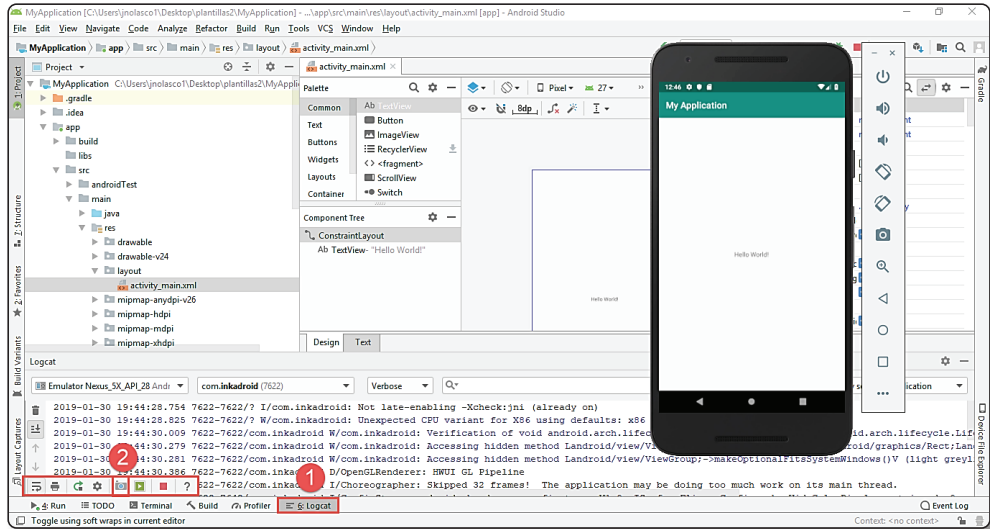
NOTA
Se recomienda utilizar un dispositivo físico.

Emulador en acción:



2.1.11 Captura de Pantalla

Usted puede observar la ejecución de su App presionando el siguiente botón (Captura de Pantalla):

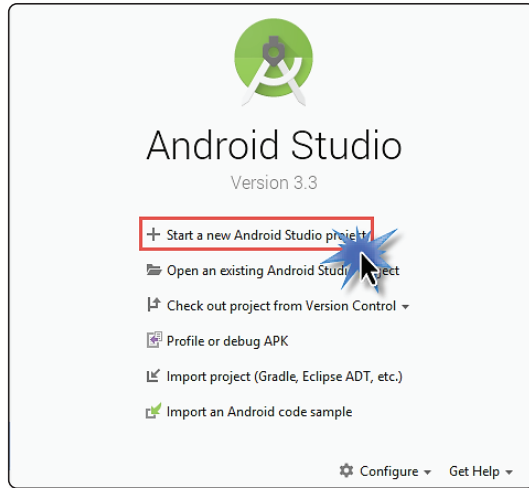


2.2 CREANDO APPS – LECTOR CÓDIGO BARRAS

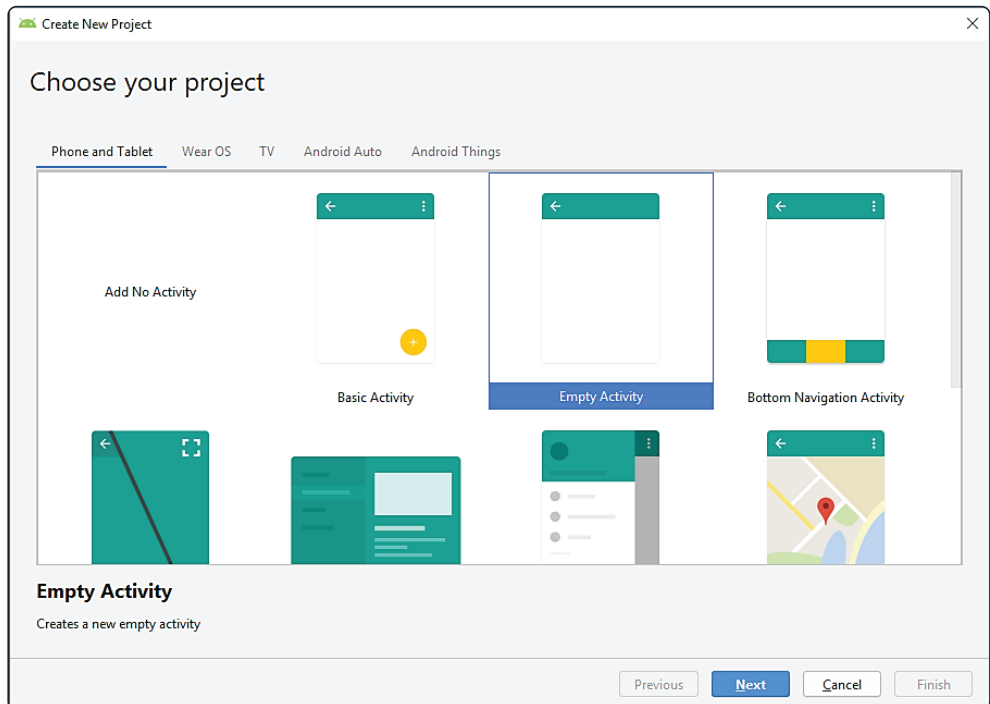
Ante de crear la primera App describiremos las distintas opciones:

- INICIO O CREACION DE UN NUEVO PROYECTO** (Start a new Android Studio project)
- ABRIR UN PROYECTO EXISTENTE** (Open an existing Android Studio project)
- CONTROL DE VERSIONES** (Check out project from Version Control)
- DEPURAR APK** (Profile or debug APK)
- IMPORTAR PROYECTO CREADO EN VERSIONES ANTIGUAS O EN ECLIPSE** (Import project (Gradle, Eclipse ADT, etc.))
- IMPORTAR PROYECTO EXISTENTE** (Import an Android code sample)

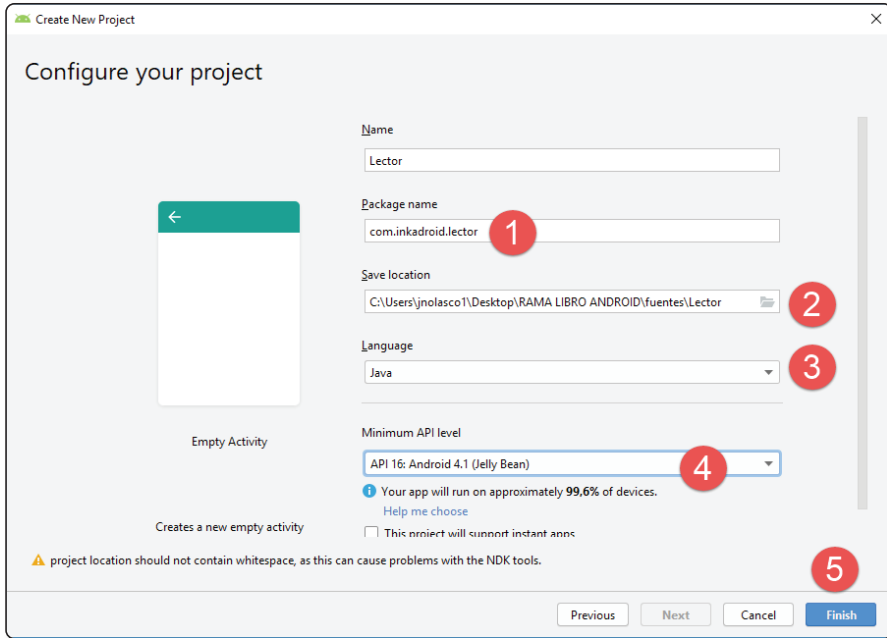
Para comenzar a crear un nuevo proyecto haga clic en Start a new Android Studio Project (Inicie un nuevo proyecto Android Studio):



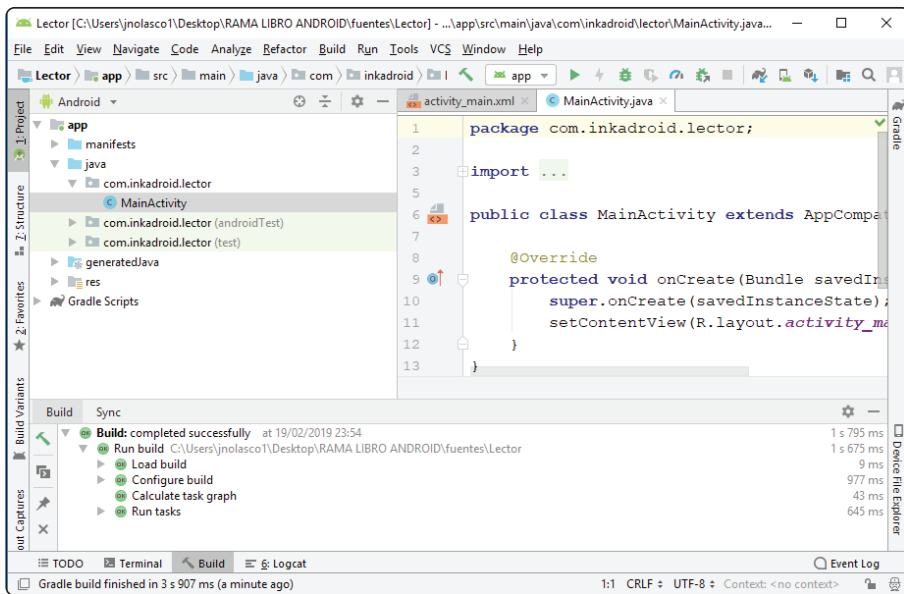
Ahora elegimos usar actividad en Blanco y luego presionamos el botón Next (Siguiete):



Ahora comenzamos a crear el proyecto indicando los siguientes datos y luego presionamos el botón Finish (Finalizar):

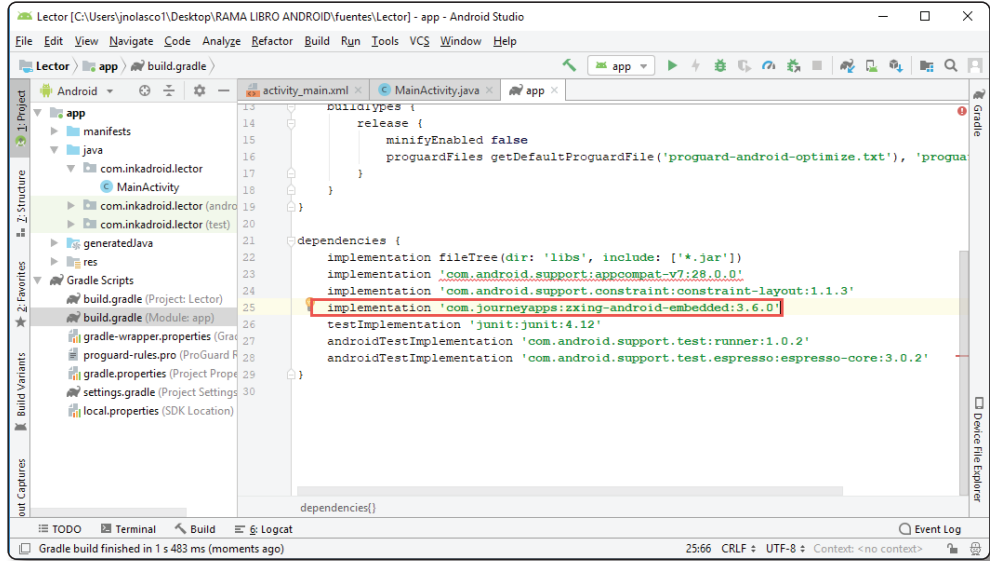


2.2.1 Obtenemos lo siguiente

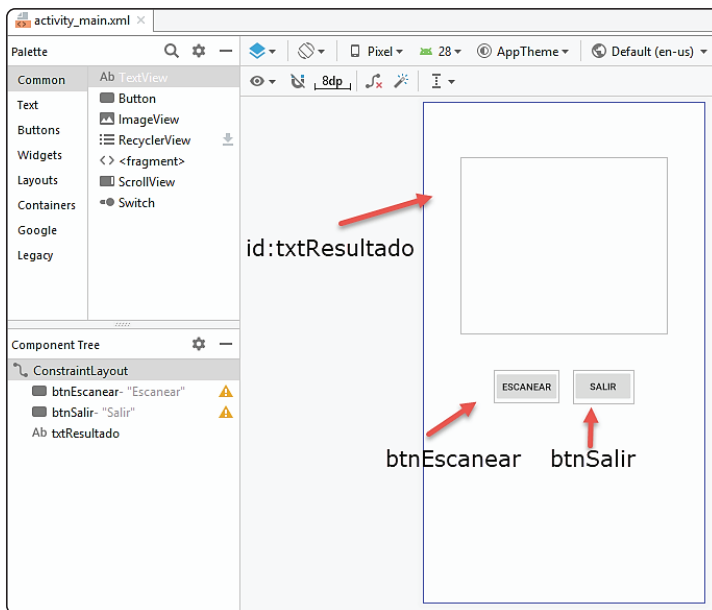


2.2.2 Biblioteca de escaneo

Hacemos referencia a la Biblioteca de escaneo de código de barras para Android, utilizando ZXing para decodificación:



2.2.3 Diseñamos la Interfaz



2.2.4 Codificación de la Actividad

```

1 package com.inkadroid.lector;
2 import ...
11 public class MainActivity extends AppCompatActivity implements View.OnClickListener {
12     private TextView txtResultado;
13     private Button btnEscanear, btnSalir;
14     @Override
15     protected void onCreate(Bundle savedInstanceState) {
16         super.onCreate(savedInstanceState);
17         setContentView(R.layout.activity_main);
18         //enlaces controles
19         txtResultado = (TextView) findViewById(R.id.txtResultado);
20         btnEscanear = (Button) findViewById(R.id.btnEscanear);
21         btnSalir = (Button) findViewById(R.id.btnSalir);
22         btnSalir.setOnClickListener(this);
23         btnEscanear.setOnClickListener(this);
24     }
25     @Override
26     public void onClick(View v) {
27
28         switch (v.getId())
29         {
30             case R.id.btnEscanear:
31                 new IntentIntegrator( this ).initiateScan(); // iniciar el escaneo
32                 break;
33             case R.id.btnSalir:
34                 finish();
35                 break;
36         }
37     }
38     @Override
39     protected final void onActivityResult(final int requestCode, final int resultCode, final Intent data) {
40         Intent result = IntentIntegrator.parseActivityResult(requestCode, resultCode, data);
41         if(result != null) {
42             if(result.getContents() == null) {
43                 Toast.makeText( context: this, text: "Cancelled", Toast.LENGTH_LONG).show();
44             } else {
45                 txtResultado.setText(result.getContents());
46             }
47         } else {
48             super.onActivityResult(requestCode, resultCode, data);
49         }
50     }
51 }

```

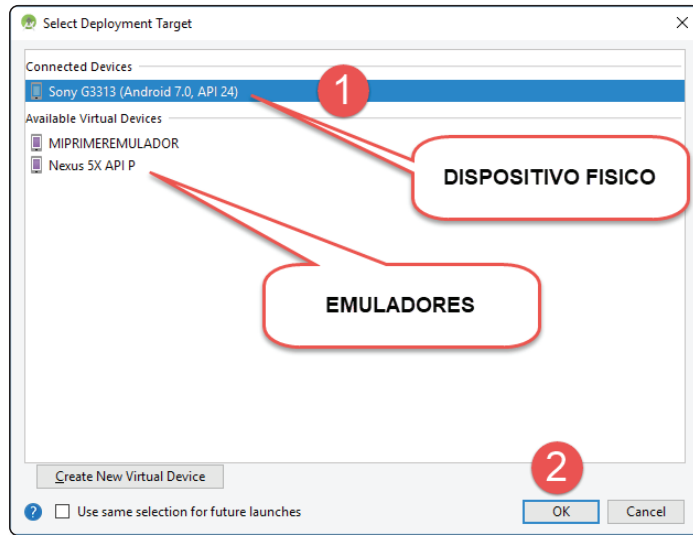
2.2.5 Ejecutando la Aplicación

Ahora para ejecutar la App podemos hacerlo utilizando el emulador o un dispositivo físico recomendando usar un dispositivo físico para ello procedemos a conectar nuestro dispositivo o celular. Hecho esto seguimos los siguientes pasos:

Presione el botón Run



Ahora indicamos si utilizamos un dispositivo físico o un emulador:



i **NOTA**

Se recomienda utilizar un dispositivo físico.



