

## Description Report TP#1

*Report By Khertish Lobine*

23/10/23

*Université des Mascareignes*

*Faculty of Information and Communication Technology*

*Department of Applied Computer Science*

*Submitted to Mr. Shiam Beeharry*

Objectives :

1. Installation of Flutter, Vscode with Flutter, Android Studio
2. Screenshots and Description of Process





<b>Downloading and Installation of Flutter.....</b>	<b>4</b>
1. Downloading Flutter on <a href="https://docs.flutter.dev/get-started/install/windows">https://docs.flutter.dev/get-started/install/windows</a> .....	4
2. Extraction of file in drive C.....	4
3. Entering the advanced system settings and entering Environment Variables.....	5
4. Pasting the bin file in Flutter SDK in the System settings.....	5
5. Opening Command Prompt.....	6
6. Type flutter in CMD.....	6
7. Now set up Android Studio.....	7
8. Allow Android Studio to install.....	7
9. Completing Installation of Android SDK through Android Studio.....	8
10. Installation of HAXM file manually from <a href="https://github.com/intel/haxm/releases">https://github.com/intel/haxm/releases</a> and following steps on <a href="https://github.com/intel/haxm/wiki/Installation-Instructions-on-Windows">https://github.com/intel/haxm/wiki/Installation-Instructions-on-Windows</a> .....	8
11. Opening command prompt and setting up licenses using command flutter doctor --android-licenses.....	9
12. Using command flutter doctor to complete installation of flutter.....	10
<b>Testing by creating a Project.....</b>	<b>10</b>
1. Creating a project on VS code.....	10



# Downloading and Installation of Flutter

## 1. Downloading Flutter on <https://docs.flutter.dev/get-started/install/windows>

The screenshot shows the Flutter documentation page for Windows installation. The page is titled "System requirements" and lists the minimum requirements for running Flutter. It also provides instructions on how to get the Flutter SDK, including downloading the installation bundle and extracting it.

**System requirements**

To install and run Flutter, your development environment must meet these minimum requirements:

- Operating Systems:** Windows 10 or later (64-bit), x86-64 based.
- Disk Space:** 2.5 GB (does not include disk space for IDE/tools).
- Tools:** Flutter depends on these tools being available in your environment.
  - Windows PowerShell 5.0 or newer (this is pre-installed with Windows 10)
  - Git for Windows 2.x, with the **Use Git from the Windows Command Prompt** option.

If Git for Windows is already installed, make sure you can run `git` commands from the command prompt or PowerShell.

**Get the Flutter SDK**

**Important:** If you're in China, read [Using Flutter in China](#).

1. Download the following installation bundle to get the latest stable release of the Flutter SDK:

[flutter\\_windows\\_3.13.0-stable.zip](#)

For other release channels, and older builds, check out the [SDK archive](#).

2. Extract the zip file and place the contained `flutter` in the desired installation location for the Flutter SDK (for example, `%USERPROFILE%\Flutter;D:\dev\Flutter`).

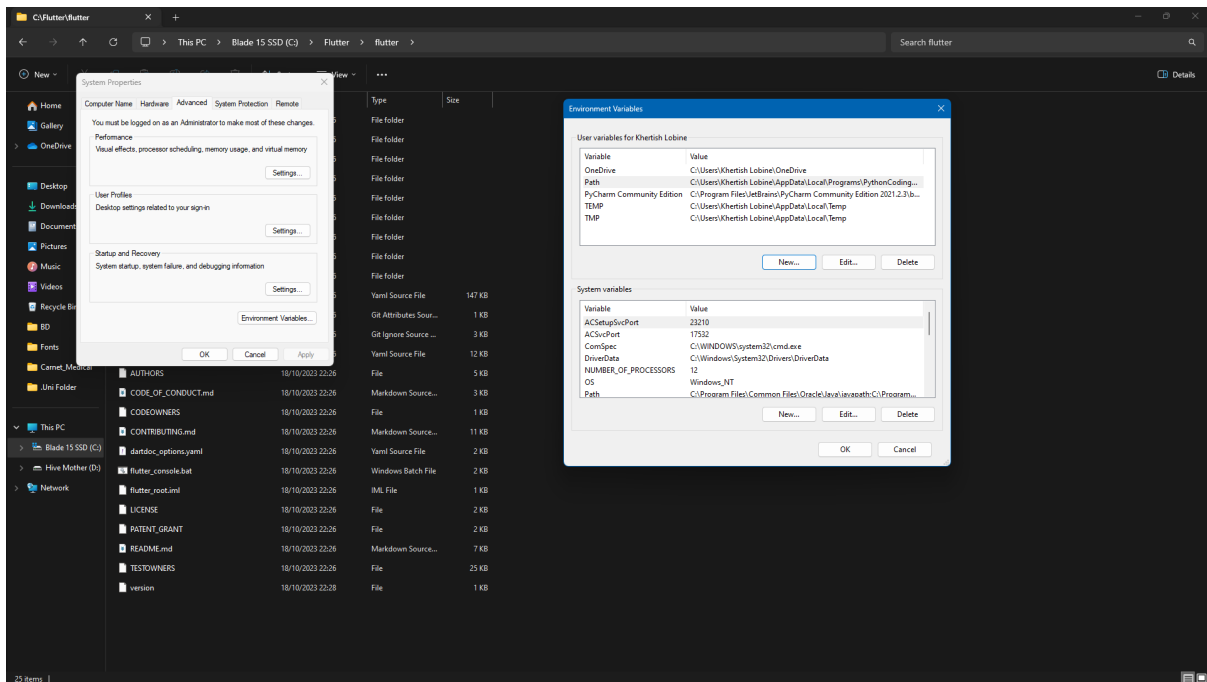
Google uses cookies to deliver its services, to personalize ads, and to analyze traffic. You can adjust your privacy controls anytime in your [Google settings](#). [Learn more](#).

## 2. Extraction of file in drive C

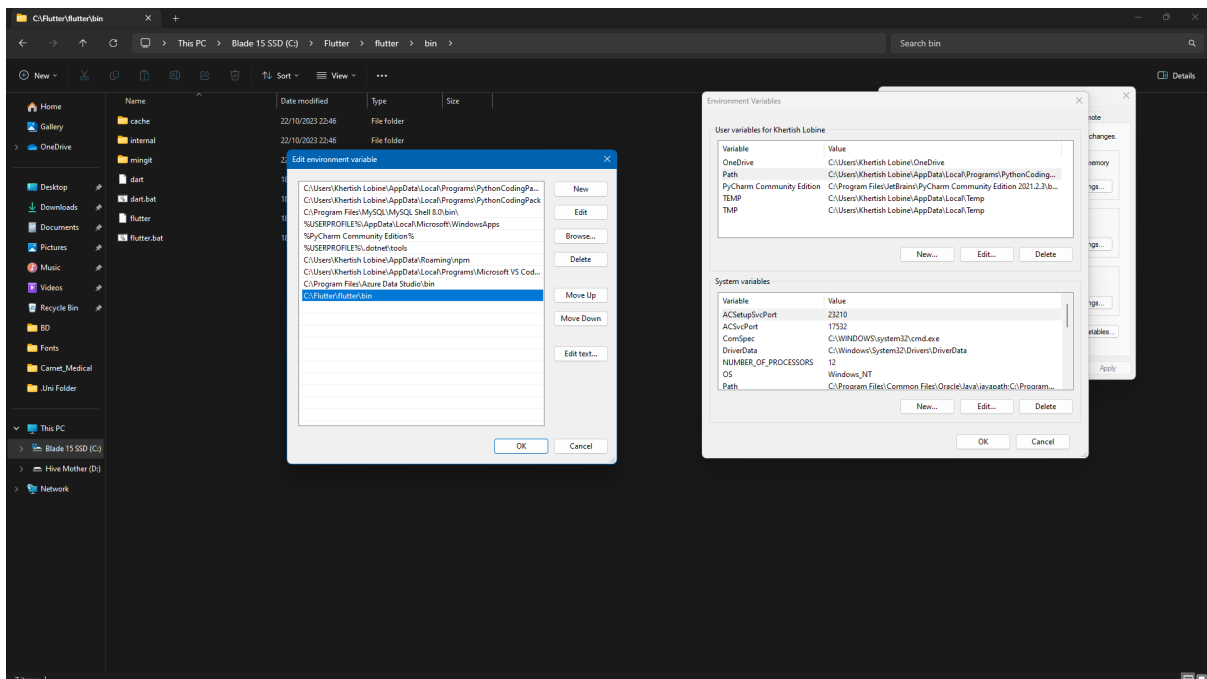
The screenshot shows a Windows File Explorer window displaying the contents of the `C:\Flutter\flutter` directory. The directory contains various files and folders related to the Flutter SDK, including source code, documentation, and build files.

Name	Date modified	Type	Size
git	22/10/2023 22:45	File folder	
github	22/10/2023 22:45	File folder	
idea	22/10/2023 22:45	File folder	
pub-preload-cache	22/10/2023 22:45	File folder	
vscode	22/10/2023 22:45	File folder	
bin	22/10/2023 22:46	File folder	
dev	22/10/2023 22:46	File folder	
examples	22/10/2023 22:46	File folder	
packages	22/10/2023 22:46	File folder	
.ci.yaml	18/10/2023 22:26	Yaml Source File	147 KB
gitattributes	18/10/2023 22:26	Git Attributes Sour...	1 KB
.gitignore	18/10/2023 22:26	Git Ignore Source ...	3 KB
analysis_options.yaml	18/10/2023 22:26	Yaml Source File	12 KB
AUTHORS	18/10/2023 22:26	File	5 KB
CODE_OF_CONDUCT.md	18/10/2023 22:26	Markdown Source...	3 KB
CODEOWNERS	18/10/2023 22:26	File	1 KB
CONTRIBUTING.md	18/10/2023 22:26	Markdown Source...	11 KB
dartdoc_options.yaml	18/10/2023 22:26	Yaml Source File	2 KB
flutter_console.bat	18/10/2023 22:26	Windows Batch File	2 KB
flutter_root.ini	18/10/2023 22:26	INI File	1 KB
LICENSE	18/10/2023 22:26	File	2 KB
PATENT_GRANT	18/10/2023 22:26	File	2 KB
README.md	18/10/2023 22:26	Markdown Source...	7 KB
TESTOWNERS	18/10/2023 22:26	File	25 KB
version	18/10/2023 22:26	File	1 KB

### 3. Entering the advanced system settings and entering Environment Variables



### 4. Pasting the bin file in Flutter SDK in the System settings





## 5. Opening Command Prompt

```
Command Prompt
Microsoft Windows [Version 10.0.22621.2428]
(c) Microsoft Corporation. All rights reserved.
C:\Users\Whertish Lobine>
```

## 6. Type flutter in CMD

```
Command Prompt - flutter
C:\Users\Whertish Lobine>flutter
Manage your Flutter app development.

Common commands:

  flutter create <output directory>
    Create a new Flutter project in the specified directory.

  flutter run [options]
    Run your Flutter application on an attached device or in an emulator.

Usage: flutter <command> [arguments]

Global options:
-h, --help                Print this usage information.
-v, --verbose              Noisy logging, including all shell commands executed.
                           If used with "--help", shows hidden options. If used with "flutter doctor", shows additional
                           diagnostic information. (Use "-vv" to force verbose logging in those cases.)
-d, --device-id            Target device id or name (prefixes allowed).
--version                 Reports the version of this tool.
--suppress-analytics       Suppress analytics reporting for the current CLI invocation.
--disable-telemetry        Disable telemetry reporting each time a Flutter or dart command runs, until it is
                           re-enabled.
--enable-telemetry         Enable telemetry reporting each time a Flutter or dart command runs.

Available commands:

Flutter SDK
bash-completion           Output command line shell completion setup scripts.
channel                   List or switch Flutter channels.
config                     Configure Flutter settings.
doctor                     Show information about the installed tooling.
downgrade                 Downgrade Flutter to the last active version for the current channel.
precache                  Populate the Flutter tool's cache of binary artifacts.
upgrade                    Upgrade your copy of Flutter.

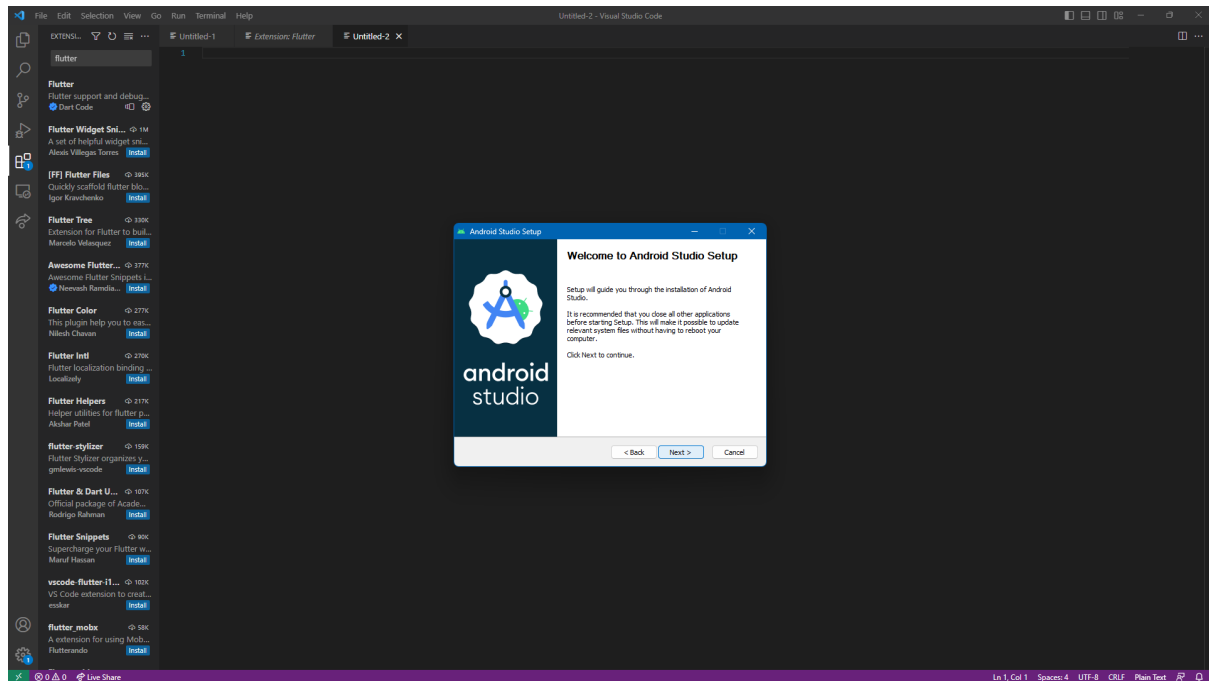
Project
analyze                   Analyze the project's Dart code.
assemble                  Assemble and build Flutter resources.
build                      Build an executable app or install bundle.
clean                      Delete the build/ and .dart_tool/ directories.
create                     Create a new Flutter project.
drive                      Run integration tests for the project on an attached device or emulator.
gen-l10n                   Generate localizations for the current project.
pub                       Commands for managing Flutter packages.
run                        Run your Flutter app on an attached device.
test                       Run Flutter unit tests for the current project.

Tools & Devices
attach                     Attach to a running app.
custom-devices             List, reset, add and delete custom devices.
devices                    List all connected devices.
emulators                  List, launch and create emulators.
install                     Install a Flutter app on an attached device.
```

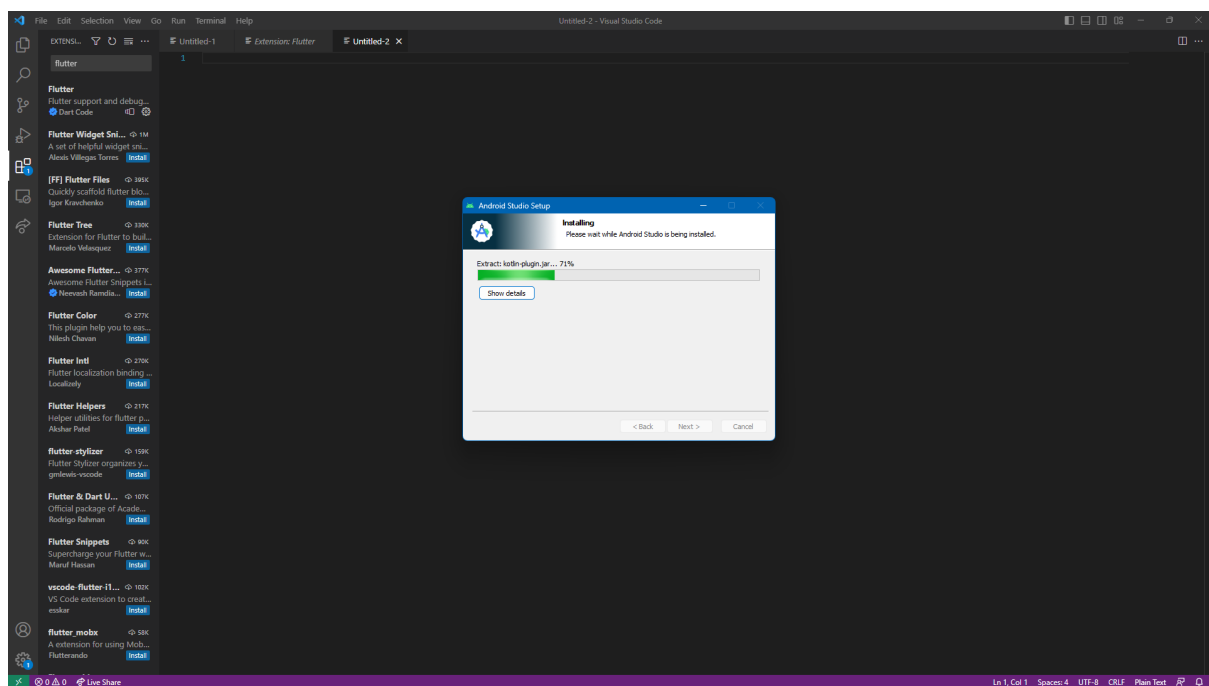
Now flutter is set up as an environment variable that can execute using command prompt



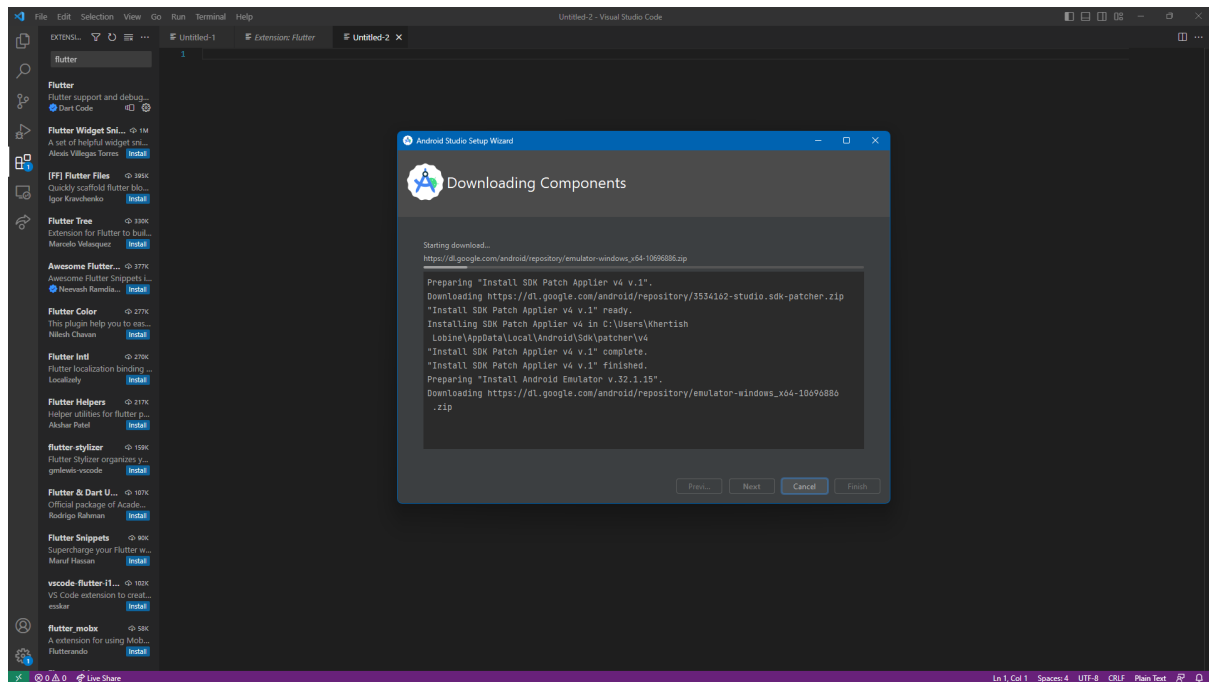
## 7. Now set up Android Studio



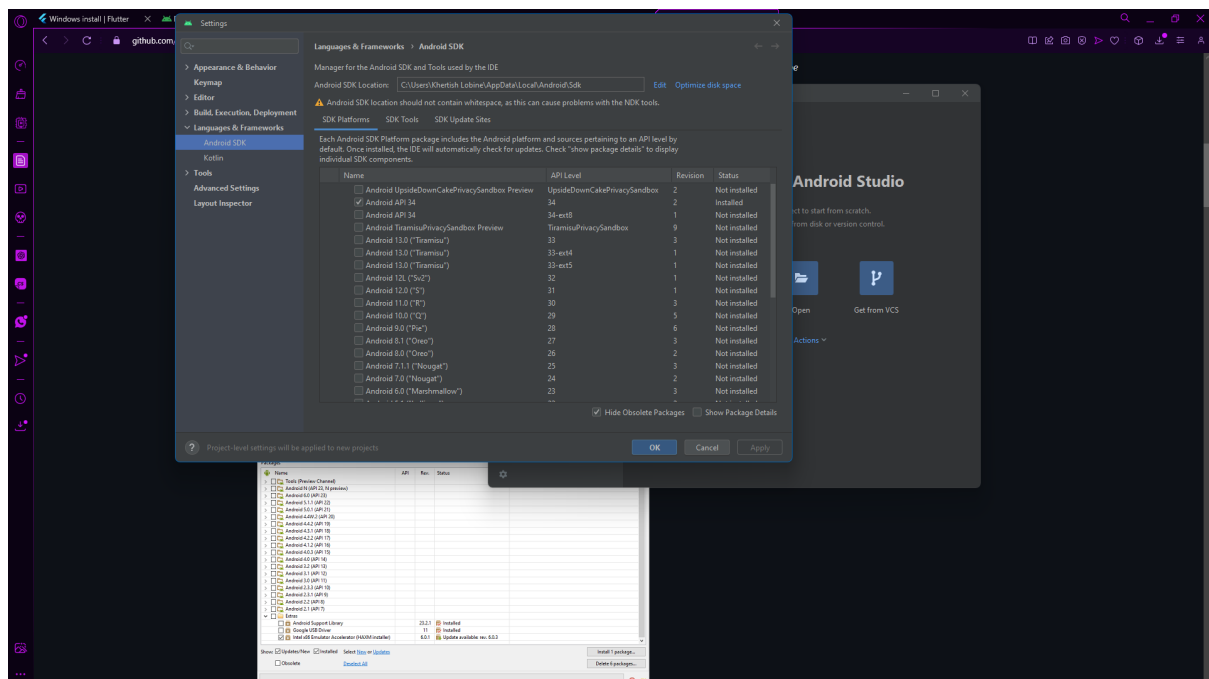
## 8. Allow Android Studio to install



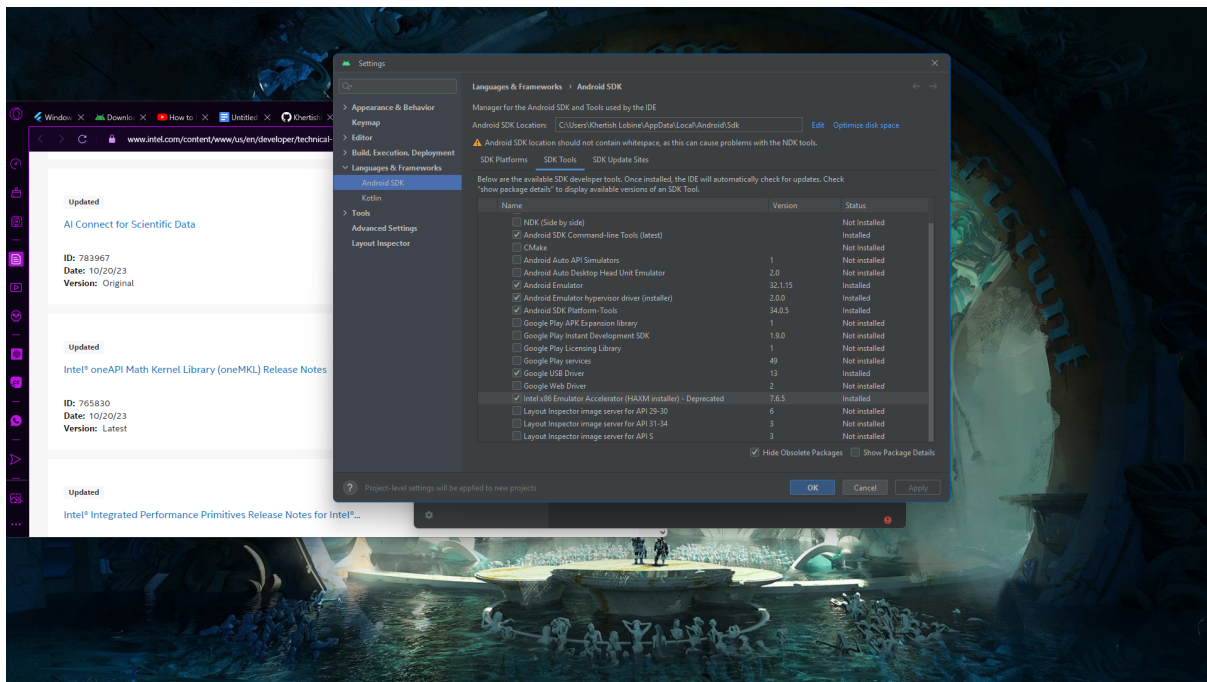
## 9. Completing Installation of Android SDK through Android Studio



## 10. Installation of HAXM file manually from <https://github.com/intel/haxm/releases> and following steps on <https://github.com/intel/haxm/wiki/Installation-Instructions-on-Windows>







## 11. Opening command prompt and setting up licences using command flutter doctor --android-licenses

```

Command Prompt - flutter d
C:\Users\Kherish Lobine>flutter doctor --android-licenses
Warning: Observed package id 'extras;intel;Hardware_Accelerated_Execution_Manager' in inconsistent location 'C:\Users\Kherish Lobine\AppData\Local\Android\Sdk\extras\intel\Hardware_Accelerated_Execution_Manag
er-2' (Expected 'C:\Users\Kherish Lobine\AppData\Local\Android\Sdk\extras\intel\Hardware_Accelerated_Execution_Manager')
Warning: Observed package id 'extras;intel;Hardware_Accelerated_Execution_Manager' in inconsistent location 'C:\Users\Kherish Lobine\AppData\Local\Android\Sdk\extras\intel\Hardware_Accelerated_Execution_Manag
er-2' (Expected 'C:\Users\Kherish Lobine\AppData\Local\Android\Sdk\extras\intel\Hardware_Accelerated_Execution_Manager')
[=====] 100% Computing updates...
4 of 7 SDK package licenses not accepted.
Review licenses that have not been accepted (y/N)? y

1/4: License android-googletv-license:
Terms and Conditions

This is the Google TV Add-on for the Android Software Development Kit License Agreement.

1. Introduction

1.1 The Google TV Add-on for the Android Software Development Kit (referred to in this License Agreement as the "Google TV Add-on" and specifically including the Android system files, packaged APIs, and Google
APIs add-ons) is licensed to you subject to the terms of this License Agreement. This License Agreement forms a legally binding contract between you and Google in relation to your use of the Google TV Add-on.

1.2 "Google" means Google Inc., a Delaware corporation with principal place of business at 1600 Amphitheatre Parkway, Mountain View, CA 94043, United States.

2. Accepting this License Agreement

2.1 In order to use the Google TV Add-on, you must first agree to this License Agreement. You may not use the Google TV Add-on if you do not accept this License Agreement.

2.2 You can accept this License Agreement by:

(A) clicking to accept or agree to this License Agreement, where this option is made available to you; or

(B) by actually using the Google TV Add-on. In this case, you agree that use of the Google TV Add-on constitutes acceptance of the License Agreement from that point onwards.

2.3 You may not use the Google TV Add-on and may not accept the Licensing Agreement if you are a person barred from receiving the Google TV Add-on under the laws of the United States or other countries includ
ng the country in which you are resident or from which you use the Google TV Add-on.

2.4 If you are agreeing to be bound by this License Agreement on behalf of your employer or other entity, you represent and warrant that you have full legal authority to bind your employer or such entity to th
is License Agreement. If you do not have the requisite authority, you may not accept the Licensing Agreement or use the Google TV Add-on on behalf of your employer or other entity.

3. Google TV Add-on License from Google

3.1 Subject to the terms of this License Agreement, Google grants you a limited, worldwide, royalty-free, non-assignable and non-exclusive license to use the Google TV Add-on solely to develop applications to
run on the Google TV platform.

3.2 You agree that Google or third parties own all legal right, title and interest in and to the Google TV Add-on, including any Intellectual Property Rights that subsist in the Google TV Add-on. "Intellectual
Property Rights" means any and all rights under patent law, copyright law, trade secret law, trademark law, and any and all other proprietary rights. Google reserves all rights not expressly granted to you.

3.3 Except to the extent required by applicable third party licenses, you may not copy (except for backup purposes), modify, adapt, redistribute, decompile, reverse engineer, disassemble, or create derivative
works of the Google TV Add-on or any part of the Google TV Add-on. Except to the extent required by applicable third party licenses, you may not load any part of the Google TV Add-on onto a mobile handset, tel
evision, or any other hardware device except a personal computer, combine any part of the Google TV Add-on with other software, or distribute any software or device incorporating a part of the Google TV Add-on

3.4 Use, reproduction and distribution of components of the Google TV Add-on licensed under an open source software license are governed solely by the terms of that open source software license and not this Li
cense Agreement.
  
```



## 12. Using command flutter doctor to complete installation of flutter

```

Command Prompt - flutter d x + v
ANY PRE-RELEASE MATERIALS ARE NON-QUALIFIED AND, AS SUCH, ARE PROVIDED *AS IS* AND *AS AVAILABLE*, POSSIBLY WITH FAULTS, AND WITHOUT REPRESENTATION OR WARRANTY OF ANY KIND.

10.8 Open Source Software. In the event Open Source software is included with Evaluation Software, such Open Source software is licensed pursuant to the applicable Open Source software license agreement identified in the Open Source software comments in the applicable source code file(s) and/or file header as indicated in the Evaluation Software. Additional detail may be available (where applicable) in the accompanying on-line documentation. With respect to the Open Source software, nothing in this Agreement limits any rights under, or grants rights that supersede, the terms of any applicable Open Source software license agreement.
-----
Accept? (y/N): y
All SDK package licenses accepted

C:\Users\Khertish Lobine>flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 2.12.0, on Microsoft Windows [Version 10.0.22621.2428], locale en-MU)
[✓] Windows Version (Installed version of Windows is version 10 or higher)
[✓] Android toolchain - develop for Android devices (Android SDK version 34.0.0)
[✓] Chrome - develop for the web
[✓] Visual Studio - develop Windows apps (Visual Studio Community 2022 17.9.0)
[✓] Android Studio (version 2022.3)
[✓] VS Code (version 1.81.1)
[✓] Connected device (3 available)
[✓] Network resources

• No issues found!

C:\Users\Khertish Lobine>

```

## Testing by creating a Project

### 1. Creating a project on VS code

```

File Edit Selection View Go Run Terminal Help
mandant - project_1_phm - Visual Studio Code

EXPLORER
main.dart x
main.dart
PROJECT_1_PHM
> .dart_tool
> .idea
> .ios
> .lib
main.dart
> linux
> macos
> test
> web
> windows
> .gitignore
E .metadata
F analysis_options.yaml
F project_1_phm.xml
F pubspec.lock
F pubspec.yaml
Q README.md

main.dart
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(const MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8   const MyApp({super.key});
9
10  // This widget is the root of your application.
11  @override
12  Widget build(BuildContext context) {
13    return MaterialApp(
14      title: 'Flutter Demo',
15      theme: ThemeData(
16        // This is the theme of your application.
17        //
18        // TRY THIS: Try running your application with "flutter run". You'll see
19        // the application has a blue toolbar. Then, without quitting the app,
20        // try changing the seedColor in the colorsScheme below to Colors.green
21        // and then invoke "hot reload" (save your changes or press the "hot
22        // reload" button in a Flutter-supported IDE, or press "r" if you used
23        // the command line to start the app).
24        //
25        // Notice that the counter didn't reset back to zero; the application
26        // state is not lost during the reload. To reset the state, use hot
27        // restart instead.
28        //
29        // This works for code too, not just values: Most code changes can be
30        // tested with just a hot reload.
31        colorsScheme: ColorsScheme.fromSeed(seedColor: Colors.deepPurple),
32        useMaterial3: true,
33      ),
34      home: const MyHomePage(title: 'Flutter Demo Home Page'),
35    );
36  }
37
38  class MyHomePage extends StatefulWidget {
39    const MyHomePage({super.key, required this.title});
40
41    // This widget is the home page of your application. It is stateful, meaning
42    // that it has a State object (defined below) that contains fields that affect
43    // how it looks.
44
45    // This class is the configuration for the state. It holds the values (in this
46    // case the title) provided by the parent (in this case the App widget) and
47    // used by the build method of the State. Fields in a Widget subclass are
48    // always marked "final".
49
50    final String title;
51
52  }

```