



# HLC

2° DAW

# FAMILIARIZARNOS CON EL IDE **ANDROID STUDIO**

Debemos tener conocimientos previos de JAVA

Trabajaremos indistintamente con Windows o Mac, pero necesitamos como mínimo de 8 Gb de RAM para poder emular sin problemas las Apps y arquitectura de 64 bits





# ¿Qué es Android?

Debemos tener claro que **Android NO ES un lenguaje de programación**, sino un Sistema Operativo

Por comodidad, requerimiento empresarial y recomendación de Google, usaremos el IDE oficial: **Android Studio**

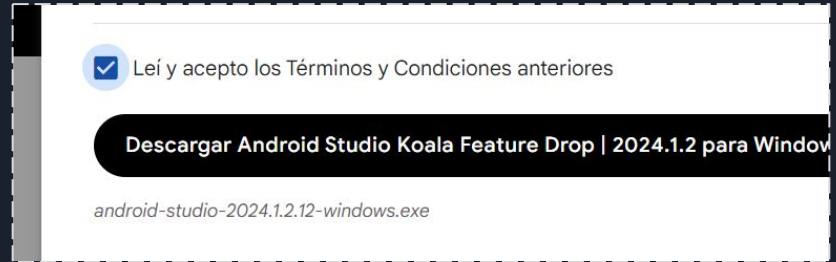
Con este IDE podemos programar nuestras Apps en

- **Java**
- **Kotlin**

# ¿Qué es Android?

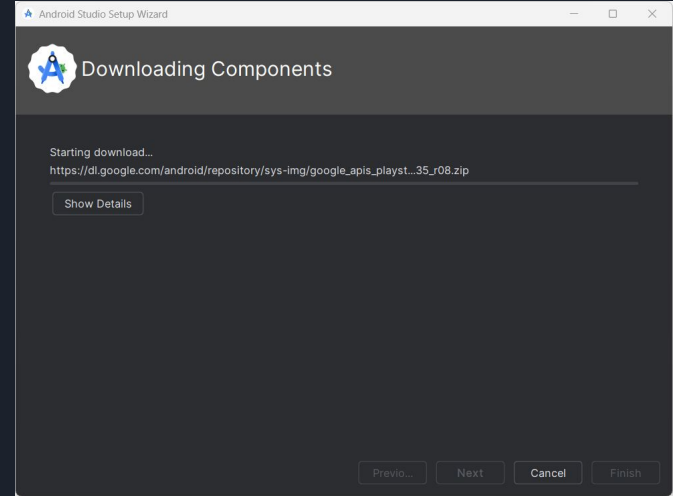
En primer lugar descargaremos el JDK de Java en caso de que no lo tengamos instalado

A continuación descargaremos la última versión de Android Studio

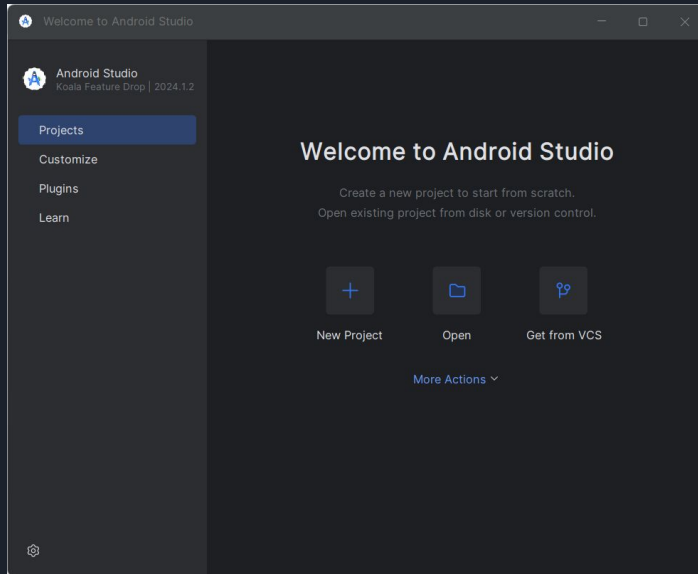


# Instalación de Android Studio

Hacemos una instalación estándar,  
dándole a Siguiente hasta que  
finalicemos.  
Empezarán a descargarse componentes.

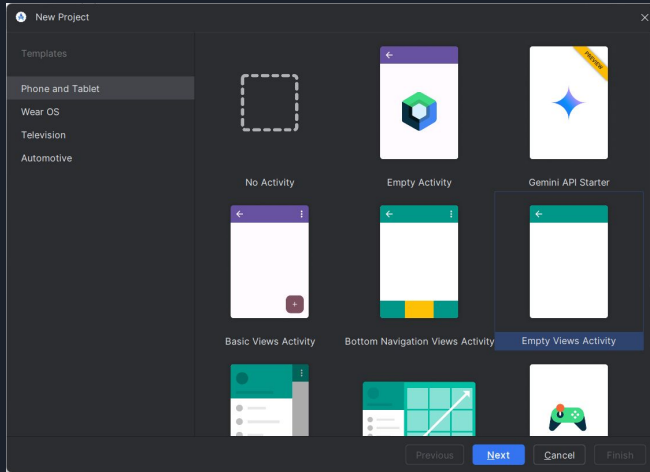


# Inicio de Android Studio



Veremos esta pantalla por primera vez, cuando no tengamos ningún proyecto creado.

# Creación de nuestro primer proyecto



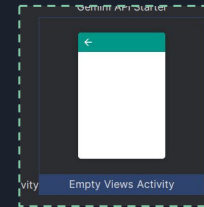
Seleccionamos una **Activity** para comenzar

¿Qué es una **Activity**?

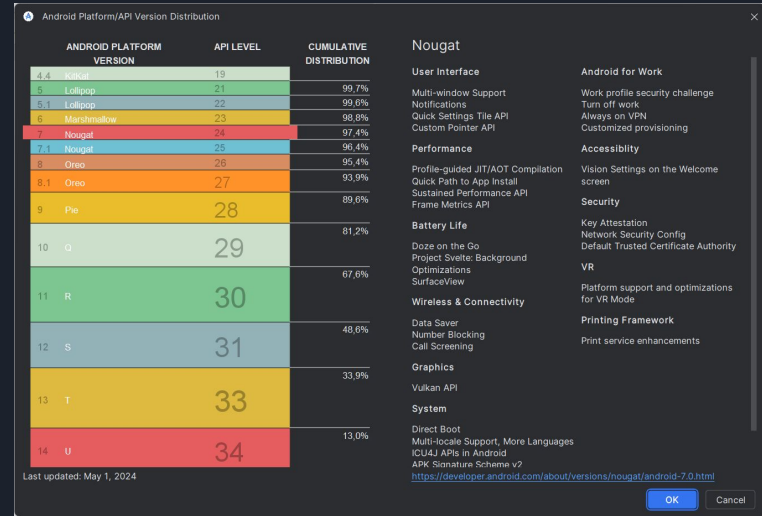
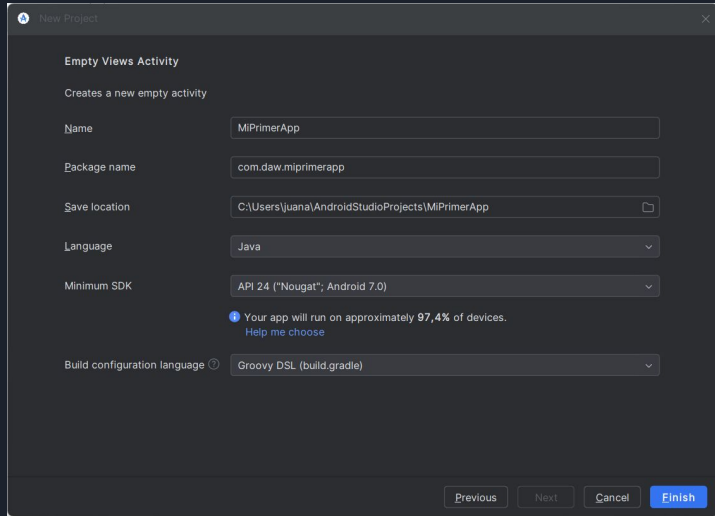
Podríamos decir que es cada una de las pantallas de nuestra App.

Cada Activity tiene una parte lógica (java) y una parte gráfica (xml).

Elegimos una **Empty Views Activity**.



# Creación de nuestro primer proyecto



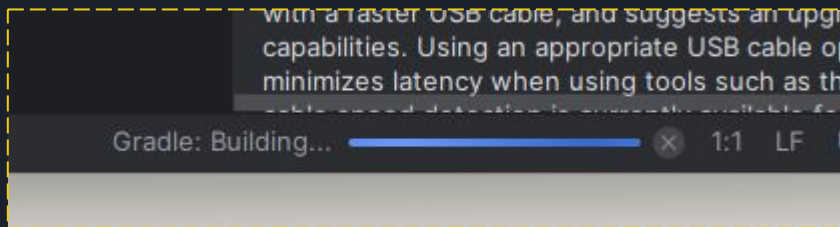
Ponemos un nombre a nuestra App: **MiPrimerApp**  
Cambiamos el nombre del Package  
Elegimos Lenguaje: **Java**  
Elegimos SDK: **Android 7.0 (Nougat)**

Consultamos la versión de Android



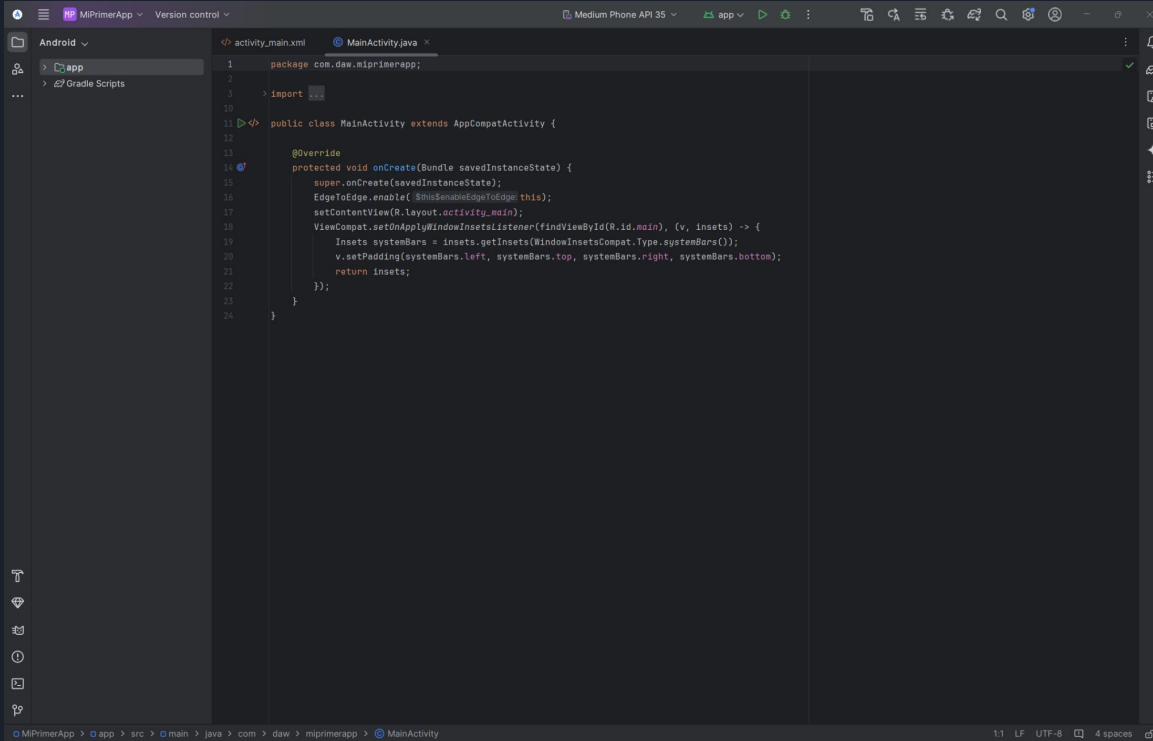


# Creación de nuestro primer proyecto



Es importante que cuando iniciemos el IDE (y sobre todo por primera vez) esperemos a que carguen todos sus componentes antes de trabajar con él. Esto es recomendable para cualquier IDE.

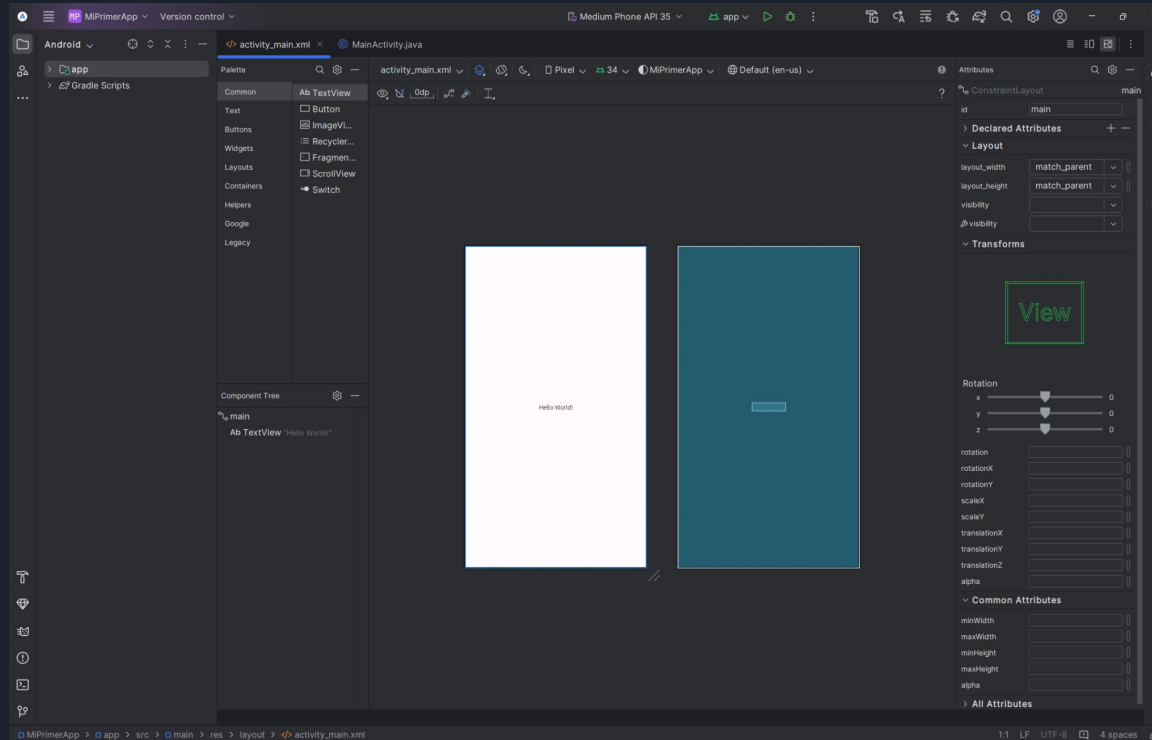
# Creación de nuestro primer proyecto



```
1 package com.daw.miprimerapp;
2
3 import androidx.appcompat.app.AppCompatActivity;
4
5 public class MainActivity extends AppCompatActivity {
6
7     @Override
8     protected void onCreate(Bundle savedInstanceState) {
9         super.onCreate(savedInstanceState);
10         EdgeToEdge.enable(this);
11         setContentView(R.layout.activity_main);
12         ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main), (v, insets) -> {
13             Insets systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars());
14             v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom);
15             return insets;
16         });
17     }
18 }
```

PARTE LÓGICA  
(JAVA)

# Creación de nuestro primer proyecto



PARTE VISUAL  
(XML)

# Creación de nuestro primer proyecto

En la parte gráfica (XML)  
podremos elegir varias vistas



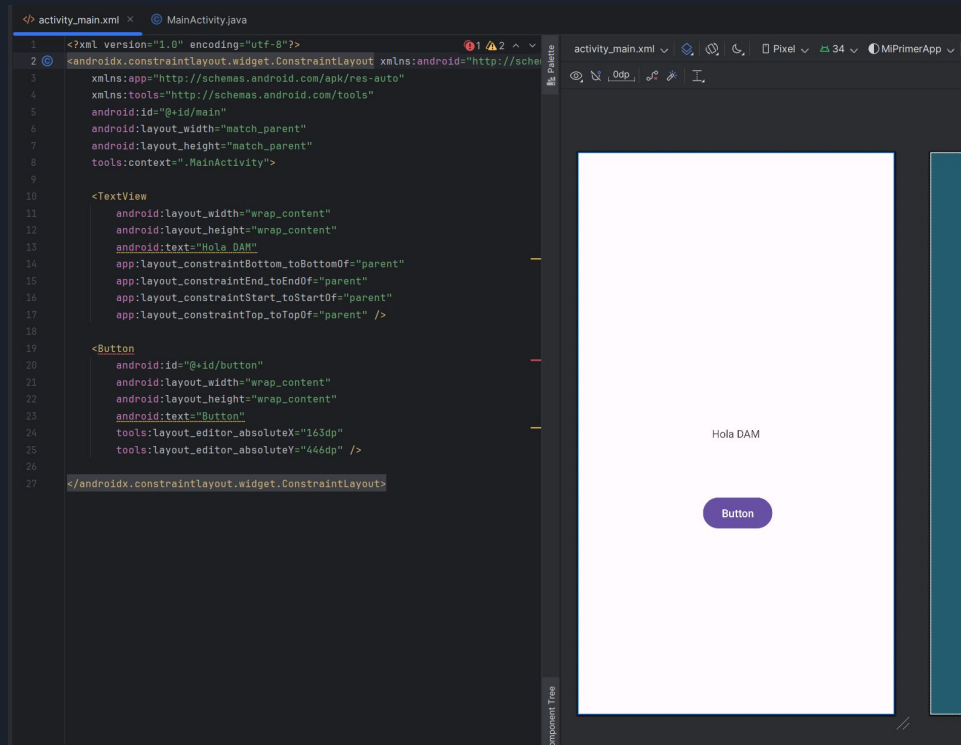
Podemos modificar el texto  
"Hello World!"

- Modo Design
- Modo Blueprint

# Creación de nuestro primer proyecto

Probamos a añadir elementos  
sin usar código.

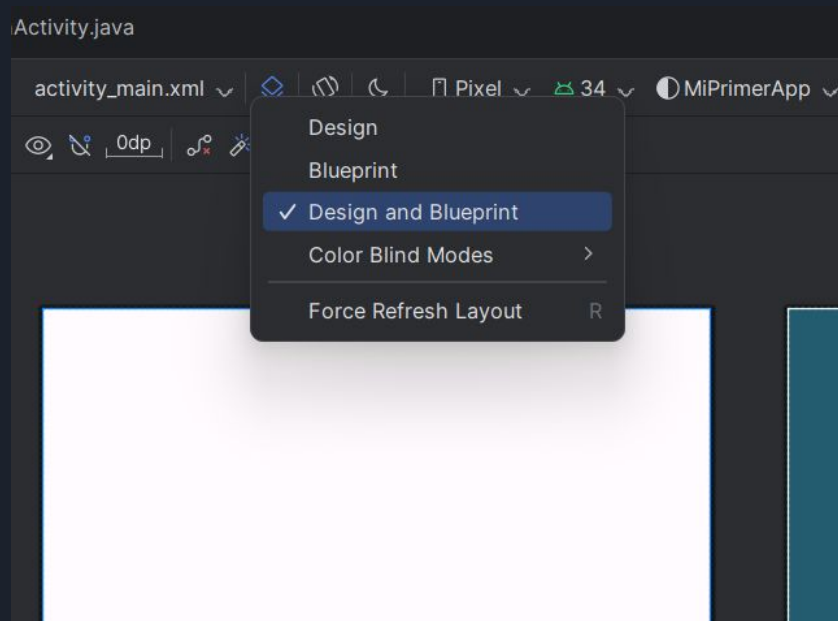
Comprobamos el código



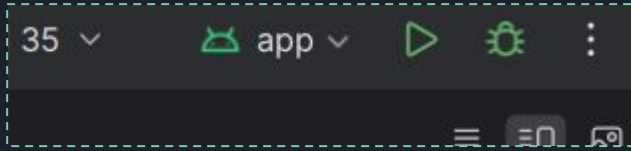
# Creación de nuestro primer proyecto

Podemos modificar el aspecto de algunos elementos sin necesidad de código también. O, incluso, modificarlo desde ese mismo panel.

Si nos molesta la vista Blueprint, podemos quitarla.



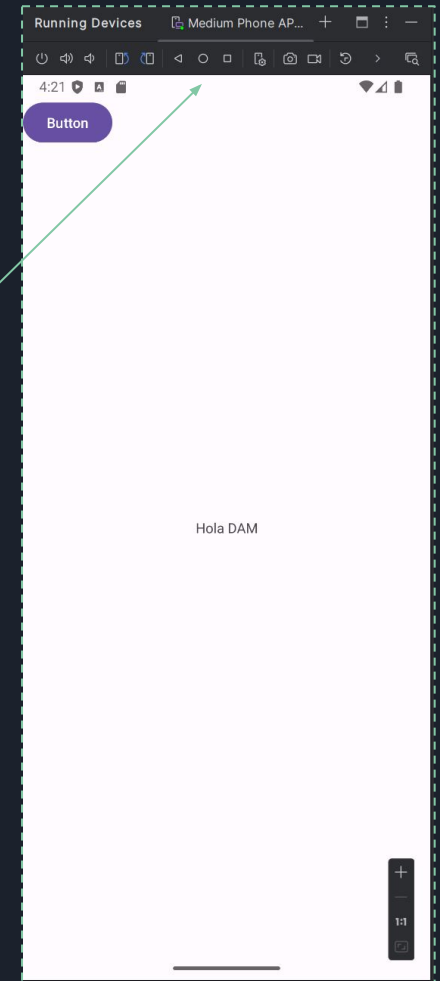
# Visualizar (emular) nuestra App



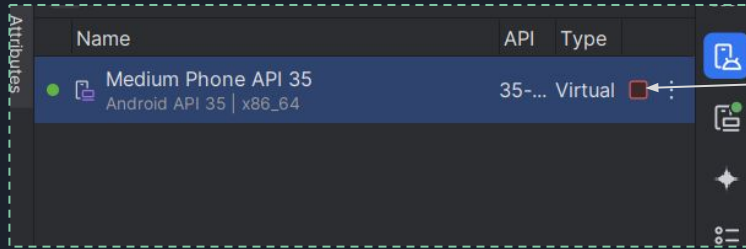
Exploramos las opciones del menú lateral

Le damos al botón PLAY y empezará a iniciarse el dispositivo Android. La primera vez tardará bastante.

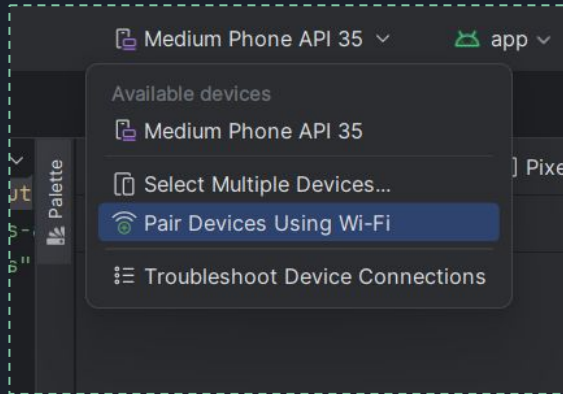
Usamos el móvil como si fuese uno de verdad.



# Visualizar (emular) nuestra App



Si queremos detenerlo.







# Visualizar nuestra App en nuestro smartphone

Es una buena solución cuando nuestro PC no cumple con los requerimientos mínimos de rendimiento

En primer lugar, el ordenador debe reconocer nuestro dispositivo, así que debemos tener especial atención si es la primera vez que lo conectamos.

Entraremos en Ajustes en nuestro teléfono.

> Acerca del dispositivo

> Versión de software ó > Número de compilación

Presionamos 7 veces

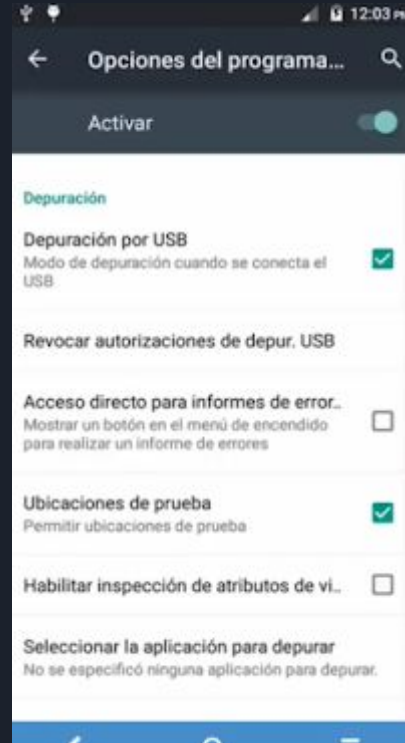
¡Ya eres programador!

Verificamos volviendo a Ajustes {}



# Visualizar nuestra App en nuestro smartphone

Activamos estas dos opciones





# Visualizar nuestra App en nuestro smartphone

Cuando demos a ejecutar, vemos como nos aparece nuestro smartphone.

Probemos nuestra Activity.

Puede que tengamos que instalar unos componentes en nuestro móvil.

Cuando la hayamos probado, hay que desinstalarla para no “generar basura”.