

# Manual de instalación y configuración



## **Estudiantes:**

Mariliny Zambrana Jiménez

Axel Andrade Villalobos

Carlos Villalobos Baltodano

Yery Marín Ortiz

Brayan Picado Aguilar

## **Curso:**

Diseño y programación de  
plataformas móviles

## **Profesor:**

Carlos Escalante Solano

UNA Campus Sarapiquí

Ciclo I 2023

## Contenido

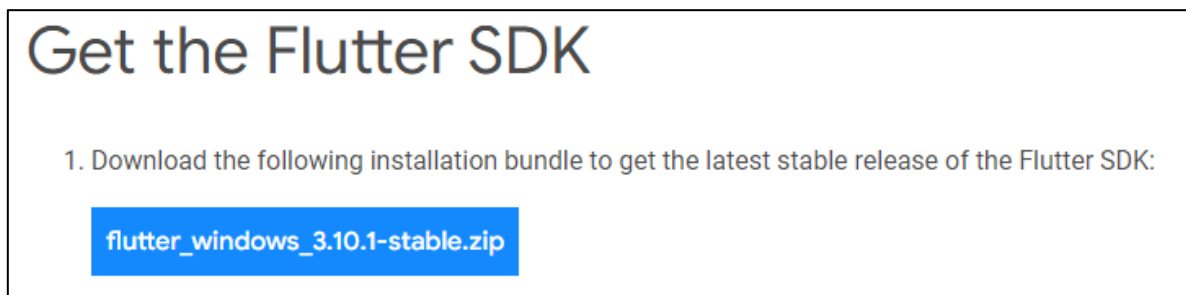
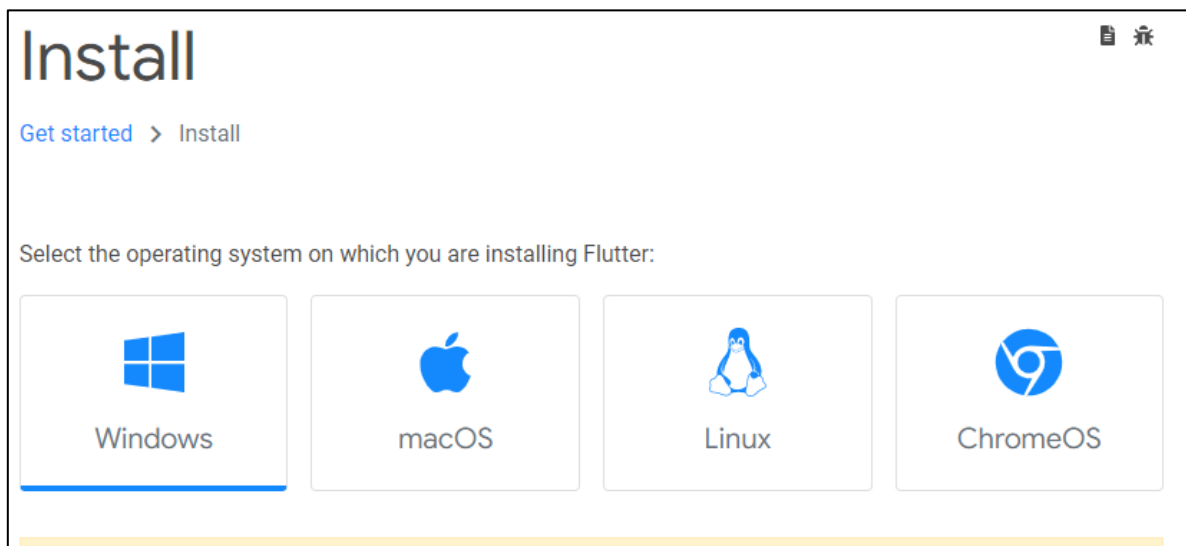
Paso 1 - Requisitos previos.....	3
Paso 2: Descargar Flutter .....	3
Paso 3: Extraer el archivo descargado .....	4
Paso 4: Configurar la variable de entorno.....	5
Paso 5: Verificar la instalación.....	6
Paso 6: Configurar un editor de código.....	7
Paso 7 – Crear y ejecutar un proyecto .....	8
Paso 9 – Creación de APK.....	9
Paso 10 – Sobre cómo configurar en Android Studio .....	9

## Paso 1 - Requisitos previos

- Asegurarse de tener una computadora con Windows, macOS o Linux.
- Verificar que el sistema cumpla con los requisitos mínimos de hardware y software para Flutter. Se pueden encontrar en la documentación oficial de Flutter.

## Paso 2: Descargar Flutter

- Visitar el sitio web oficial de **Flutter** en <https://docs.flutter.dev/get-started/install> y descargar la última versión estable para el sistema operativo. En este caso, se descargará para Windows.

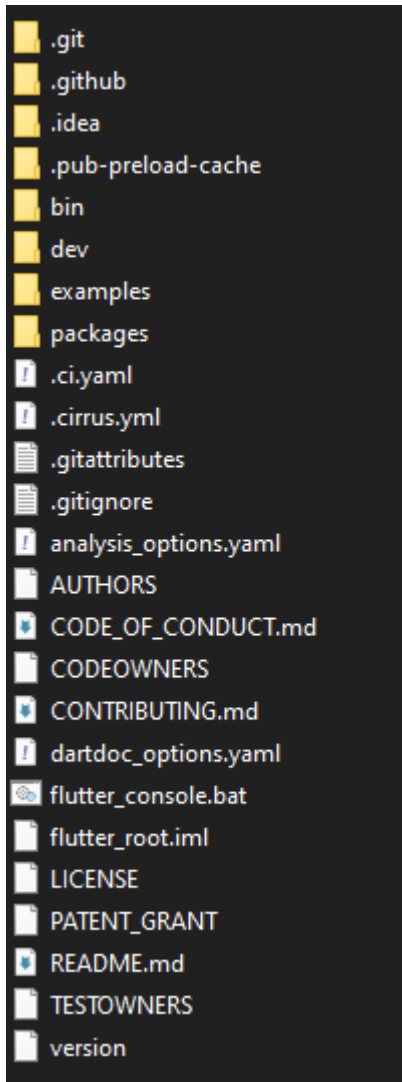


### Paso 3: Extraer el archivo descargado

- Descomprimir el archivo descargado en una ubicación conveniente del disco duro. Por ejemplo, extraerlo en **C:\flutter** (en Windows) o **/Users/tu-usuario/flutter** (en macOS o Linux).



Archivos de la carpeta:



## Paso 4: Configurar la variable de entorno

- Agregar la ruta de Flutter al **PATH** del sistema. Esto permitirá que se pueda ejecutar comandos de Flutter desde cualquier ubicación en tu terminal o línea de comandos.
- **En Windows:**
  - Abrir la configuración del sistema.
  - Buscar "**Variables de entorno**" y hacer clic en "**Editar las variables de entorno del sistema**".
  - En la sección "**Variables del sistema**", selecciona la variable "**Path**" y hacer clic en "**Editar**".
  - Agregar la ruta completa de **Flutter**, por ejemplo: **C:\flutter\bin**.
  - Guardar los cambios y cerrar la ventana de configuración.



C:\flutter\bin

- **En macOS y Linux:**
  - Abrir una terminal o línea de comandos.
  - Ejecutar el siguiente comando para abrir el archivo de perfil de tu usuario:  
**\$ nano ~/.bash\_profile**
  - Agregar la siguiente línea al archivo y guardar los cambios:  
**export PATH="\$PATH:/Users/tu-usuario/flutter/bin"**
  - Cerrar y volver a abrir la terminal o línea de comandos.

## Paso 5: Verificar la instalación

- Abrir una terminal o línea de comandos.
- Ejecutar el siguiente comando para verificar que Flutter se haya instalado correctamente:

### **flutter doctor**

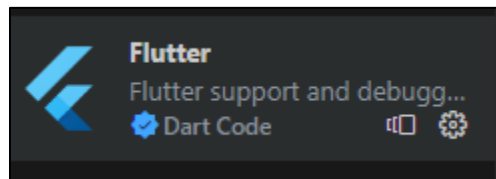
- Este comando analizará el sistema y mostrará si hay alguna configuración faltante o problemas conocidos. Si hay problemas, seguir las instrucciones proporcionadas para solucionarlos. Dependiendo de que editor o IDE se utilice se mostrará si carece de configuración o no para ejecutar correctamente flutter.

```
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 3.7.12, on Microsoft Windows [Version 10.0.19045.2965], locale es-CR)
[✗] Windows Version (Unable to confirm if installed Windows version is 10 or greater)
[!] Android toolchain - develop for Android devices (Android SDK version 33.0.2)
    ✗ cmdline-tools component is missing
      Run `path/to/sdkmanager --install "cmdline-tools;latest"`
      See https://developer.android.com/studio/command-line for more details.
    ✗ Android license status unknown.
      Run `flutter doctor --android-licenses` to accept the SDK licenses.
      See https://flutter.dev/docs/get-started/install/windows#android-setup for more details.
[✓] Chrome - develop for the web
[!] Visual Studio - develop for Windows (Visual Studio Community 2022 17.5.4)
    ✗ Visual Studio is missing necessary components. Please re-run the Visual Studio installer for the "Desktop
      development with C++" workload, and include these components:
        MSVC v142 - VS 2019 C++ x64/x86 build tools
          - If there are multiple build tool versions available, install the latest
        C++ CMake tools for Windows
        Windows 10 SDK
[✓] Android Studio (version 2022.2)
[✓] VS Code (version 1.78.2)
[✓] Connected device (3 available)
[✓] HTTP Host Availability
```

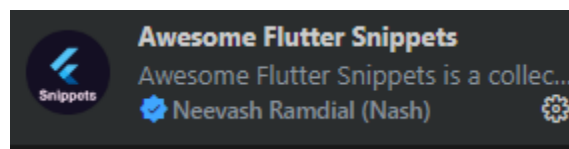
## Paso 6: Configurar un editor de código

**Flutter** es compatible con varios editores de código. Se puede elegir el que se prefiera, como **Visual Studio Code**, **Android Studio** o **IntelliJ IDEA**. Instalar el editor de tu elección y asegurarse de tener las extensiones o complementos de Flutter instalados. En este caso se configurará para **Visual Studio Code**:

- Instalar los plugins de Flutter y Dart
  - Abrir VSCode
  - Ir **View > Command Palette**
  - Escribir “install” y seleccionar “**Extensions: Install Extensions**”
  - Escribir luego “flutter” en las extensiones, seleccionar **Flutter** y presionar sobre **Install**. Esto también instalará el plugin de **Dart**.
- Igualmente, en consola, se puede validar la instalación con **Flutter Doctor**
  - Ir **View > Command Palette**
  - Escribir “doctor”, luego seleccionar **Flutter: Run Flutter Doctor**
  - Revisar el output para ver la información.



Otra extensión extra, con snippets para autocompletado más rápido (Ver más detalles en details de la extensión):

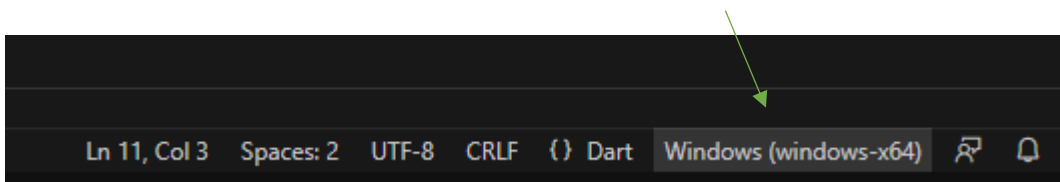


## Paso 7 – Crear y ejecutar un proyecto

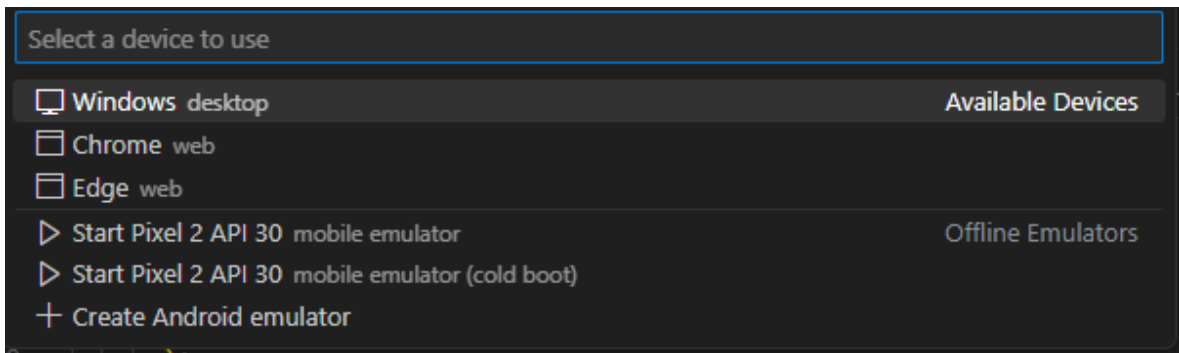
Con esto se ha completado la instalación y configuración de Flutter 2.10 en el sistema. Se puede empezar creando un nuevo proyecto Flutter con el comando **flutter create nombre-del-proyecto**. También se puede consultar la documentación oficial de **Flutter** para obtener más información sobre cómo comenzar con el desarrollo de aplicaciones.

Documentación y tutoriales: <https://docs.flutter.dev/reference/tutorials>

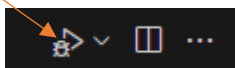
Para ejecutar el proyecto se puede elegir los dispositivos o emuladores, dando clic en la parte inferior derecha:



Inmediatamente va desplegar la lista:



En este caso, se usará Chrome. Para ejecutar solamente se presiona:



Y automáticamente, se abrirá el navegador mostrando la aplicación.



## Paso 9 – Creación de APK

Para poder generar un apk de Android, se hace uso de la terminal, usando el siguiente comando sobre la carpeta raíz del proyecto:

**flutter build apk**

Este comando compilará el proyecto y generará el archivo APK en la carpeta **"build/app/outputs/apk/release/app-release.apk"** del proyecto.

## Paso 10 – Sobre cómo configurar en Android Studio

Para más información sobre cómo configurar Flutter en Android Studio, se puede revisar la documentación en la página oficial de Flutter: <https://docs.flutter.dev/tools/android-studio>