

# Introducción a la programación Móvil

Capítulo I, primera parte.

Prof. Oriel A. Cedeño

# Agenda

- ▶ Introducción a la Programación Móvil.
- ▶ Los Sistemas Operativos de los dispositivos Móviles.
- ▶ Instalación del Android Studio.
- ▶ Configuración de un dispositivo móvil virtual.
- ▶ Estructura de un Proyecto Android.
- ▶ Componentes de una aplicación Android

# Dispositivos móviles

- ▶ "Dispositivo móvil" es un término general para cualquier computadora de mano o teléfono inteligente. Las tabletas, los lectores electrónicos, los teléfonos inteligentes, las PDA y los reproductores portátiles de música con capacidades inteligentes son todos dispositivos móviles.
- ▶ Algunos ejemplos
  - ▶ Smartphones,
  - ▶ Tablets,
  - ▶ E-Readers
  - ▶ Ipods



# Que es el Desarrollo de aplicaciones móviles?

- Es el conjunto de procesos y procedimientos involucrados en la escritura de software para pequeños dispositivos inalámbricos de cómputo, como teléfonos inteligentes o tabletas.



# Sistemas operativos de DM

- ▶ Un **sistema operativo móvil** o **SO móvil** es un conjunto de programas de bajo nivel que permite la abstracción de las peculiaridades del hardware específico del teléfono móvil y provee servicios a las aplicaciones móviles, que se ejecutan sobre él.
- ▶ Algunos Ejemplos
  - ▶ Android
  - ▶ iOS
  - ▶ Windows Phone
  - ▶ BlackBerry OS
  - ▶ Symbian
  - ▶ Firefox O.S
  - ▶ Ubuntu Touch
  - ▶ Amazon Fire OS



# Algunas características que se deben cumplir

- ▶ Los dispositivos móviles tienen características similares. Entre ellos están:
- ▶ Wi-Fi o acceso celular a internet
- ▶ Una batería que alimenta el dispositivo durante varias horas
- ▶ Un teclado físico o en pantalla para ingresar información
- ▶ Tamaño y peso que permite llevarlo en una mano y manipularlo con la otra mano
- ▶ Interfaz de pantalla táctil en casi todos los casos
- ▶ Un asistente virtual, como Siri, Cortana o Google Assistant
- ▶ La capacidad de descargar datos de Internet, incluidas aplicaciones y libros
- ▶ Funcionamiento inalámbrico

# Breve historia

- ▶ Allá por octubre del año 2003, Andy Rubin, Rich Miner, Nick Sears y Chris White daban forma a Android Inc.
- ▶ Andy es muñeco verde, logotipo del sistema operativo Android. Salió por primera vez en el año 2005, cuando Google compró la empresa Android Inc.
- ▶ Desde la versión 1.6, comenzaron a nombrar sus versiones con nombre de postres famosos según el orden alfabético de la versión.
- ▶ <https://histinf.blogs.upv.es/2012/12/14/android/> < Lean este blog para mas curiosidades he historia de Android



Andy Rubin





## Android 8.0

### Oreo

API Level 26  
SDK for Android 8.0  
Security Patches - Supported  
Release Date - Aug 21, 2017



## Android 7.0

### Nougat

API Level 24-25  
SDK for Android 7.0  
Security Patches - Supported  
Release Date - Aug 22, 2016



## Android 6.0

### Marshmallow

API Level 23  
SDK for Android 6.0  
Security Patches - Supported  
Release Date - Oct 5, 2015



## Android 5.0

### Lollipop

API Level 21-22  
SDK for Android 5.0  
Security Patches - Supported  
Release Date - Nov 12, 2014



## Android 4.4

### KitKat

API Level 19-20  
SDK for Android 4.4  
Security Patches - Supported  
Release Date - Oct 31, 2013



## Android 4.1

### Jelly Bean

API Level 16-18  
SDK for Android 4.1  
Based on Linux kernel 3.0.31  
Security Patches - Unsupported  
Release Date - Jul 9, 2012



## Android 4.0

### Ice Cream Sandwich

API Level 14-15  
SDK for Android 4.0.1  
Based on Linux kernel 3.0.1  
Security Patches - Unsupported  
Release Date - Oct 18, 2011



## Android 3.0

### Honeycomb

API Level 11-13  
SDK for Android 3.0  
Based on Linux kernel 2.6.30  
Security Patches - Unsupported  
Release Date - Feb 12, 2011



## Android 2.3

### Gingerbread

API Level 9-10  
SDK for Android 2.3  
Based on Linux kernel 2.6.35  
Security Patches - Unsupported  
Release Date - Dec 6, 2010



## Android 2.2

### Froyo

API Level 8  
SDK for Android 2.2  
Based on Linux kernel 2.6.32  
Security Patches - Unsupported  
Release Date - May 20, 2010



## Android 2.0

### Eclair

API Level 5-7  
SDK for Android 2.0  
Based on Linux kernel 2.6.29  
Security Patches - Unsupported  
Release Date - Oct 6, 2009



## Android 1.6

### Donut

API Level 4  
SDK for Android 1.6  
Based on Linux kernel 2.6.29  
Security Patches - Unsupported  
Release Date - Sep 15, 2009



## Android 1.5

### Cupcake

API Level 3  
SDK for Android 1.5  
Based on Linux kernel 2.6.27  
Security Patches - Unsupported  
Release Date - Apr 27, 2009



## Android 1.1

### Beta

API Level 2  
SDK for Android 1.1  
Security Patches - Unsupported  
Release Date - Feb 9, 2009



## Android 1.0

### Alpha

API Level 1  
SDK for Android 1.0  
Security Patches - Unsupported  
Release Date - Sep 23, 2008



# Tecnologías y herramientas para el desarrollo de aplicaciones Android



# Herramientas

- ▶ Para desarrollar aplicaciones móviles, vamos a tomar en consideraciones las siguientes herramientas:
- ▶ Android Studio > IDE de desarrollo
- ▶ JDK > Java Developer Kit
- ▶ Android SDK > Administrador de las versiones (API) de Android
- ▶ Genymotion > Emulador de teléfonos celular (opcional)



Android  
Studio

# Instalación del Android Studio.

Vamos al sitio: <https://developer.android.com/studio/> y descarguemos la opción Por defecto que nos aparece

## Windows

- Microsoft® Windows® 7/8/10 (64-bit)
- 4 GB RAM minimum, 8 GB RAM recommended
- 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

For use with the Emulator, see the [Emulator system requirements](#).



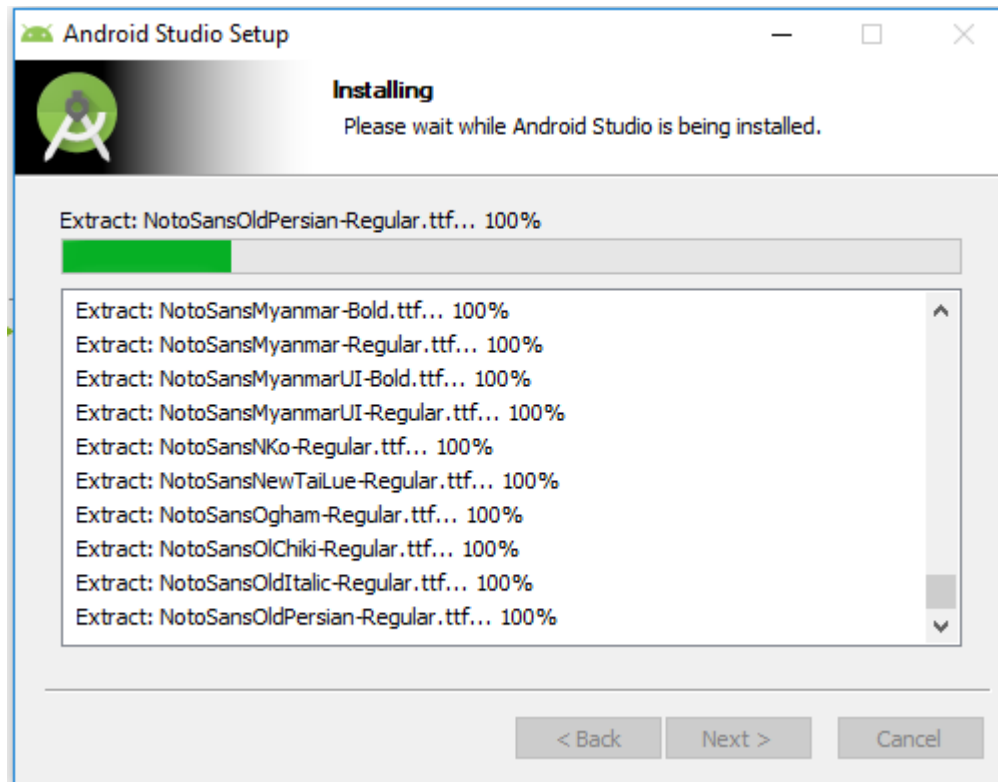
Android Studio provides the fastest tools for building apps on every type of Android device.

**DOWNLOAD ANDROID STUDIO**

4.1.3 for Windows 64-bit (896 MiB)

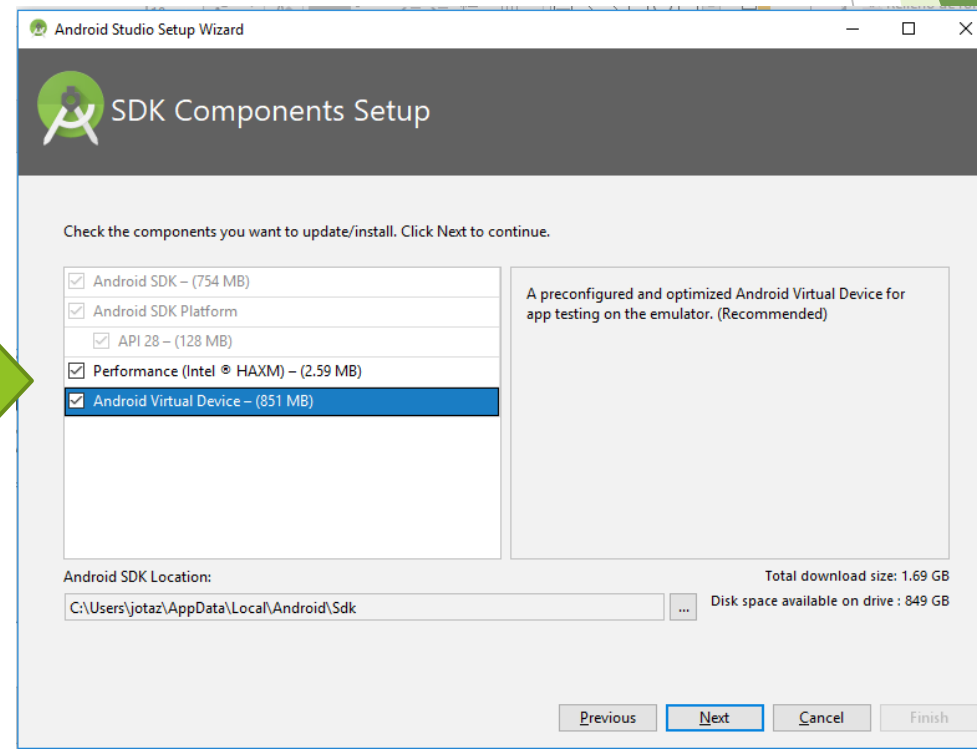
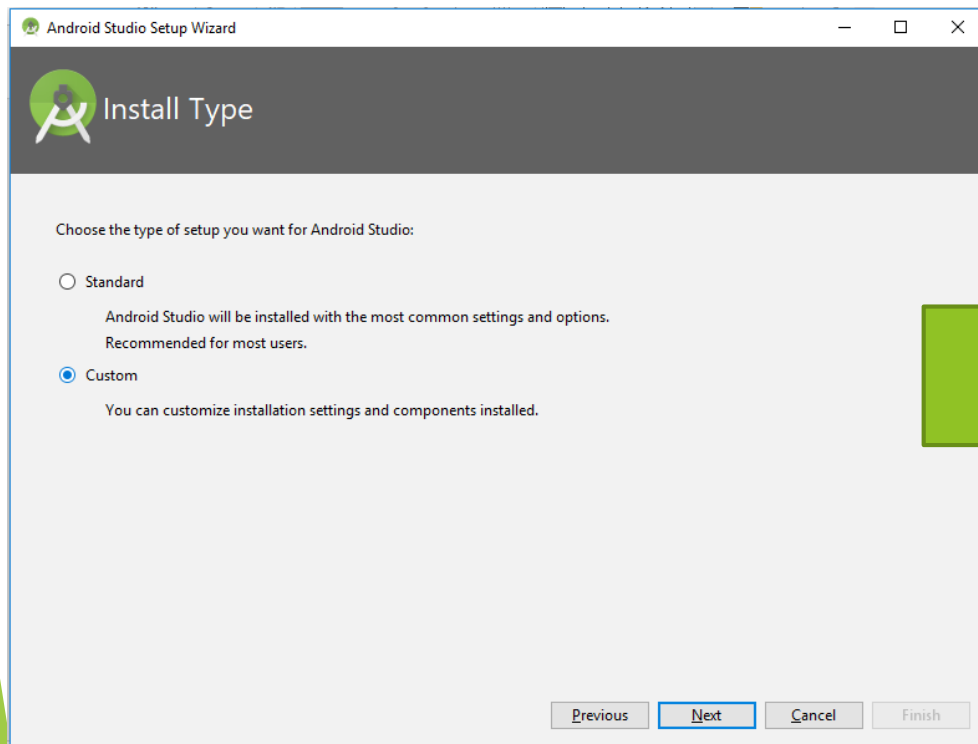
# Instalación del Android Studio.

- Básicamente, la instalación es un clásico Next...Next...Next Install



# Al iniciar Android Studio

- Se nos presenta un asistente que debemos seguir.



# Al iniciar Android Studio

- ▶ Verificamos los componentes a instalar y le damos finish... y esperamos que se instale
- ▶ Luego de instalado, preferiblemente instalamos las API de Android 9, 8, 7 y 6 en el SDK manager

The image shows the initial setup of Android Studio. On the left, the 'Android Studio Setup Wizard' is open to the 'Verify Settings' screen. It lists the components to be downloaded, including the Android Emulator, SDK Build-Tools, Platform 28, Platform-Tools, Tools, Support Repository, Google APIs, Google Repository, Intel x86 Emulator Accelerator, and SDK Patch Applier. A large green arrow points from this window to the 'Android SDK Manager' window on the right. The SDK Manager shows a list of Android SDK platforms with checkboxes for installation. The 'Android 6.0 (Marshmallow)' is highlighted. The 'Appearance & Behavior' sidebar on the left of the SDK Manager is also visible.

**Android Studio Setup Wizard - Verify Settings**

If you want to review or change any of your installation settings, click Previous.

Current Settings:

SDK Components to Download:	
Android Emulator	288 MB
Android SDK Build-Tools 28.0.2	55.6 MB
Android SDK Platform 28	72.1 MB
Android SDK Platform-Tools	4.55 MB
Android SDK Tools	149 MB
Android Support Repository	339 MB
Google APIs Intel x86 Atom System Image	851 MB
Google Repository	205 MB
Intel x86 Emulator Accelerator (HAXM installer)	2.59 MB
SDK Patch Applier v4	1.74 MB

**Android SDK Manager**

Appearance & Behavior > System Settings > Android SDK

Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: C:\Users\jotaz\AppData\Local\Android\Sdk

SDK Platforms | SDK Tools | SDK Update Sites

Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.

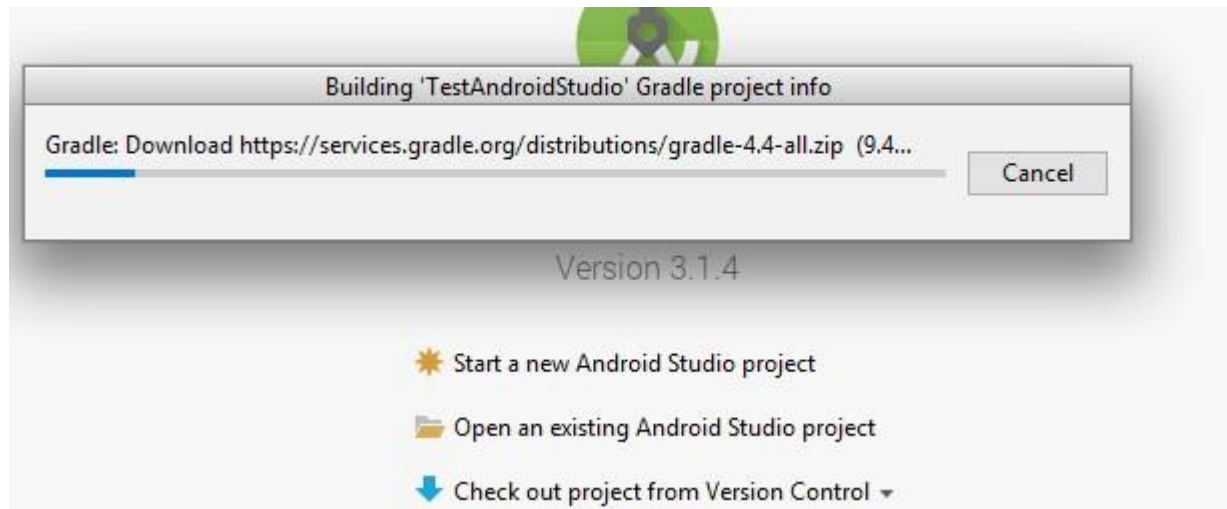
	Name	API Level	Revision	Status
<input checked="" type="checkbox"/>	Android API 28	28	6	Partially installed
<input checked="" type="checkbox"/>	Android 8.1 (Oreo)	27	3	Not installed
<input checked="" type="checkbox"/>	Android 8.0 (Oreo)	26	2	Not installed
<input checked="" type="checkbox"/>	Android 7.1.1 (Nougat)	25	3	Not installed
<input checked="" type="checkbox"/>	Android 7.0 (Nougat)	24	2	Not installed
<input checked="" type="checkbox"/>	Android 6.0 (Marshmallow)	23	3	Not installed
<input type="checkbox"/>	Android 5.1 (Lollipop)	22	2	Not installed
<input type="checkbox"/>	Android 5.0 (Lollipop)	21	2	Not installed
<input type="checkbox"/>	Android 4.4W (KitKat Wear)	20	2	Not installed
<input type="checkbox"/>	Android 4.4 (KitKat)	19	4	Not installed
<input type="checkbox"/>	Android 4.3 (Jelly Bean)	18	3	Not installed
<input type="checkbox"/>	Android 4.2 (Jelly Bean)	17	3	Not installed
<input type="checkbox"/>	Android 4.1 (Jelly Bean)	16	5	Not installed
<input type="checkbox"/>	Android 4.0.3 (IceCreamSandwich)	15	5	Not installed
<input type="checkbox"/>	Android 4.0 (IceCreamSandwich)	14	4	Not installed
<input type="checkbox"/>	Android 3.2 (Honeycomb)	13	1	Not installed
<input type="checkbox"/>	Android 3.1 (Honeycomb)	12	3	Not installed
<input type="checkbox"/>	Android 3.0 (Honeycomb)	11	2	Not installed
<input type="checkbox"/>	Android 2.3.3 (Gingerbread)	10	2	Not installed
<input type="checkbox"/>	Android 2.3 (Gingerbread)	9	2	Not installed
<input type="checkbox"/>	Android 2.2 (Froyo)	8	3	Not installed
<input type="checkbox"/>	Android 2.1 (Eclair)	7	3	Not installed

☐ Show Package Details

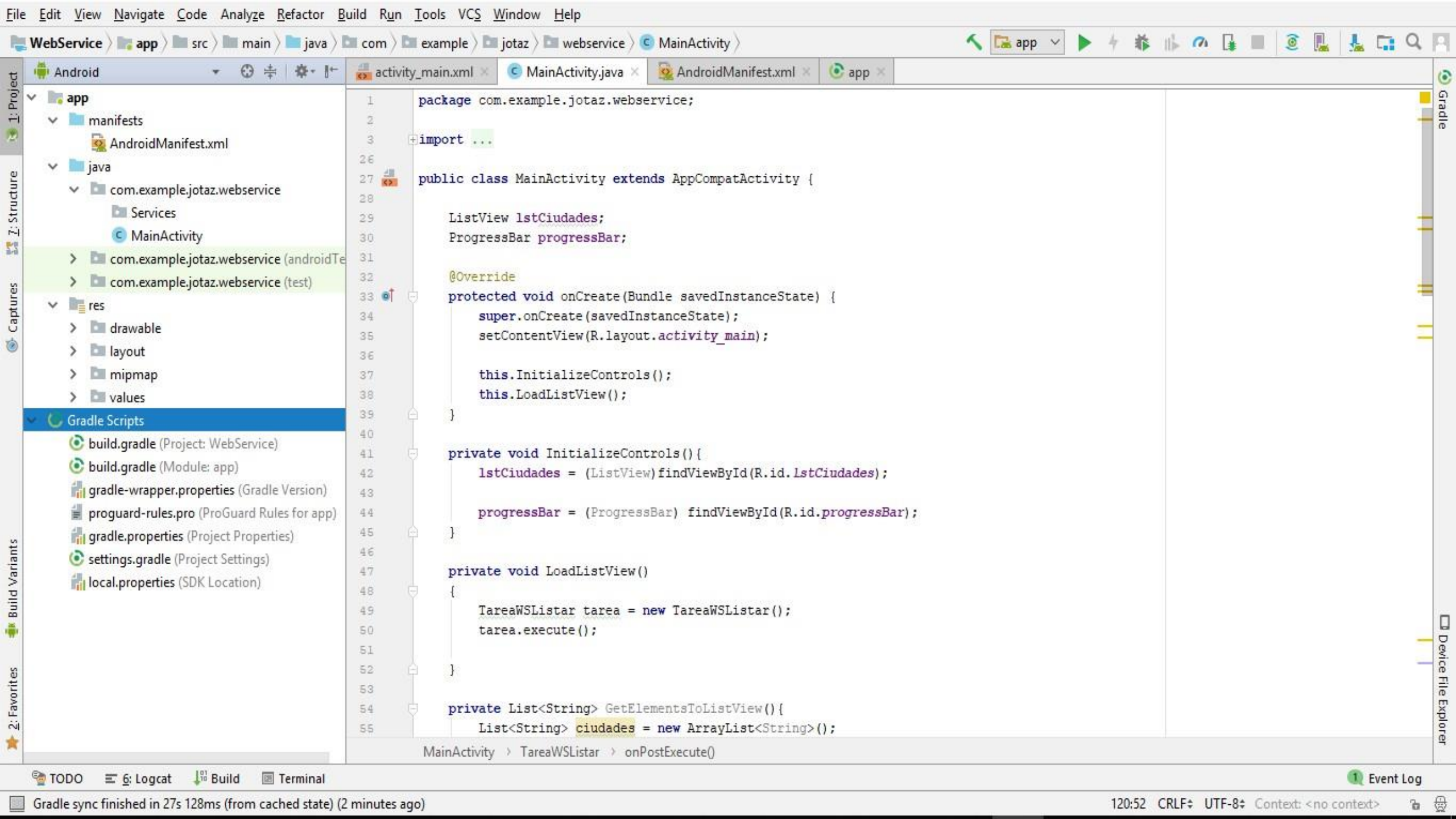
OK Cancel Apply

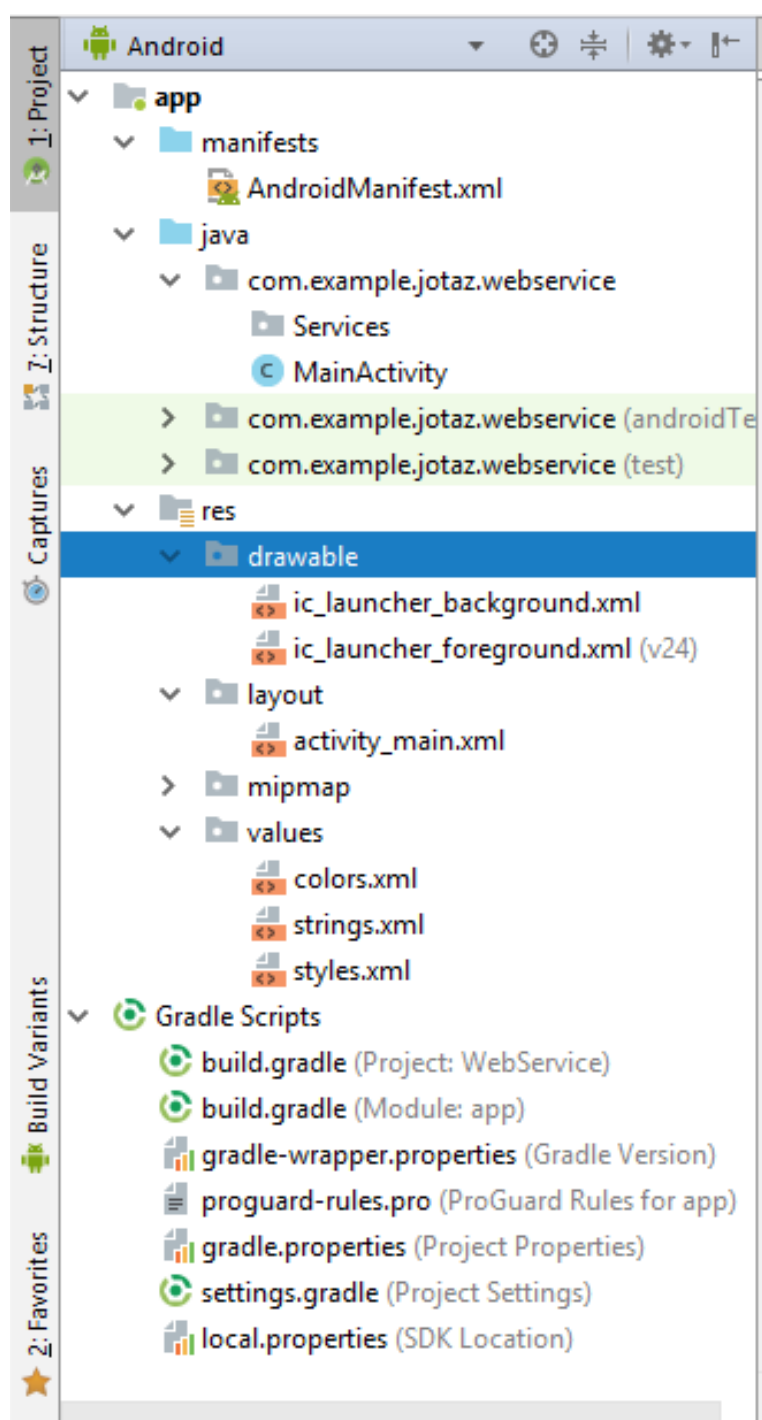
# Al iniciar Android Studio

- Cuando todo esto haya acabado, debemos asegurarnos de estar conectados a internet por lo menos la primera vez que se abre el proyecto, ya que se debe descargar y compilar el gradle de Android Studio









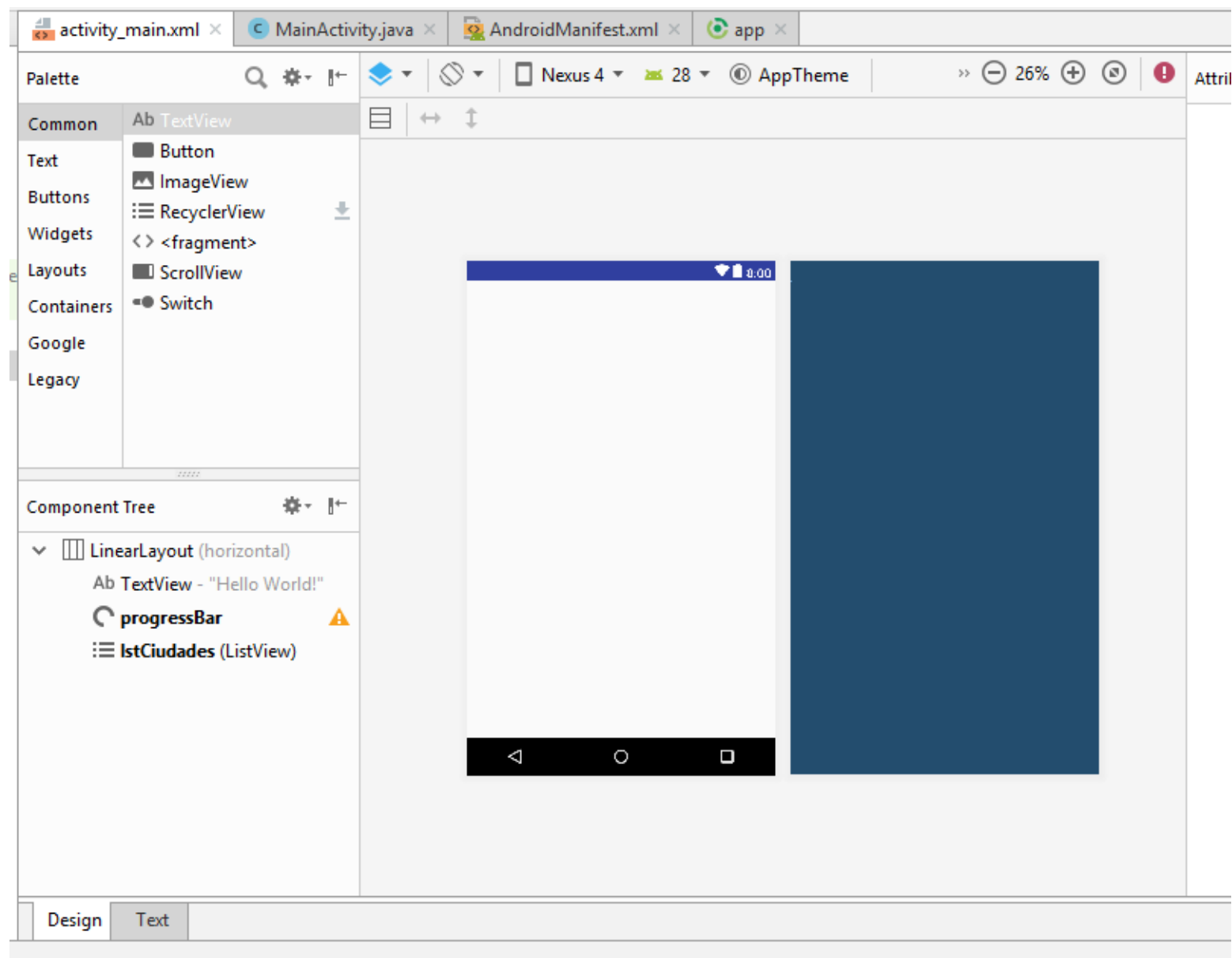
# Área del Proyecto

- ▶ Podemos encontrar las carpetas:
  - ▶ Manifest: contiene el archivo manifiesto.
  - ▶ Java: contiene el código Fuente de nuestras actividades
  - ▶ Res/drawable: almacena las imágenes estáticas
  - ▶ Res/layout: contiene el código de diseño de nuestras actividades
  - ▶ Res/values: contiene archivos xml de configuración o uso en la aplicación

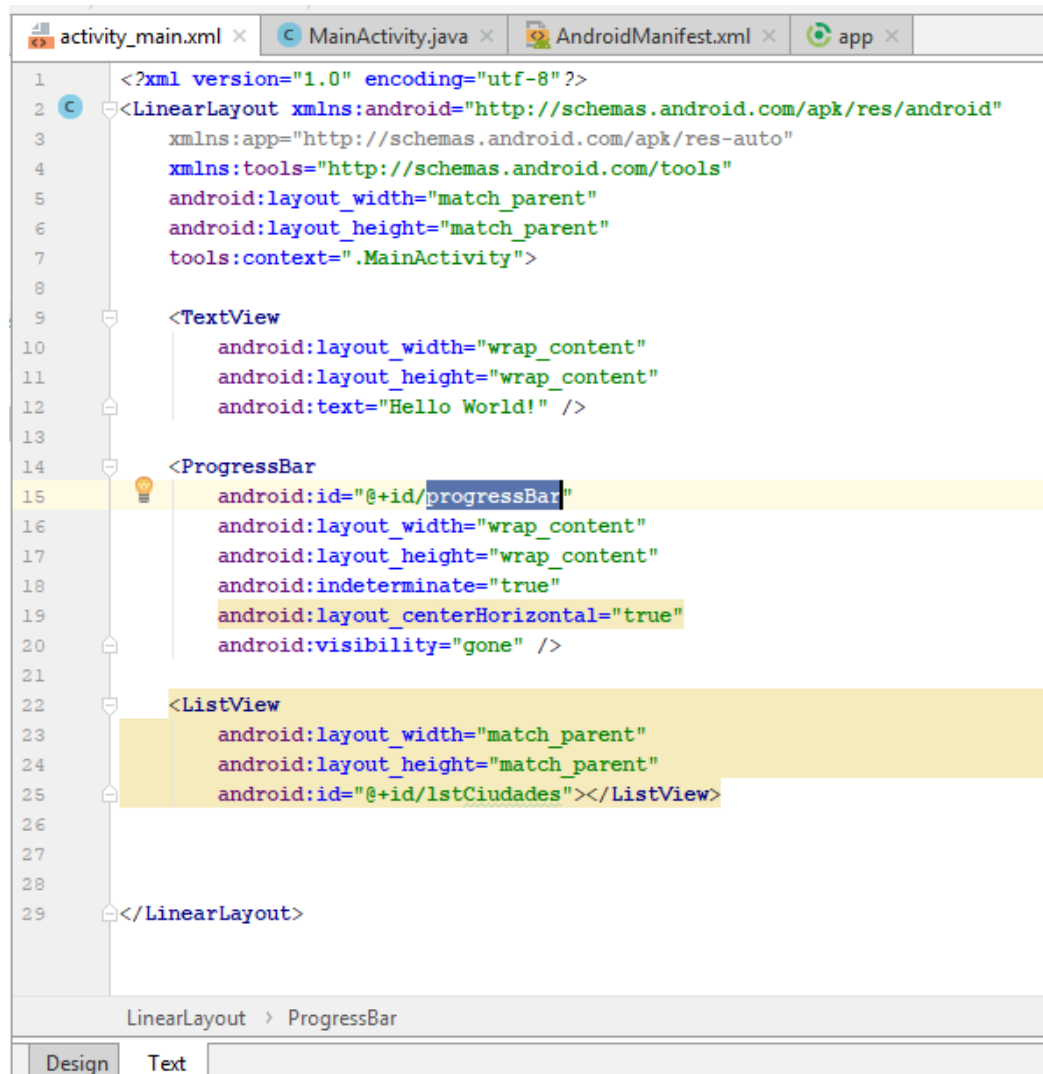
```
1 package com.example.jotaz.webservice;
2
3 import ...
4
26
27 public class MainActivity extends AppCompatActivity {
28
29     ListView lstCiudades;
30     ProgressBar progressBar;
31
32     @Override
33     protected void onCreate(Bundle savedInstanceState) {
34         super.onCreate(savedInstanceState);
35         setContentView(R.layout.activity_main);
36
37         this.InitializeControls();
38         this.LoadListView();
39     }
40
41     private void InitializeControls(){
42         lstCiudades = (ListView) findViewById(R.id.lstCiudades);
43
44         progressBar = (ProgressBar) findViewById(R.id.progressBar);
45     }
46
47     private void LoadListView()
48     {
49         TareaWSListar tarea = new TareaWSListar();
50         tarea.execute();
51     }
52
53
54     private List<String> GetElementsToListView(){
55         List<String> ciudades = new ArrayList<String>();
```

MainActivity > TareaWSListar > onPostExecute()

# Área de código fuente



Área del código de interfaz 1



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".MainActivity">
8
9     <TextView
10         android:layout_width="wrap_content"
11         android:layout_height="wrap_content"
12         android:text="Hello World!" />
13
14     <ProgressBar
15         android:id="@+id/progressBar"
16         android:layout_width="wrap_content"
17         android:layout_height="wrap_content"
18         android:indeterminate="true"
19         android:layout_centerHorizontal="true"
20         android:visibility="gone" />
21
22     <ListView
23         android:layout_width="match_parent"
24         android:layout_height="match_parent"
25         android:id="@+id/lstCiudades"></ListView>
26
27
28
29 </LinearLayout>
```

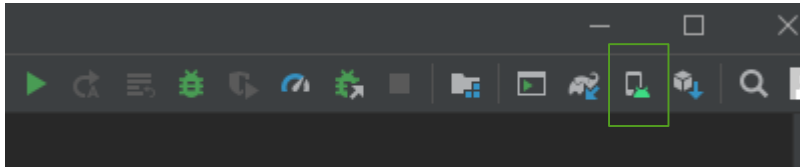
LinearLayout > ProgressBar

Design Text

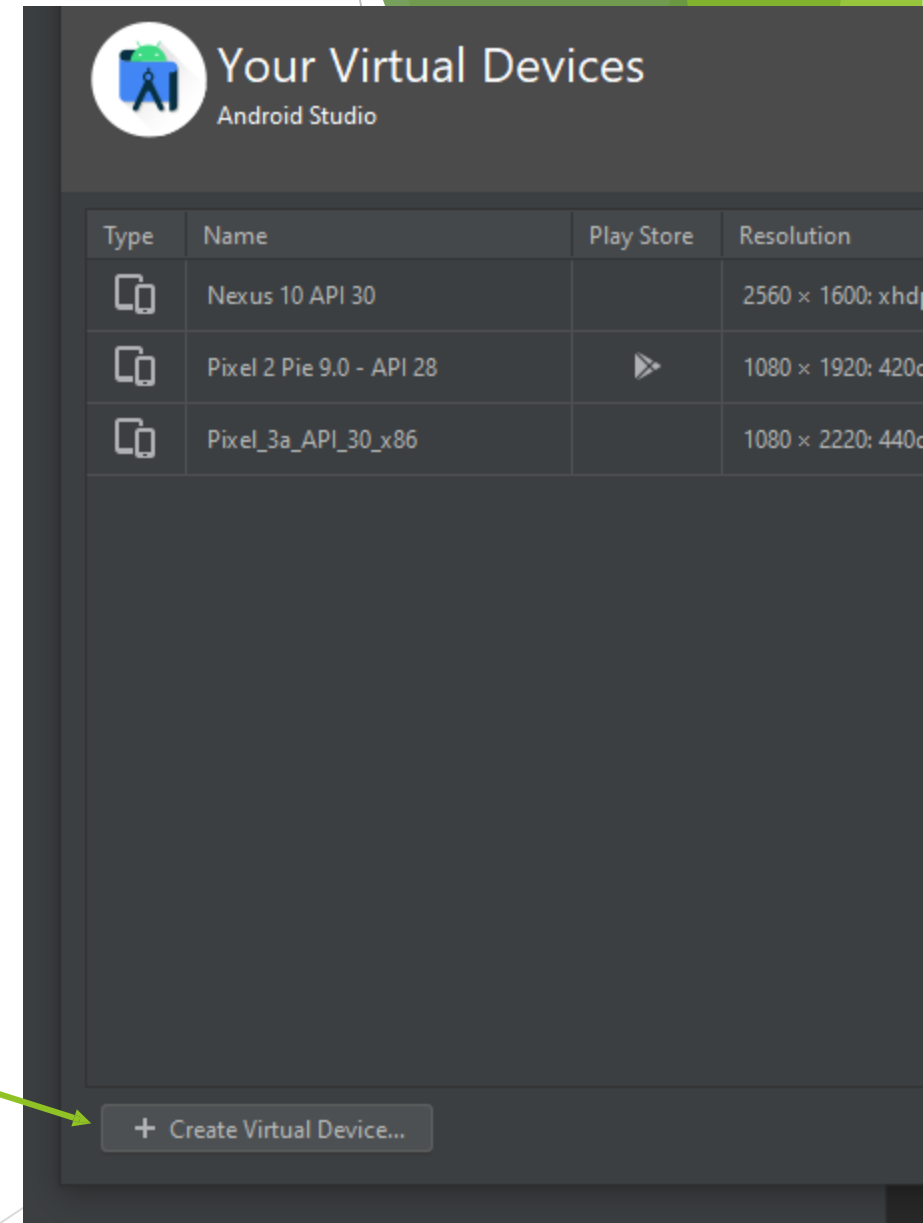
# Área del código de interfaz 2

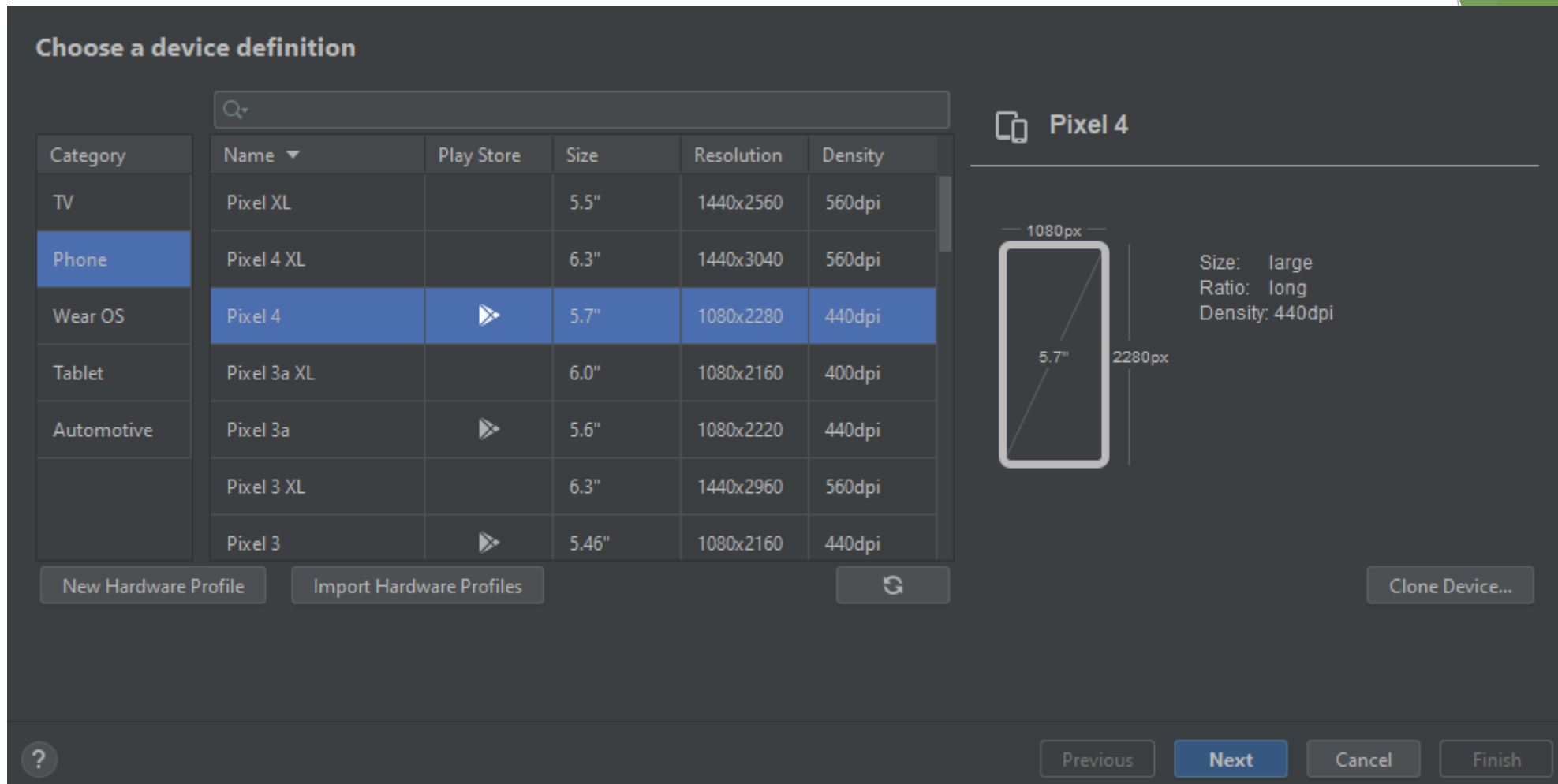
# Configurando Dispositivos Virtuales

- ▶ Para configurar dispositivos virtuales, debemos ir a la barra de tarea y ubicar el icono del AVD Manager



- ▶ Luego, en la ventana emergente, le damos a la opción de Crear nuevo dispositivo (Create new virtual Device) en la parte inferior izquierda





- ▶ En la siguiente pantalla, procedemos a establecer las características que tendrá nuestro dispositivo virtual.
- ▶ Para este ejemplo, seleccionamos un teléfono predefinido en el listado => Pixel 4




- Procedemos a seleccionar una versión de sistema operativo que deseamos tener en nuestro dispositivo virtual

### Select a system image

Recommendedx86 ImagesOther Images

Release Name	API Level ▼	ABI	Target
<a href="#">R Download</a>	30	x86	Android 11.0 (Google Play)
<a href="#">Q Download</a>	29	x86	Android 10.0 (Google Play)
<b>Pie</b>	<b>28</b>	<b>x86</b>	<b>Android 9.0 (Google Play)</b>
<a href="#">Oreo Download</a>	27	x86	Android 8.1 (Google Play)
<a href="#">Oreo Download</a>	26	x86	Android 8.0 (Google Play)
<a href="#">Nougat Download</a>	25	x86	Android 7.1.1 (Google Play)
<a href="#">Nougat Download</a>	24	x86	Android 7.0 (Google Play)

### Pie



API Level

**28**

Android

**9.0**

Google Inc.

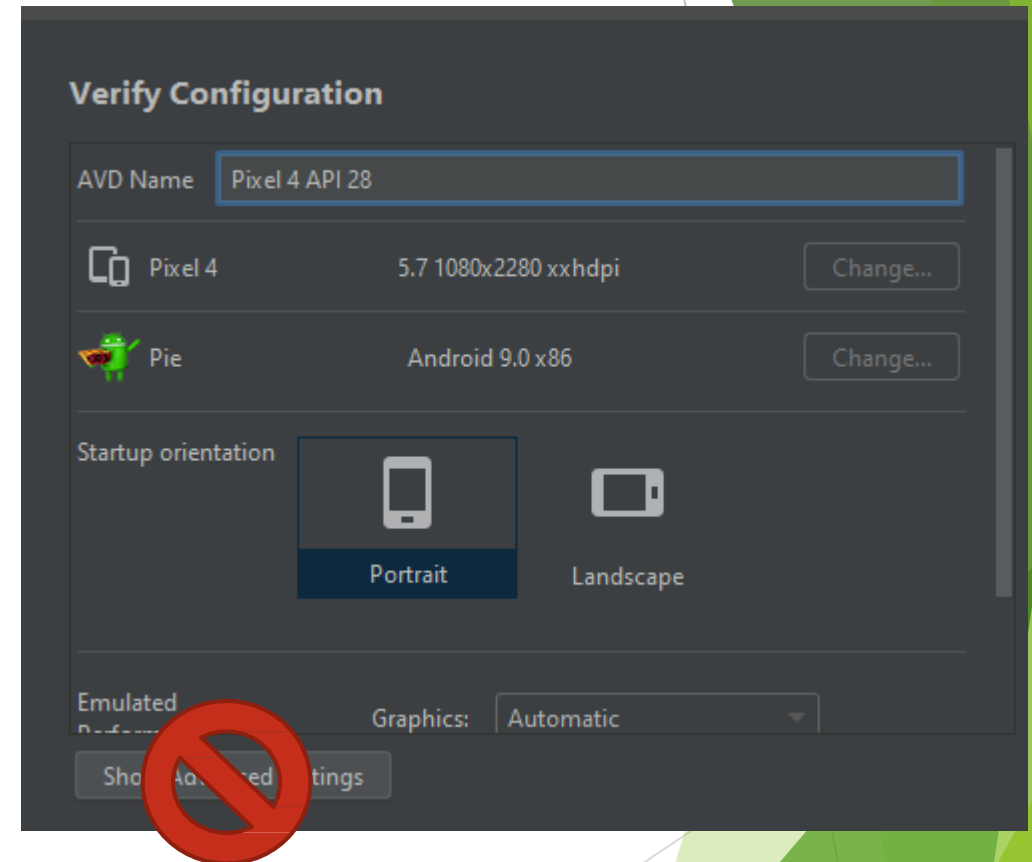
System Image

**x86**

We recommend these Google Play images because this device is compatible with Google Play.

[Questions on API level?](#)

- Establecemos las ultimas características como el nombre y orientación del dispositivo por defecto.
- Nota: se sugiere no tocar ninguna característica avanzada para que su máquina pueda trabajar de manera mas optima.
- Luego, damos click en el boton de Finalizar.



# Dispositivo listo

- Visualizaremos nuestro dispositivo en la pantalla inicial en el listado y le damos al botón de play.

