

# Les outils indispensables pour Flutter

## Flutter :

Documentation officielle : <https://docs.flutter.dev/>

Télécharger : <https://docs.flutter.dev/get-started/install>

## Android Studio :

Android Studio est indispensable pour Flutter et langages mobiles en général. En effet, il permet de générer des simulateurs Android permettant de tester l'app. Il est aussi requis pour la gestion des Sdks.

Télécharger : <https://developer.android.com/studio>

## Xcode :

Xcode est nécessaire pour réaliser des tests sur iOS. C'est avec ce logiciel que l'on peut gérer des simulateurs iOS.

Télécharger : <https://developer.apple.com/xcode/>

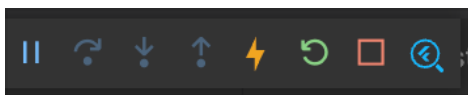
## Choisir son IDE :

Les deux meilleurs options, sont soit de choisir Android Studio soit partir sur VS code. C'est selon vos préférence, il est mieux de choisir un IDE sur lequel on se sent à l'aise.

Il est plus simple de partir sur Android Studio étant donné qu'il nécessairement installé pour gérer vos appareils virtuels.

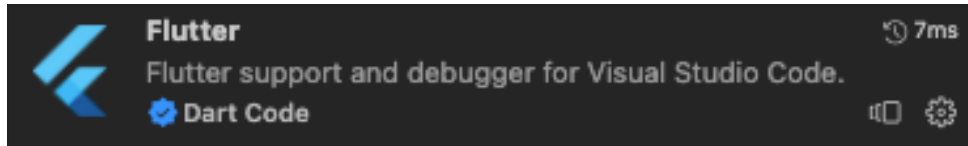
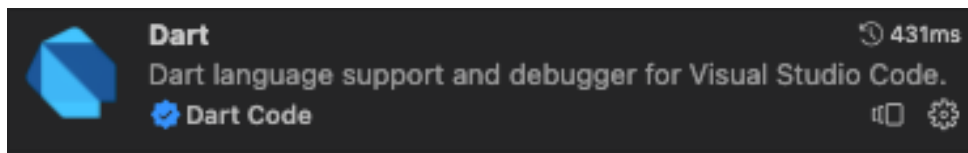
Pour autant VS code est aussi un très bon IDE pour travailler avec Flutter.

Tous deux propose la possibilité de hot reload : C'est la fonction la plus simple et la plus rapide qui vous aide à appliquer des modifications, à corriger des bogues, à créer des interfaces utilisateur et à ajouter des fonctionnalités. Il faut environ une seconde pour exécuter sa fonctionnalité.

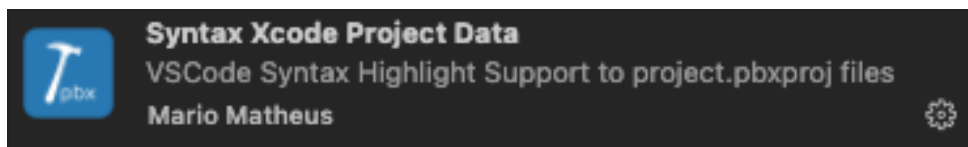
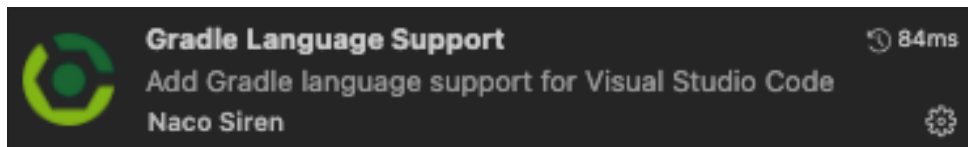


Concernant VS code, il est nécessaire d'ajouter des packages supplémentaires pour une utilisation plus agréable de Flutter. En effet, il est nécessaire d'ajouter

des extensions Dart et Flutter, cela vous facilitera grandement la vie.



D'autres extension sont recommandées tel que Gradle Language Support pour votre gestion de build côté Android. Ou encore Syntax Xcode Project Data permettant d'intégrer project.pbxproj dans Visual Studio Code permettant la colorisation et la connaissance de la grammaire.



D'autres extensions peuvent être intéressantes tel que Xml Tools, Kotlin, SVG...

## fvm / sidekick :

Fvm (Flutter Version Management) répond au besoin de builds d'applications cohérentes en référençant la version du SDK Flutter utilisée pour chaque projet. Il vous permet également d'installer plusieurs versions de Flutter pour valider et tester rapidement les prochaines versions de Flutter avec vos applications sans attendre l'installation de Flutter à chaque fois.

Documentation fvm : [https://fvm.app/docs/getting\\_started/overview](https://fvm.app/docs/getting_started/overview)

Associé à FVM, Sidekick vous permet de gérer et de mettre en cache plusieurs versions du SDK Flutter, de basculer rapidement entre les canaux et les versions et d'afficher les canaux et les versions disponibles. De plus, Sidekick vous permet de configurer et de spécifier une version du SDK Flutter par projet. Les chemins SDK dynamiques pour le débogage IDE sont pris en charge, ce qui permet une cohérence entre les équipes et les environnements CI et vous permet de gérer les versions globales du SDK Flutter, en engendrant des processus dans n'importe quelle version du SDK Flutter avec Playground.

Non seulement cela, mais Sidekick agit comme votre hub Flutter central, vous permettant d'avoir tous vos projets en un seul endroit et de les lancer dans votre IDE préféré en un seul clic.

## vérification de l'installation de flutter

flutter doctor -v

Cette commande vérifie votre environnement et affiche un rapport sur l'état de votre installation Flutter. Vérifiez attentivement la sortie pour d'autres logiciels que vous pourriez avoir besoin d'installer ou d'autres tâches à effectuer

```
al@al-macbook-pro:~$ flutter doctor -v
[✓] Flutter (Channel stable, 3.3.9, on macOS 12.6.1 21G217 darwin-x64, locale fr-FR)
    • Flutter version 3.3.9 on channel stable at /Users/abel.de-smedt/fvm/versions/3.3.9
    • Upstream repository https://github.com/flutter/flutter.git
    • Framework revision b8f7f1f986 (il y a 2 semaines), 2022-11-23 06:43:51 +0900
    • Engine revision 8f2221fbef
    • Dart version 2.18.5
    • DevTools version 2.15.0

[✓] Android toolchain - develop for Android devices (Android SDK version 33.0.0)
    • Android SDK at /Users/abel.de-smedt/Library/Android/sdk
    • Platform android-33, build-tools 33.0.0
    • Java binary at: /Applications/Android Studio.app/Contents/jre/Contents/Home/bin/java
    • Java version OpenJDK Runtime Environment (build 11.0.12+0-b1504.28-7817840)
    • All Android licenses accepted.

[✓] Xcode - develop for iOS and macOS (Xcode 14.1)
    • Xcode at /Applications/Xcode.app/Contents/Developer
    • Build 14B47b
    • CocoaPods version 1.11.3

[✓] Chrome - develop for the web
    • Chrome at /Applications/Google Chrome.app/Contents/MacOS/Google Chrome

[✓] Android Studio (version 2021.2)
    • Android Studio at /Applications/Android Studio.app/Contents
    • 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
    • Java version OpenJDK Runtime Environment (build 11.0.12+0-b1504.28-7817840)

[✓] VS Code (version 1.47.3)
    • VS Code at /Applications/Visual Studio Code.app/Contents
    • Flutter extension version 3.54.0

[✓] VS Code (version 1.70.0)
    • VS Code at /Users/abel.de-smedt/Downloads/Visual Studio Code.app/Contents
    • Flutter extension version 3.54.0

[✓] Connected device (4 available)
    • sdk gphone64 x86_64 (mobile) • emulator-5554 • android-x64 • Android 12 (API 31) (emulator)
    • iPhone14 (mobile) • 97383F57-C559-4A2A-8E3A-8D3F62BCAFB2 • ios • com.apple.CoreSimulator.SimRuntime.iOS-16-1 (simulator)
    • macOS (desktop) • macos • darwin-x64 • macOS 12.6.1 21G217 darwin-x64
    • Chrome (web) • chrome • web-javascript • Google Chrome 108.0.5359.94

[✓] HTTP Host Availability
    • All required HTTP hosts are available

• No issues found!
```