

Programación móviles 8B

Docente:

M.A Fued Alejandro Majul Ramírez

Manual de instalación de Flutter y creación de primera aplicación

Alumnos:

Diego Manuel Martínez Molina

José Luis García Dávila

Jorge Noriega Rivera



Contenido

Responsable.....	3
Introducción	4
Lista de requisitos	4
Flutter	5
Instalación:.....	5
Configuración:.....	7
Visual Studio Code	10
Instalación:.....	10
Configuración:.....	11
Android Studio	12
Instalación:.....	12
Configuración:.....	13
Creación de una aplicación:	17

Responsable

Editor	Cambio	Versión
Diego Martínez	<ul style="list-style-type: none">• Instalación y configuración de Flutter.• Instalación y configuración de Visual Studio Code.• Instalación y configuración de Android Studio.• Instalación y configuración de FIREBASE.	1.0.0

Introducción

Bienvenido al manual de instalación, configuración y creación de primera aplicación en Flutter. Flutter es un poderoso framework de desarrollo de aplicaciones móviles que le permite crear aplicaciones nativas de alta calidad para iOS y Android en un solo código base. Este manual lo guiará a través de los pasos para instalar y configurar Flutter en su sistema y luego lo llevará a través del proceso de creación de su primera aplicación. Aprenderá cómo crear una aplicación simple y elegante utilizando los widgets de Flutter. Al final de este manual, tendrá las habilidades y el conocimiento necesario para comenzar a crear sus propias aplicaciones móviles con Flutter.

Lista de requisitos

Antes de instalar Flutter, es importante asegurarse de que su sistema cumpla con los siguientes requisitos mínimos:

Sistema operativo: Flutter es compatible con Windows (versión 7 o posterior), macOS (versión 10.12 o posterior) y Linux (cualquier distribución que pueda ejecutar el compilador de C++). También se puede instalar en Chrome OS.

Espacio en disco: asegúrese de tener al menos 400 MB de espacio en disco disponible para la instalación de Flutter.

Herramientas de desarrollo: Flutter se integra con varios editores de código, incluidos Android Studio, IntelliJ IDEA y Visual Studio Code. Asegúrese de tener uno de estos editores instalados antes de instalar Flutter.

Dependencias adicionales: Flutter requiere que se instalen algunas dependencias adicionales, como el compilador de C++, Git y Java Development Kit (JDK) versión 8 o posterior.

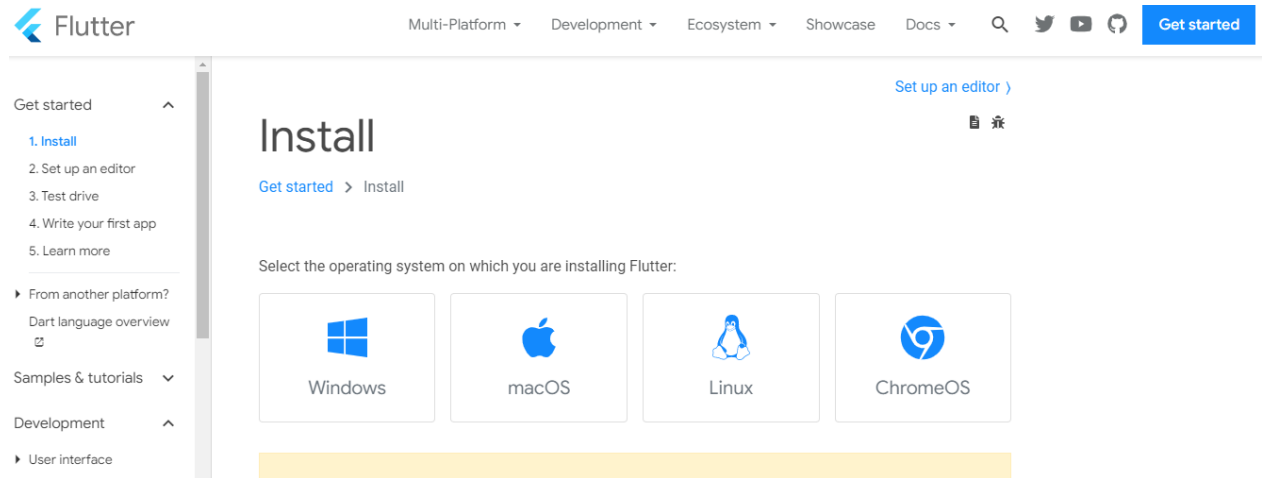
Una vez que su sistema cumple con estos requisitos, puede descargar e instalar Flutter siguiendo los pasos detallados en la documentación.

Flutter

A continuación, se describen a detalle los pasos para instalar y configuración de Flutter.

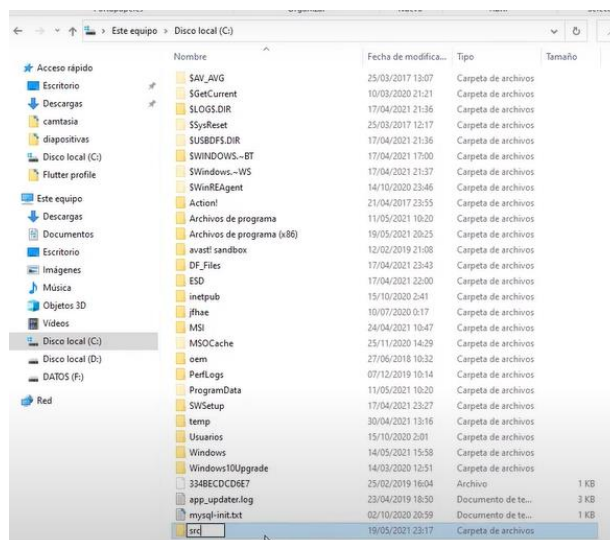
Instalación:

1. El recurso es descargado de la página oficial de Flutter
<https://docs.flutter.dev/get-started/install>

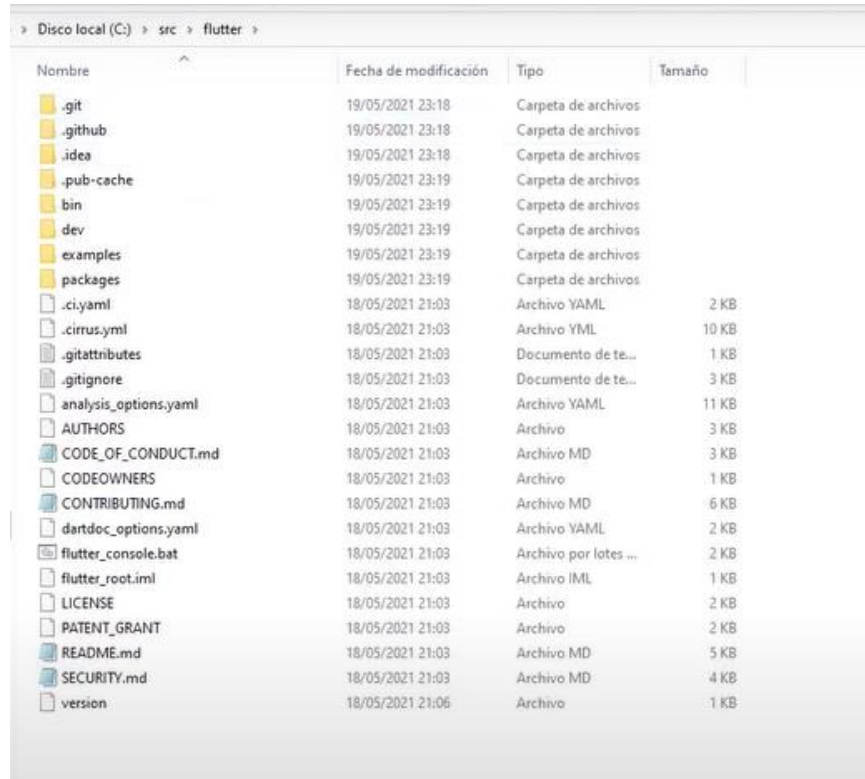


1. Listado de recursos para las diferentes plataformas

2. Ya descargado Flutter, procedemos a ir al disco local C y creamos una nueva carpeta llamada src.



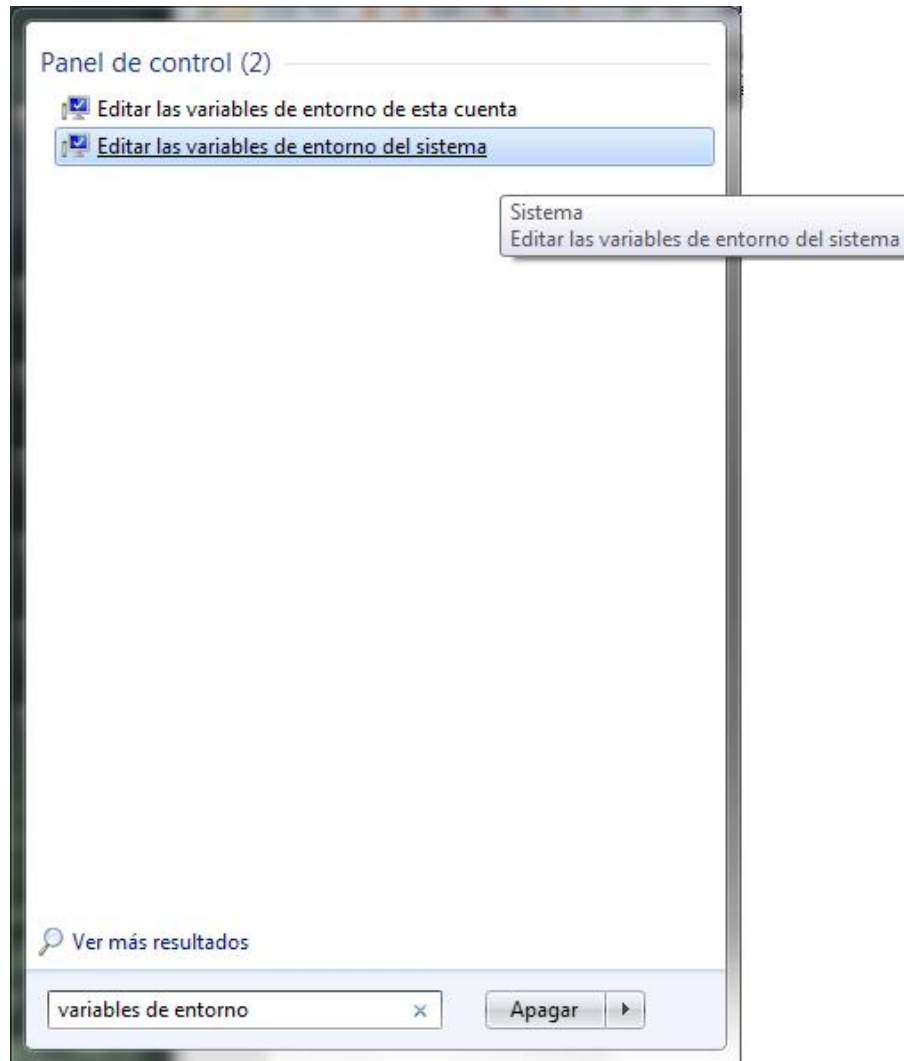
3. Arrastramos el Zip descargado anteriormente y lo descomprimos en la carpeta src.



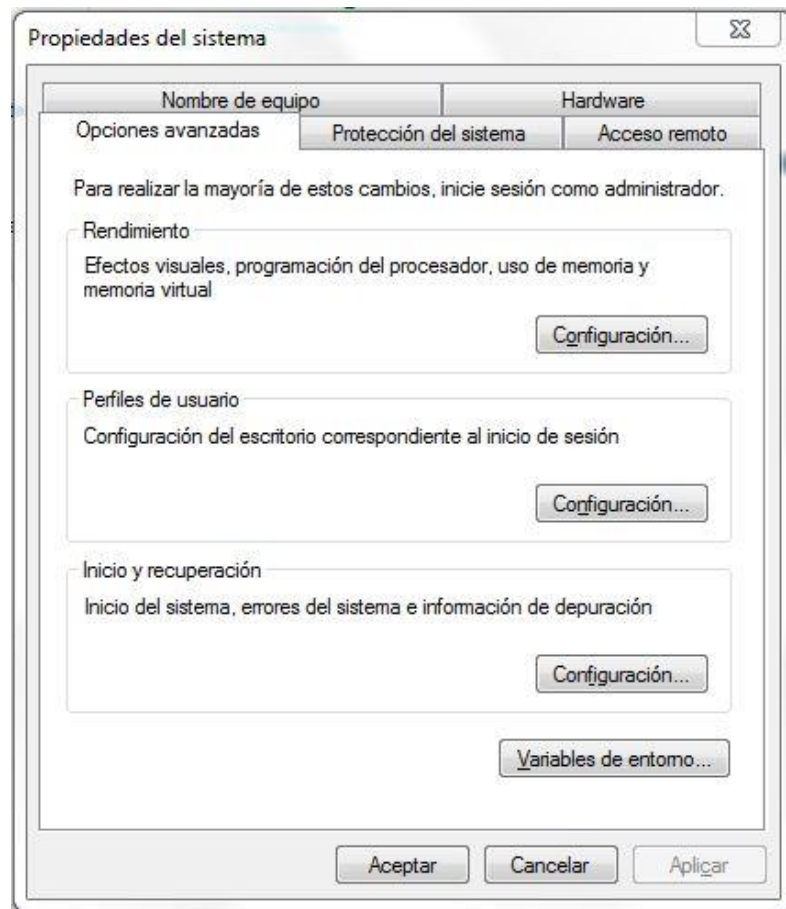
Nombre	Fecha de modificación	Tipo	Tamaño
.git	19/05/2021 23:18	Carpeta de archivos	
.github	19/05/2021 23:18	Carpeta de archivos	
.idea	19/05/2021 23:18	Carpeta de archivos	
.pub-cache	19/05/2021 23:19	Carpeta de archivos	
bin	19/05/2021 23:19	Carpeta de archivos	
dev	19/05/2021 23:19	Carpeta de archivos	
examples	19/05/2021 23:19	Carpeta de archivos	
packages	19/05/2021 23:19	Carpeta de archivos	
.ci.yaml	18/05/2021 21:03	Archivo YAML	2 KB
.cirrus.yml	18/05/2021 21:03	Archivo YML	10 KB
.gitattributes	18/05/2021 21:03	Documento de te...	1 KB
.gitignore	18/05/2021 21:03	Documento de te...	3 KB
analysis_options.yaml	18/05/2021 21:03	Archivo YAML	11 KB
AUTHORS	18/05/2021 21:03	Archivo	3 KB
CODE_OF_CONDUCT.md	18/05/2021 21:03	Archivo MD	3 KB
CODEOWNERS	18/05/2021 21:03	Archivo	1 KB
CONTRIBUTING.md	18/05/2021 21:03	Archivo MD	6 KB
dartdoc_options.yaml	18/05/2021 21:03	Archivo YAML	2 KB
flutter_console.bat	18/05/2021 21:03	Archivo por lotes ...	2 KB
flutter_root.iml	18/05/2021 21:03	Archivo IML	1 KB
LICENSE	18/05/2021 21:03	Archivo	2 KB
PATENT_GRANT	18/05/2021 21:03	Archivo	2 KB
README.md	18/05/2021 21:03	Archivo MD	5 KB
SECURITY.md	18/05/2021 21:03	Archivo MD	4 KB
version	18/05/2021 21:06	Archivo	1 KB

Configuración:

1. Al finalizar la instalación se procede a configurar la variable de ambiente en el equipo, esto lo logramos accediendo al menú "Propiedades del Sistema" por medio de la búsqueda con el asistente del botón inicio de Windows.

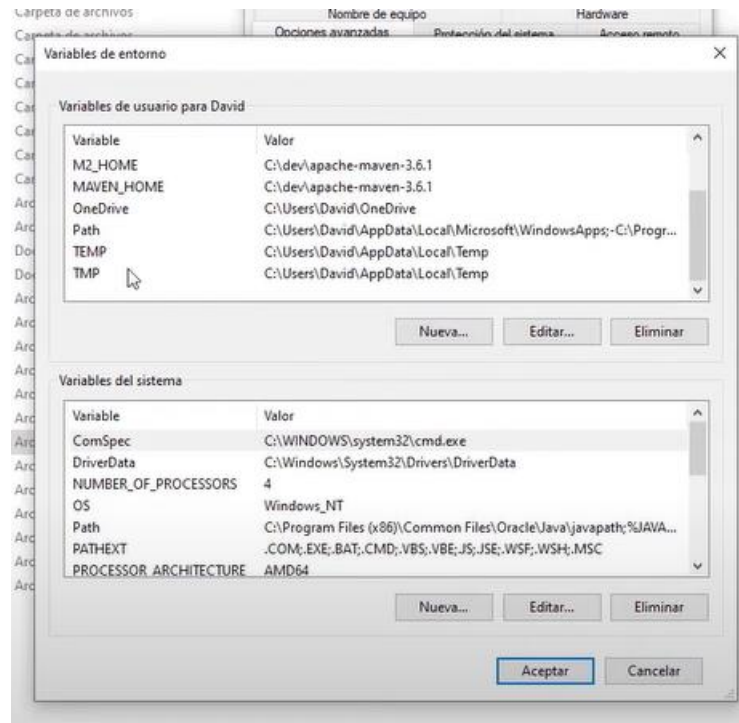


2. Al acceder al menú se muestra una pantalla como se muestra en la imagen.



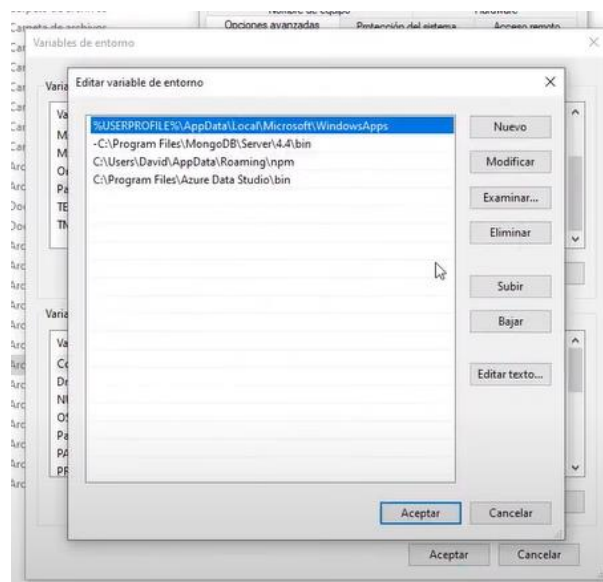
Damos clic en el botón “Variables de entorno...” para poder acceder al listado de las variables del sistema y poder modificar/agregar variables.

3. El menú “Variables de entorno”, como se muestra en la imagen, permite agregar nuevas variables y/o modificar las ya existentes.



Pulsamos sobre la variable Path y después damos en Editar.

4. Ahora nos iremos a la carpeta bin que esta dentro de Flutter y copiaremos la dirección y la pegamos en el siguiente recuadro y pulsamos aceptar.

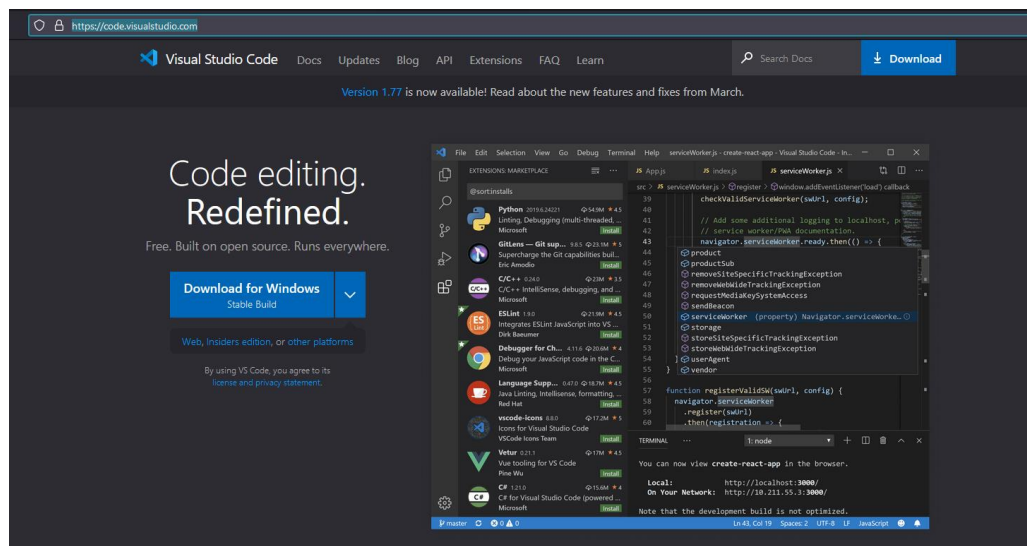


Visual Studio Code

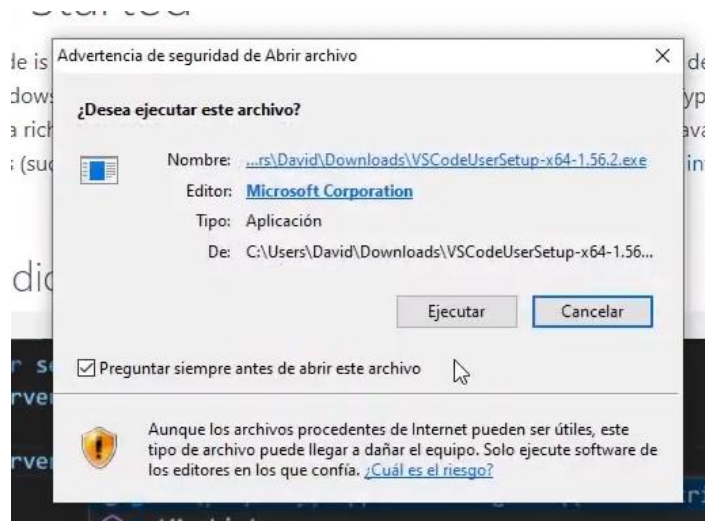
Instalación:

1. El recurso es descargado de la página oficial de Visual Studio Code

<https://code.visualstudio.com/>

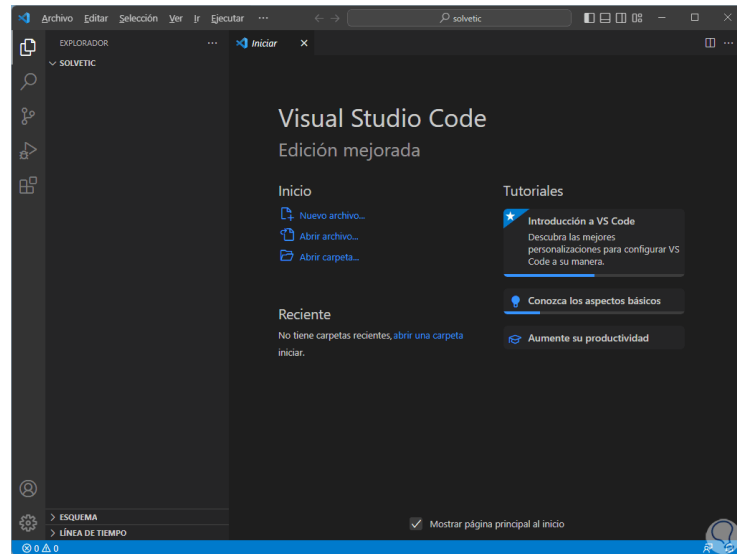


2. Lo siguiente será abrir el instalador anteriormente descargado y presionar sobre el botón ejecutar y seguimos los pasos del instalador.

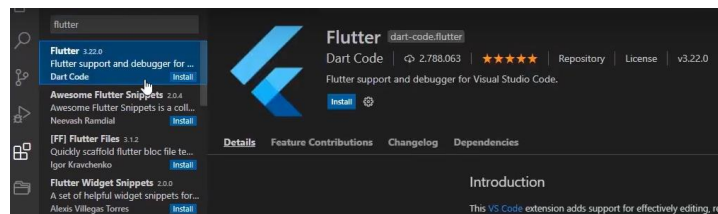


Configuración:

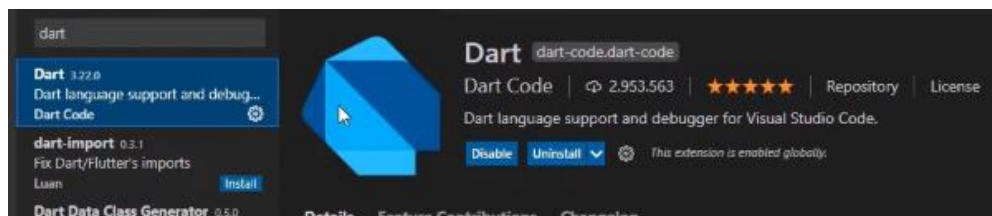
1. Ahora que tenemos instalado Visual Studio Code, tendremos que abrir el acceso directo que se nos genero en el escritorio para proceder a configurar lo necesario para usar Flutter



2. El siguiente paso será instalar el plugin de Flutter, por lo cual en la barra de búsqueda buscaremos Flutter e instalaremos la primera opción que nos aparecerá.



Una vez instalado automáticamente se instalará el plugin Dart



3. Con lo anterior podemos comprobar que todo se instaló correctamente ejecutando en una terminal de Visual Studio Code el comando flutter doctor.



```
David@DESKTOP-LG8702G MINGW64 ~
$ flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 2.2.0, on Microsoft Windows [Version 10.0.19042.985], locale es-ES)
[!] Android toolchain - develop for Android devices (Android SDK version 30.0.3)
    X Cannot execute C:\Program Files\Java\jdk-15.0.2\bin\java to determine the version
[✓] Chrome - develop for the web
[!] Android Studio (not installed)
[✓] VS Code (version 1.56.2)
[✓] Connected device (2 available)

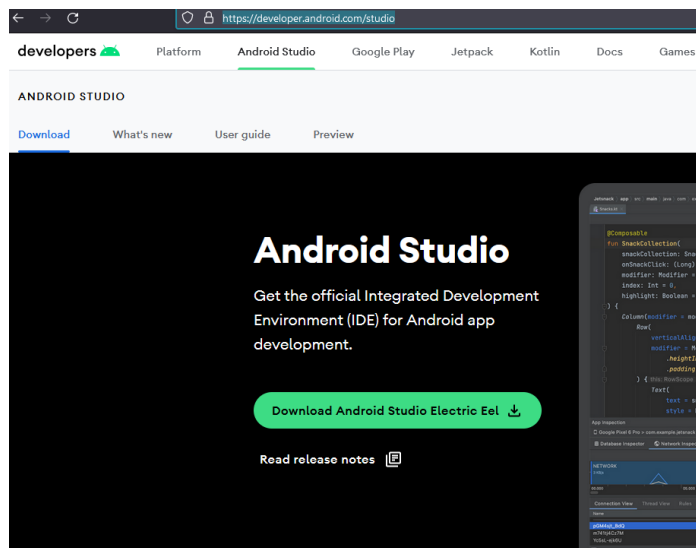
! Doctor found issues in 2 categories.

David@DESKTOP-LG8702G MINGW64 ~
$
```

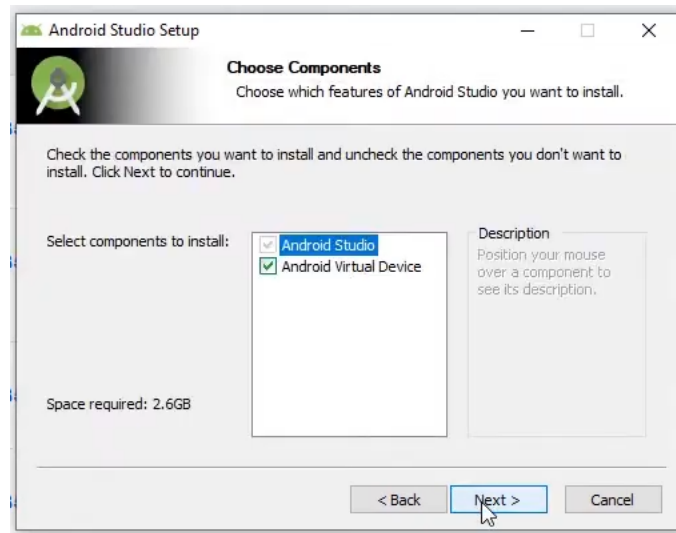
Android Studio

Instalación:

1. El recurso es descargado de la página oficial de Android Studio <https://developer.android.com/studio>



2. Lo siguiente será abrir el instalador anteriormente descargado y presionar sobre el botón ejecutar y seguimos los pasos del instalador.



Configuración:

1. Tendremos que abrir una consola en Visual Studio Code, si ejecutas el comando flutter doctor te daras cuenta de que no tenemos las licencias de Android Studio, por lo cual tendremos que aceptar las licencias ejecutando el comando flutter doctor --android-licenses, posteriormente pulsamos la tecla "Y" a todo lo que nos pregunte para aceptar.

```
David@DESKTOP-LG8702G MINGW64 ~  
$ flutter doctor --android-licenses  
4 of 7 SDK package licenses not accepted. 100% Computing updates...  
Review licenses that have not been accepted (y/N)?
```

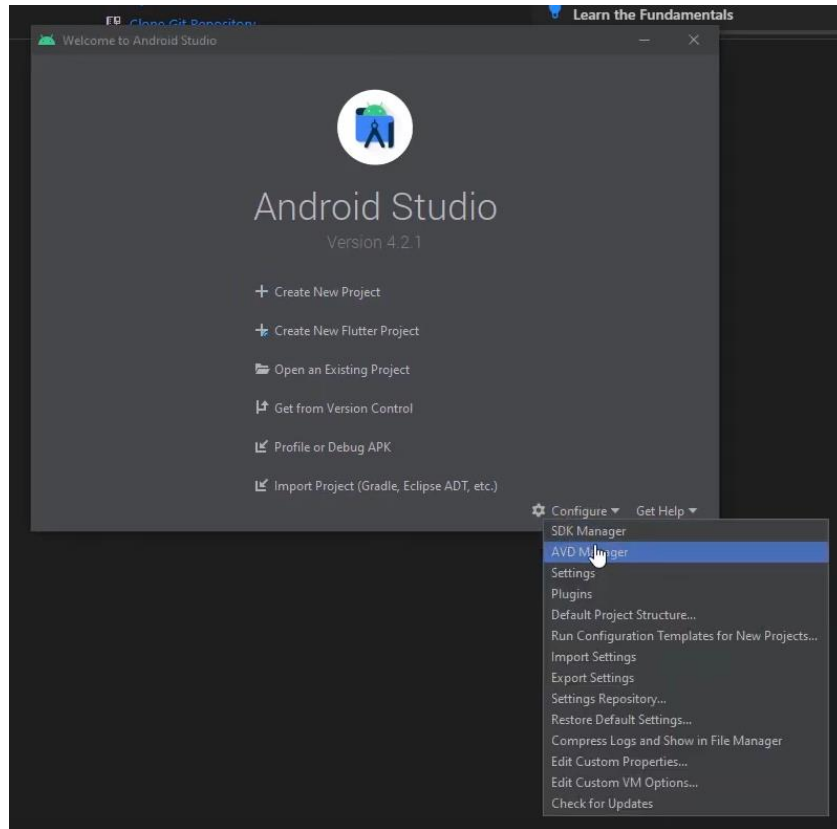
2. Si ejecutas flutter doctor te darás cuenta que ya no tienes problema con las licencias, pero aún no detectara Android Studio, para solucionarlo tendremos que ir a la carpeta de instalación de Android Studio y copiar la dirección y ejecutamos el siguiente comando y reiniciamos Visual Studio Code.

```
David@DESKTOP-LG8702G MINGW64 ~  
$ flutter config --android-studio-dir="C:\Program Files\Android\Android Studio"  
Setting "android-studio-dir" value to "C:\Program Files\Android\Android Studio".  
  
You may need to restart any open editors for them to read new settings.  
  
David@DESKTOP-LG8702G MINGW64 ~  
$
```

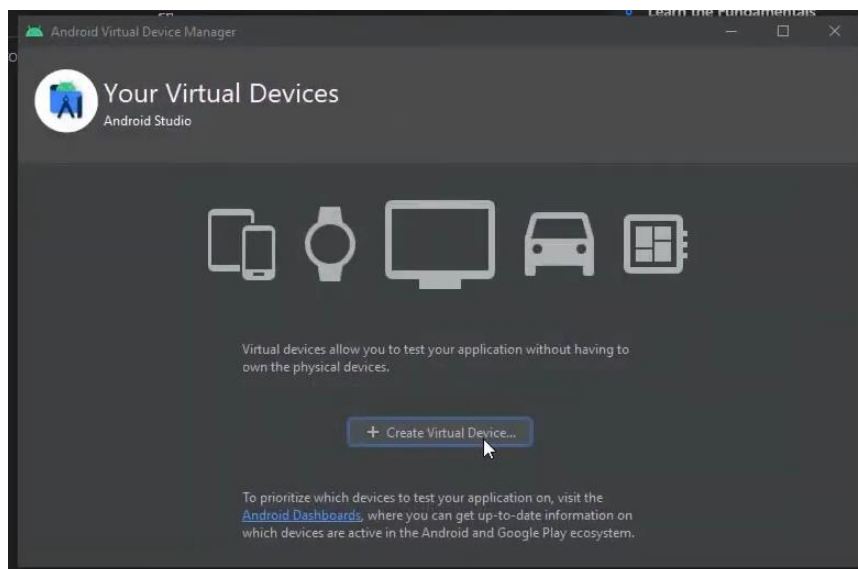
Ahora si ejecutamos el comando flutter doctor ya no tendremos ningún error.

```
David@DESKTOP-LG8702G MINGW64 ~  
$ flutter doctor -v  
[✓] Flutter (Channel stable, 2.2.0, on Microsoft Windows [Version 10.0.19042.985], locale es-ES)  
    • Flutter version 2.2.0 at C:\src\flutter  
    • Framework revision b22742018b (10 days ago), 2021-05-14 19:12:57 -0700  
    • Engine revision a9d88a4d18  
    • Dart version 2.13.0  
  
[✓] Android toolchain - develop for Android devices (Android SDK version 30.0.3)  
    • Android SDK at C:\Users\David\AppData\Local\Android\sdk  
    • Platform android-30, build-tools 30.0.3  
    • Java binary at: C:\Program Files\Android\Android Studio\jre\bin\java  
    • Java version OpenJDK Runtime Environment (build 11.0.8+10-b944.6842174)  
    • All Android licenses accepted.  
  
[✓] Chrome - develop for the web  
    • Chrome at C:\Program Files (x86)\Google\Chrome\Application\chrome.exe  
  
[✓] Android Studio  
    • Android Studio at C:\Program Files\Android\Android Studio  
    • Flutter plugin can be installed from:  
      https://plugins.jetbrains.com/plugin/9212-flutter  
    • Dart plugin can be installed from:  
      https://plugins.jetbrains.com/plugin/6351-dart  
    • android-studio-dir = C:\Program Files\Android\Android Studio  
    • Java version OpenJDK Runtime Environment (build 11.0.8+10-b944.6842174)  
  
[✓] VS Code (version 1.56.2)  
    • VS Code at C:\Users\David\AppData\Local\Programs\Microsoft VS Code  
    • Flutter extension version 3.22.0  
  
[✓] Connected device (2 available)  
    • Chrome (web) • chrome • web-javascript • Google Chrome 90.0.4430.212  
    • Edge (web)   • edge   • web-javascript • Microsoft Edge 90.0.818.66  
  
• No issues found!
```

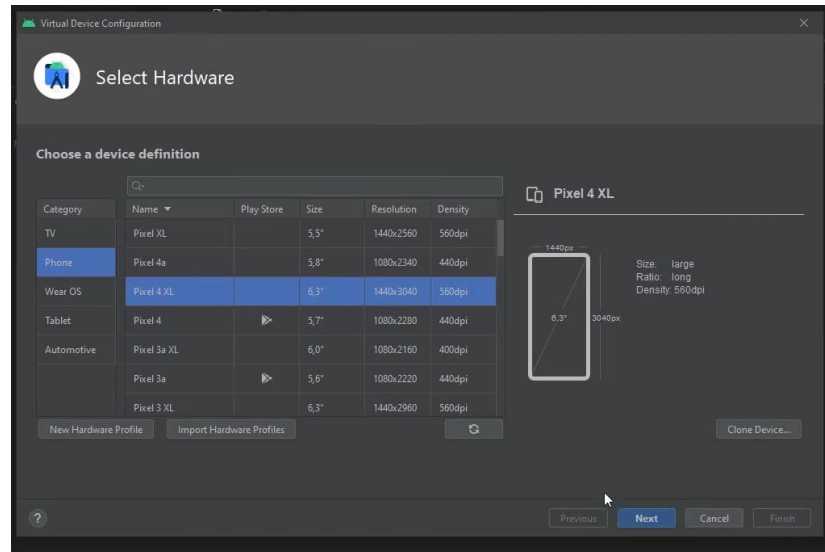
3. El siguiente paso será abrir Android Studio y pulsaremos sobre el botón de configuración y seleccionaremos la opción AVD Manager.



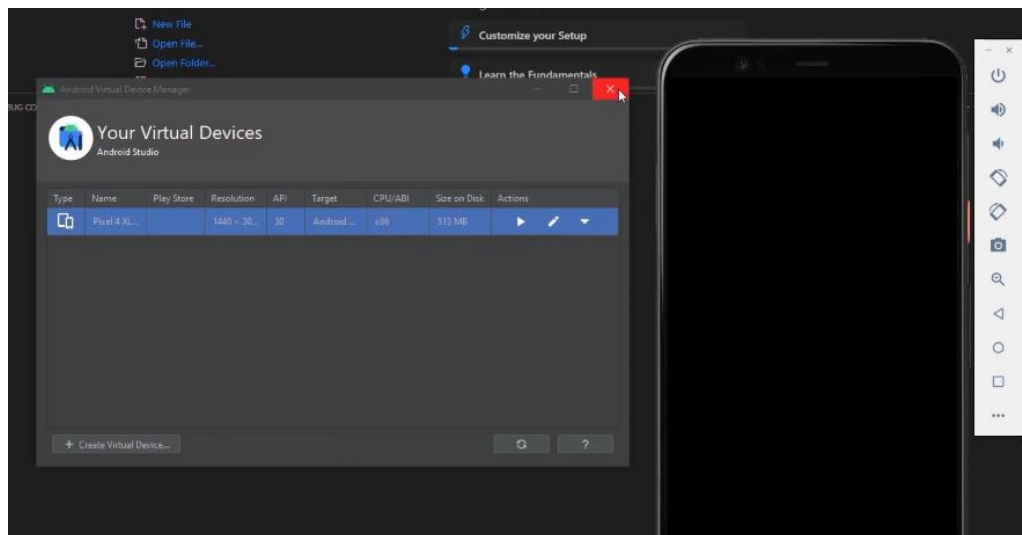
4. Creamos un nuevo dispositivo que será el que aparecerá cuando ejecutemos la aplicación.



5. Seleccionaremos el dispositivo que deseemos utilizar y pulsamos siguiente.

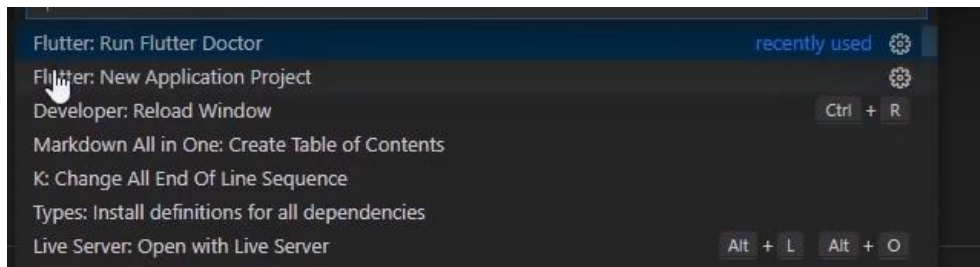


6. Una vez seleccionado damos finalizar y presionamos sobre el botón play para ejecutar.

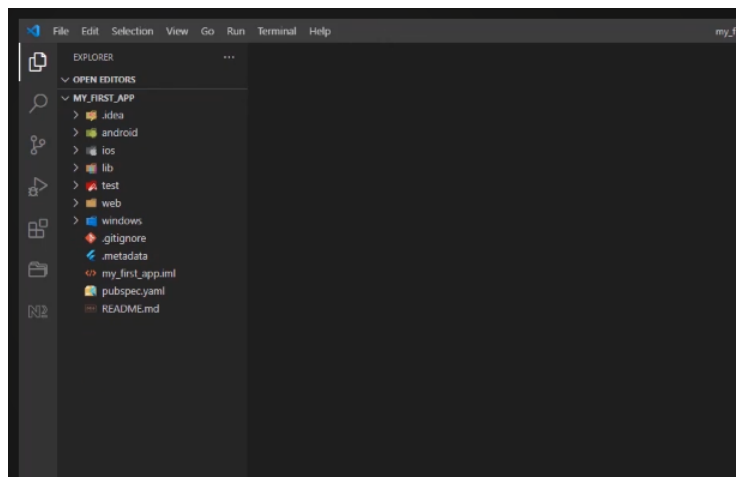


Creación de una aplicación:

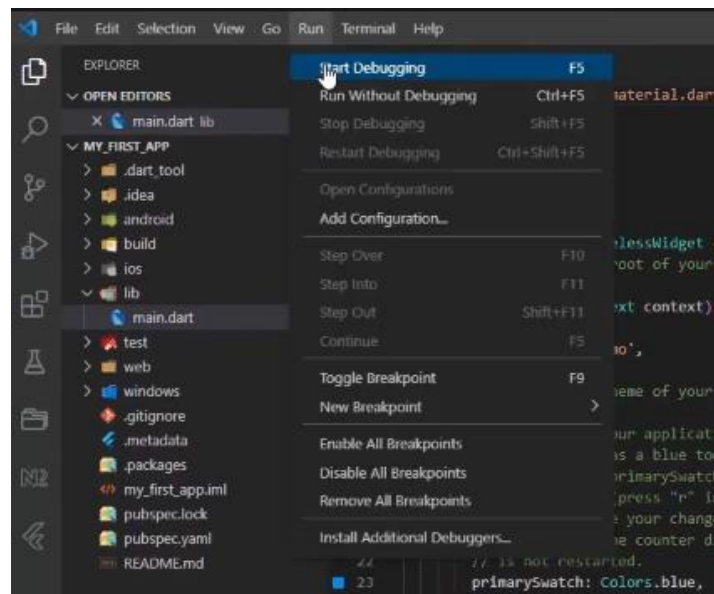
1. Nos dirigimos a Visual Studio Code y presionaremos la combinación de teclas Ctrl+Shift+P y seleccionaremos la opción Flutter: New Application Project, después escogemos la carpeta donde queremos guardar el proyecto y el nombre del proyecto.



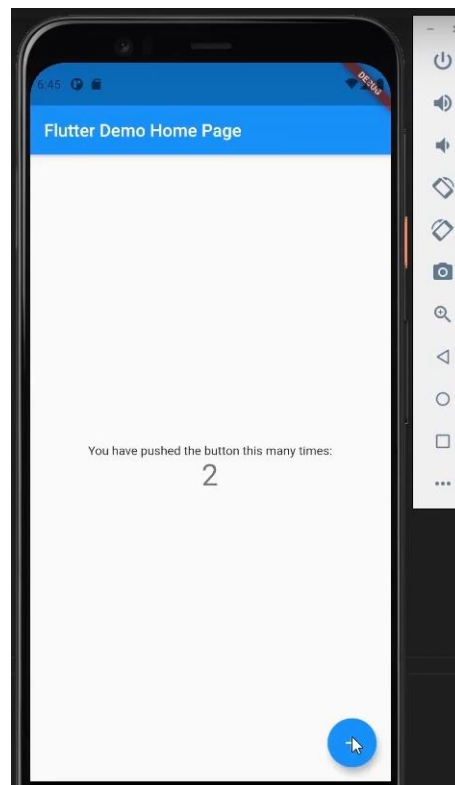
Se generará la estructura de un proyecto de Flutter.



2. Iremos al fichero main.dart que se encuentra en la carpeta lib, pulsamos sobre Run y después Start Debuggin.



Si abrimos el dispositivo y esperamos nos daremos cuenta que Flutter creó y ejecuto una aplicación automática.



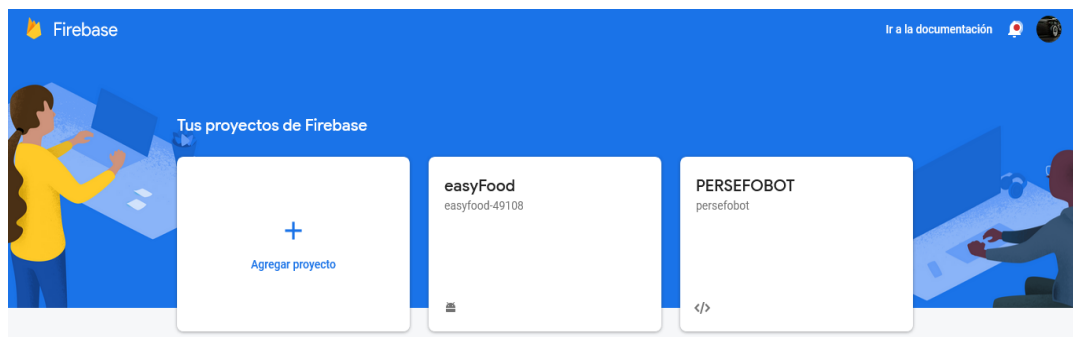
Flutter + FIREBASE

Configurar FIREBASE:

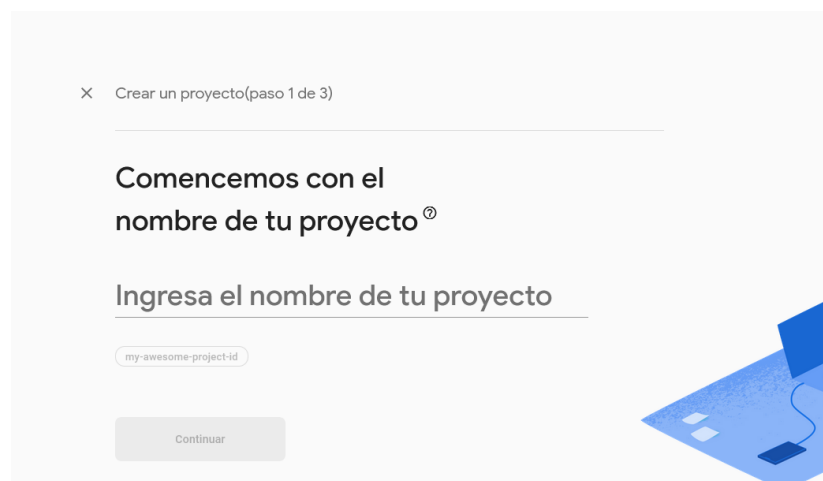
1. Ingresar a la página de FIREBASE:
<https://firebase.google.com>



2. Seleccionar "Ir a consola".



3. Ingresar un nombre a la aplicación.



X Crear un proyecto(paso 1 de 3)

Comencemos con el nombre de tu proyecto[?]

Ingresa el nombre de tu proyecto

my-awesome-project-id

Continuar



X Crear un proyecto(paso 3 de 3)

Configurar Google Analytics

Elige o crea una cuenta de Google Analytics[?]

Default Account for Firebase

Crear una cuenta nueva

Propiedad de Google Analytics en la cuenta de Google Analytics que elijas y se vinculará a tu proyecto de Firebase. Este vínculo permitirá el flujo de datos entre los productos. Los datos que se



Creando tu proyecto. Espera...

Conexion



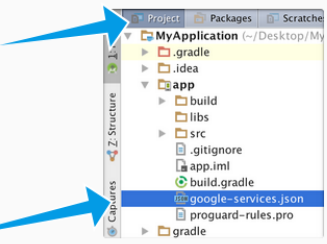
- Donde colocar este archivo: Android/app

2 Descargar y, luego, agregar el archivo de configuración para Android Studio a continuación | [Unity](#) [C++](#)

[Descargar google-services.json](#)

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

Mueve el archivo `google-services.json` descargado al directorio raíz de tu módulo (nivel de app).



`google-services.json`

[Siguiente](#)

6. Agregar SDK FIREBASE.

```
buildscript {
    repositories {
        // Make sure that you have the following two repositories
        google() // Google's Maven repository
        mavenCentral() // Maven Central repository
    }
    dependencies {
        ...
        // Add the dependency for the Google services Gradle plugin
        classpath 'com.google.gms:google-services:4.3.15'
    }
}

allprojects {
    ...
    repositories {
        // Make sure that you have the following two repositories
        google() // Google's Maven repository
        mavenCentral() // Maven Central repository
    }
}

plugins {
    id 'com.android.application'
    // Add the Google services Gradle plugin
    id 'com.google.gms.google-services'
    ...
}

dependencies {
    // Import the Firebase BoM
    implementation platform('com.google.firebase:firebase-bom:31.2.2')

    // TODO: Add the dependencies for Firebase products you want to use
    // When using the BoM, don't specify versions in Firebase dependencies
    implementation 'com.google.firebase:firebase-analytics-ktx'

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

7. Colocar: Android/app/build.gradle

```
apply plugin: 'com.android.application'
apply plugin: 'kotlin-android'
apply plugin: 'com.google.gms.google-services'
apply from: "$flutterRoot/packages/flutter_tools/gradle/flutter.gradle"
```

```
dependencies {
    implementation "org.jetbrains.kotlin:kotlin-stdlib-jdk7:$kotlin_version"
    implementation platform('com.google.firebase:firebase-bom:31.2.2')
}
```

8. Colocar en: pubspec.yaml

```
cupertino_icons: ^1.0.2
firebase_core: any
firebase_auth: any
```

9. Como importar FIREBASE en Flutter.

```
import 'package:firebase_core/firebase_core.dart';
```

10. Detectar la conexión en FIREBASE.

```
future: _initFirebase(),
```

```
Future<FirebaseApp> _initFirebase() async {
    FirebaseApp firebaseApp = await FirebaseApp.initializeApp();
    return firebaseApp;
}
```