

[T2.3] COPIAR DIVERSAS PANTALLAS DE LA APP DE FACEBOOK PARA CELULAR

- Para esta actividad trabajaremos en equipo Germán y yo.

Es el trabajo para fin de semestre.

FECHA DE ENTREGA

Viernes, 11 de junio del 2021, 5PM

NOMBRE DEL PAQUETE DE LA APP

El nombre del paquete de la app lo cambié para registrarlo con Firebase, siguiendo la nomenclatura:

com.company.appname

- NOMBRE DEL PAQUETE:
 - com.uaslp.facebook_simple_replication

TABLA DE CONTENIDOS

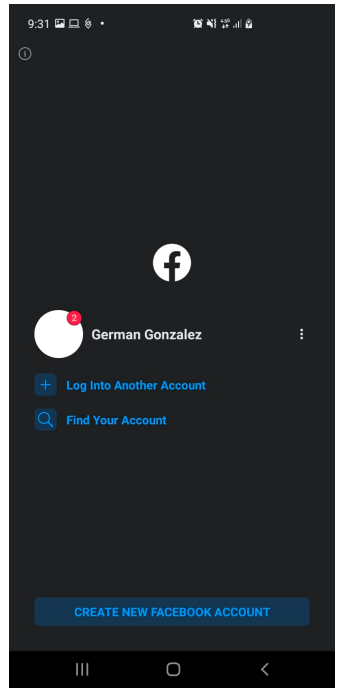
- 1. **FECHA DE ENTREGA**
- 2. **NOMBRE DEL PAQUETE DE LA APP**
- 4. **PANTALLAS OBJETIVO**
- 5. **RESULTADOS DE LAS PANTALLAS**
- 6. DOCUMENTACIÓN AUTOMÁTICA CON *dartdoc*
- 7. **INSTALACIONES**
 - 7.1. **Android Studio**
 - 7.1.1. **CONFIGURACIÓN OBLIGATORIA ANDROID STUDIO**
 - 7.1.1.1. **INSTALACIÓN DE PAQUETES**
 - 7.1.1.2. **INSTALACIÓN DE PLUGINS**
 - 7.1.1.3. **COMPROBACIÓN DE ACTUALIZACIONES**
 - 7.1.2. **CREAR UN EMULADOR DE ANDROID (ANDROID VIRTUAL DEVICE)**
 - 7.1.2.1. **RECOMENDACIÓN IMPORTANTE PARA EL AVD**
 - 7.2. **Flutter**
 - 7.2.1. **EN QUÉ DIRECTORIO COLOCAR FLUTTER**
 - 7.2.2. **FLUTTER *SOUND NULL SAFETY***
 - 7.2.3. **VARIABLE DE USUARIO**
 - 7.3. **Chocolatey**
 - 7.3.1. **INSTRUCCIONES INSTALACIÓN DESDE POWERSHELL**
 - 7.4. **scrcpy**
 - 7.4.1. **INSTRUCCIONES INSTALACIÓN DESDE POWERSHELL con CHOCOLATEY**
 - 7.5. **JAVA**
 - 7.5.1. **INFORMACIÓN SOBRE LAS VERSIONES DE JAVA**
 - 7.5.2. **CONCEPTOS EN LOS ACRÓNIMOS DE JAVA: *JDK, JVM, JRE***

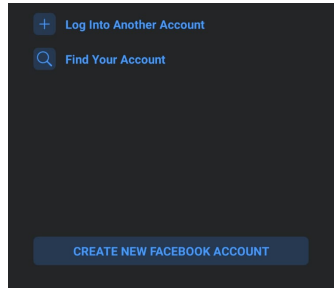
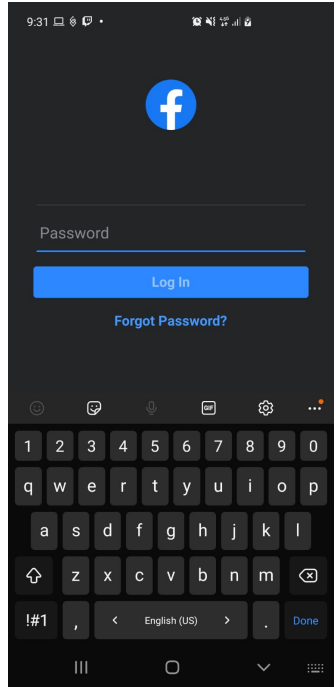
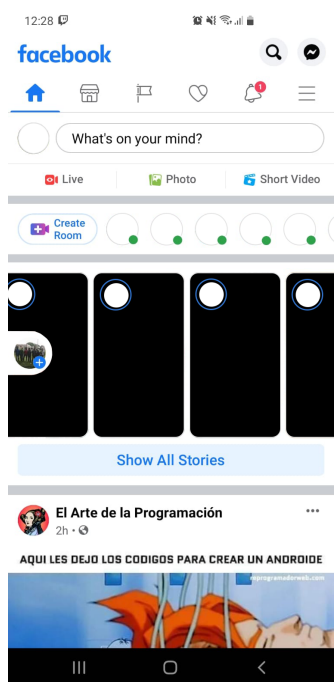
- 7.5.3. **DESCARGA DE JAVA**
- 8. **VARIABLES DEL SISTEMA o de USUARIO**
 - 8.1. **ACCEDER A LA SECCIÓN PARA EDITAR LAS VARIABLES**
 - 8.2. **VARIABLE DEL SISTEMA: JAVA_HOME**
 - 8.2.1. VERIFICACIÓN DE QUE JAVA_HOME ESTÁ ESTABLECIDO
 - 8.2.2. OTRAS VARIABLES DE USUARIO PARA FLUTTER y DART
- 9. **FIREBASE**
 - 9.1. SCREENSHOTS CONFIGURACIÓN FIREBASE
 - 9.2. AGREGAR LAS SIGUIENTES LÍNEAS
- 10. **RECURSOS DE FACEBOOK APP (SVG, ÍCONOS, ...)**
 - 10.1. LOGO DE FACEBOOK
 - 10.2. **REACCIONES DE FACEBOOK**
 - 10.3. USO DE ÍCONOS
 - 10.3.1. GOOGLE FONTS | MATERIAL ICONS
 - 10.3.1.1. Íconos que podríamos utilizar
 - 10.3.2. **Íconos con Font Awesome**
 - 10.3.2.1. **Íconos que podríamos utilizar**
- 11. **ENLACES EXTRA QUE FUERON ÚTILES**

1.4. PANTALLAS OBJETIVO

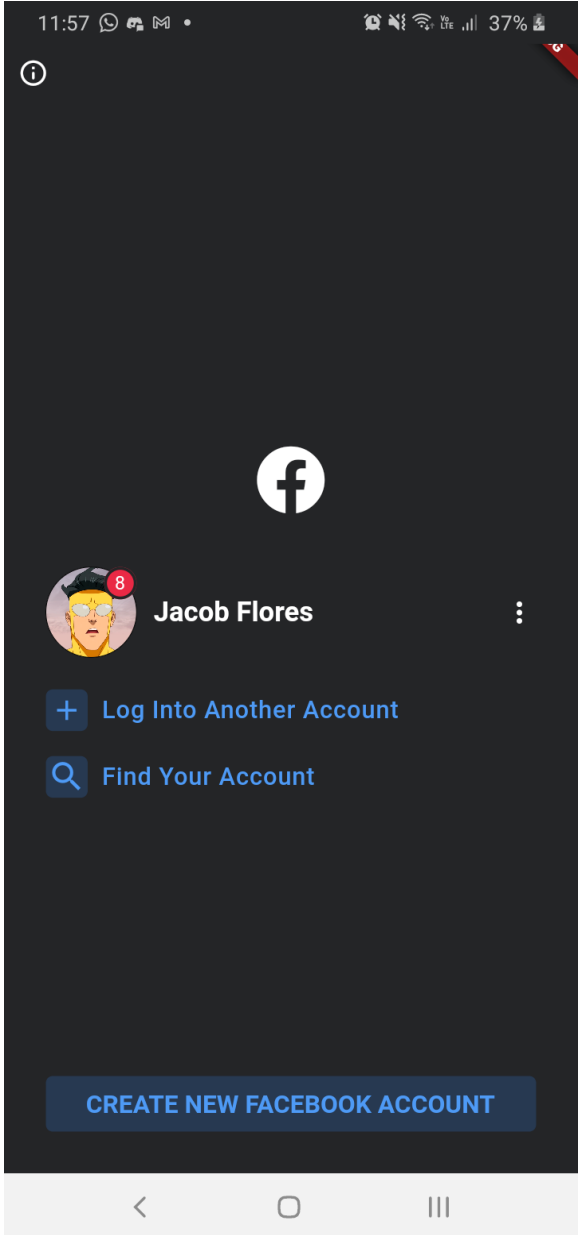
Hay dos pantallas que tenemos como objetivo, pero una es opcional: pantalla de inicio de sesión, y pantalla después de dar click al botón de inicio de sesión.

- **Pantalla principal de inicio de sesión (los colores se modificaron al exportar desde Photoshop, ya que le quité la imagen original).**

NÚMERO DE PANTALLA	DESCRIPCIÓN DE PANTALLA	IMAGEN DE LA PANTALLA
1	PRIMERA PANTALLA DE INICIO DE SESIÓN CON LOS COLORES MODIFICADOS	

NÚMERO DE PANTALLA	DESCRIPCIÓN DE PANTALLA	IMAGEN DE LA PANTALLA
1	COLORES ORIGINALES DE LA PRIMERA PANTALLA DE INICIO DE SESIÓN	 A screenshot of the Facebook login screen. It features a dark background with a blue Facebook logo at the top. Below the logo, there are two links: 'Log Into Another Account' and 'Find Your Account'. At the bottom, there is a blue button that says 'CREATE NEW FACEBOOK ACCOUNT'.
2	PANTALLA PARA INICIAR SESIÓN DESPUÉS DE DAR CLICK AL BOTÓN "Log Into Another Account"	 A screenshot of the Facebook login screen after clicking 'Log Into Another Account'. It shows a password input field with the label 'Password' and a blue 'Log In' button. Below the button is a link for 'Forgot Password?'. A virtual keyboard is visible at the bottom of the screen.
3	FEED PRINCIPAL DESPUÉS DE HABER INICIADO SESIÓN (LOS ELEMENTOS SON IMPRECISOS RESPECTO A POSICIÓN Y TAMAÑO PORQUE LA IMAGEN FUE EDITADA PARA OCULTAR INFORMACIÓN PERSONAL. NOS BASAMOS EN LA IMAGEN SIN EDITAR.)	 A screenshot of the Facebook main feed. It shows the top navigation bar with icons for home, search, create, activity, and menu. Below the navigation bar is a text input field with the placeholder 'What's on your mind?'. There are tabs for 'Live', 'Photo', and 'Short Video'. A 'Create Room' button is also visible. The feed itself shows a post from 'El Arte de la Programación' with a video thumbnail and the text 'AQUI LES DEJO LOS CODIGOS PARA CREAR UN ANDROIDO'.

1.5. RESULTADOS DE LAS PANTALLAS

NÚMERO DE PANTALLA	DESCRIPCIÓN DE PANTALLA	IMAGEN DE LA PANTALLA
1	PRIMERA PANTALLA DE INICIO DE SESIÓN QUE MUESTRA NÚMERO DE NOTIFICACIONES [SAMSUNG A20S]	

1.6. DOCUMENTACIÓN AUTOMÁTICA CON *dartdoc*

En dart, gracias a un comando llamado *dartdoc*, es posible generar la documentación de *Dart* y *Flutter* de forma automática, pero hay que seguir ciertos pasos:

1. Hay que establecer la siguiente variable de usuario en **Path**:

- **C:\SDKs\flutter\bin\cache\dart-sdk\bin**

2. Hay que seguir una convención de comentarios específica para que se generen descripciones en los elementos de la documentación. Estas convenciones se encuentran definidas en **Effective Dart: Documentation**, en donde lo más importante es lo siguiente:

- Los comentarios deben de iniciar con **3 backslashes (\\)** si queremos que aparezca el texto en la documentación. Si lo hacemos con 2 (\\), no saldrá.

- Podemos agregar descripciones de archivos, clases, atributos, funciones, y no estoy seguro de si se puede de más cosas, pero eso se puede encontrar en la misma documentación.
- Hay ciertas convenciones para que estén estructurados de una mejor forma (según **Effective Dart: Documentation**), por lo que si quieres saber más al respecto, tendrás que revisar la documentación oficial.

3. Entrar al directorio inicial de tu aplicación.

```
cd carpeta_prinicipal_app_flutter
```

Dentro podrás encontrar (entre otros más) los siguientes directorios:

- **android/**
- **build/**
- **ios/**
- **lib/**

4. Ejecutar el siguiente comando para activar **dartdoc**:

```
flutter pub global activate dartdoc
```

5. Ejecutar el siguiente comando para generar la documentación:

```
dartdoc
```

6. Una vez realizados los pasos anteriores, encontraremos la documentación en formato **html** junto con más documentos en el siguiente directorio:

```
carpeta_prinicipal_app_flutter\doc\api\
```

El archivo específico es el siguiente:

```
carpeta_prinicipal_app_flutter\doc\api\index.html
```

7. Para ver la documentación, solamente hay que abrir el archivo y se abrirá una ventana en nuestro navegador. El documento tiene enlaces a los elementos, además de la descripción de cada uno a los que le agregamos con los comentarios en el formato mencionado anteriormente (\\\).

1.7. INSTALACIONES

Para que todo funcione correctamente, hay que instalar varias cosas:

1.7.1. **Android Studio**

Simplemente descarga el archivo que se encuentra en el enlace y ejecutalo. Después podrás configurarlo como tú quieras.

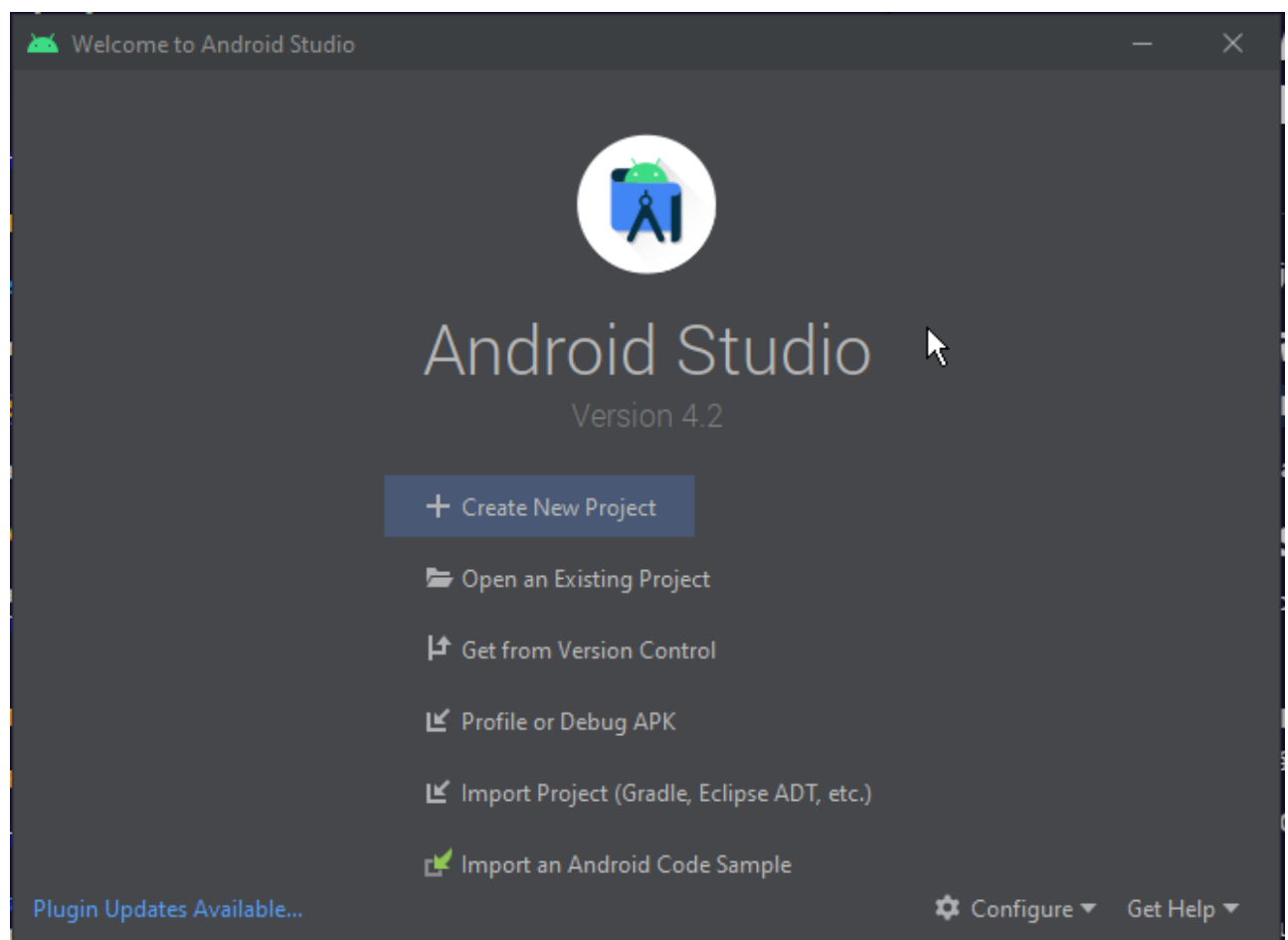
1.7.1.1. CONFIGURACIÓN OBLIGATORIA ANDROID STUDIO

Para que Android Studio y Flutter funcionen de manera correcta, hay que configurar unos elementos que serán obligatorios, siguiendo los siguientes pasos:

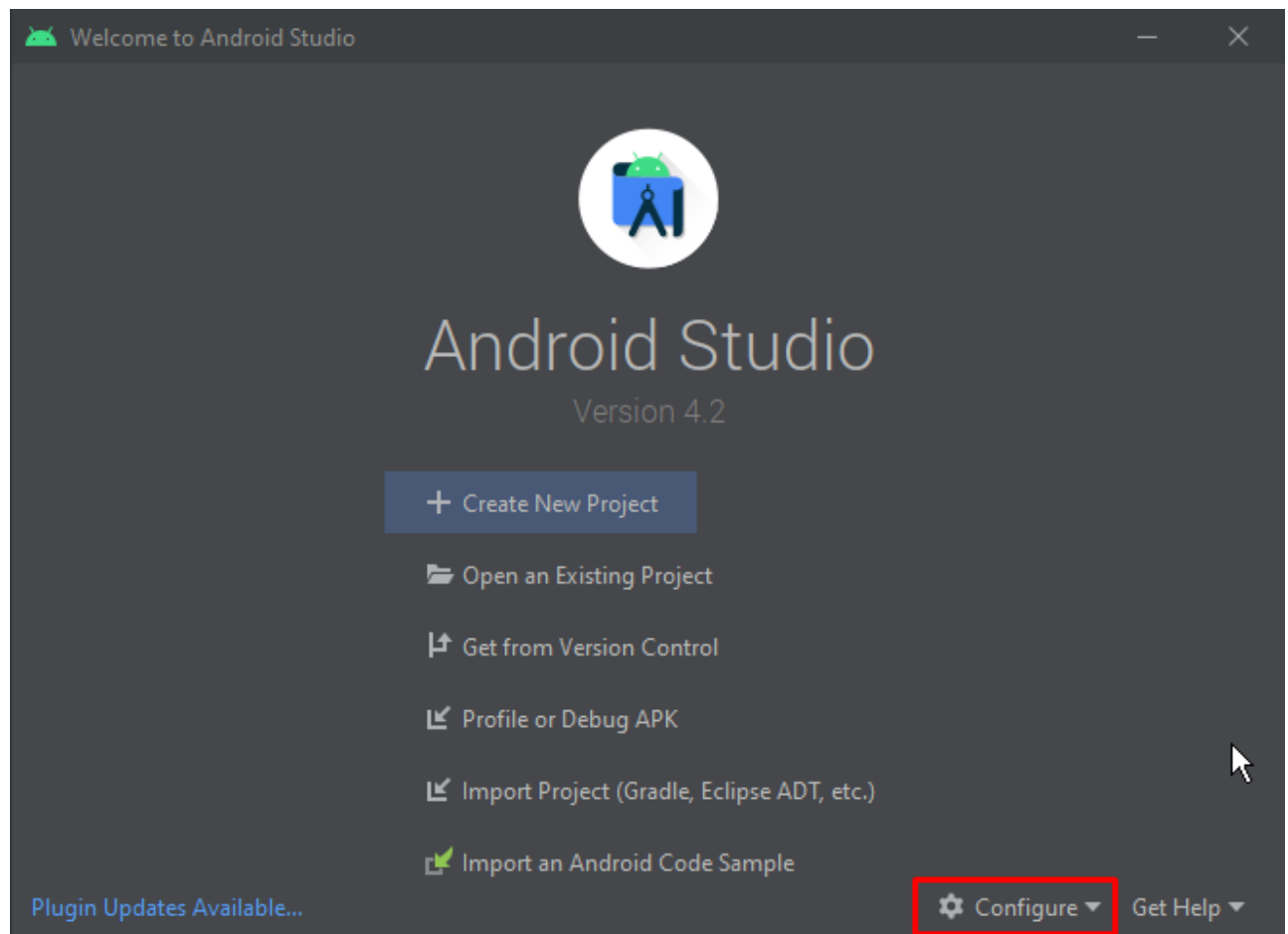
1.7.1.1.1. INSTALACIÓN DE PAQUETES

Hay una serie de paquetes necesarios para trabajar con Flutter y el emulador de Android, entre otras cosas más.

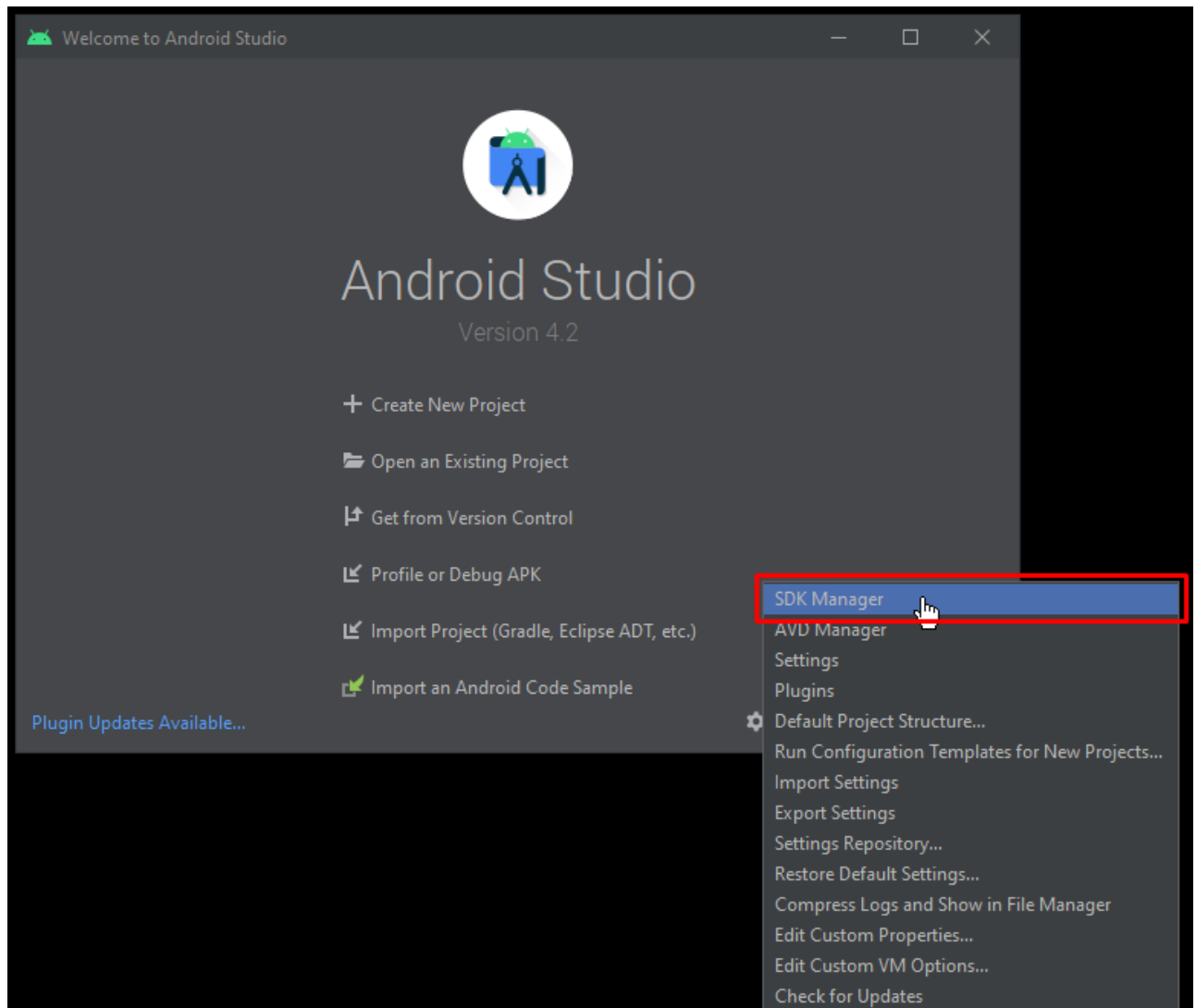
1. Abre Android Studio.



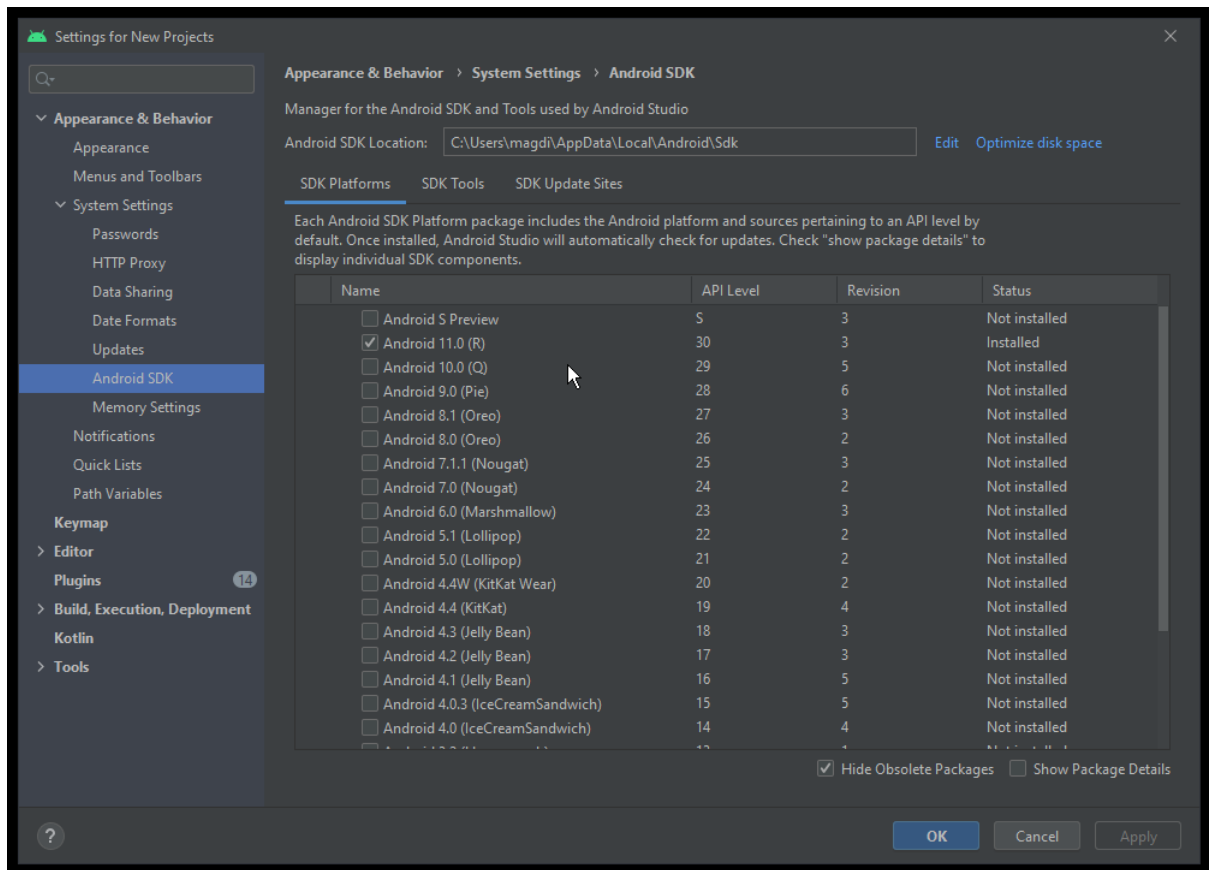
2. Da click en el botón  **Configure**.



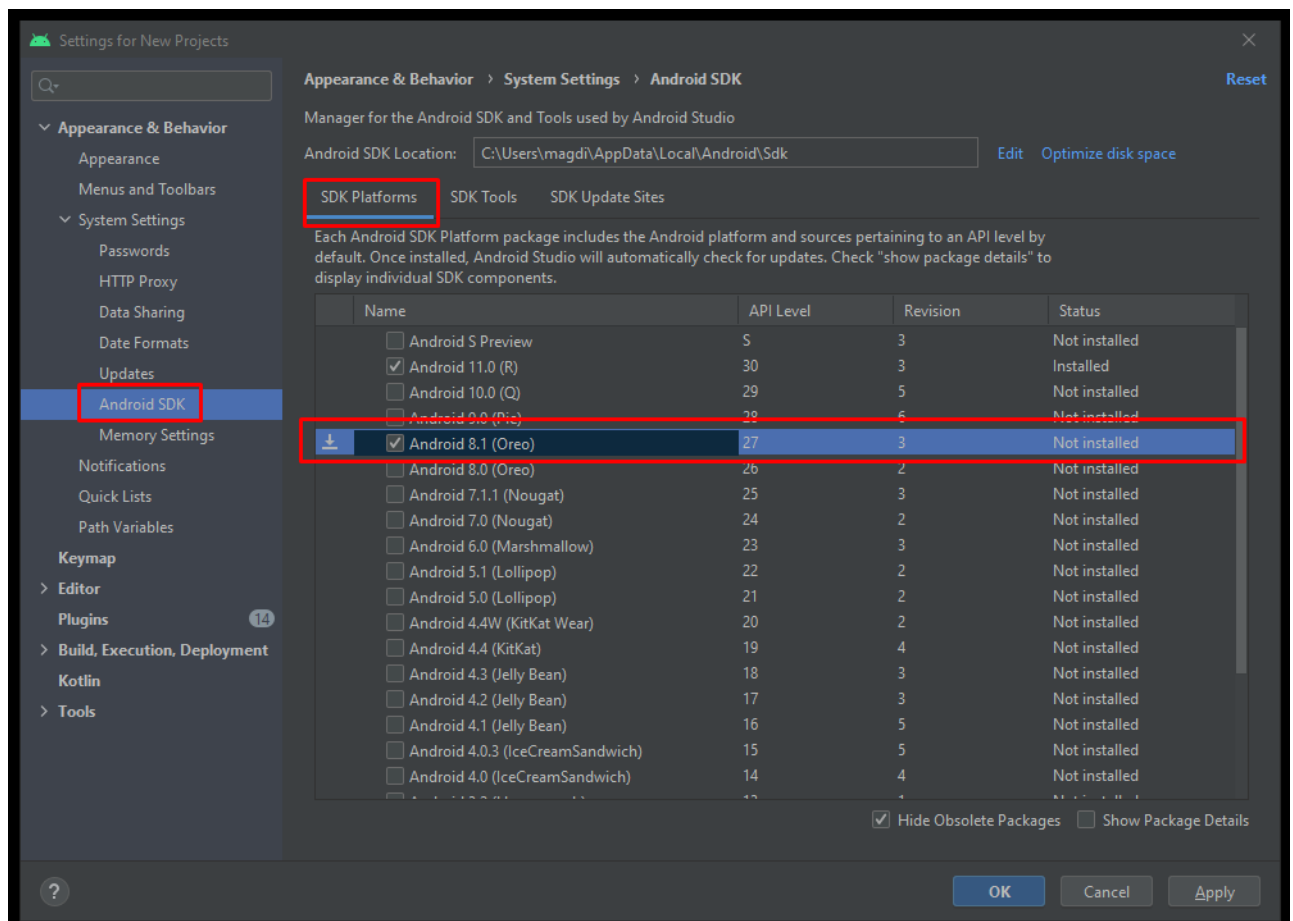
3. Abre la ventana de configuraciones del **SDK (Software Developer Kit)** dando click en el botón **SDK Manager**.



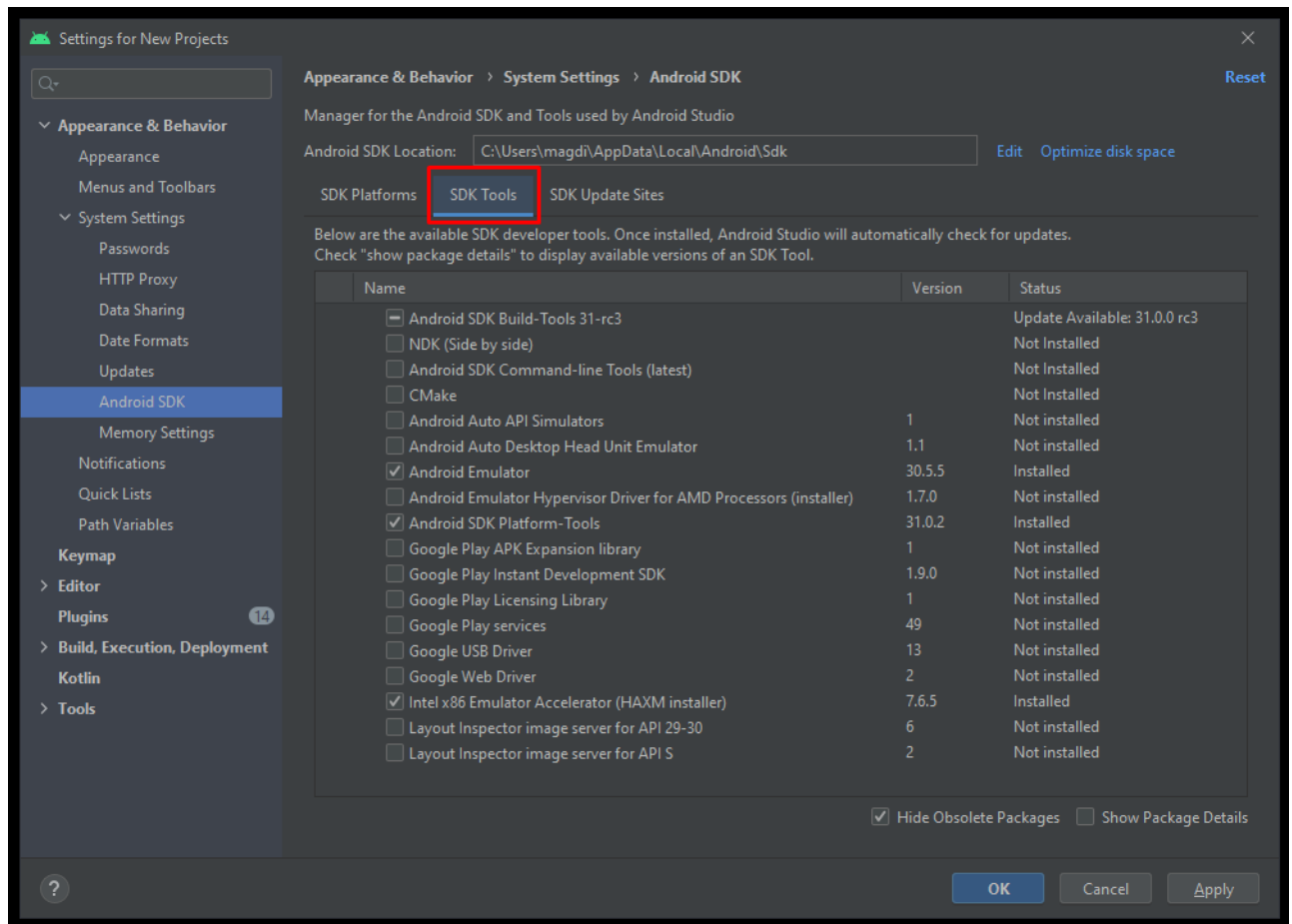
- Te aparecerá una ventana con la lista de los **Android SDK**, los cuales te permitirán desarrollar aplicaciones en ese sistema operativo.



4. De la lista de **SDK Platforms** puedes elegir el que quieras, pero en mi caso tuve que elegir el **Android 8.1 (Oreo) | API Level: 27** porque el profesor así lo indicó por su estabilidad.

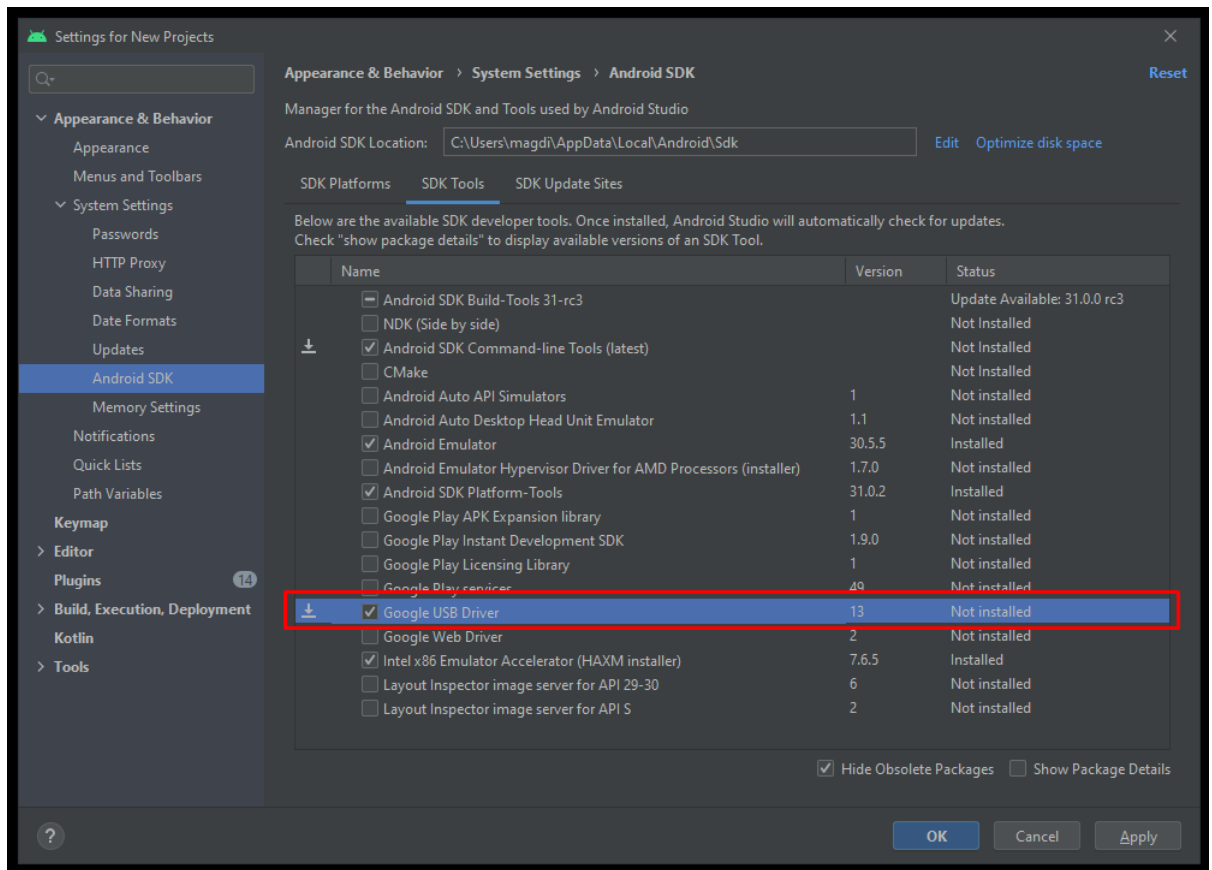


5. Ahora dirígete a la pestaña de **SDK Tools** dentro del mismo submenú. Está a la derecha de la pestaña del paso anterior (**SDK Platforms**).



6. Tendrás que seleccionar una serie de elementos (*sin quitar la selección de los seleccionados por default*) que te permitirán trabajar con el desarrollo de tus aplicaciones y el uso del **emulador de Android**. Estos elementos son los siguientes:

- **Google USB Driver**



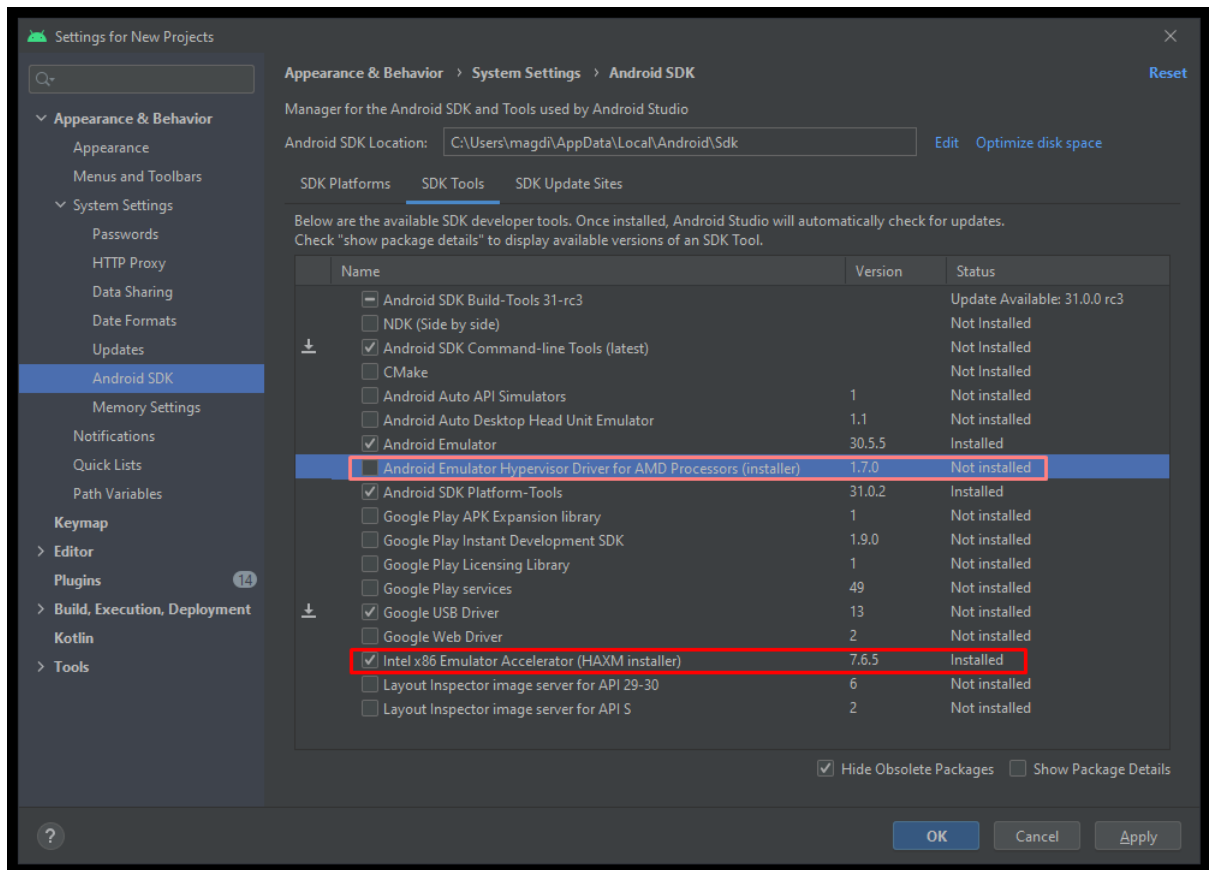
- **El siguiente elemento depende de tu procesador:**

- **Si tienes un procesador de Intel, selecciona la siguiente opción:**

Intel x86 Emulator Accelerator (HAXM Installer)

- **Si tienes un procesador de AMD, selecciona la siguiente opción:**

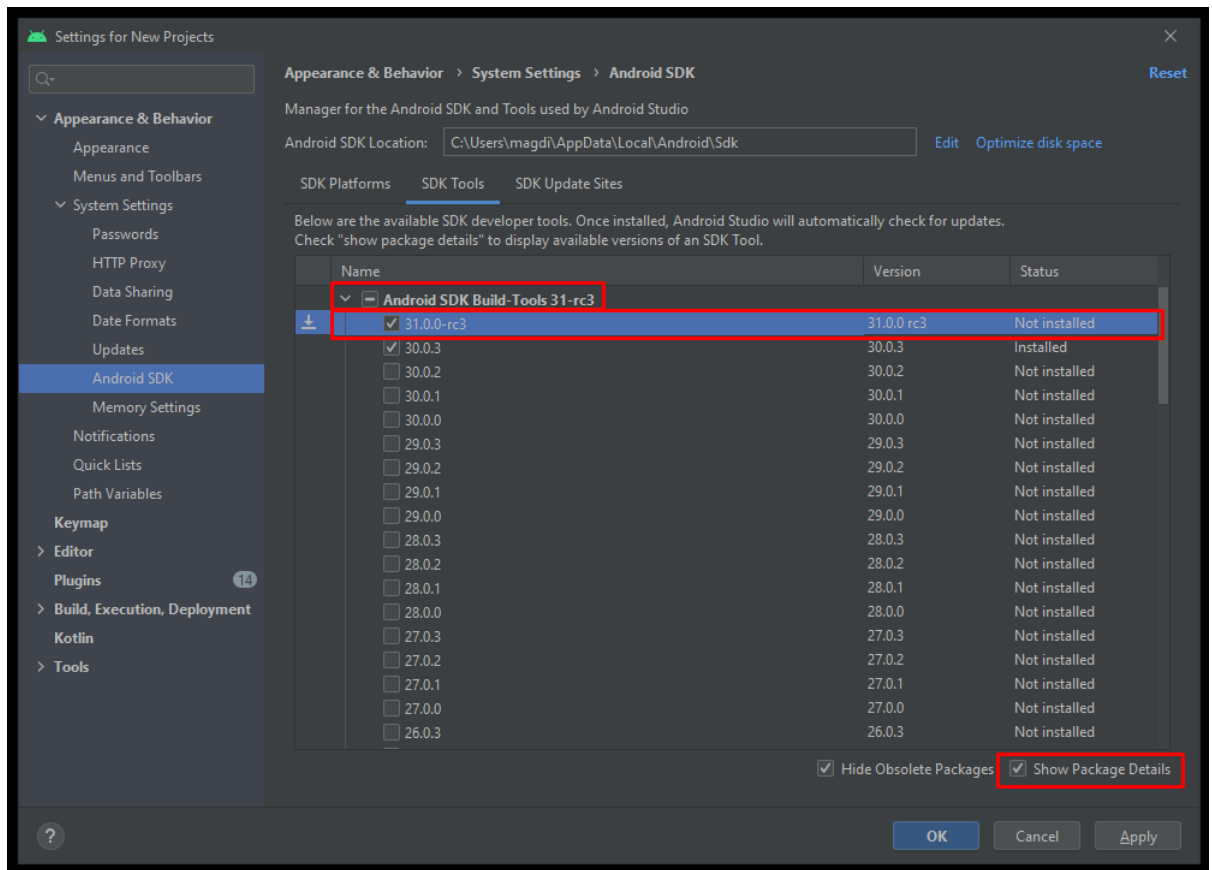
Android Emulator Hypervisor Driver for AMD Processors (installer)



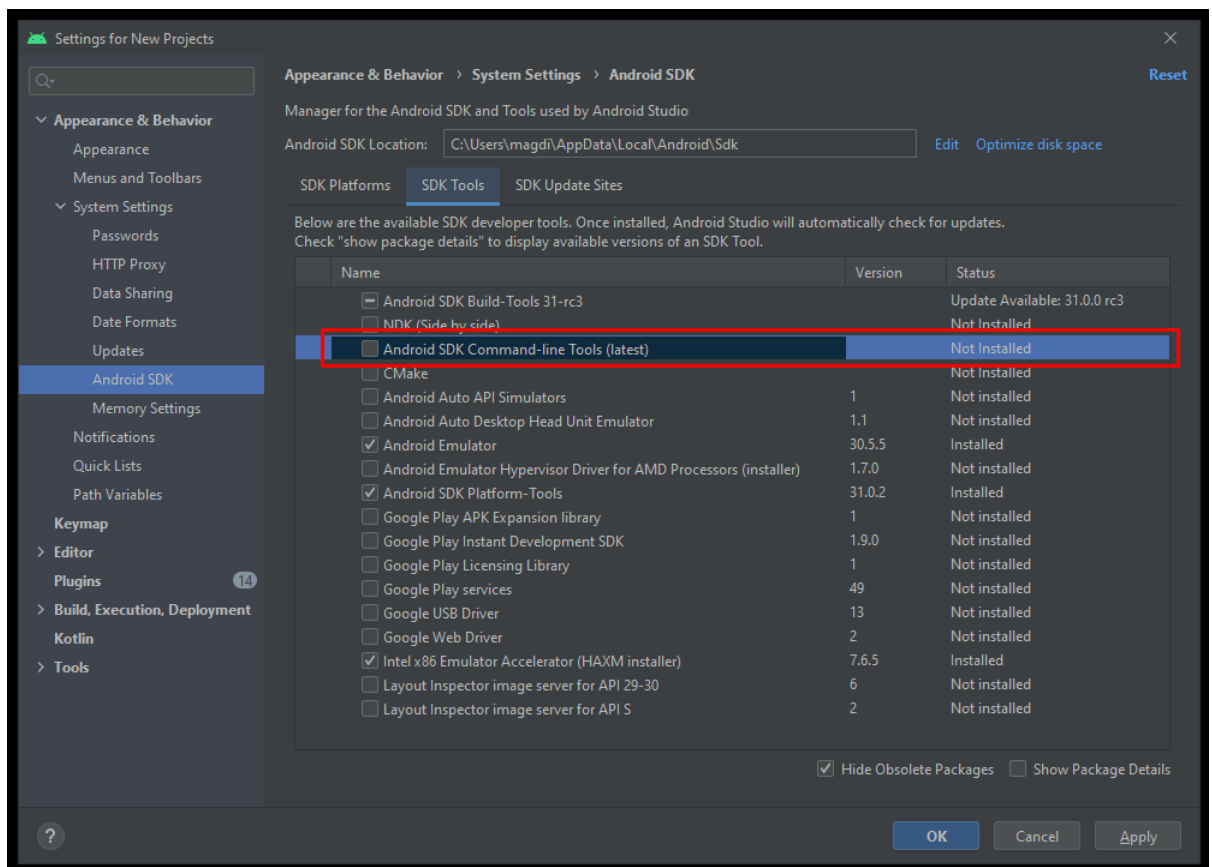
- [OPCIONAL POR SI QUIERES LA ÚLTIMA VERSION (NO ESTABLE) DE ESTE ELEMENTO]

Android SDK Build-Tools 31-rc3 (YO NO LA SELECCIONARÉ, A MENOS QUE TENGA PROBLEMAS CON LA ÚLTIMA VERSIÓN ESTABLE)

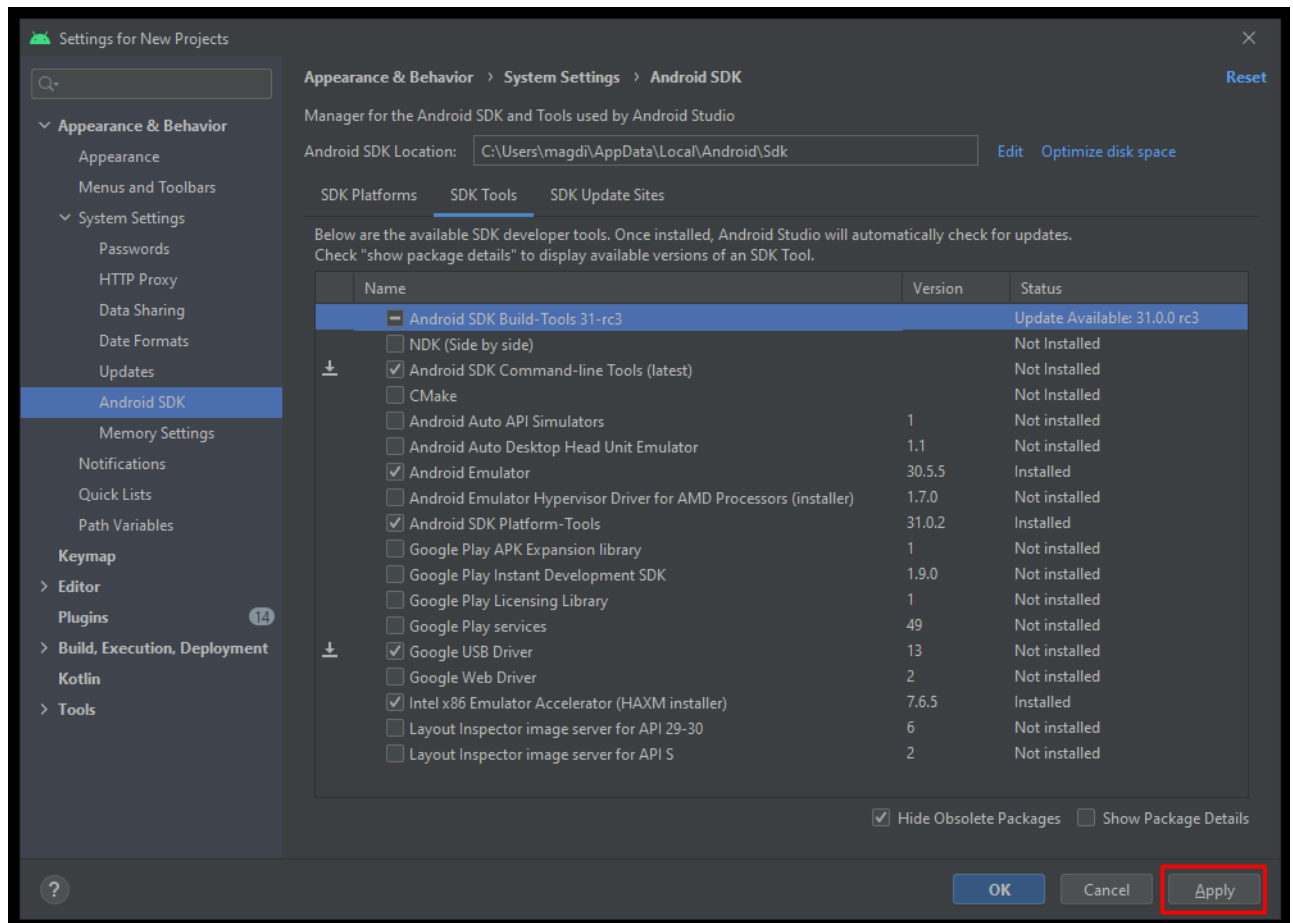
1. Da click en el botón de la parte inferior derecha que dice "**Show Package Details**" para mostrar TODAS las versiones de cada paquete.
2. Da click en la última versión del paquete, la cual puede estar hasta o hasta abajo de la lista, dependiendo del orden en el que estén mostrados. Es la versión con los números más altos.



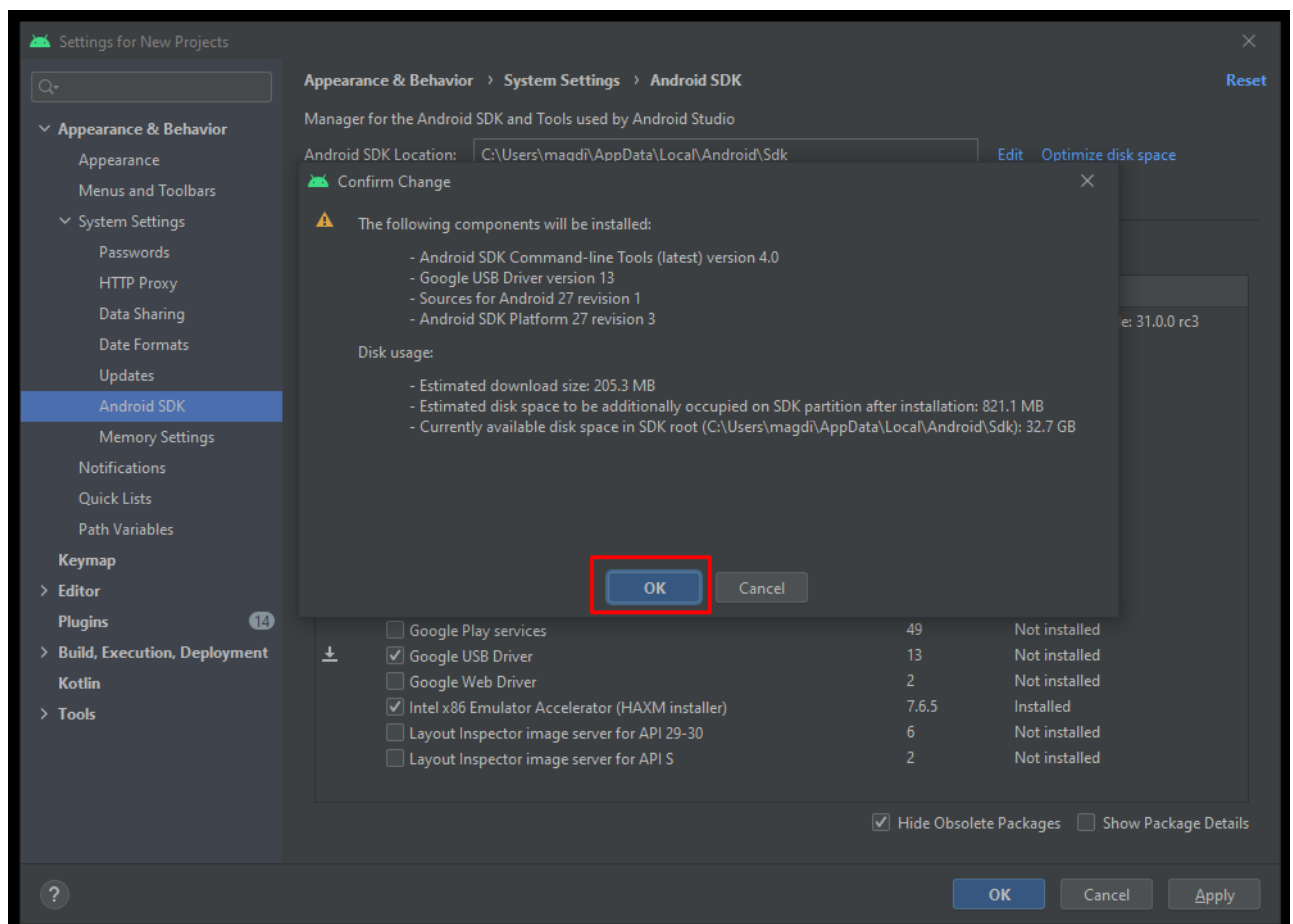
- [OPCIONAL POR SI QUIERES PODER UTILIZAR LA LÍNEA DE COMANDOS] **Android SDK Command-line Tools (latest)**



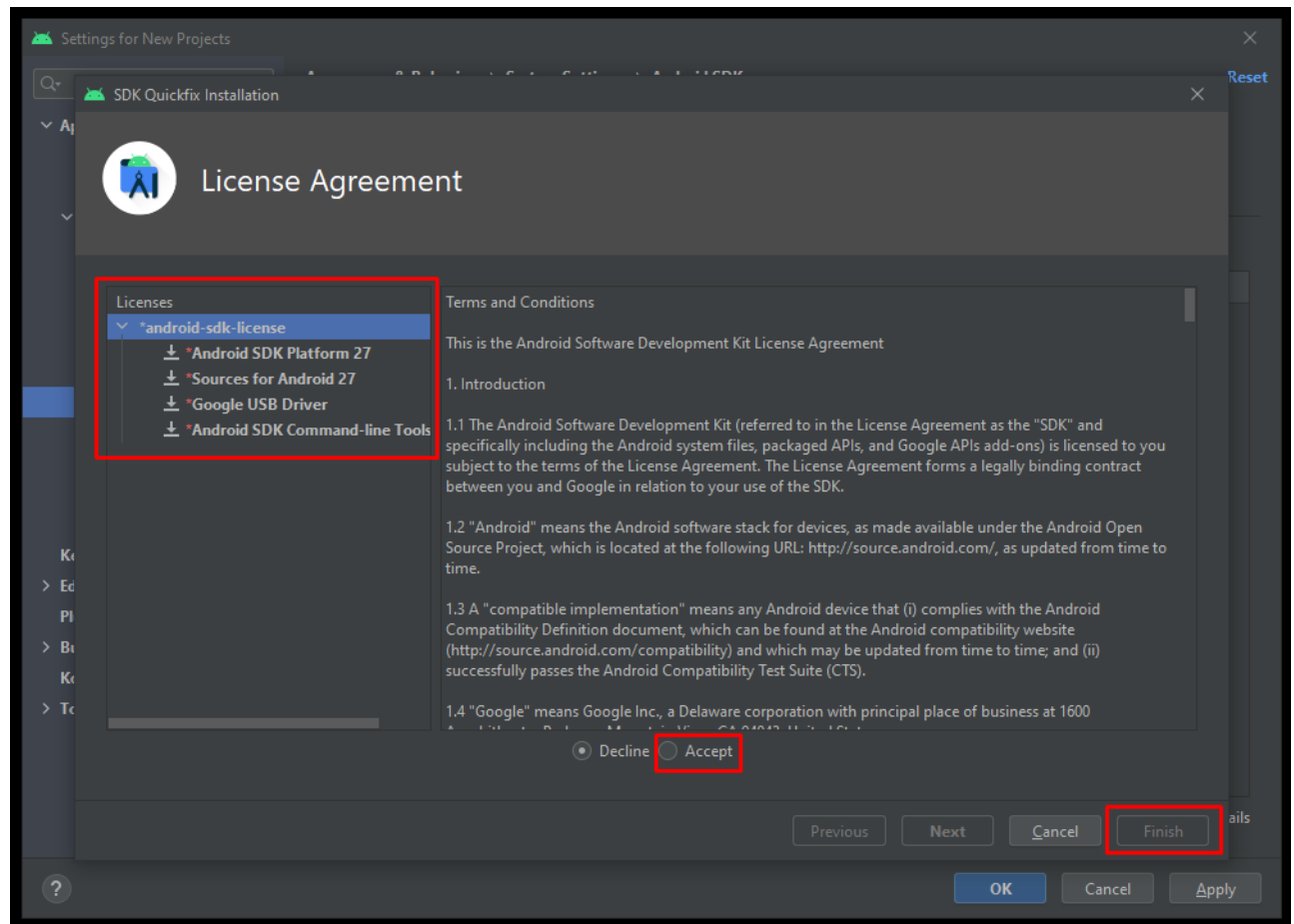
7. Ahora, para que todo sea instalado, da click en el botón "Apply".



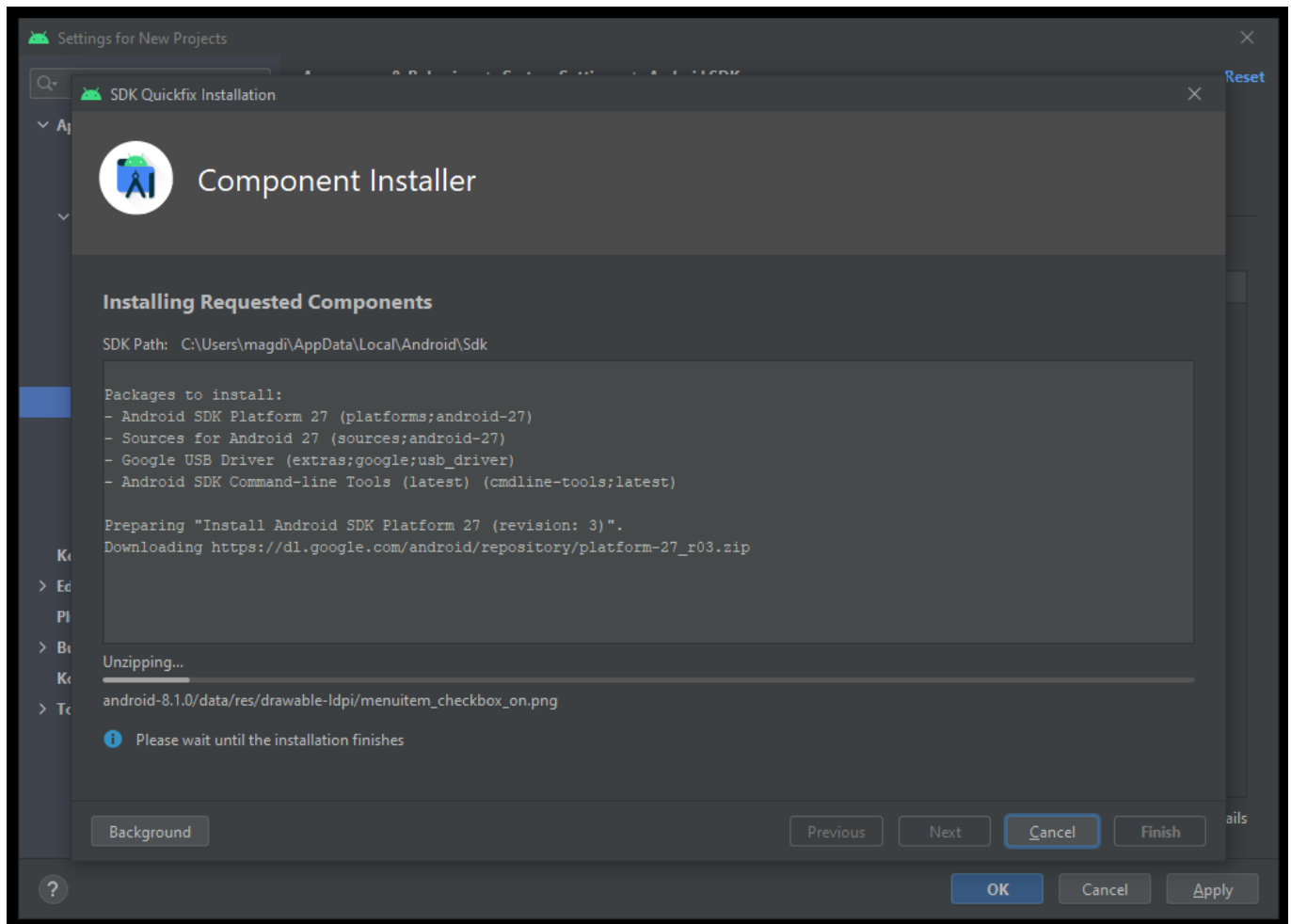
8. Aparecerá una ventana que te pedirá confirmar los cambios. Da click en el botón "OK".



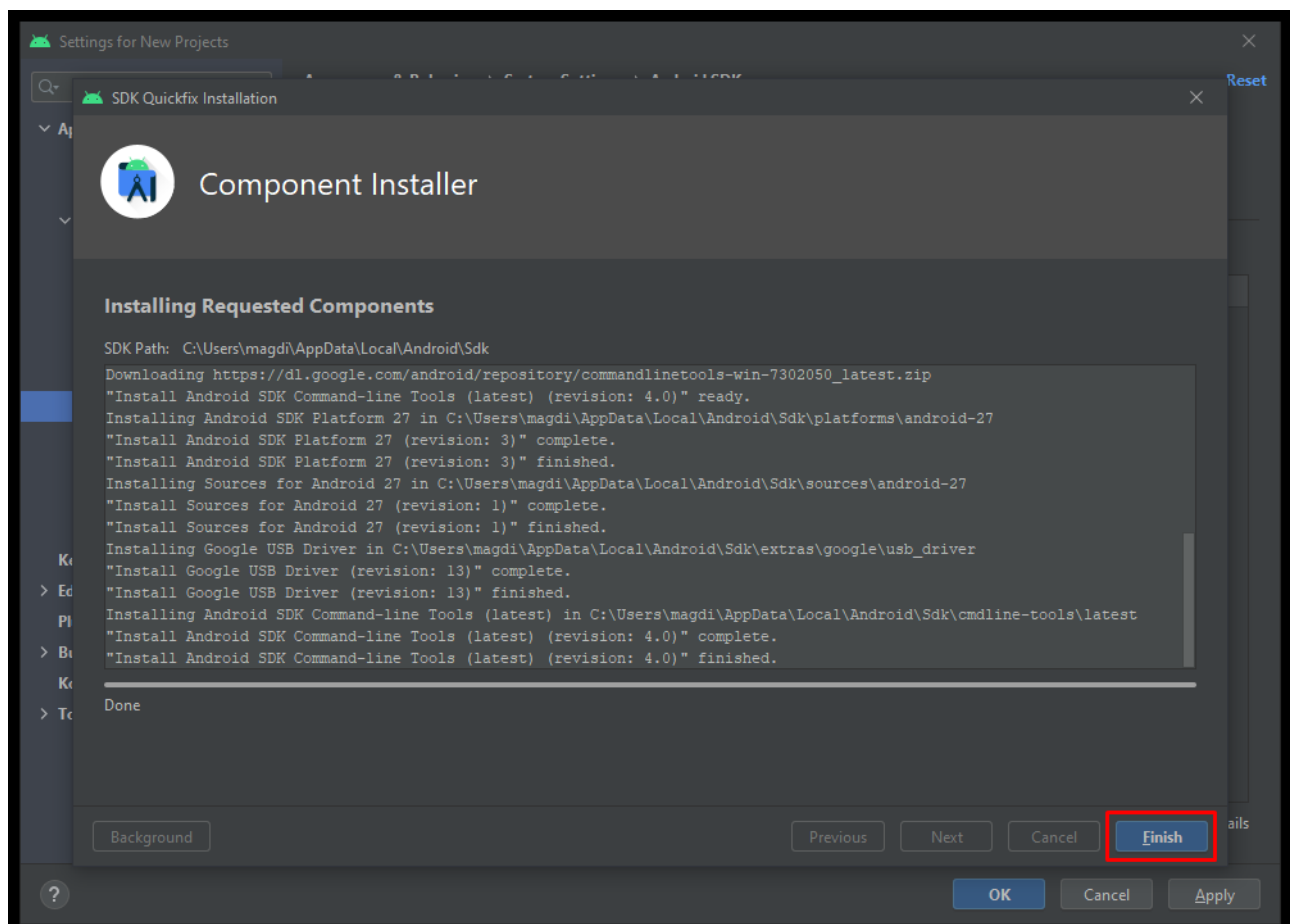
9. Aparecerá otra ventana en donde tendrás que aceptar los términos y condiciones de cada licencia. Acéptalos y da click al botón **"Finish"**.



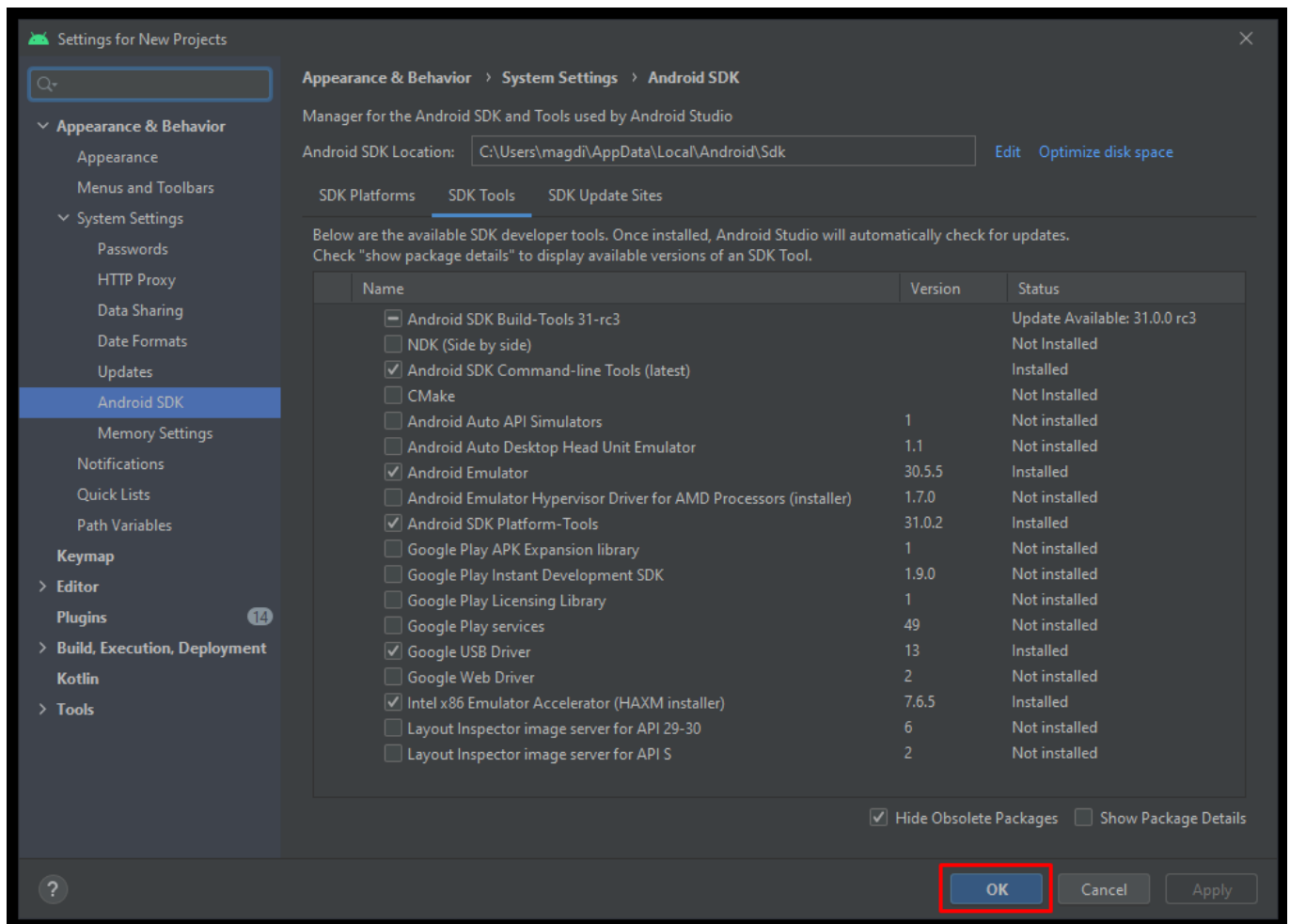
10. Después de esto, te mostrará el progreso de instalación de los paquetes.



11. Una vez terminada la instalación, da click en el botón "**Finish**".



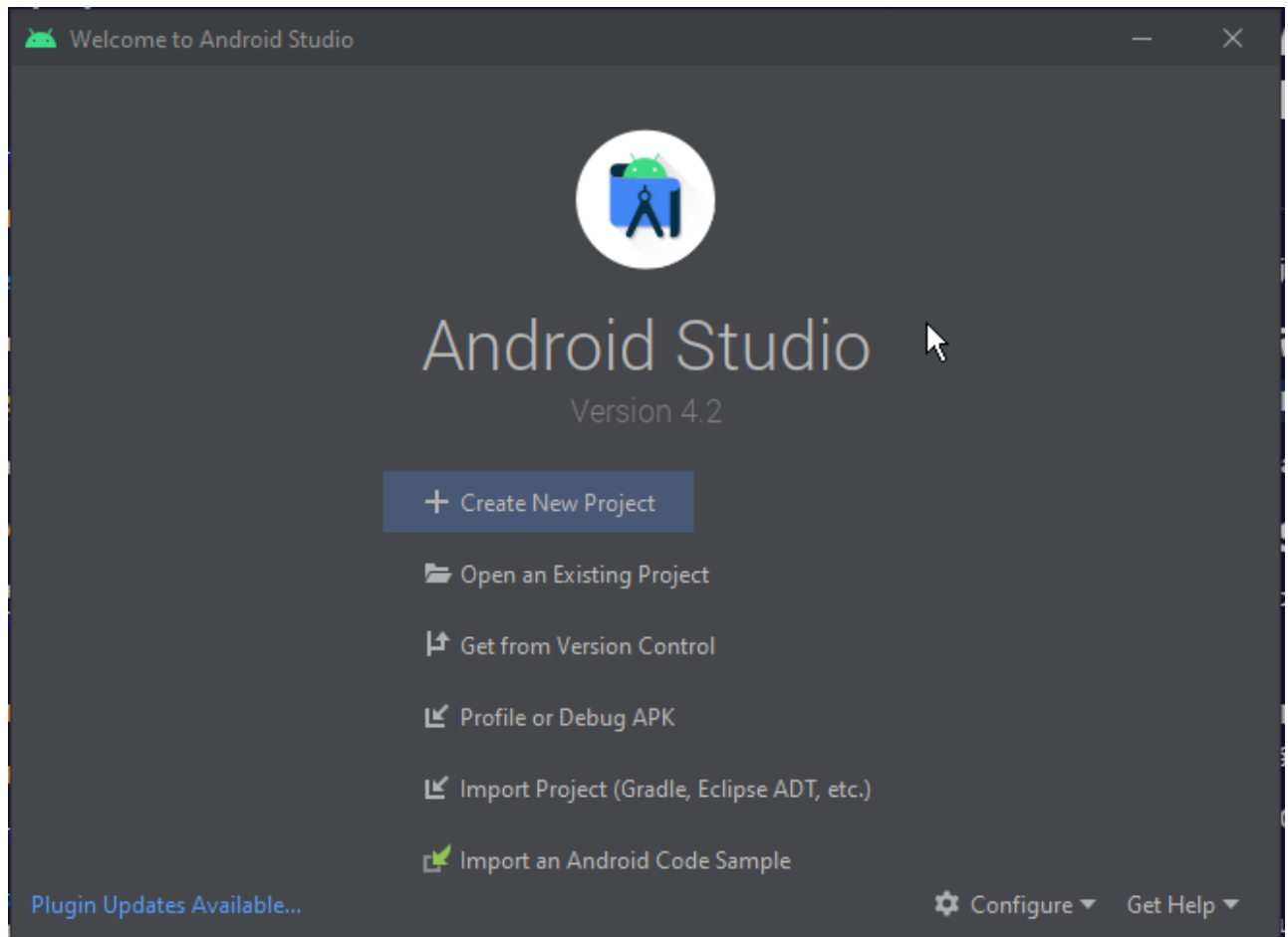
12. Ahora da click en el botón "OK".



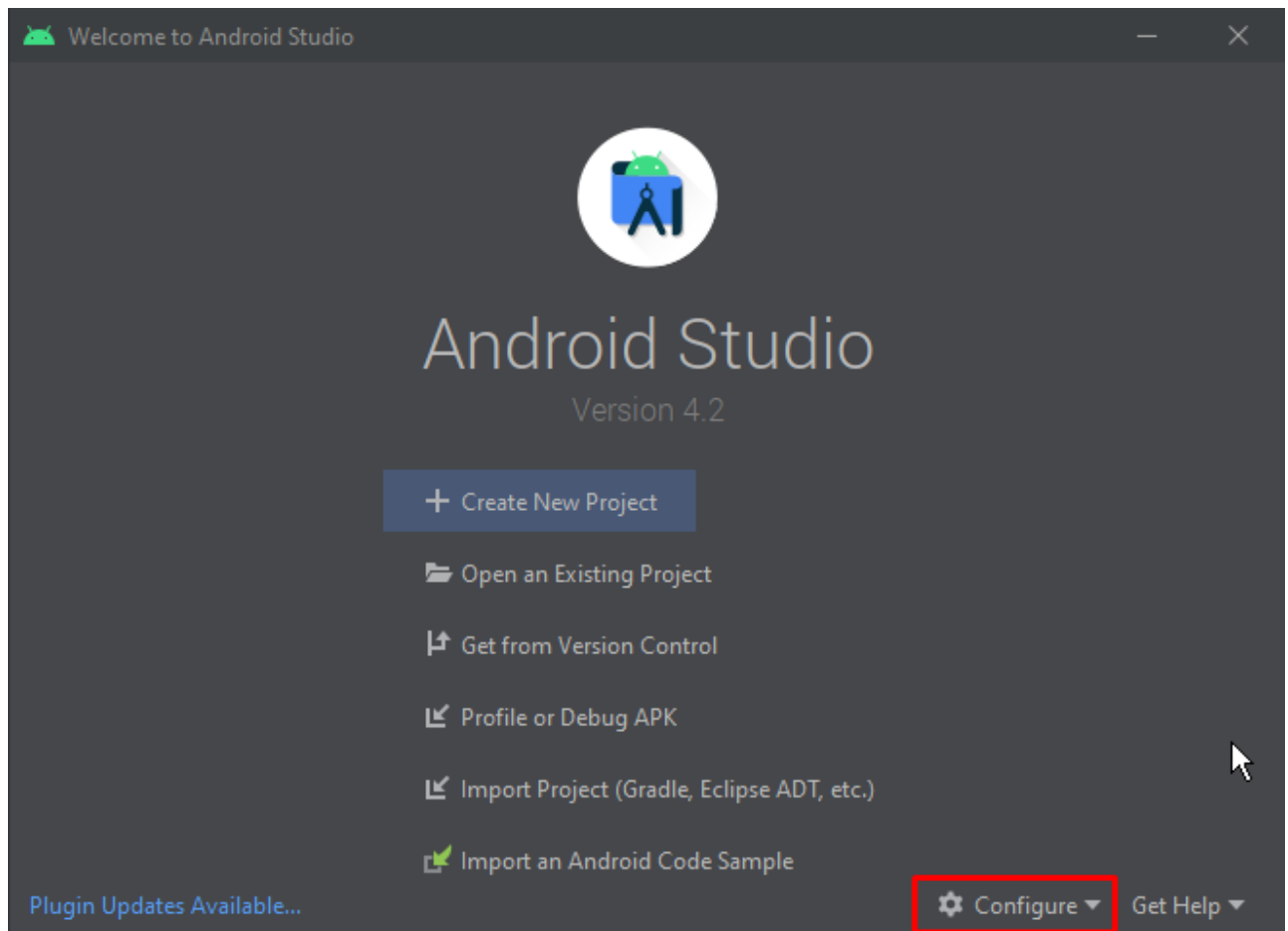
1.7.1.1.2. INSTALACIÓN DE PLUGINS

Para trabajar con Flutter de forma cómoda, hay que instalar una serie de Plugins que nos facilitarán el desarrollo de los programas.

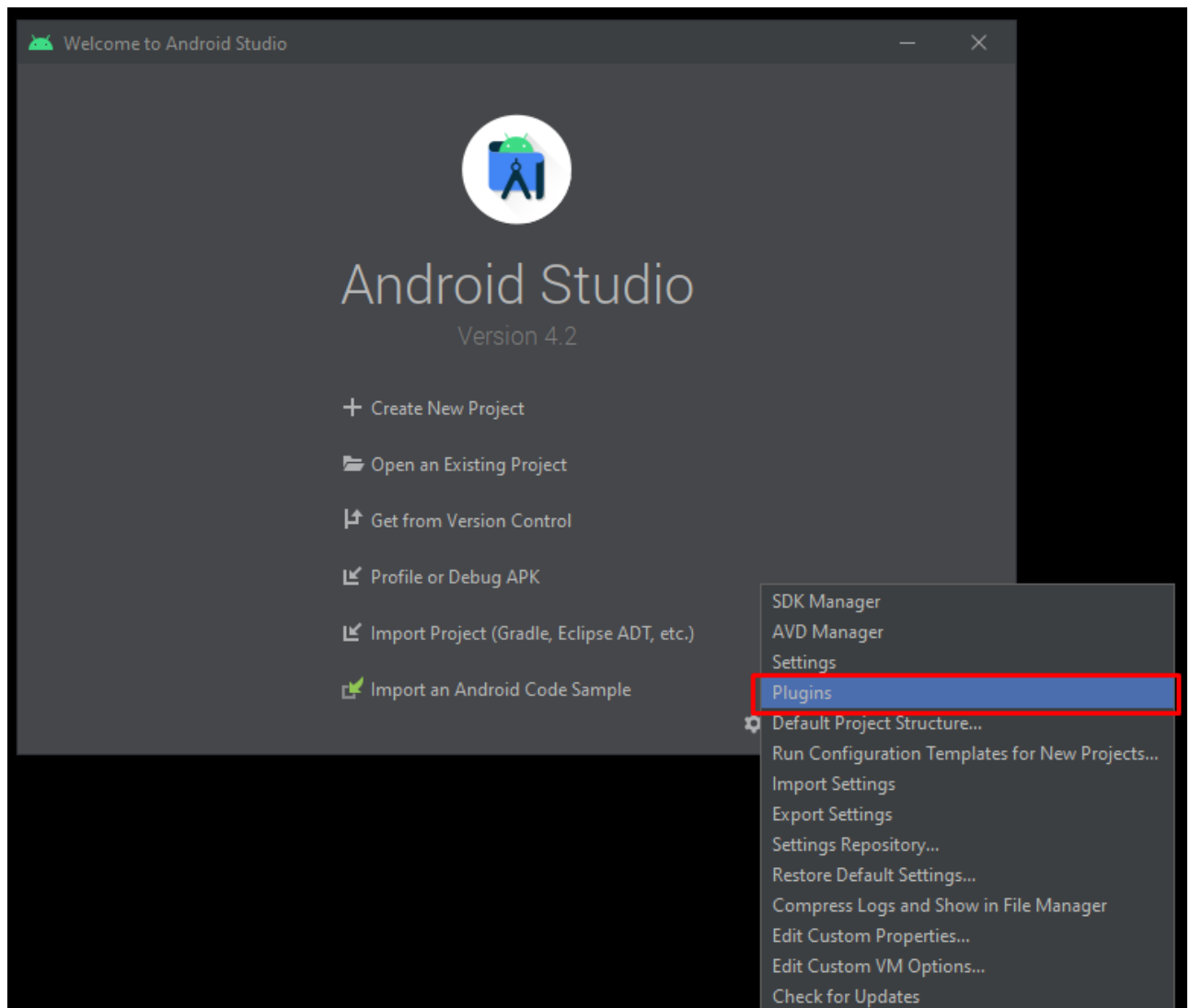
1. Abre Android Studio.



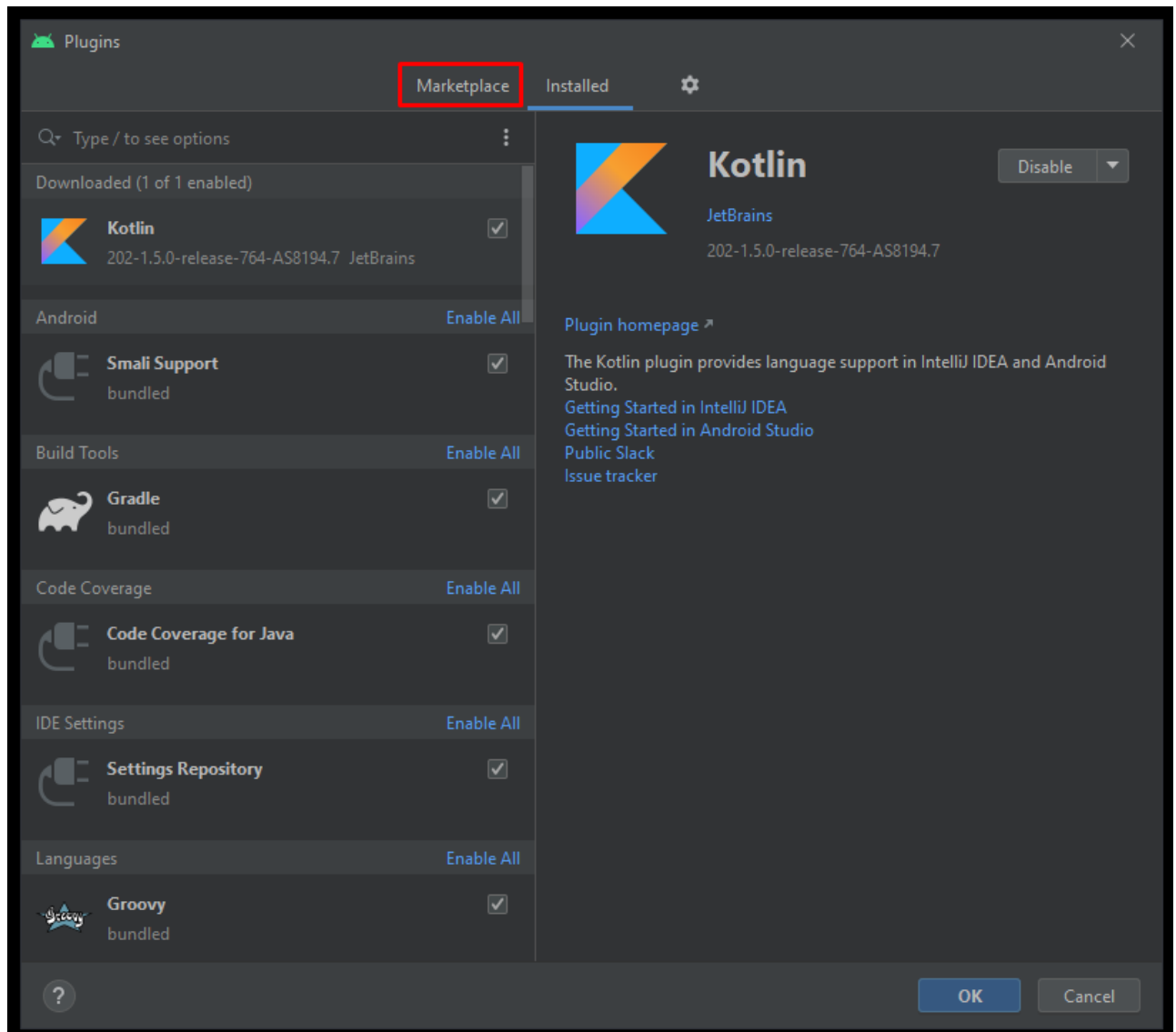
2. Da click en el botón  **Configure**.



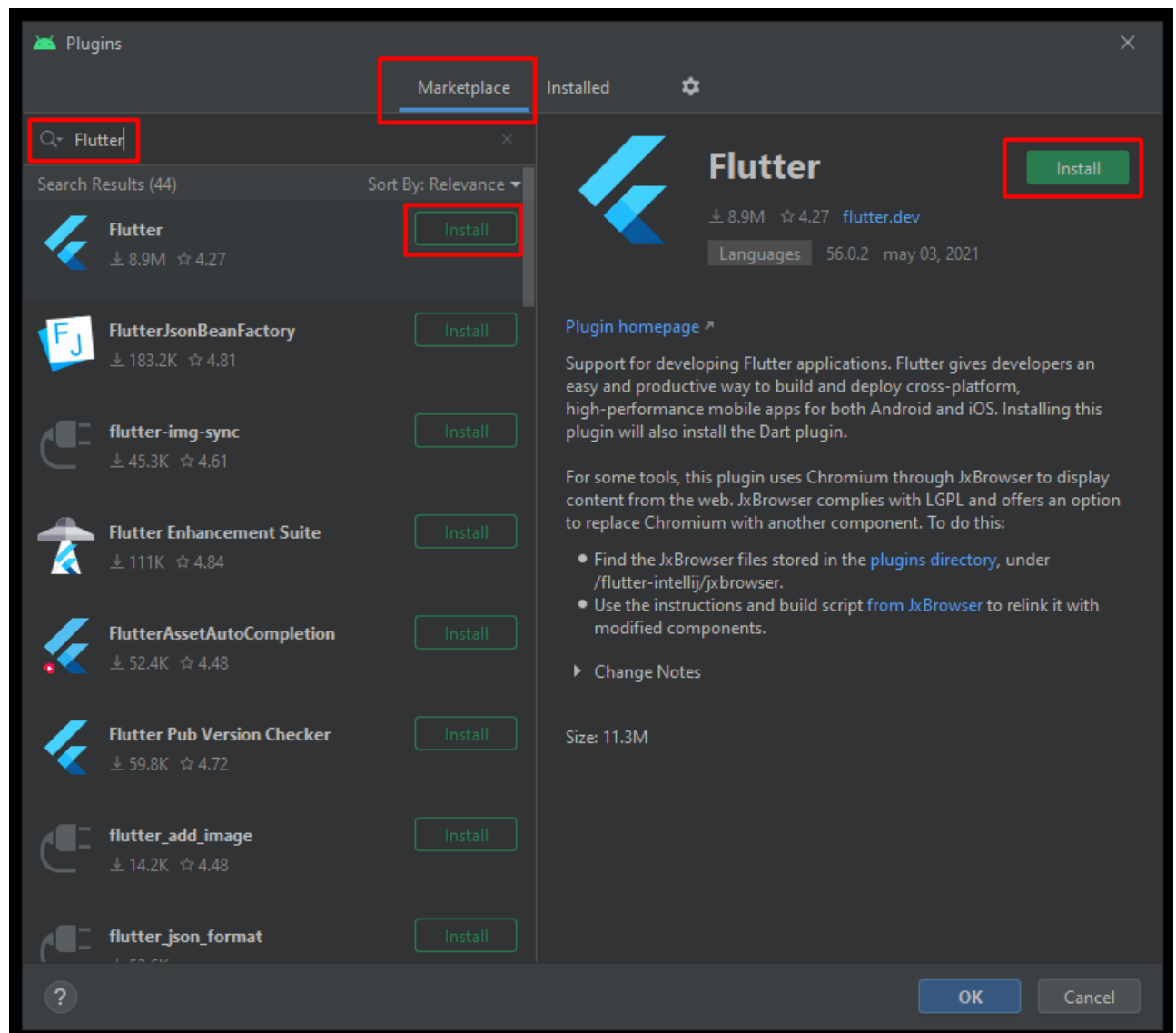
3. Da click en el botón "*Plugins*".



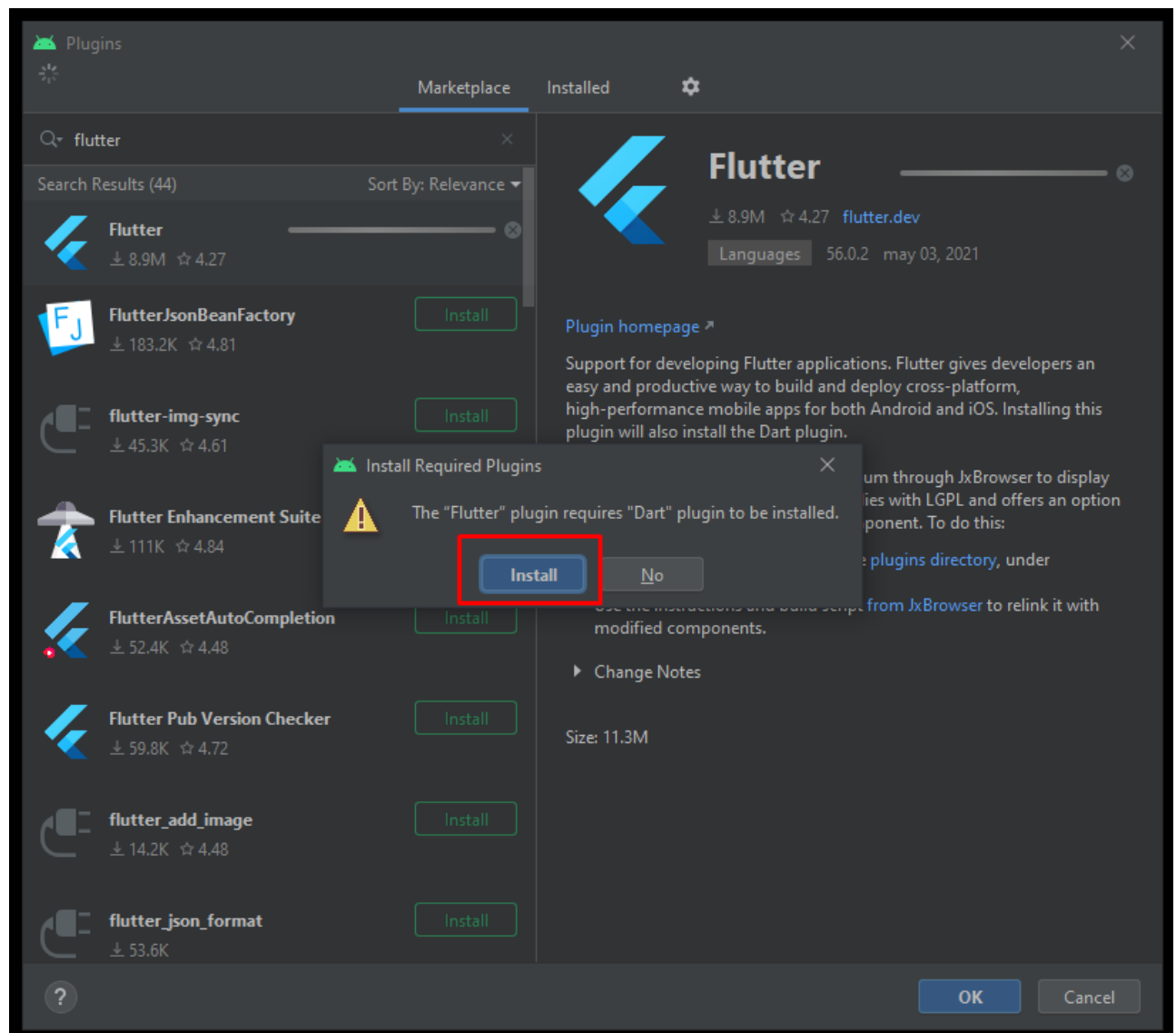
4. Te aparecerá una nueva ventana. Selecciona el apartado de "*Marketplace*".



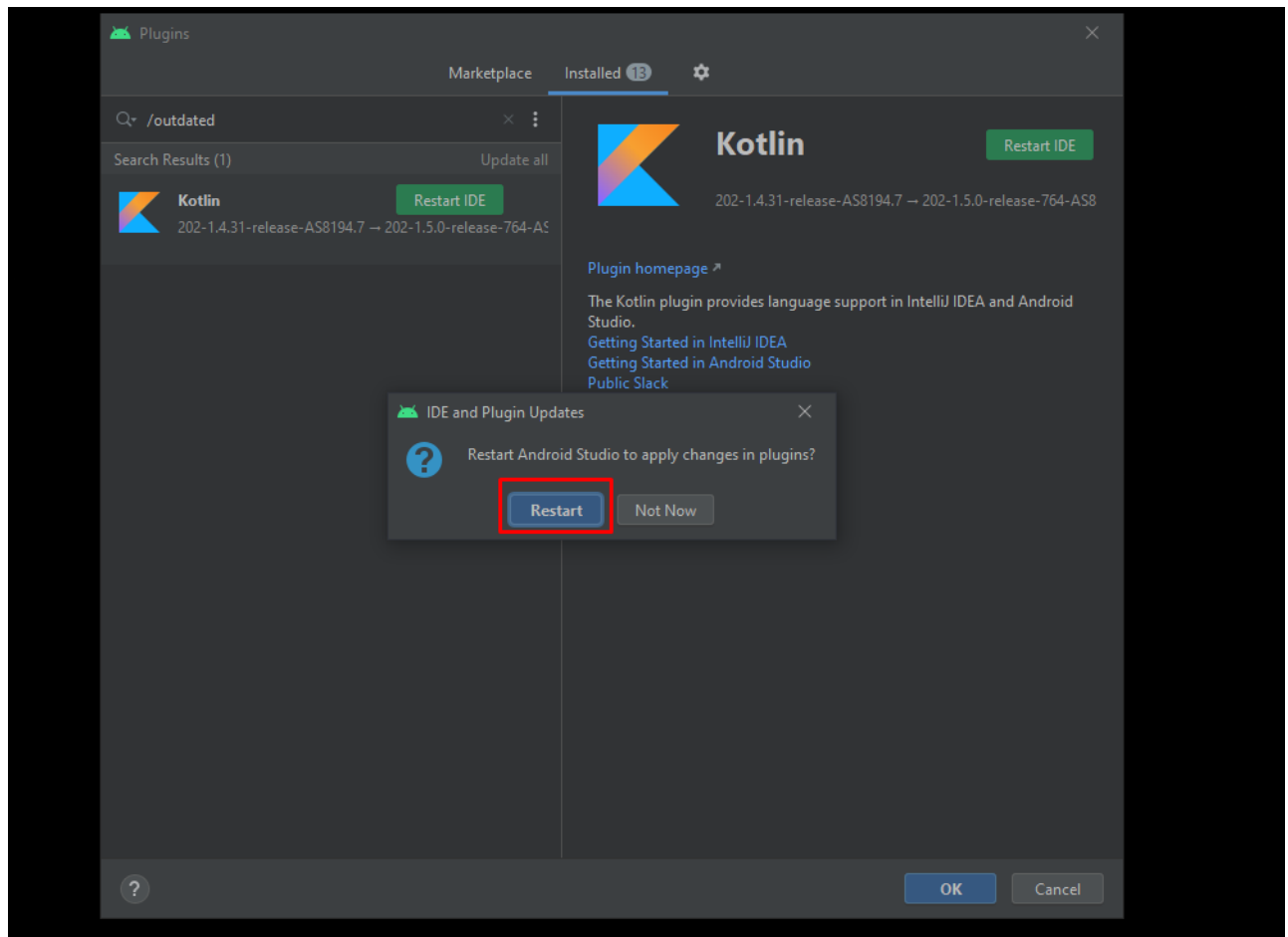
5. Busca "**Flutter**", seleccionalo y da click en "**Install**".



6. Te indicará que también tienes que instalar el plugin de "**Dart**". Da click en "**Install**", ya que **Flutter se escribe en Dart**, por lo que es necesario.



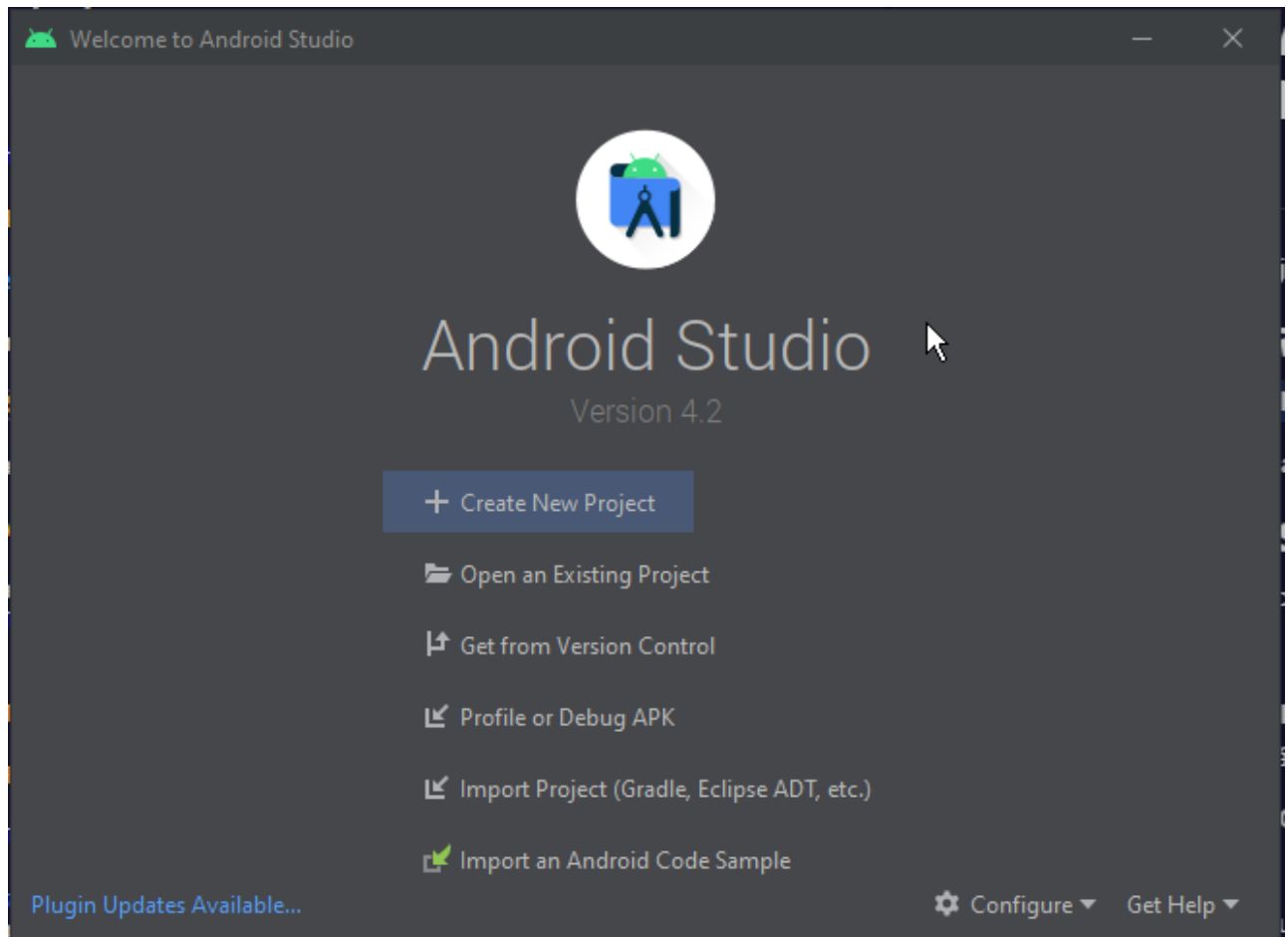
7. Si es que te lo pide, da click en "**Restart IDE**" para que se apliquen los cambios.



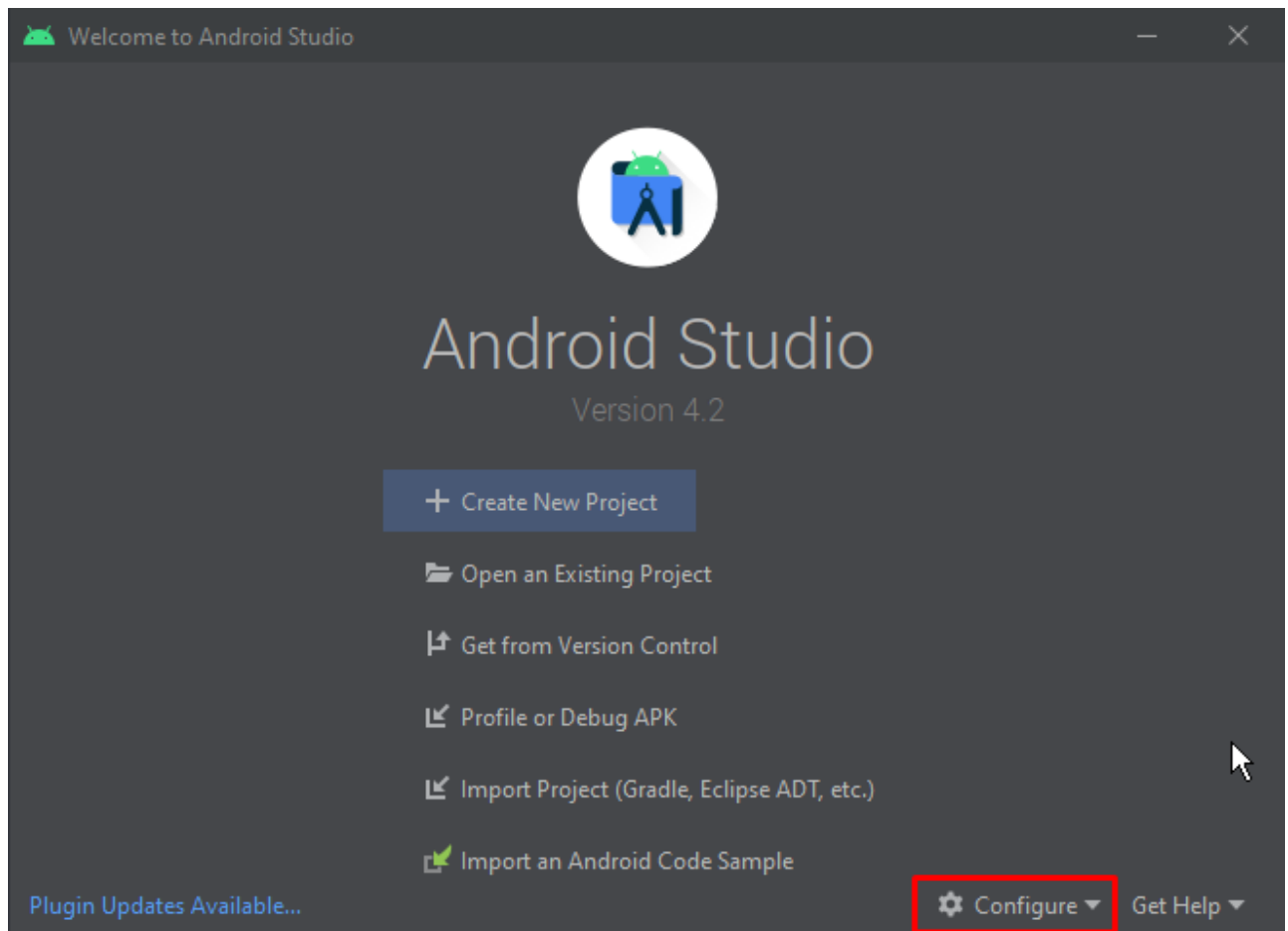
1.7.1.1.3. COMPROBACIÓN DE ACTUALIZACIONES

Hay que comprobar si hay actualizaciones disponibles.

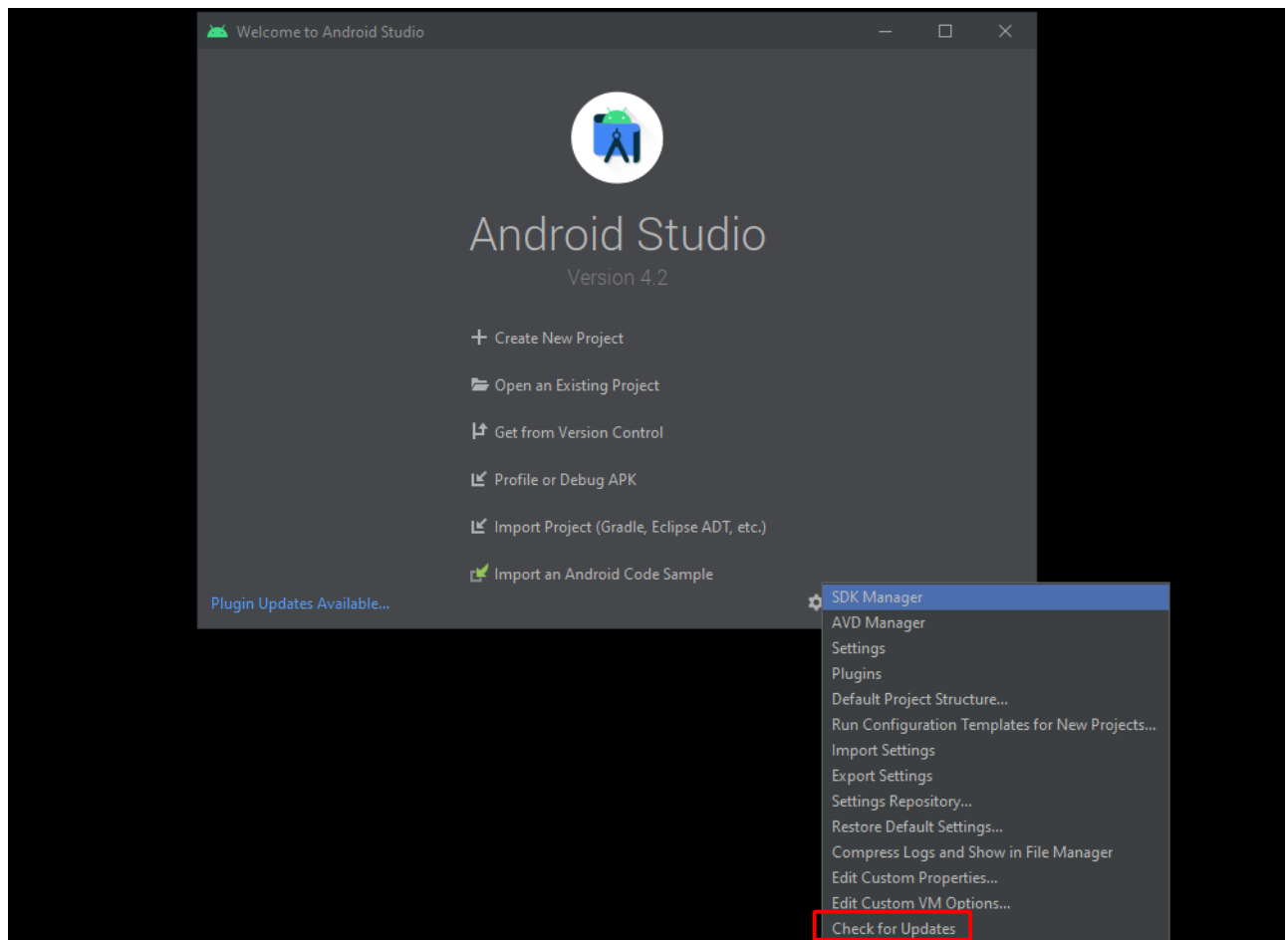
1. Abre Android Studio.



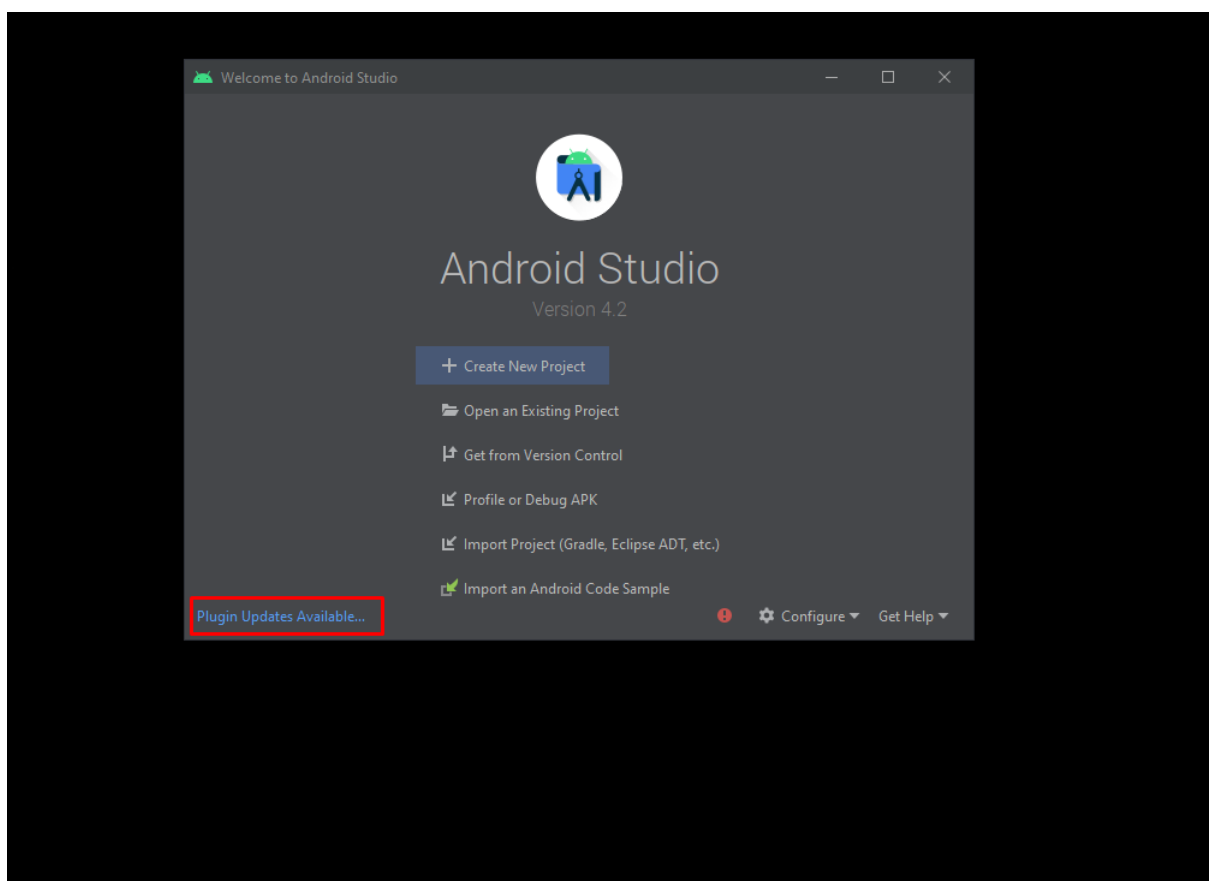
2. Da click en el botón  **Configure**.



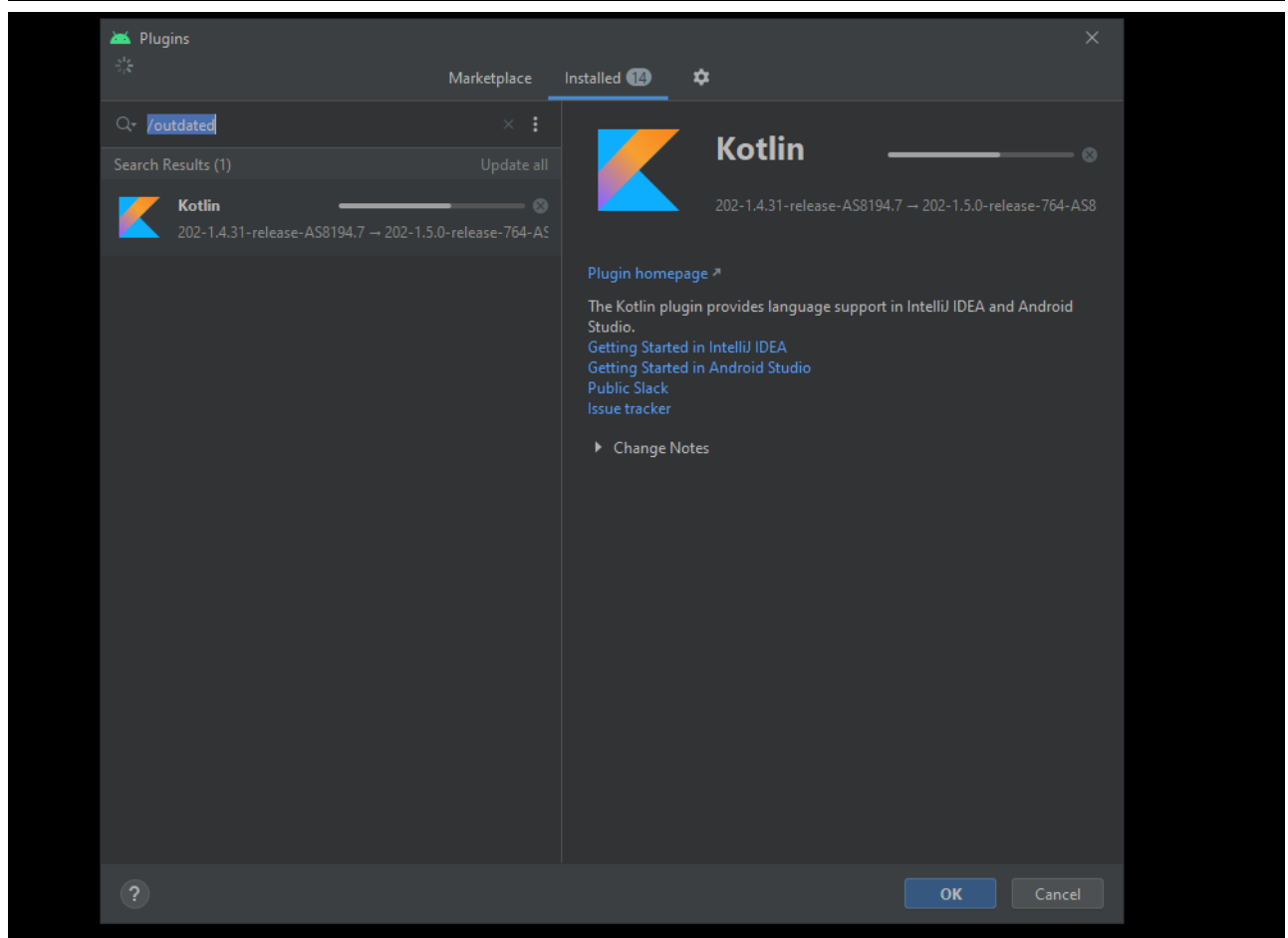
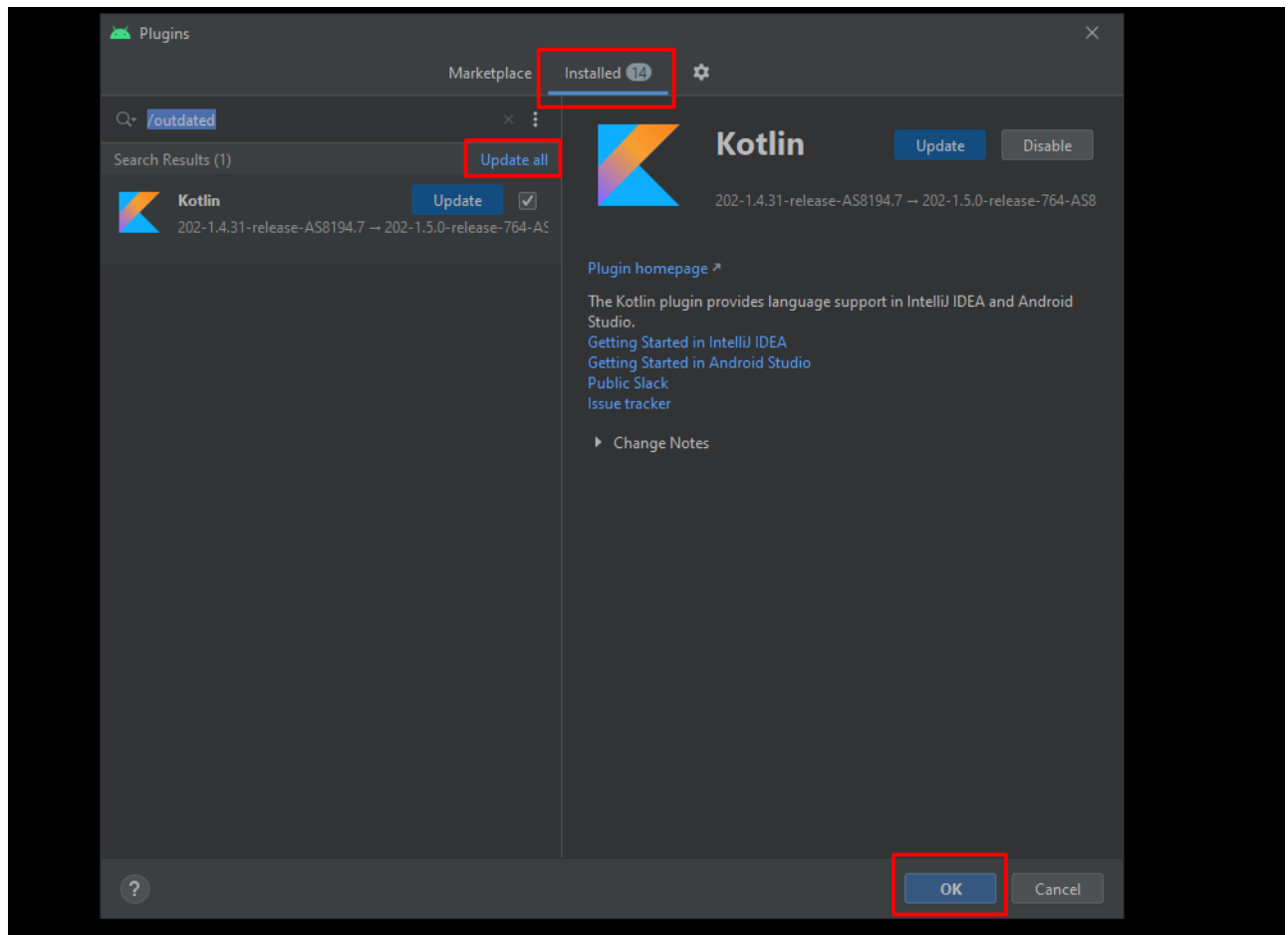
3. Da click en el botón "*Check for Updates*".



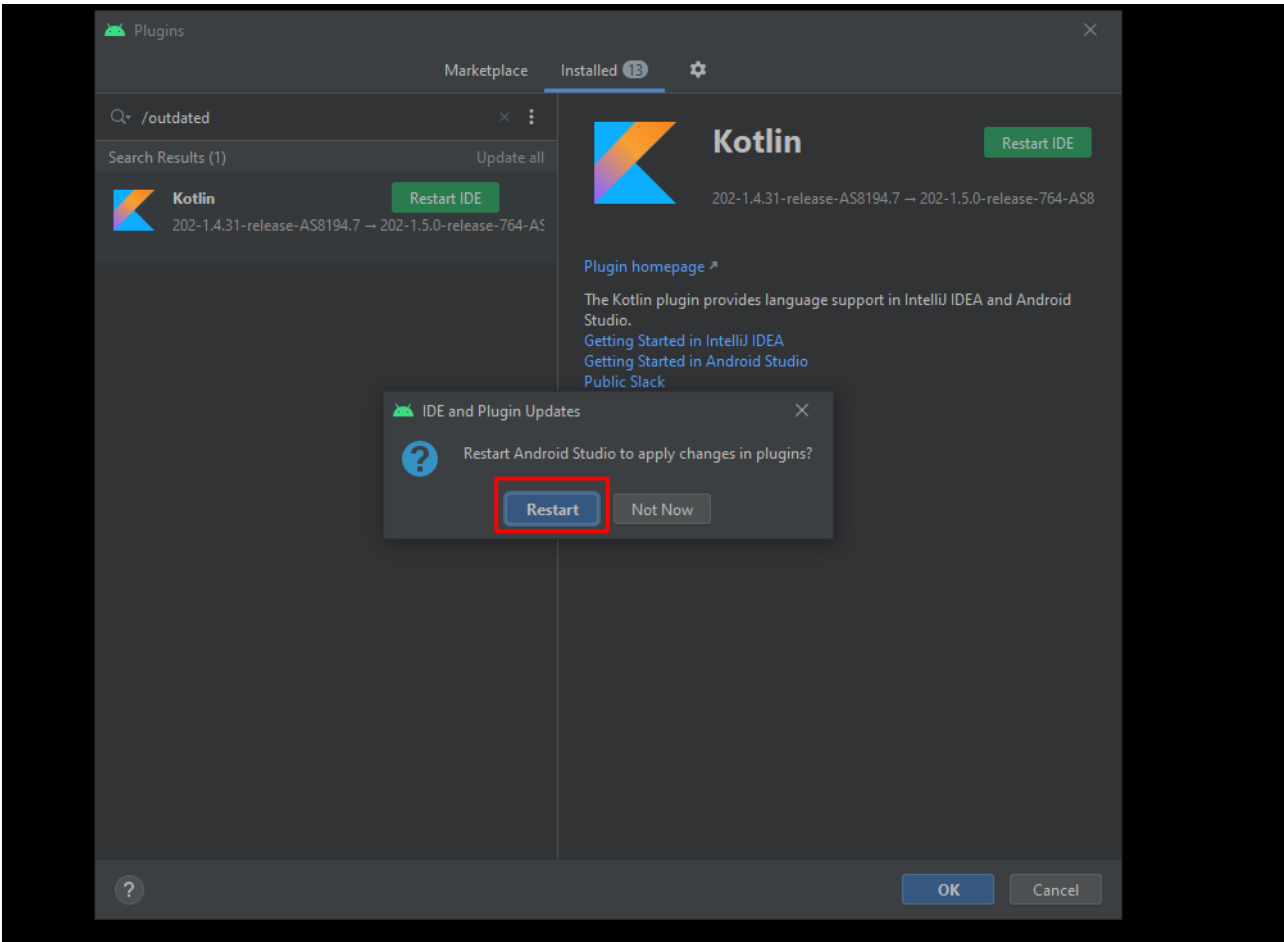
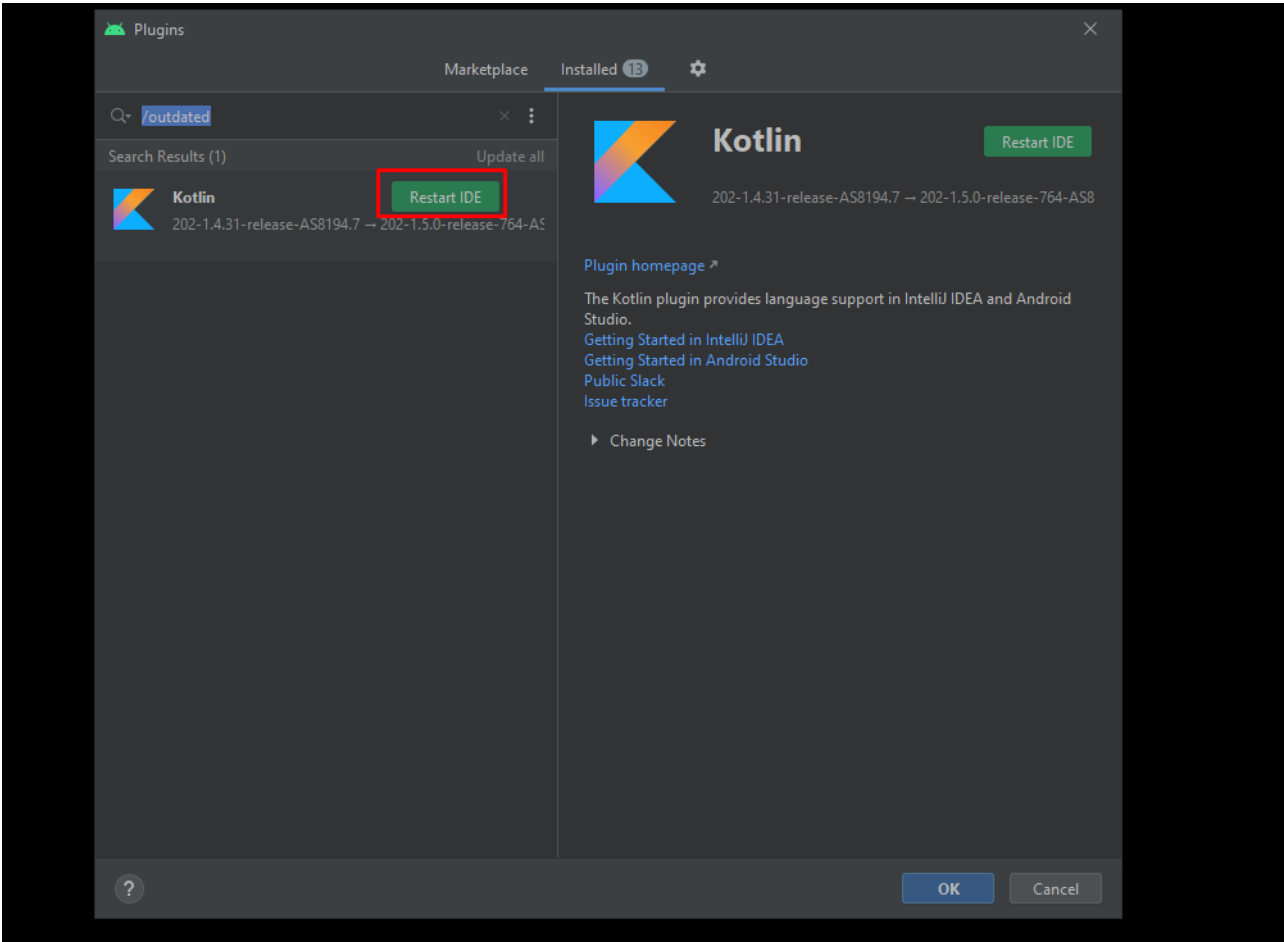
- Si hay actualizaciones te lo indicará como en la siguiente captura. Da click a ese botón/hipervínculo.



4. Te aparecerá una nueva ventana. Da click a "*Update*" de forma individual si solo quieres actualizar elementos en específico, o da click en "Update all" si quieres que todos los paquetes se actualicen automáticamente.



5. Si es que te lo pide, da click en "**Restart IDE**" para que se apliquen los cambios.

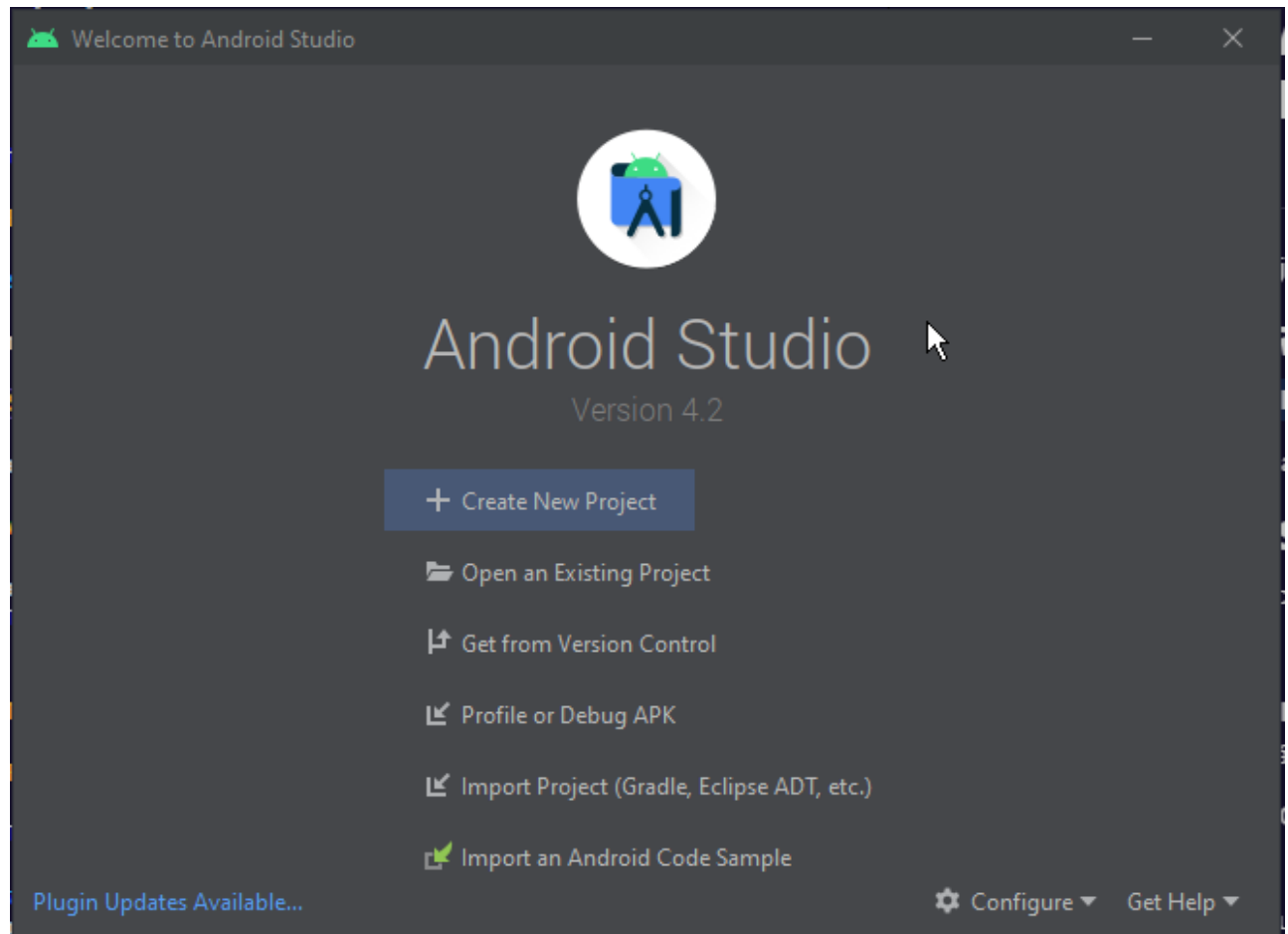


6. Listo.

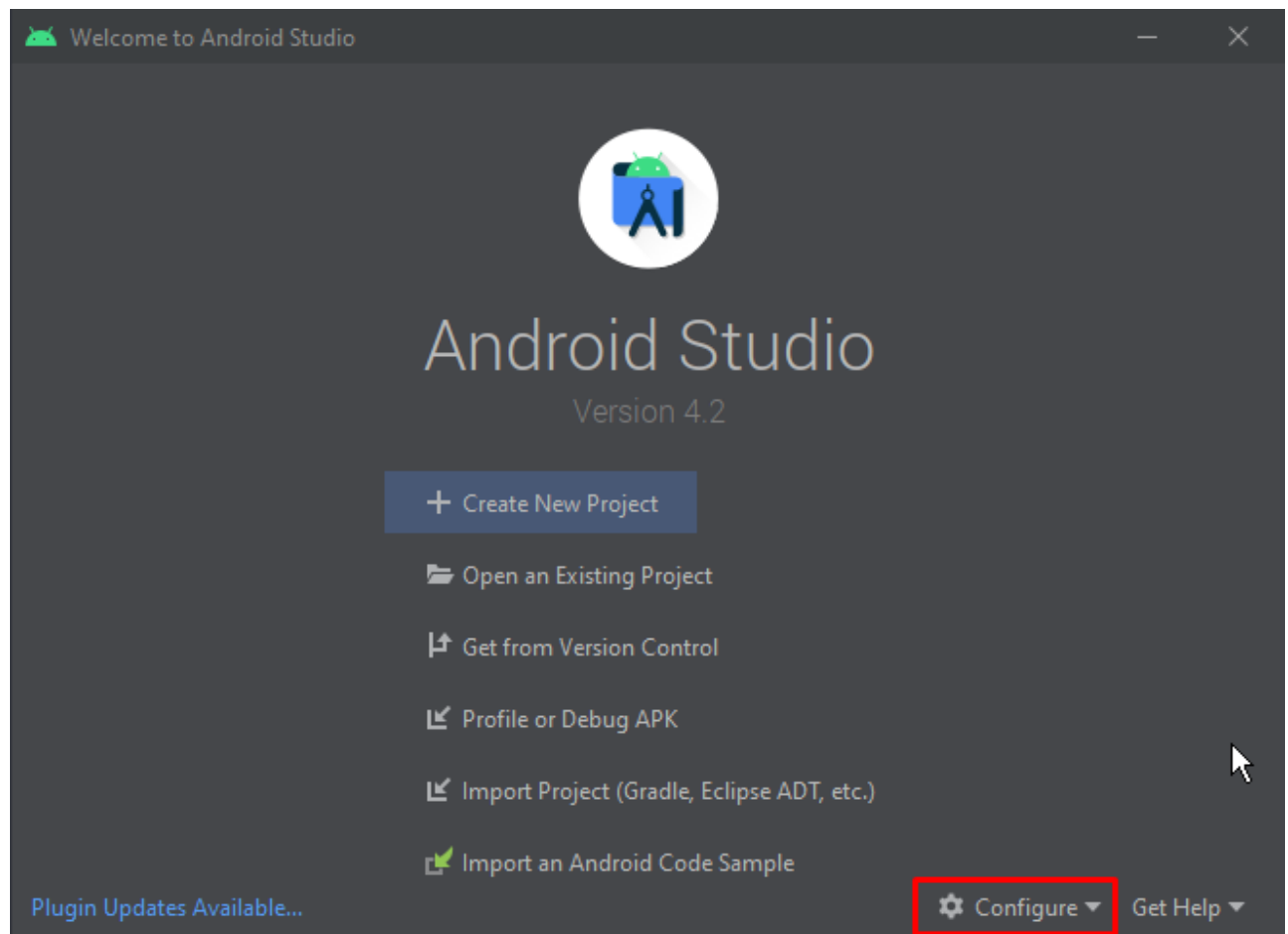
1.7.1.2. CREAR UN EMULADOR DE ANDROID (ANDROID VIRTUAL DEVICE)

Para correr un programa podemos utilizar nuestro celular conectándolo a la computadora mediante cable USB, o mediante un emulador de Android creado con Android Studio.

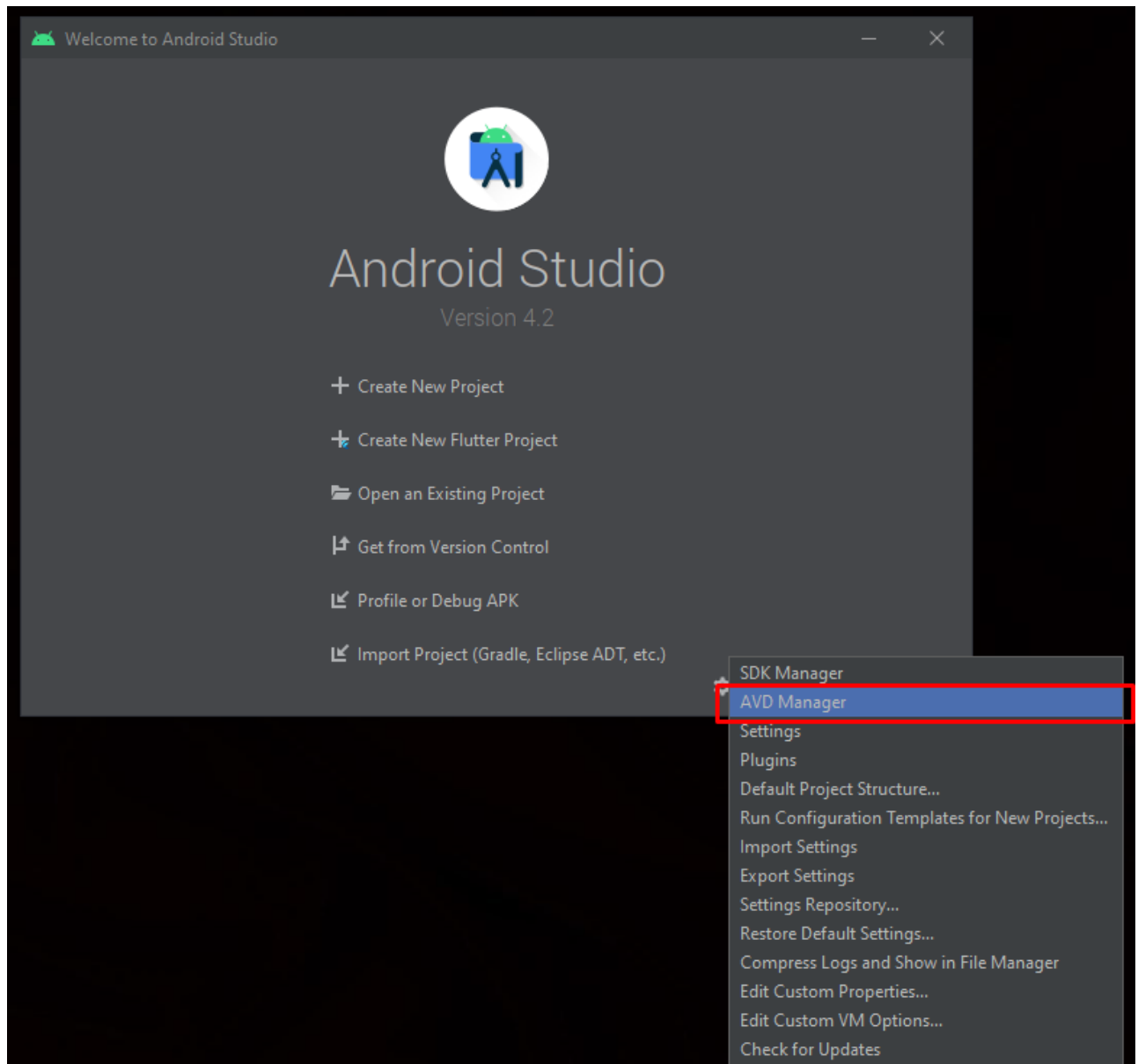
1. Abre Android Studio.



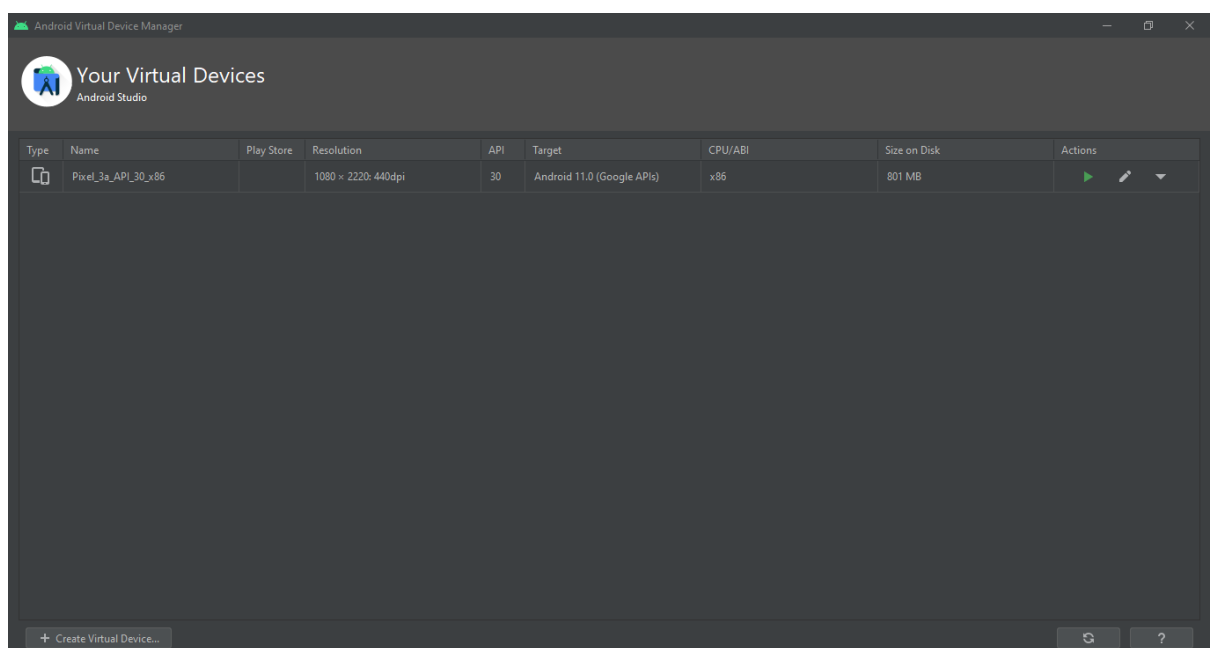
2. Da click en el botón  **Configure**.



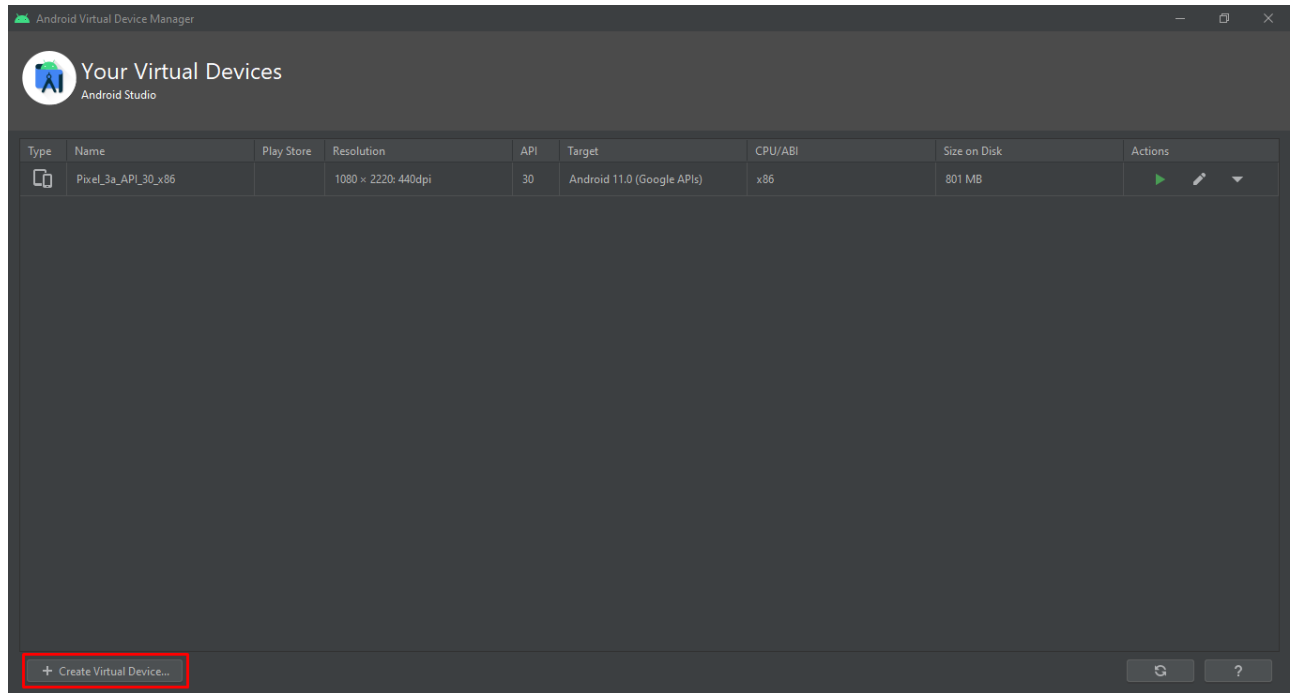
3. Da click en la opción de **AVD Manager**.



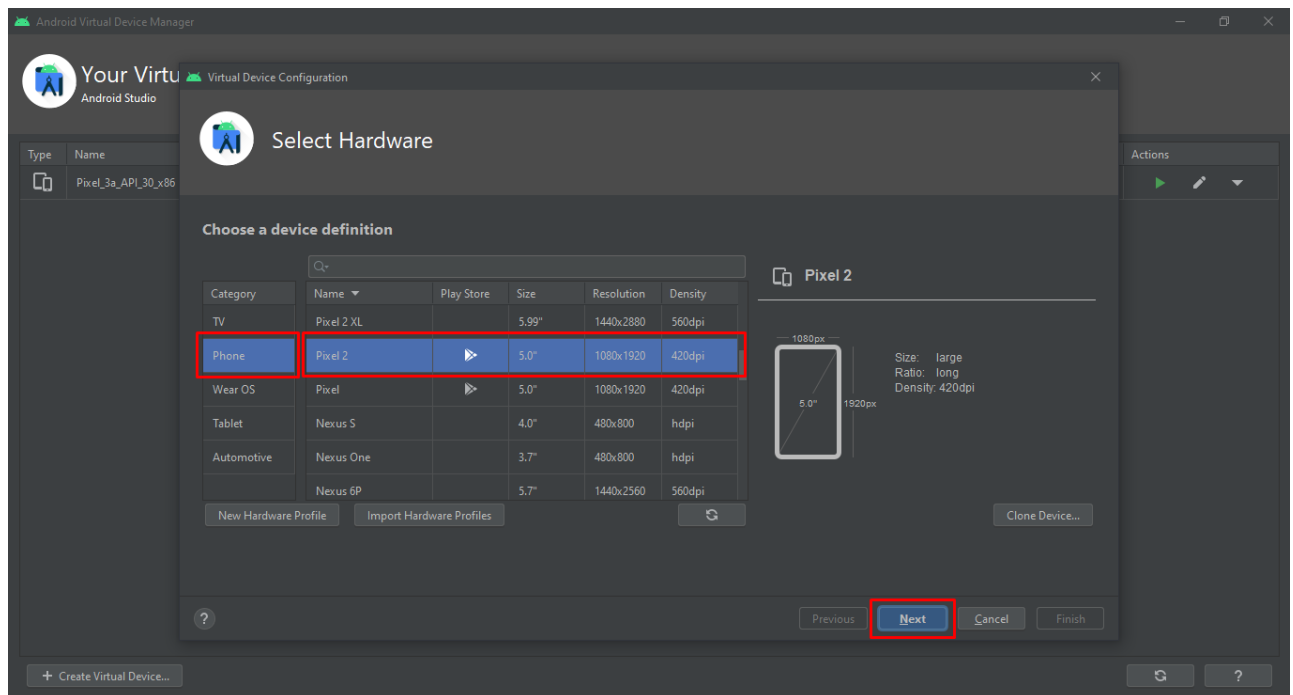
- Se abrirá una nueva ventana "**Your Virtual Devices**", en donde se encontrará una lista de dispositivos virtuales (emuladores de android).



4. Da click en **Create Virtual Device**.

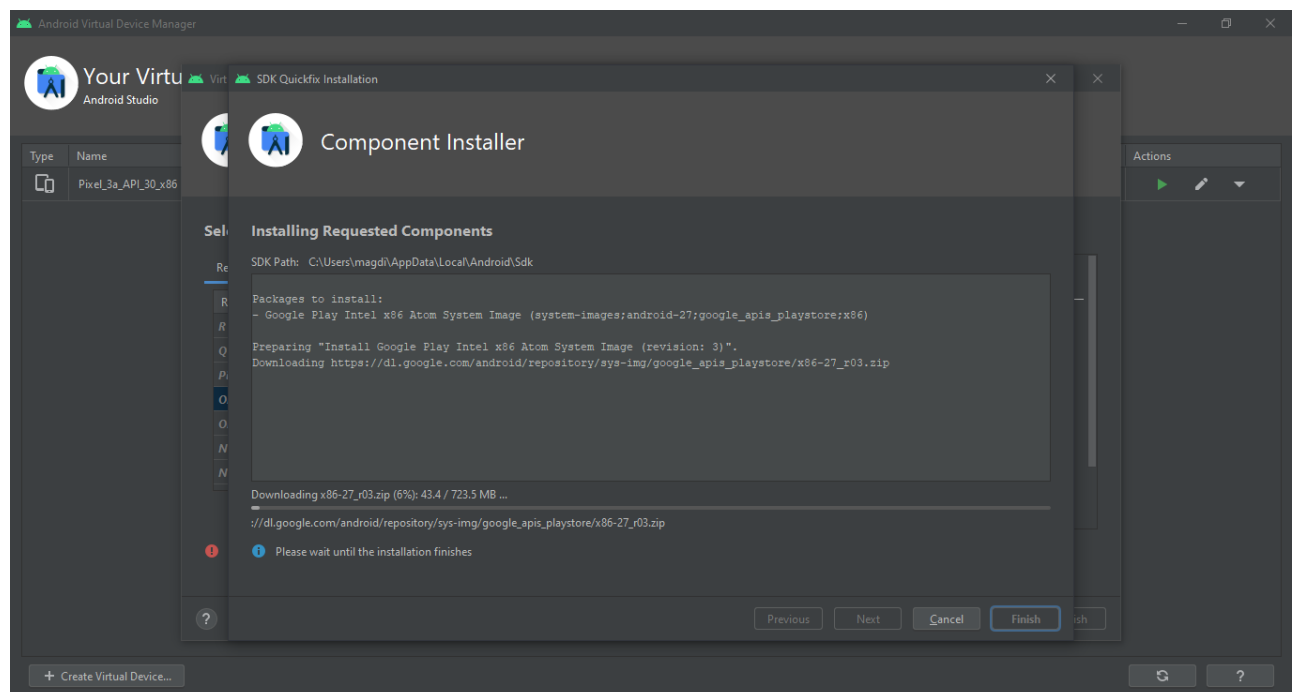
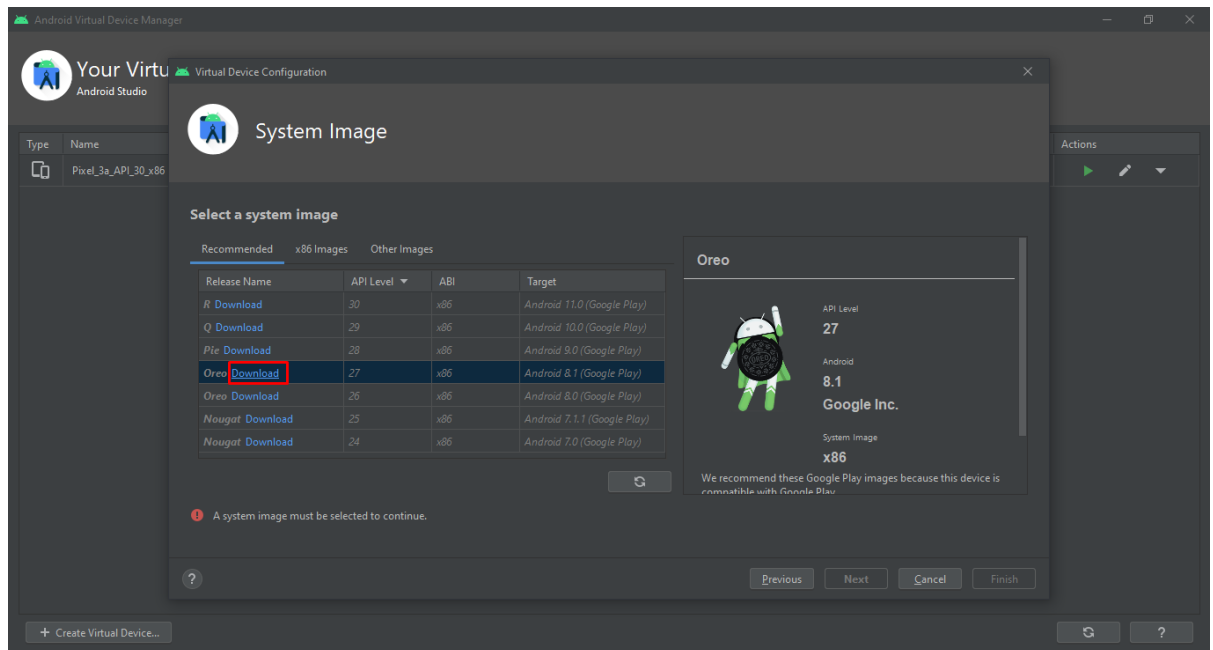


5. Aparecerá una nueva ventana: **Virtual Device Configuration**. Elige el dispositivo que quieras emular. En mi caso elegí el **Pixel 2**, que es el que recomendó el profesor. Después das click en el botón **Next**.

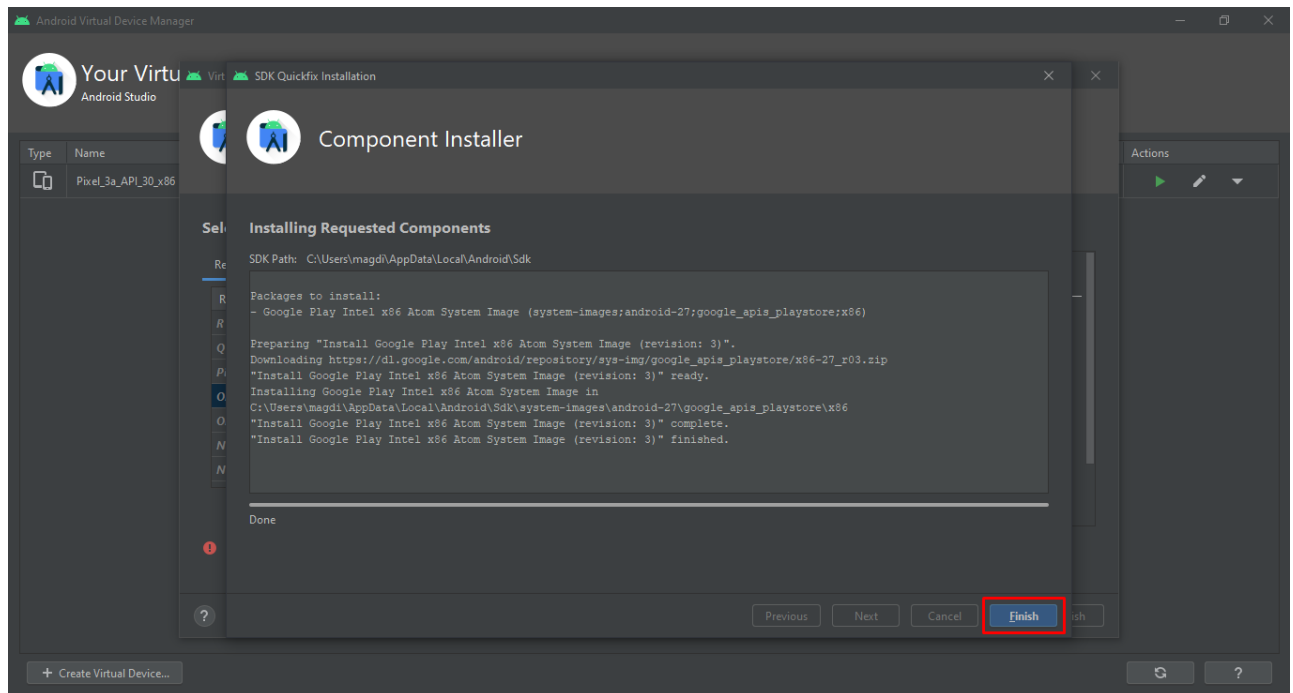


6. Selecciona el Sistema Operativo que quieres que emule el dispositivo virtual. En mi caso elegí **Android 8.1 (Oreo) | API Level: 27**.

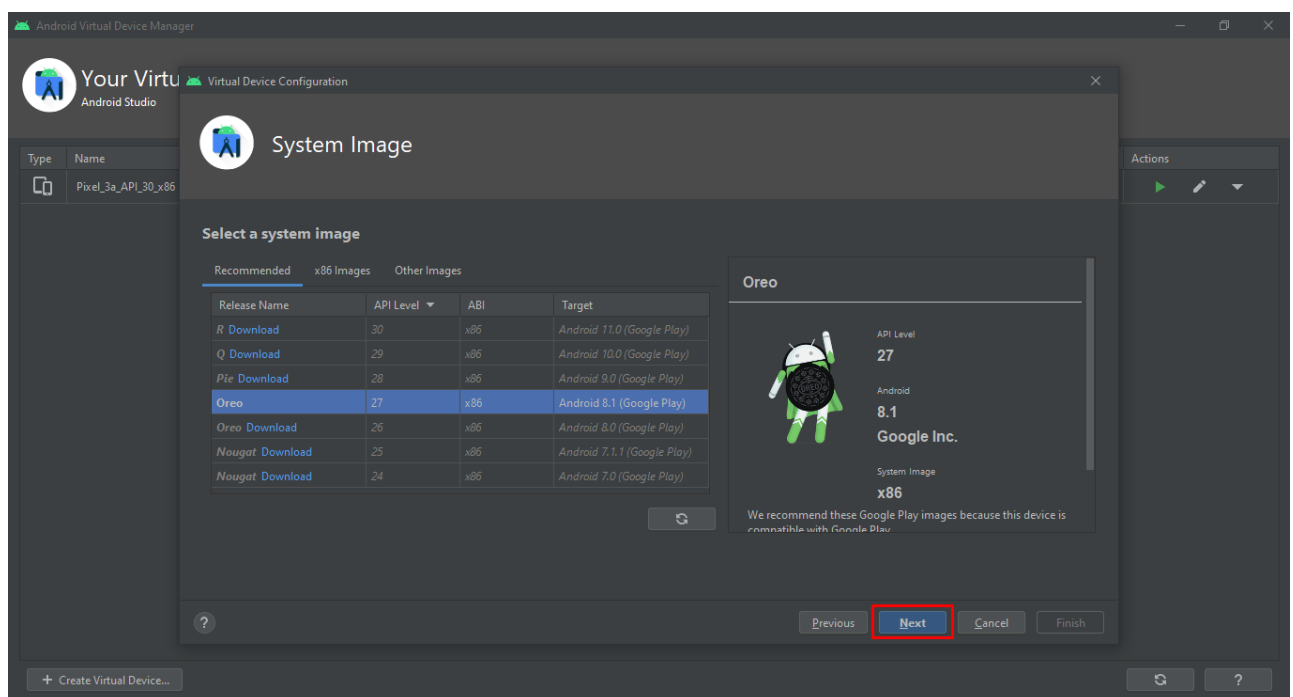
- Si no lo tienes descargado, lo tendrás que descargar dando click al **Download** que está a la derecha del nombre del Sistema Operativo.



7. Una vez terminada la instalación del API que elegiste, da click en **Finish**.

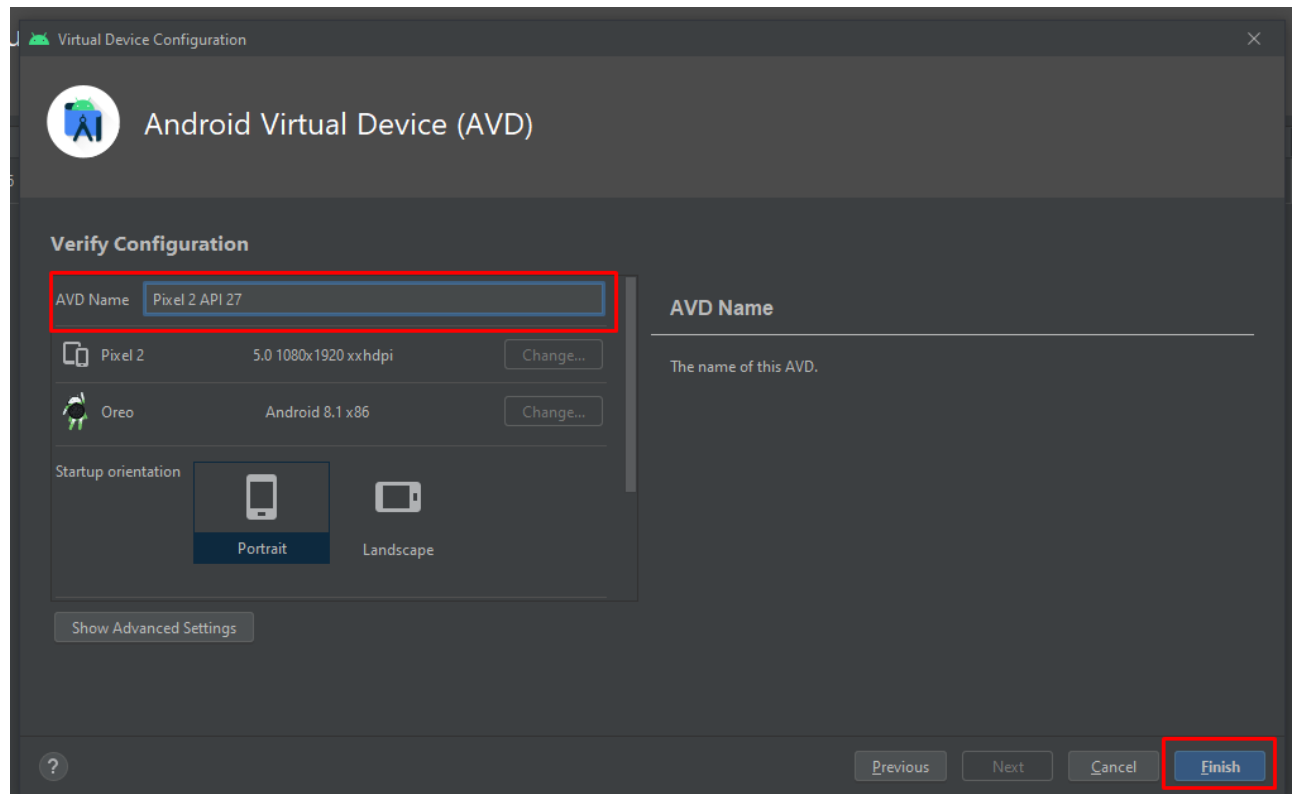


8. Da click en **Next** después de elegir la API a utilizar.

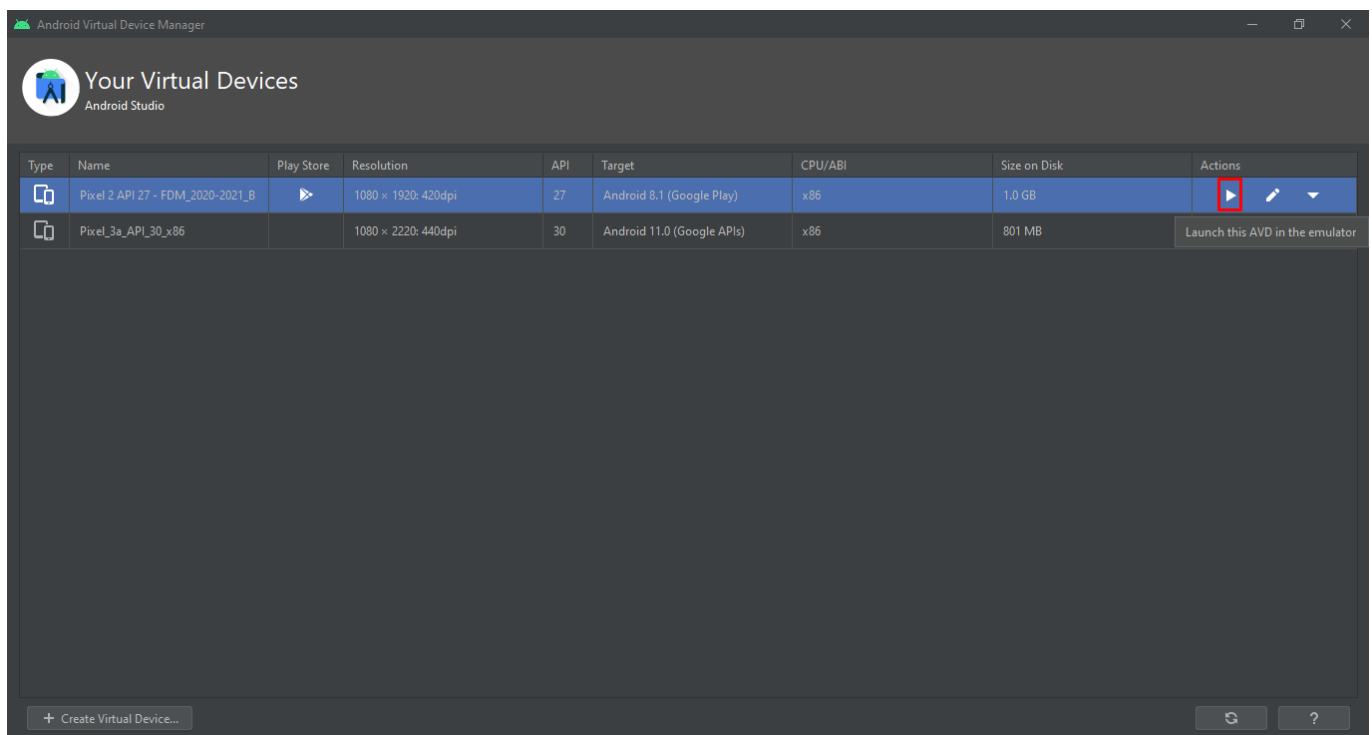


9. Verifica tu configuración y cuando todo esté como lo deseas, da click en **Finish**.

- En esta sección puedes cambiar el nombre del dispositivo y algunas configuraciones avanzadas.

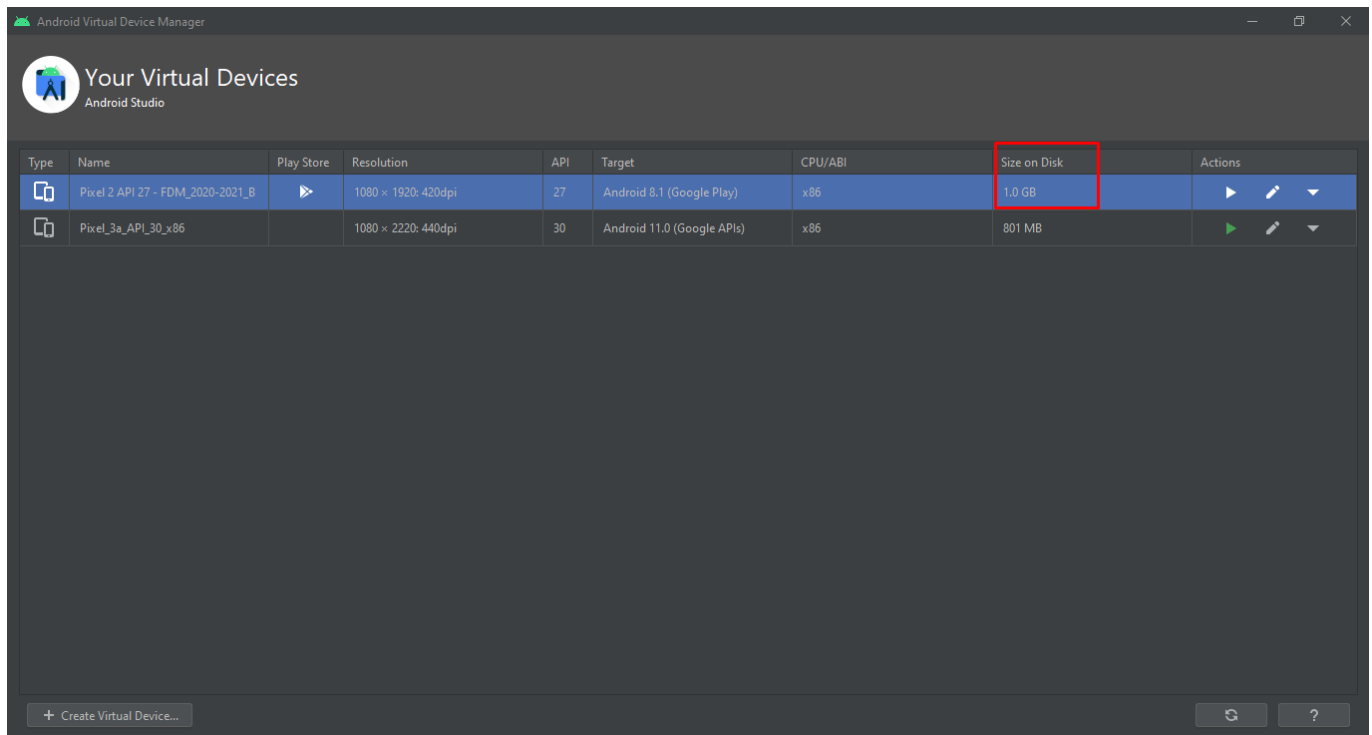


10. Ya está todo listo. Ahora puedes iniciar tu dispositivo dando click en el botón verde bajo la sección "Actions".

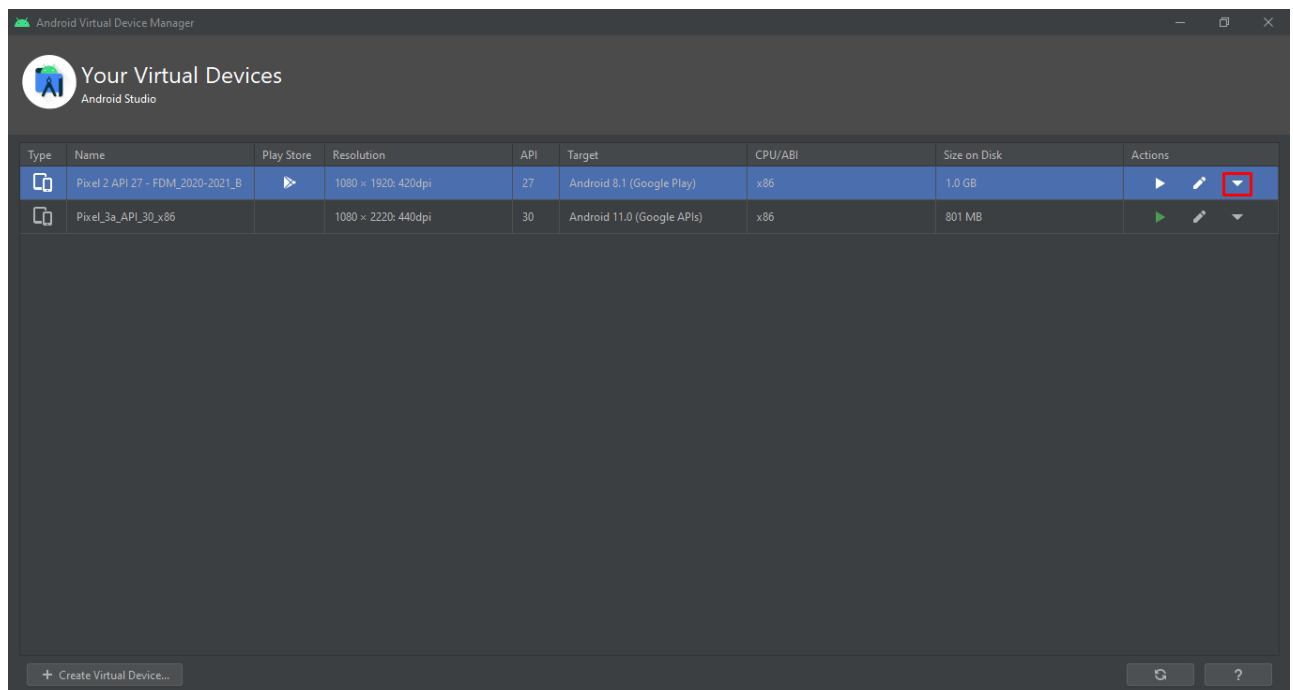


1.7.1.2.1. RECOMENDACIÓN IMPORTANTE PARA EL AVD

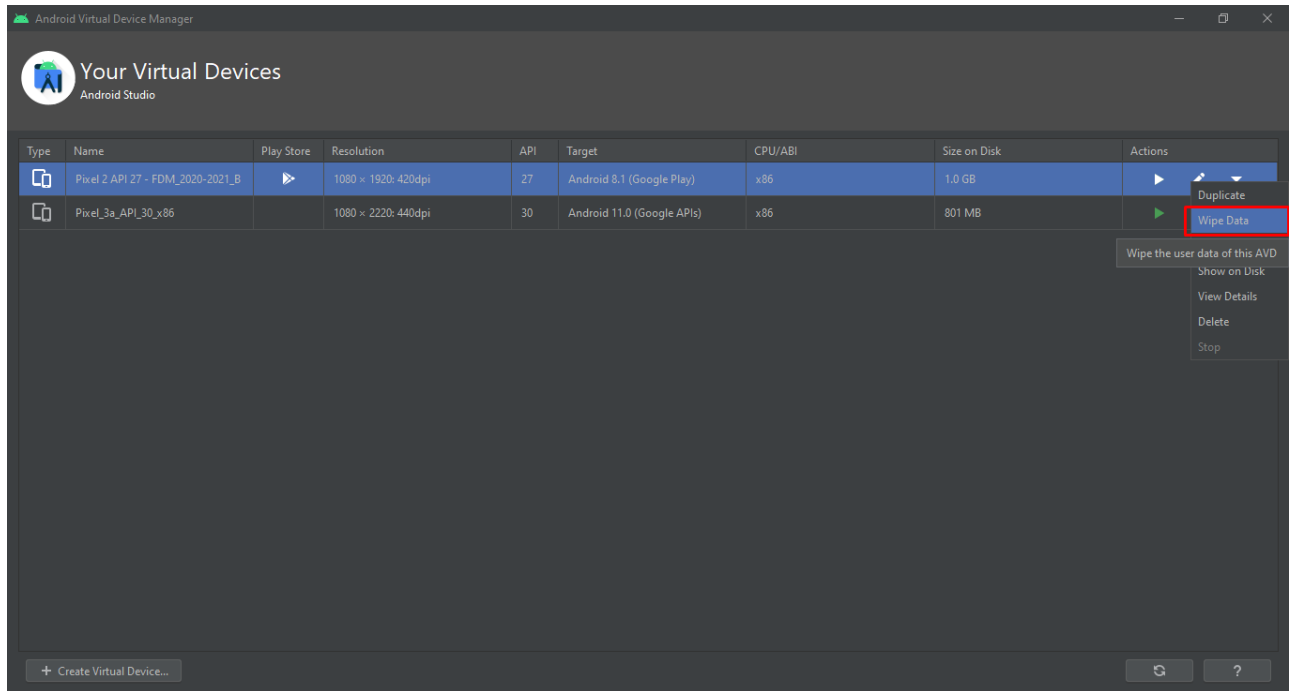
Cada vez, el AVD va a ir ocupando una mayor cantidad de espacio en el disco duro, el cual se puede ver en la sección de "Size on Disk", así que viene muy bien la liberación de ese espacio.



1. En la sección "**Actions**" da click al botón que tiene la flecha para abajo.



2. Da click a "**Wipe Data**".



3. Listo. Ahora has liberado una buen cantidad de espacio en el disco duro. Puedes hacer esto en cualquier momento (mientras no utilices el dispositivo).

1.7.2. Flutter

Entra al enlace de la documentación oficial de [Flutter](#) y en la sección de "**Get the Flutter SDK**" da click en el botón de descarga, el cual dice *flutter_windows_2.0.6-stable.zip*.

1.7.2.1. EN QUÉ DIRECTORIO COLOCAR FLUTTER

Para no tener problemas con espacios y demás, coloqué Flutter directamente en el directorio principal del disco duro (**C:/**), creando una carpeta dentro y extrayéndolo. Entonces quedaría de la siguiente forma:

```
C:\SDKs\flutter
```

Así, cada vez que quiera encontrar Flutter, será más sencillo.

1.7.2.2. FLUTTER **SOUND NULL SAFETY**

A día de hoy, Viernes, 21 de mayo del 2021, aún no lo implementamos, pero dejamos las instrucciones para un futuro.

Habilitamos la opción **Sound Null Safety** de Flutter para evitar problemas con valores null. Esto lo podemos encontrar en la documentación:

[Dart | Samples & tutorials/Null Safety](#)

Como se indica en el sitio web:

To make Dart treat your code as null safe, the SDK constraints must require a language version that has null safety support. For example, your *pubspec.yaml* file might have the following constraints:

```
environment:  
  sdk: ">=2.12.0 <3.0.0"
```

Siguiendo lo anterior, `Sound null safety` estaría activado.

1.7.2.3. VARIABLE DE USUARIO

Para ver cómo configurar la variable de usuario, dirígete a la **sección del siguiente enlace** (te enviará dentro del README):

[CONFIGURACIÓN DE VARIABLE DE USUARIO PARA FLUTTER](#)

1.7.3. Chocolatey

[Chocolatey](#) es un administrador de paquetes en Windows, con el que podemos instalar una diversidad de paquetes (valga la redundancia) de forma sencilla corriendo comandos desde una terminal.

Este es solo para instalar [scrcpy](#)

1.7.3.1. INSTRUCCIONES INSTALACIÓN DESDE POWERSHELL

De acuerdo con las instrucciones del sitio web de [Chocolatey](#), hay que seguir los siguientes pasos:

REQUIREMENTS

- Windows 7+ / Windows Server 2003+
- PowerShell v2+ (minimum is v3 for install from this website due to TLS 1.2 requirement)
- .NET Framework 4+ (the installation will attempt to install .NET 4.0 if you do not have it installed) (minimum is 4.5 for install from this website due to TLS 1.2 requirement)

1. First, ensure that you are using an [administrative shell](#) - you can also install as a non-admin, check out [Non-Administrative Installation](#).
2. Install with powershell.exe

NOTE: Please inspect <https://chocolatey.org/install.ps1> prior to running any of these scripts to ensure safety. We already know it's safe, but you should verify the security and contents of any script from the internet you are not familiar with. All of these scripts download a remote PowerShell script and execute it on your machine. We take security very seriously. [Learn more about our security protocols](#) .

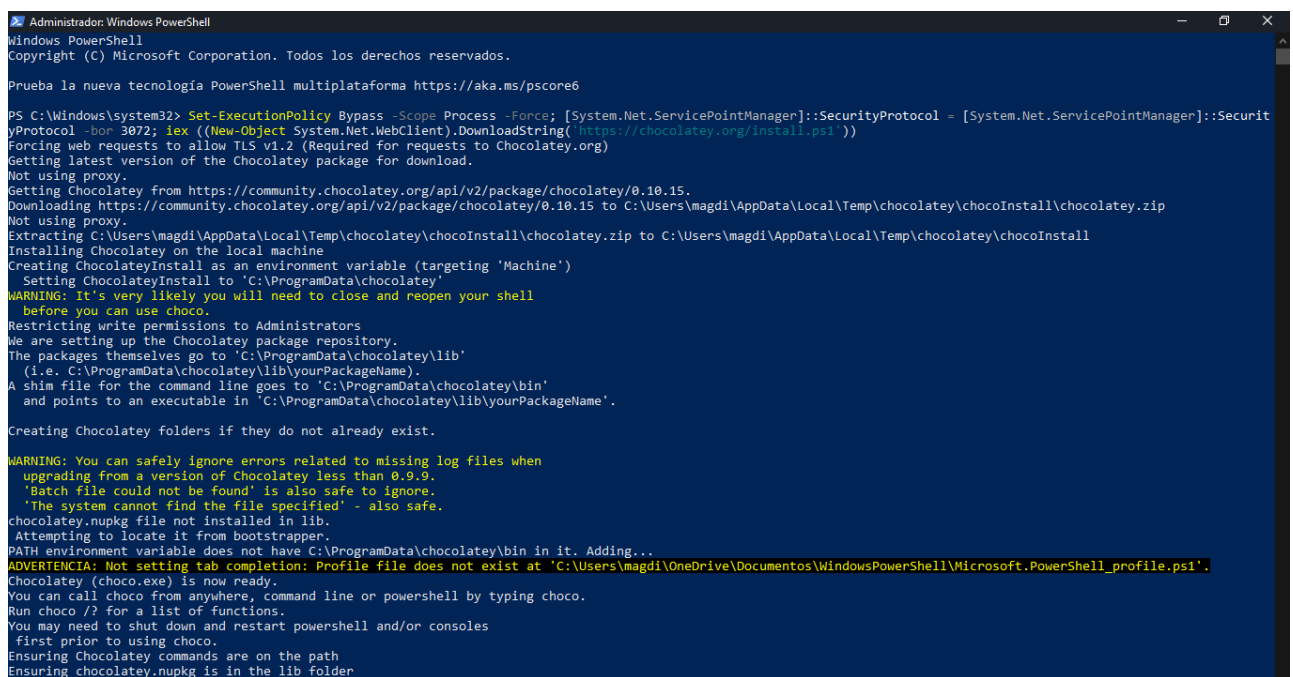
With PowerShell, you must ensure [Get-ExecutionPolicy](#) is not Restricted. We suggest using [Bypass](#) to bypass the policy to get things installed or [AllSigned](#) for quite a bit more security.

- Run `Get-ExecutionPolicy`. If it returns `Restricted`, then run `Set-ExecutionPolicy AllSigned` or `Set-ExecutionPolicy Bypass -Scope Process`.

Now run the following command:

```
Set-ExecutionPolicy Bypass -Scope Process -Force;
[System.Net.ServicePointManager]::SecurityProtocol =
[System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex
((New-Object
System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps
1'))
```

3. Paste the copied text into your shell and press Enter.
4. Wait a few seconds for the command to complete.
5. If you don't see any errors, you are ready to use Chocolatey! Type `choco` or `choco -?` now, or see [Getting Started](#) for usage instructions.



```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> Set-ExecutionPolicy Bypass -Scope Process -Force; [System.Net.ServicePointManager]::SecurityProtocol = [System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex ((New-Object System.Net.WebClient).DownloadString('https://chocolatey.org/install.ps1'))
Forcing web requests to allow TLS v1.2 (Required for requests to Chocolatey.org)
Getting latest version of the Chocolatey package for download.
Not using proxy.
Getting Chocolatey from https://community.chocolatey.org/api/v2/package/chocolatey/0.10.15.
Downloading https://community.chocolatey.org/api/v2/package/chocolatey/0.10.15 to C:\Users\magdi\AppData\Local\Temp\chocolatey\chocoInstall\chocolatey.zip
Not using proxy.
Extracting C:\Users\magdi\AppData\Local\Temp\chocolatey\chocoInstall\chocolatey.zip to C:\Users\magdi\AppData\Local\Temp\chocolatey\chocoInstall
Installing Chocolatey on the local machine
Creating ChocolateyInstall as an environment variable (targeting 'Machine')
Setting ChocolateyInstall to 'C:\ProgramData\chocolatey'
WARNING: It's very likely you will need to close and reopen your shell
before you can use choco.
Restricting write permissions to Administrators
We are setting up the Chocolatey package repository.
The packages themselves go to 'C:\ProgramData\chocolatey\lib'
(i.e. C:\ProgramData\chocolatey\lib\yourPackageName).
A shim file for the command line goes to 'C:\ProgramData\chocolatey\bin'
and points to an executable in 'C:\ProgramData\chocolatey\lib\yourPackageName'.
Creating Chocolatey folders if they do not already exist.

WARNING: You can safely ignore errors related to missing log files when
  upgrading from a version of Chocolatey less than 0.9.9.
  'Batch file could not be found' is also safe to ignore.
  'The system cannot find the file specified' - also safe.
chocolatey.nupkg file not installed in lib.
  Attempting to locate it from bootstrapper.
PATH environment variable does not have C:\ProgramData\chocolatey\bin in it. Adding...
ADVERTENCIA: Not setting tab completion: Profile file does not exist at 'C:\Users\magdi\OneDrive\Documentos\WindowsPowerShell\Microsoft.PowerShell_profile.ps1'.
Chocolatey (choco.exe) is now ready.
You can call choco from anywhere, command line or powershell by typing choco.
Run choco /? for a list of functions.
You may need to shut down and restart powershell and/or consoles
first prior to using choco.
Ensuring Chocolatey commands are on the path
Ensuring chocolatey.nupkg is in the lib folder
```

```
PS C:\Windows\system32> choco
Chocolatey v0.10.15
Please run 'choco -?' or 'choco <command> -?' for help menu.
```

1.7.4. [scrcpy](#)

[scrcpy](#) es una aplicación para poder ver la pantalla de tu celular en tu computadora, conectándolo con un cable USB. Se puede instalar de diversas formas, indicadas en el GitHub, al que puedes hacer dando click aquí: [scrcpy](#).

1.7.4.1. INSTRUCCIONES INSTALACIÓN DESDE POWERSHELL con CHOCOLATEY

En mi caso, decidí instalarlo con [Chocolatey](#), ya que es más rápido y sencillo que tener que descomprimir carpetas manualmente y cosas así.

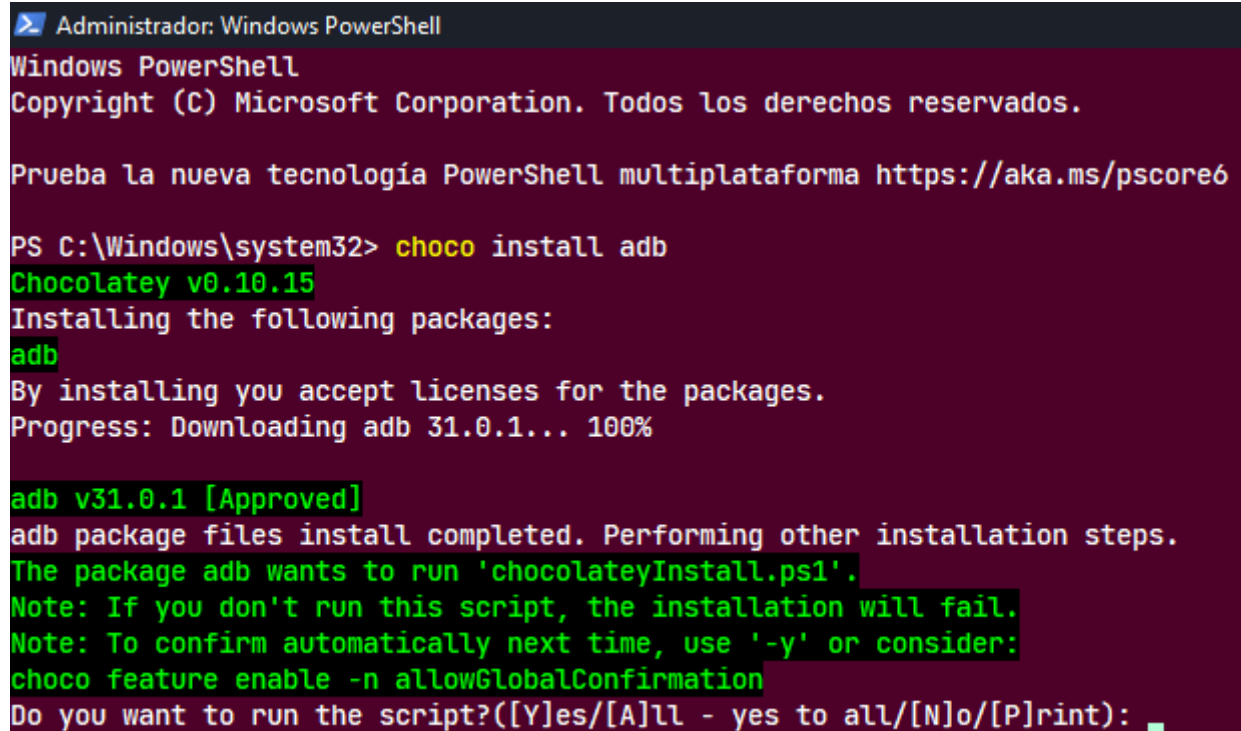
Hay que ejecutar los siguientes comandos en la terminal de [Windows Powershell](#), que ya viene instalada en Windows 10.

NOTA: Va a pedir que aceptes varias cosas, por lo que tendrás que hacerlo para que los programas se puedas descargar e instalar.

- **Y** <- **Correr solo el script actual.**
- **A** <- **Correr todos los scripts.**

1. Hay que instalar una cosa antes, para hacerlo funcionar, que es el **Android Debug Bridge (adb)**:

```
choco install adb
```



```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> choco install adb
Chocolatey v0.10.15
Installing the following packages:
adb
By installing you accept licenses for the packages.
Progress: Downloading adb 31.0.1... 100%

adb v31.0.1 [Approved]
adb package files install completed. Performing other installation steps.
The package adb wants to run 'chocolateyInstall.ps1'.
Note: If you don't run this script, the installation will fail.
Note: To confirm automatically next time, use '-y' or consider:
choco feature enable -n allowGlobalConfirmation
Do you want to run the script?([Y]es/[A]ll - yes to all/[N]o/[P]rint):
```



```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> choco install adb
Chocolatey v0.10.15
Installing the following packages:
adb
By installing you accept licenses for the packages.
Progress: Downloading adb 31.0.1... 100%

adb v31.0.1 [Approved]
adb package files install completed. Performing other installation steps.
The package adb wants to run 'chocolateyInstall.ps1'.
Note: If you don't run this script, the installation will fail.
Note: To confirm automatically next time, use '-y' or consider:
choco feature enable -n allowGlobalConfirmation
Do you want to run the script?([Y]es/[A]ll - yes to all/[N]o/[P]rint): A

Downloading adb
  from 'https://dl-ssl.google.com/android/repository/platform-tools_r31.0.1-windows.zip'
Progress: 100% - Completed download of C:\Users\magdi\AppData\Local\Temp\chocolatey\adb\31.0.1\platform-tools_r31.0.1-windows.zip (11.66 MB).
Download of platform-tools_r31.0.1-windows.zip (11.66 MB) completed.
Hashes match.
Extracting C:\Users\magdi\AppData\Local\Temp\chocolatey\adb\31.0.1\platform-tools_r31.0.1-windows.zip to C:\ProgramData\chocolatey\lib\adb\tools...
C:\ProgramData\chocolatey\lib\adb\tools
ShimGen has successfully created a shim for adb.exe
ShimGen has successfully created a shim for fastboot.exe
ShimGen has successfully created a shim for make_f2fs.exe
ShimGen has successfully created a shim for make_f2fs_casefold.exe
ShimGen has successfully created a shim for mke2fs.exe
The install of adb was successful.
Software installed to 'C:\ProgramData\chocolatey\lib\adb\tools'

Chocolatey installed 1/1 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).
```

2. Ahora ya habría que instalar [scrcpy] escribiendo lo siguiente:

```
choco install scrcpy
```

```
Administrador: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnología PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Windows\system32> choco install scrcpy
Chocolatey v0.10.15
Installing the following packages:
scrcpy
By installing you accept licenses for the packages.
Progress: Downloading chocolatey-core.extension 1.3.5.1... 100%
Progress: Downloading scrcpy 1.17... 100%

chocolatey-core.extension v1.3.5.1 [Approved]
chocolatey-core.extension package files install completed. Performing other installation steps.
Installed/updated chocolatey-core extensions.
The install of chocolatey-core.extension was successful.
Software installed to 'C:\ProgramData\chocolatey\extensions\chocolatey-core'

scrcpy v1.17 [Approved]
scrcpy package files install completed. Performing other installation steps.
The package scrcpy wants to run 'chocolateyinstall.ps1'.
Note: If you don't run this script, the installation will fail.
Note: To confirm automatically next time, use '-y' or consider:
choco feature enable -n allowGlobalConfirmation
Do you want to run the script?([Y]es/[A]ll - yes to all/[N]o/[P]rint):
```

```

Administrador: Windows PowerShell
chocolatey-core.extension v1.3.5.1 [Approved]
chocolatey-core.extension package files install completed. Performing other installation steps.
Installed/updated chocolatey-core extensions.
The install of chocolatey-core.extension was successful.
Software installed to 'C:\ProgramData\chocolatey\extensions\chocolatey-core'

scrcpy v1.17 [Approved]
scrcpy package files install completed. Performing other installation steps.
The package scrcpy wants to run 'chocolateyinstall.ps1'.
Note: If you don't run this script, the installation will fail.
Note: To confirm automatically next time, use '-y' or consider:
choco feature enable -n allowGlobalConfirmation
Do you want to run the script?([Y]es/[A]ll - yes to all/[N]o/[P]rint): A

Downloading scrcpy
  from 'https://github.com/Genymobile/scrcpy/releases/download/v1.17/scrcpy-win64-v1.17.zip'
Progress: 100% - Completed download of C:\Users\magdi\AppData\Local\Temp\chocolatey\scrcpy\1.17\scrcpy-win64-v1.17.zip (24.93 MB).
Download of scrcpy-win64-v1.17.zip (24.93 MB) completed.
Hashes match.
Extracting C:\Users\magdi\AppData\Local\Temp\chocolatey\scrcpy\1.17\scrcpy-win64-v1.17.zip to C:\ProgramData\chocolatey\lib\scrcpy\tools...
C:\ProgramData\chocolatey\lib\scrcpy\tools
Downloading android.ico
  from 'https://www.iconfinder.com/icons/3185263/download/icon/512'
Download of android.ico (-1 B) completed.
Hashes match.
C:\ProgramData\chocolatey\lib\scrcpy\tools\android.ico
ShimGen has successfully created a shim for adb.exe
ShimGen has successfully created a shim for scrcpy.exe
The install of scrcpy was successful.
Software installed to 'C:\ProgramData\chocolatey\lib\scrcpy\tools'

Chocolatey installed 2/2 packages.
See the log for details (C:\ProgramData\chocolatey\logs\chocolatey.log).

Enjoy using Chocolatey? Explore more amazing features to take your
experience to the next level at
https://chocolatey.org/compare

```

1.7.5. JAVA

Es necesario tener instalado Java y tenerlo en las variables del sistema para poder ejecutar los programas.

1.7.5.1. INFORMACIÓN SOBRE LAS VERSIONES DE JAVA

Hay diversas versiones de Java, por lo que hay que saber elegir cuál utilizar. En el siguiente enlace esto viene bien explicado:

- [stackchief | Which Version of Java Should You Use?](#)

1.7.5.2. CONCEPTOS EN LOS ACRÓNIMOS DE JAVA: **JDK**, **JVM**, **JRE**

Hay diversos acrónimos que se manejan en Java, por lo que identificar cuál significa qué cosa es complicado.

La información para conocer esto la encontré en el siguiente enlace (mismo del punto anterior):

- [stackchief | Which Version of Java Should You Use?](#)
- **JDK: Java Developer Kit**

The **Java Developer Kit (JDK)** is a development environment that you download for developing Java applications. It includes the JRE as well as an interpreter, compiler, and other tools for debugging and development. You'll need the JDK to develop Java applications, but not to run them. This is because the JRE is included with Java installs.

- **JVM: Java Virtual Machine**

The **Java Virtual Machine (JVM)** which is an abstract computing machine that enables your computer to run a Java program.

- **JRE: Java Runtime Environment**

Java requires a runtime environment **to execute the code you write**. When you download Java, you also get the **Java Runtime Environment (JRE)**. This JRE includes a Java Virtual Machine (JVM) which is an abstract computing machine that enables your computer to run a Java program.

1.7.5.3. DESCARGA DE JAVA

En mi caso descargué la versión **Java SE 11 (LTS)**, ya que es la última versión que se ha lanzado con **LTS (Long Term Support - Soporte a largo plazo)**, lo que significa que es una versión que tendrá soporte por mucho más tiempo de lo habitual. Esto no debería ser un problema y podría instalar la última versión que se ha lanzado, la **Java SE 16**, pero por ahora instalaré esta.

- [TheServerSide | Don't ever put a non-Java LTS release into production](#)

Por lo que leí, la próxima versión de **Java LTS** será la versión **17**, y llegará en este año 2021, pero a día de hoy, Mayo del 2021 aún no ha salido. Cada versión de Java es lanzada cada 6 meses.

ENLACE DE DESCARGA DE VERSIONES DE JAVA SE (Standard Edition)

- [Oracle | Java SE Downloads | Java Platform, Standard Edition](#)

DESCARGAS DE Java SE Development Kit (JDK) 11

- [Oracle | Java SE Development Kit 11 Downloads](#)

- **Descargué la versión de Windows:**

Product / File Description	File Size	Download
Windows x64 Installer	152.05 MB	jdk-11.0.11_windows-x64_bin.exe

NOTA: REQUIERES TENER UNA CUENTA DE ORACLE PARA LA INSTALACIÓN.

- **DIRECTORIO EN MI PC:** `C:\Program Files\Java\jdk-11.0.11\#`

1.8. VARIABLES DEL SISTEMA o de USUARIO

Para que los programas funcionen de la mejor forma posible, hay que establecer las variables del sistema que identifiquen a los programas. Esto se explicará en esta sección.

Además, las variables del sistema permiten ejecutar ciertas instrucciones desde la línea de comandos.

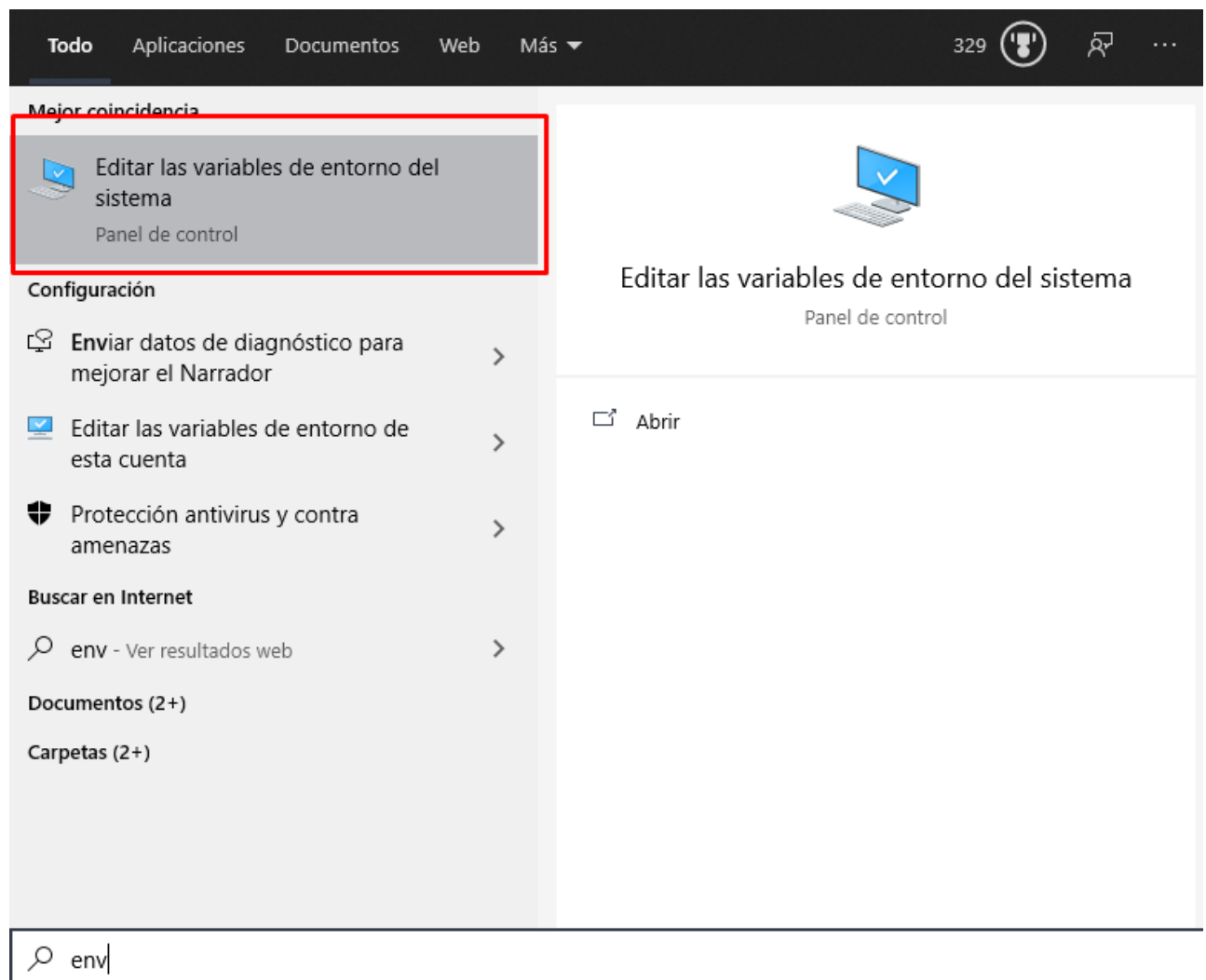
1.8.1. ACCEDER A LA SECCIÓN PARA EDITAR LAS VARIABLES

Para esto hay que seguir una serie de pasos que serán indicados a continuación.

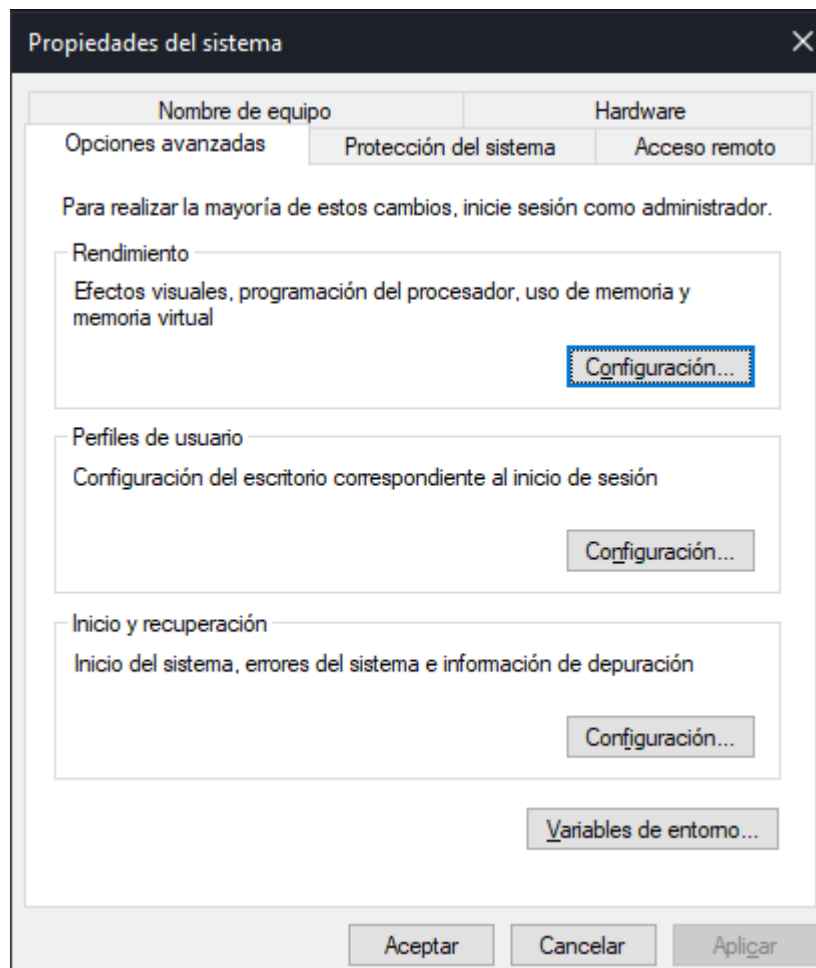
1. Abre el buscador de Windows. Hay 2 formas de hacerlo.

1. Presiona la lupa que está al lado del ícono de Windows en la barra de tareas.
2. Presiona la combinación de teclas: **Windows + S**

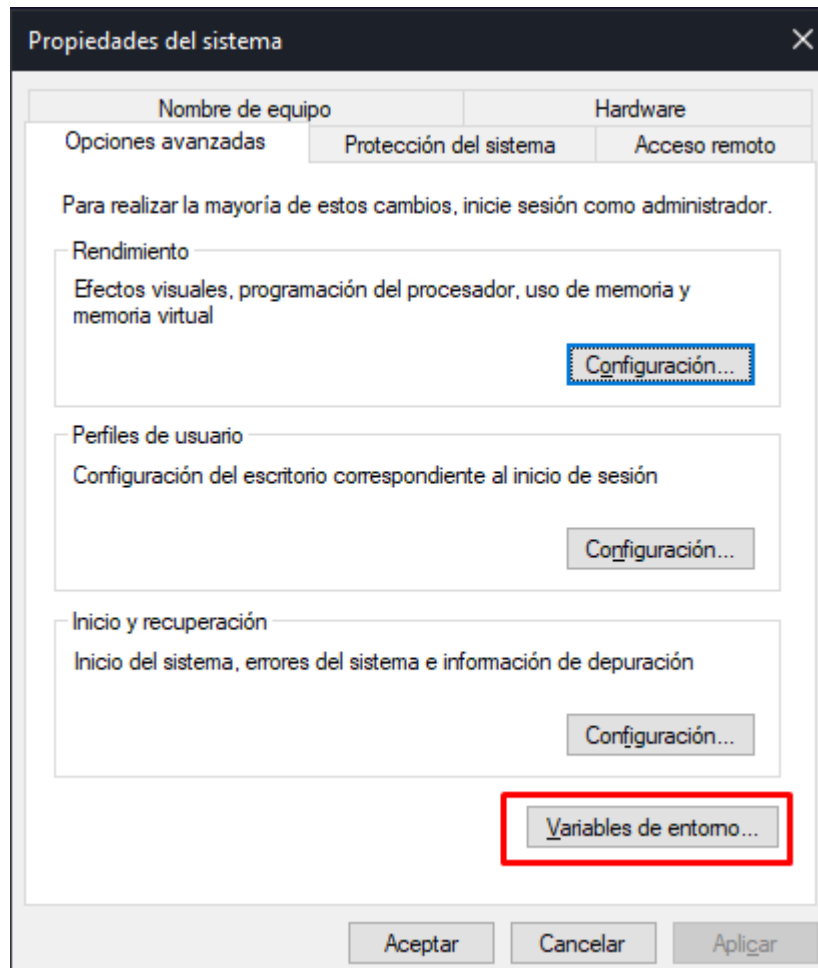
2. Escribir en la barra lo siguiente: **env**.
3. Selecciona la opción: "**Editar Las variables de entorno del sistema**".



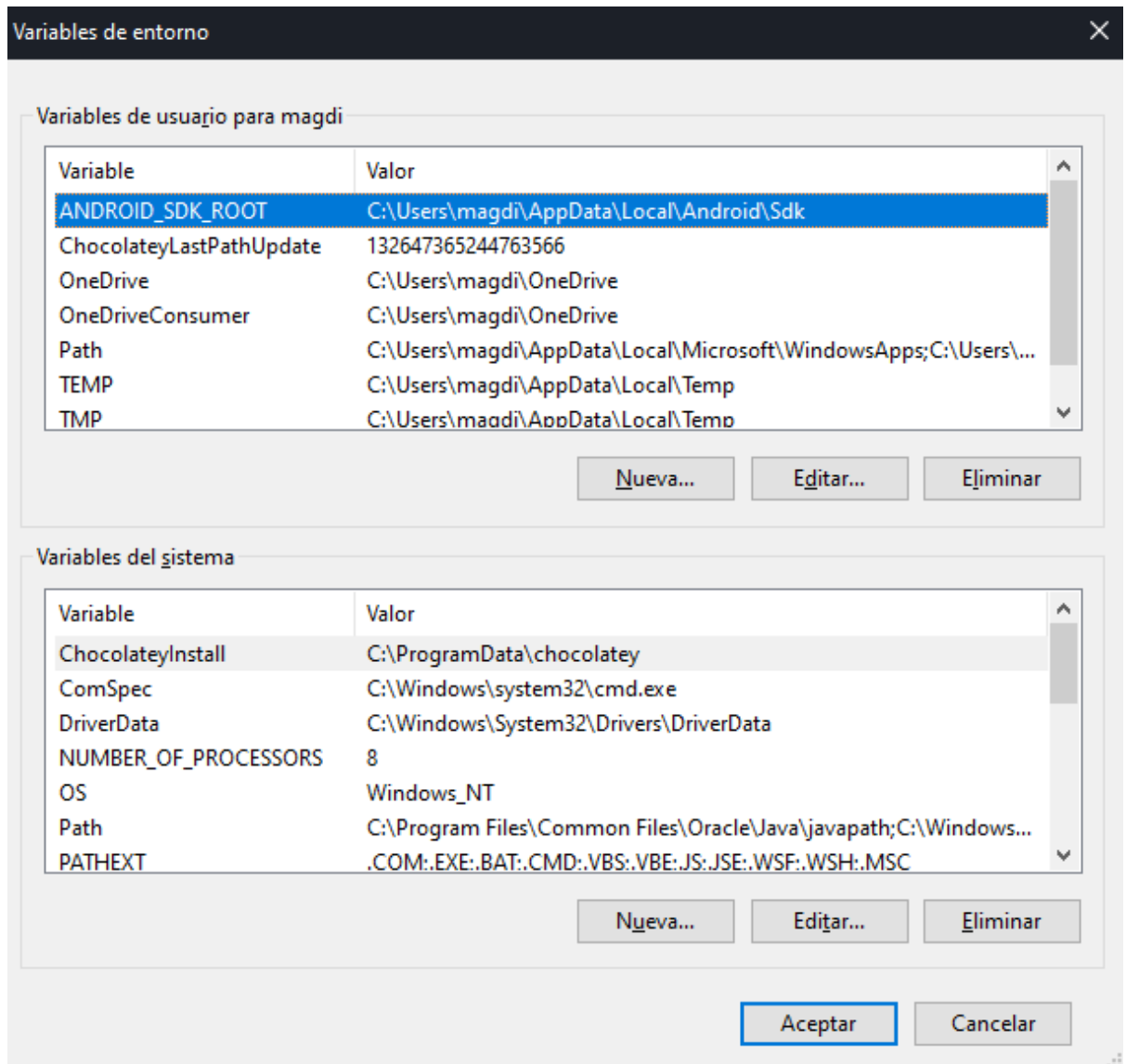
- Te aparecerá una ventana: **Propiedades del sistema**.



4. Selecciona el botón: "Variables de entorno...".



- Te aparecerá una nueva ventana: "Variables de entorno", que muestra y da la opción de manipular todas las variables de usuario y del sistema.



5. En esta sección ya puedes modificar las **variables de usuario o del sistema** como lo requieras.

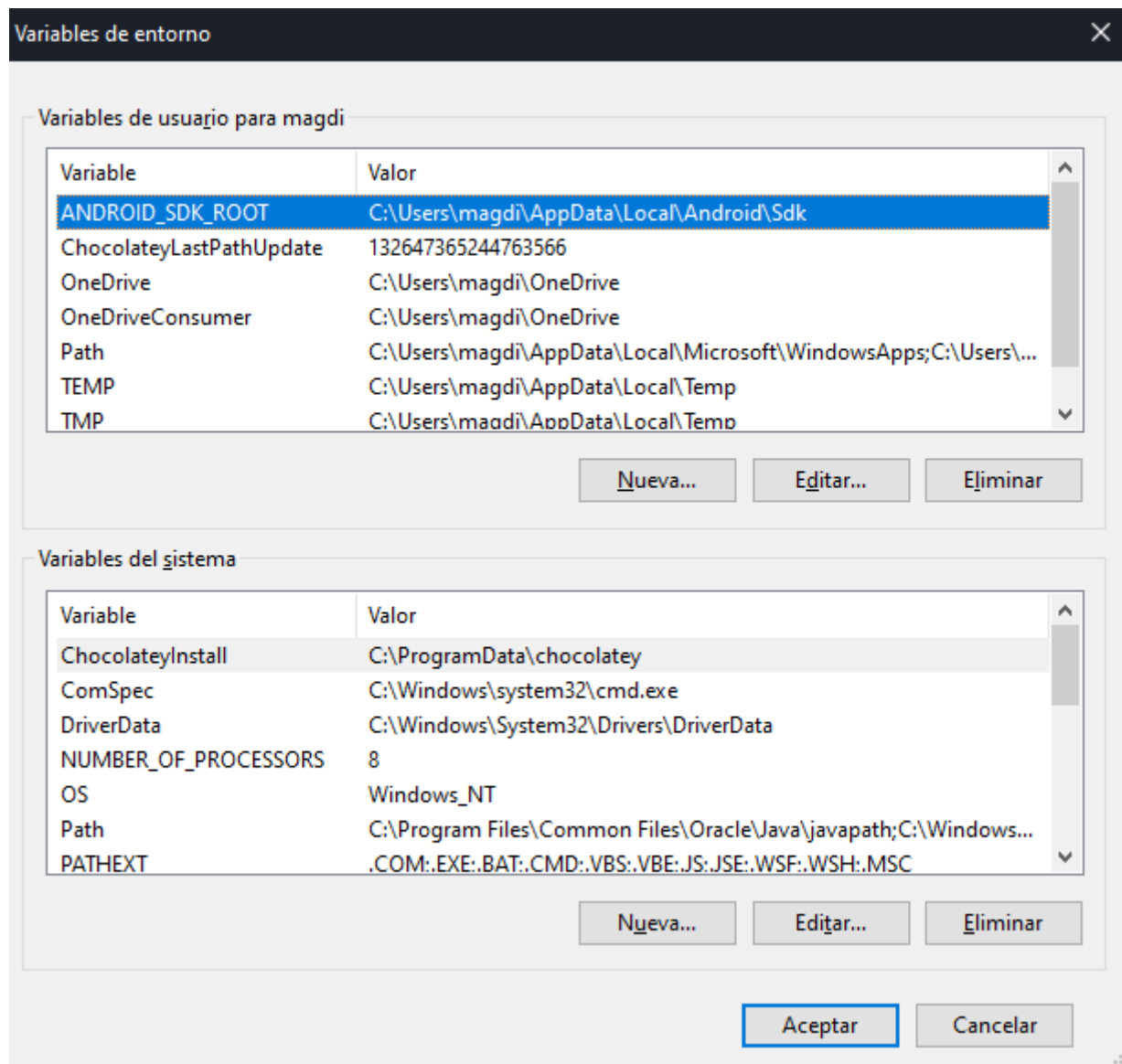
1.8.2. VARIABLE DEL SISTEMA: JAVA_HOME

Para poder ejecutar los programas necesitamos tener instalado **Java** y agregado a **JAVA_HOME**.

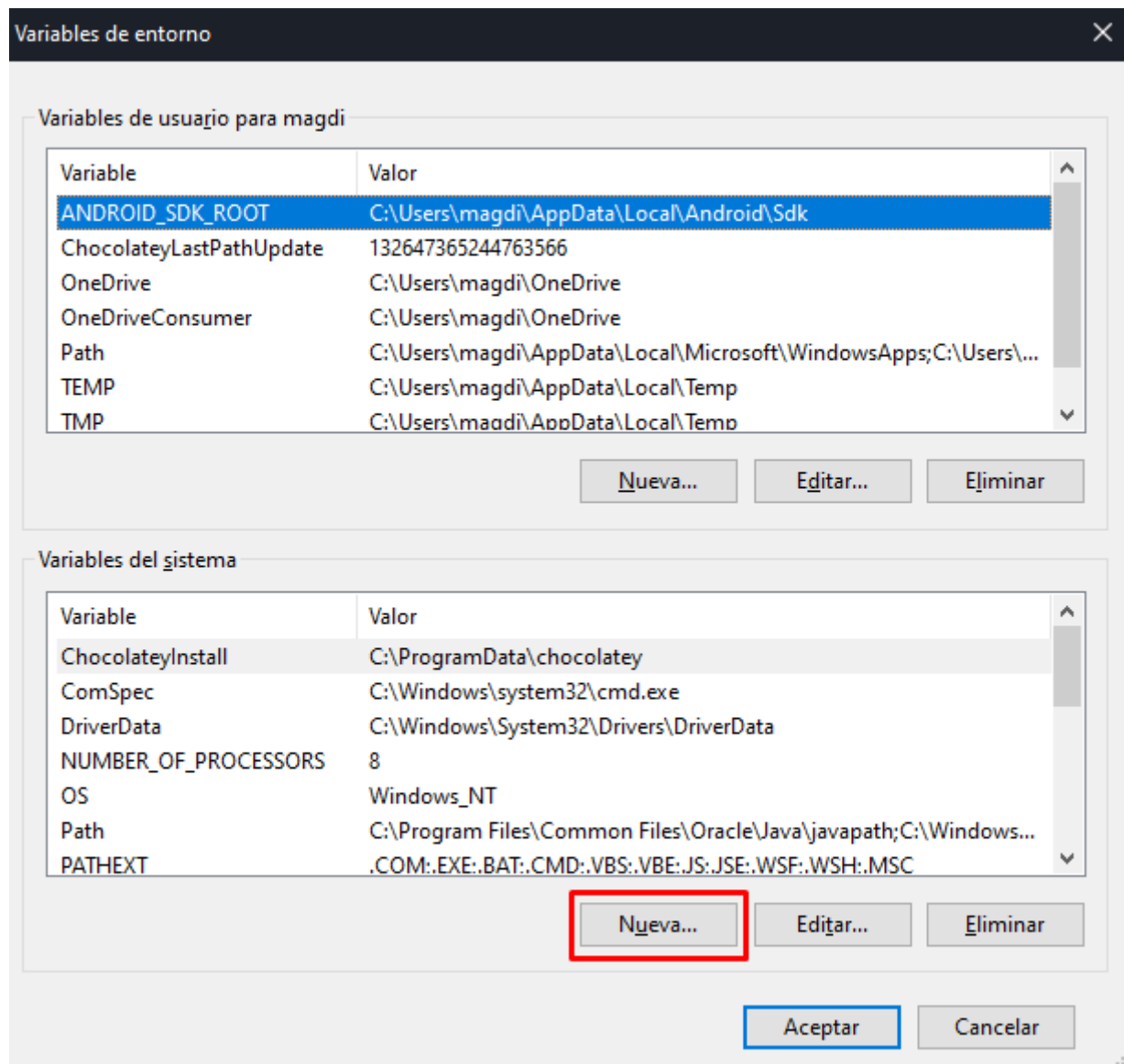
En el siguiente enlace están las instrucciones para establecer **JAVA_HOME**:

- [Confluence Support | Setting the JAVA_HOME Variable in Windows](#)

1. Accede a la ventana de "**Variables del entorno**" como indican los pasos del inicio de la sección.

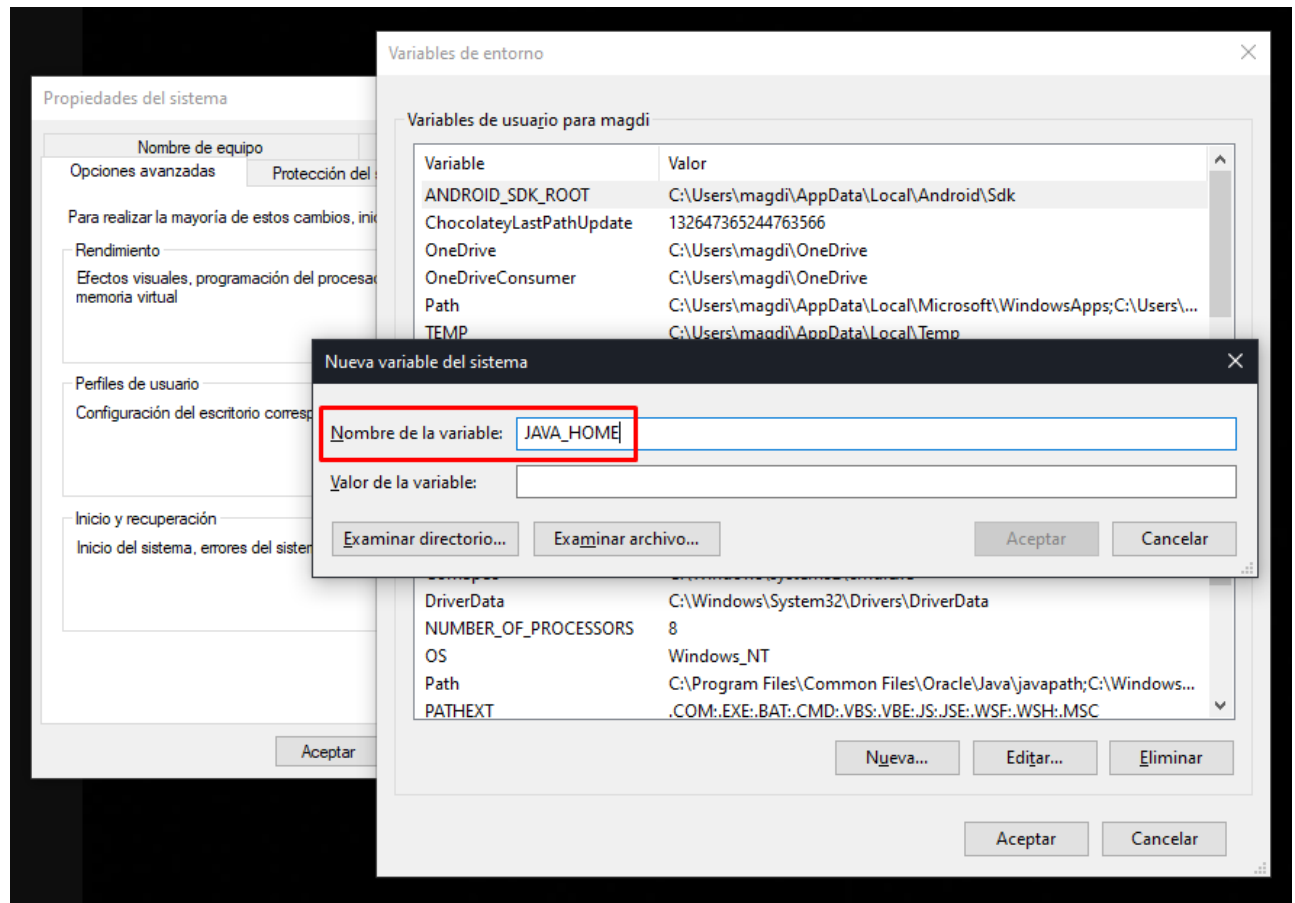


2. En las *variables del sistema* da click en "Nueva...".



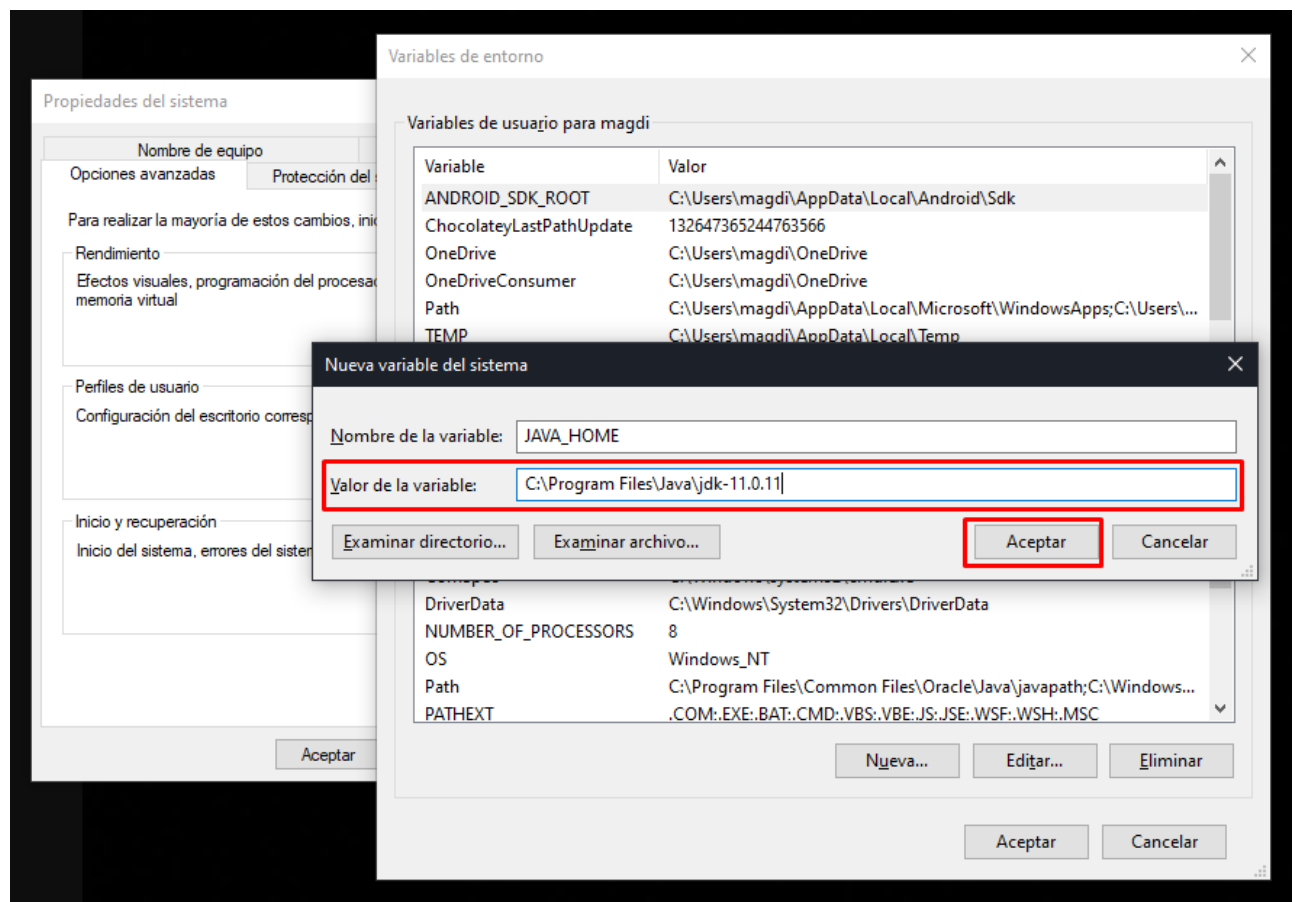
3. En el campo "*Nombre de la variable:*" ingresa lo siguiente:

JAVA_HOME

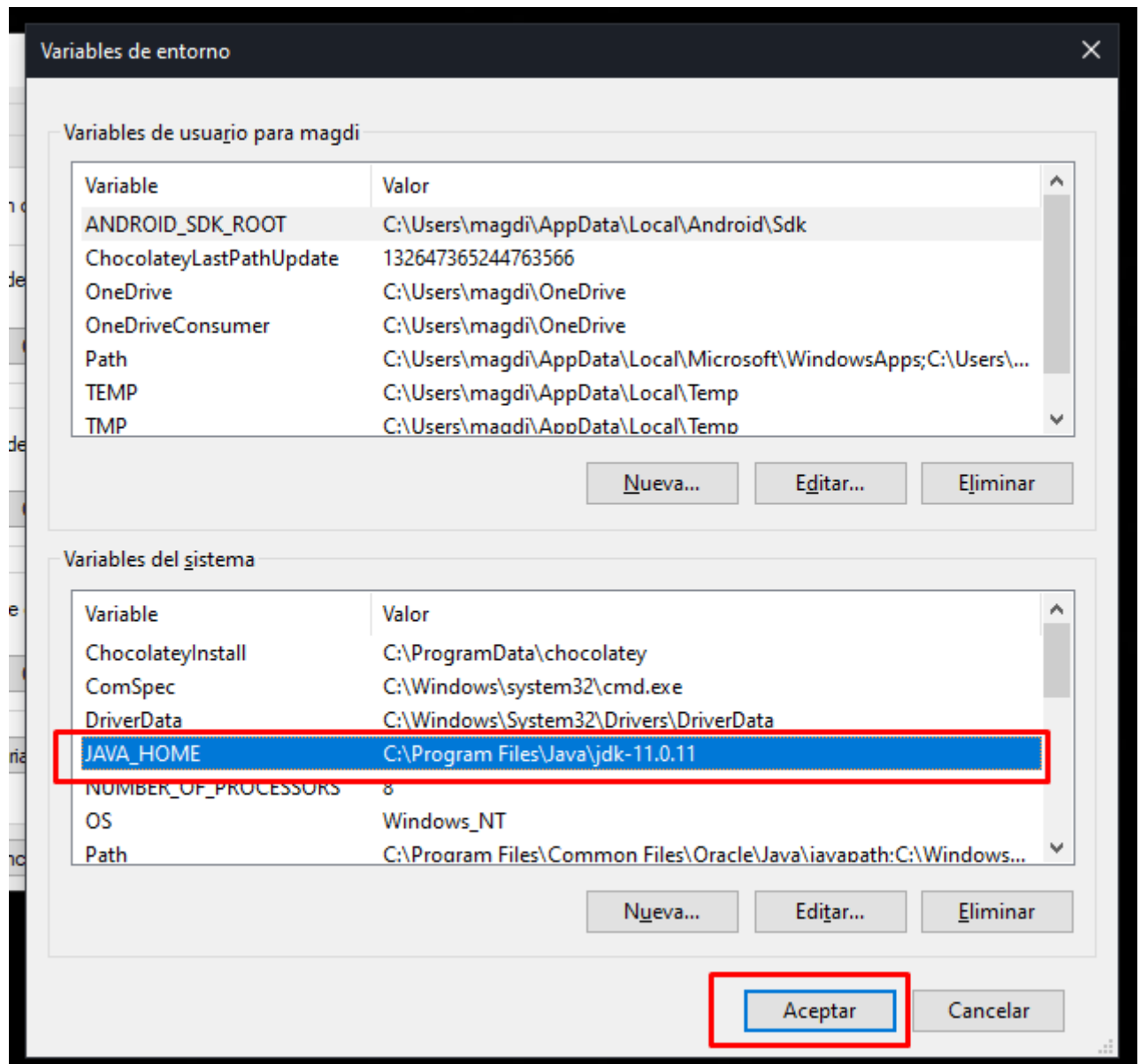


4. En el campo "**Valor de la variable:**" ingresa el directorio de instalación de tu JDK. En mi caso fue el siguiente (no lleva backslash "\" al final):

C:\Program Files\Java\jdk-11.0.11



5. Da click en "**Aceptar**".
6. Listo, tu variable del sistema ya está establecida.



1.8.2.1. VERIFICACIÓN DE QUE JAVA_HOME ESTÁ ESTABLECIDO

Hay que verificar que todo haya salido bien y el sistema identifica la variable **JAVA_HOME**. Esto se logra ejecutando un comando después de haber reiniciado cualquier terminal que haya abierta, ya que si no, las variables del sistema no se actualizan, y si no hay efecto, entonces hay que reiniciar el sistema.

El comando es el siguiente:

```
java --version
```

Si todo se configuró correctamente, te mostrará un mensaje con tu versión instalada de Java.

```
$ java --version
java 11.0.11 2021-04-20 LTS
Java(TM) SE Runtime Environment 18.9 (build 11.0.11+9-LTS-194)
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.11+9-LTS-194, mixed mode)
```

4.2. ANDROID STUDIO

Para [Android Studio](#) solo hay una **VARIABLE DE USUARIO** que yo utilicé:

- `ANDROID_SDK_ROOT=C:\Users\...\AppData\Local\Android\Sdk`

4.3. FLUTTER

Para que flutter sea reconocido en el sistema, hay que agregarlo a las variables de usuario, [como es indicado en la documentación](#).

En la sección de **variables de usuario** hay que revisar si ya hay una variable llamada **Path**.

- Si no la hay, habrá que crearla.
- Si la hay, seguir los siguientes pasos:
 1. Dar click en la variable **Path**.
 2. Dar click en el botón "**Editar...**".
 3. Dar click en el botón "**Nuevo**".
 4. En el campo agregar el directorio de **flutter\bin** dependiendo de en dónde instalaste **Flutter**.

En mi caso puse lo siguiente en **Path**:

- `C:\SDKs\flutter\bin`

1.8.2.2. OTRAS VARIABLES DE USUARIO PARA FLUTTER y DART

Además, es necesario agregar otras **variables de usuario** para poder trabajar bien con flutter, que son las siguientes:

- **Path**
 - `C:\SDKs\flutter\pub-cache\bin`
 - `C:\SDKs\flutter\bin\cache\dart-sdk\bin`

Esta variable de usuario es requerida para poder ejecutar **dartdoc** desde la terminal.

- **FLUTTER_ROOT**
 - `C:\SDKs\flutter`

1.9. FIREBASE

Para poder utilizar Firebase se tuvieron que realizar algunas modificaciones en algunos archivos, además de hacer la configuración en línea.

- `com.uaslp.facebook_simple_replication`

1.9.1. SCREENSHOTS CONFIGURACIÓN FIREBASE

× Agrega Firebase a tu app para Android

✓ Registrar app

Nombre del paquete de Android: com.uaslp.facebook_simple_replication; sobrenombre de la app: Facebook Mobile

✎ Descargar archivo de configuración

3 Agregar el SDK de Firebase

[Instrucciones para Gradle](#) | [Unity](#) [C++](#)

El complemento de Google Services para [Gradle](#) carga el archivo `google-services.json` que acabas de descargar. Para usar el complemento, debes modificar los archivos `build.gradle`.

Archivo `build.gradle` de nivel de proyecto (<project>/`build.gradle`):

```
buildscript {  
    repositories {  
        // Check that you have the following line (if not, add it):  
        google() // Google's Maven repository  
    }  
    dependencies {  
        ...  
        // Add this line  
        classpath 'com.google.gms:google-services:4.3.8'  
    }  
}
```

✓ Registrar app

Nombre del paquete de Android: com.uaslp.facebook_simple_replication; sobrenombre de la app: Facebook Mobile

2 Descargar archivo de configuración

[Instrucciones para Android Studio a continuación](#) | [Unity](#) [C++](#)

⬇ Descargar `google-services.json`

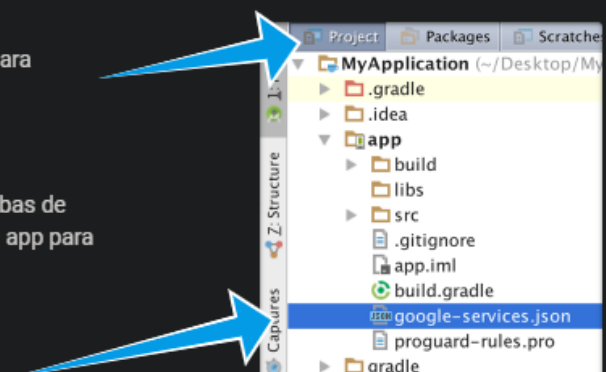
Cambia a la vista **Proyecto** de Android Studio para ver el directorio raíz de tu proyecto.

Coloca el archivo `google-services.json` que acabas de descargar en el directorio raíz del módulo de tu app para Android.

google-services.json

Siguiente

✎ Agregar el SDK de Firebase



3 Agregar el SDK de Firebase

Instrucciones para Gradle | [Unity](#) [C++](#)

El complemento de Google Services para [Gradle](#) carga el archivo `google-services.json` que acabas de descargar. Para usar el complemento, debes modificar los archivos `build.gradle`.

Archivo `build.gradle` de nivel de proyecto (<project>/`build.gradle`):

```
buildscript {
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
    }
    dependencies {
        ...
        // Add this line
        classpath 'com.google.gms:google-services:4.3.8'
    }
}

allprojects {
    ...
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
        ...
    }
}
```

Archivo build.gradle de nivel de app (<project>/<app-module>/build.gradle):

```
apply plugin: 'com.android.application'
// Add this line
apply plugin: 'com.google.gms.google-services'

dependencies {
    // Import the Firebase BoM
    implementation platform('com.google.firebase:firebase-bom:28.0.1')

    // Add the dependencies for the desired Firebase products
    // https://firebase.google.com/docs/android/setup#available-libraries
}
```

Si usas la BoM de Firebase para Android, tu app siempre utilizará versiones compatibles de la biblioteca de Firebase. [Más información](#)

Por último, presiona "Sincronizar ahora" en la barra que aparece en el entorno IDE:

Gradle files have changed since last sync. [Sync now](#)

[Anterior](#) [Siguiente](#)

4 Próximos pasos

× **Agrega Firebase a tu app para Android** [Ir a la documentación](#)

- ✓ Registrar app
Nombre del paquete de Android: com.uaslp.facebook_simple_replication; sobrenombre de la app: Facebook Mobile
- ✓ Descargar archivo de configuración
- ✓ Agregar el SDK de Firebase
- 4 Próximos pasos

¡Listo!

Asegúrate de revisar la [documentación](#) para conocer los primeros pasos con cada producto de Firebase que quieras usar en tu app.

También puedes explorar las [apps de muestra de Firebase](#).

También puedes ir a console para explorar Firebase

[Anterior](#) [Ir a la consola](#)

1.9.2. AGREGAR LAS SIGUIENTES LÍNEAS

De las instrucciones de **Agrega Firebase a tu app para Android**:

3. Agregar el SDK de Firebase

El complemento de Google Services para [Gradle](#) carga el archivo google-services.json que acabas de descargar. Para usar el complemento, debes modificar los archivos **build.gradle**.

Archivo `*build.gradle*` de nivel de proyecto (`<project>/build.gradle`):

```
buildscript {
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
    }
    dependencies {
        ...
        // Add this line
        classpath 'com.google.gms:google-services:4.3.8'
    }
}

allprojects {
    ...
    repositories {
        // Check that you have the following line (if not, add it):
        google() // Google's Maven repository
        ...
    }
}
```

Archivo `*build.gradle*` de nivel de app (`<project>/<app-module>/build.gradle`):

```
apply plugin: 'com.android.application'
// Add this line
apply plugin: 'com.google.gms.google-services'

dependencies {
    // Import the Firebase BoM
    implementation platform('com.google.firebase:firebase-bom:28.0.1')

    // Add the dependencies for the desired Firebase products
    // https://firebase.google.com/docs/android/setup#available-libraries
}
```

Después de haber realizado estos pasos anteriores, utilizar el comando `flutter packages get` (equivalente a `flutter pub get`) para que quede todo actualizado, el cual es indicado en el siguiente sitio web:

- [Firebase | Documentación | Firebase / Documentos / Información básica / Agrega Firebase a tu app de Flutter](#)

1.10. RECURSOS DE FACEBOOK APP (SVG, ÍCONOS, ...)

Para poder maquetar de forma más precisa las pantallas de la app de Facebook para celulares, necesitamos diversos recursos, como los íconos de Facebook, reacciones y demás.

En esta sección se encontrarán diversos enlaces, en los cuales se podrán encontrar los recursos que utilizamos.

1.10.1. LOGO DE FACEBOOK

Los logos de Facebook los descargamos de la página oficial de los recursos (assets):

- [Brand Resources / Facebook App / "f" Logo](#)

Los logotipos descargados se encuentran en el siguiente directorio:

[Directorio con todos los logos descargados | assets/fb_official/logos/](#)

Los logos con la F blanca y el círculo azul, se encuentran en diversos directorios y con diversos tamaños, por lo que los listaré para encontrarlos con mayor facilidad y utilizar los que más nos convengan:

- [Carpeta principal de los siguientes logos:](#)
[assets/fb_official/logos/f_Logo_Online_04_2019/Color/PNG/](#)





o



o



o



o

- Carpeta principal de los siguientes logos: [assets/fb_official/logos/f_Logo_TVFilm_04_2019/Color/](#)



1.10.2. REACCIONES DE FACEBOOK

Facebook tiene su propio sitio web en donde provee de estos íconos, pero se tienen que seguir ciertas pautas para utilizarlos de forma que no se rompan las normas de Facebook. El enlace es el siguiente:

- [Brand Resources / Facebook App / Reactions](#)

El problema es que se descargan como una imagen en PNG y los necesitamos por separado, por lo que recurrimos al siguiente enlace:

- [iconscout | Facebook Reaction Icon Pack](#)

- **Y para poder renderizar y modificar los elementos en SVG podemos utilizar el siguiente paquete:**

- [pub.dev | flutter_svg 0.22.0](#)

1.10.3. USO DE ÍCONOS

Podemos utilizar varios paquetes de íconos. En nuestro caso consideramos Font Awesome y Material Design.

1.10.3.1. GOOGLE FONTS | MATERIAL ICONS

[Google Fonts | Material Icons](#)

1.10.3.1.1. Íconos que podríamos utilizar

- [Facebook](#)
- [Search](#)
- [Add](#)
- [More Vert](#)
- [info_outlined](#)

1.10.3.2. Íconos con Font Awesome

Font Awesome, nos provee de una gran cantidad de elementos.

Utilizamos el paquete:

```
pub.dev | font_awesome_flutter 9.0.0
```

Para que funcione hay que agregar el nombre del paquete y la última versión (9.0.0) a las dependencias del `pubspec.yaml`.

1.10.3.2.1. Íconos que podríamos utilizar

- [search](#)
 - El problema es que es más robusto que el de facebook.
 - El de Facebook parece ser el **LIGHT**, pero ese requiere versión **PRO**.
- [facebook](#)
- [plus](#)
 - También es algo robusto.
 - El de Facebook parece ser la versión **LIGHT**.
- [elipsis-v](#)
 - También parece que la versión de Facebook es la **LIGHT** o la **PRO**.
- [info-circle](#)
 - **LIGHT**.

1.11. **ENLACES EXTRA QUE FUERON ÚTILES**

- <https://www.youtube.com/watch?v=d5PpeNb-dOY>
- <https://stackoverflow.com/questions/50081213/how-do-i-use-hexadecimal-color-strings-in-flutter>
- https://www.tutorialspoint.com/dart_programming/dart_programming_map.htm
- <https://stackoverflow.com/questions/29182581/global-variables-in-dart>
- <https://stackoverflow.com/questions/49541354/disable-defaultfocushighlightenabled-in-the-android-part-for-a-xamarin-forms-app>
 - <https://github.com/flutter/flutter/issues/16810>
- <https://aschilken.medium.com/flutter-best-practices-colors-and-textstyles-6e14b06fc3a1>
- <https://stackoverflow.com/questions/52489458/how-to-change-status-bar-color-in-flutter>

- <https://stackoverflow.com/questions/64873410/how-to-get-status-bar-height-in-flutter>
 - <https://stackoverflow.com/questions/61969143/no-mediaquery-ancestor-could-be-found>
- <https://stackoverflow.com/questions/50522237/flutter-circle-design/50524531>
- [Stack Overflow | What is the Dart null checking idiom or best practice?](#)
 - [Stack Overflow | What is the Dart null checking idiom or best practice? | atreeon's answer](#)
- [Stack Overflow | Button Width Match Parent](#)
- [GitHub | Genymobile/scrcpy | Screenshot/Record Screen #21](#)
 - [GitHub | Genymobile/scrcpy | Screenshot/Record Screen #21 | Command to take Screenshot or record video](#)
- [Stack Overflow | How to clear Flutter's Build cache?](#)
- [Stack Overflow | How can I add a border to a widget in Flutter?](#)
- [Stack Overflow | Flutter Image Variable](#)
- [Stack Overflow | How to make a widget fill remaining space in a Column](#)
- [Dart | Effective Dart: Style](#)
- [Medium | DARTLANG | Dart \(DartLang\) Introduction: String Interpolation](#)
- [Dart getters and setters](#)
- [Dart | A tour of the Dart language](#)
- [GitHub Gist | aruld/foreachlistset.dart](#)
- [Stack Overflow | What is the difference between named and positional parameters in Dart? | Seth Ladd](#)