Introduction and Setup to Flutter

WHAT IS FLUTTER?
Flutter is Google's UI toolkit for building beautiful, natively compiled applications for Mobile, Web, and Desktop from a single codebase. It is very easy to learn and currently it is getting more and more popular. With this blog post, you will learn some basic stuff about Flutter and after reading it, you will be able to create a simple application using this technology.

Tools and Setup

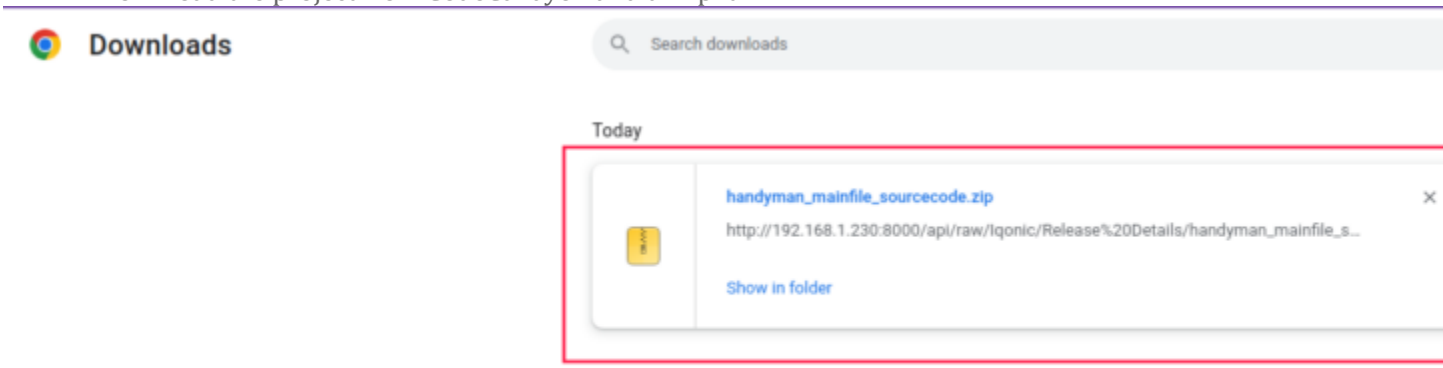
Prerequisite

- Flutter & Dart SDK
- Anyone IDE Android Studio (Recommended), Visual Studio Code or IntelliJ IDEA
- To edit this project you must have Flutter and Dart installed and configured successfully on your computer.
- Set up your editor – Install the Flutter and Dart plugins.
- If you have got Android SDK installed and configured, to install Flutter you only need to:
    - Download Flutter SDK from official website and extract it.
    - Add path to previously extracted SDK to your PATH variable
    - Run flutter doctor tool to check if everything is configured correctly.
    - All above steps are mentioned here: https://flutter.dev/docs/get-started/install/

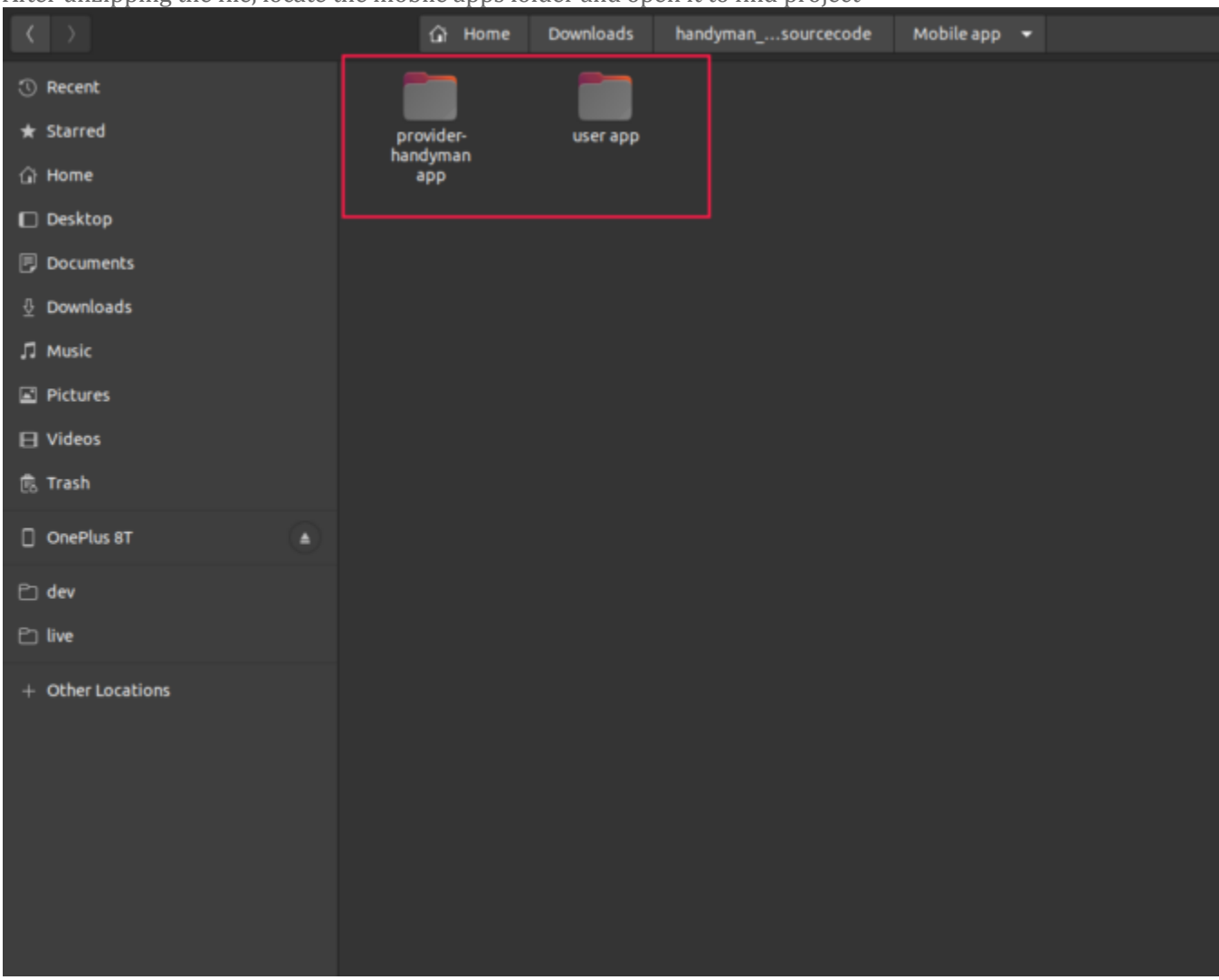
Getting Started (Build & Run)

Follow Below Steps To Build And Run Your Application.

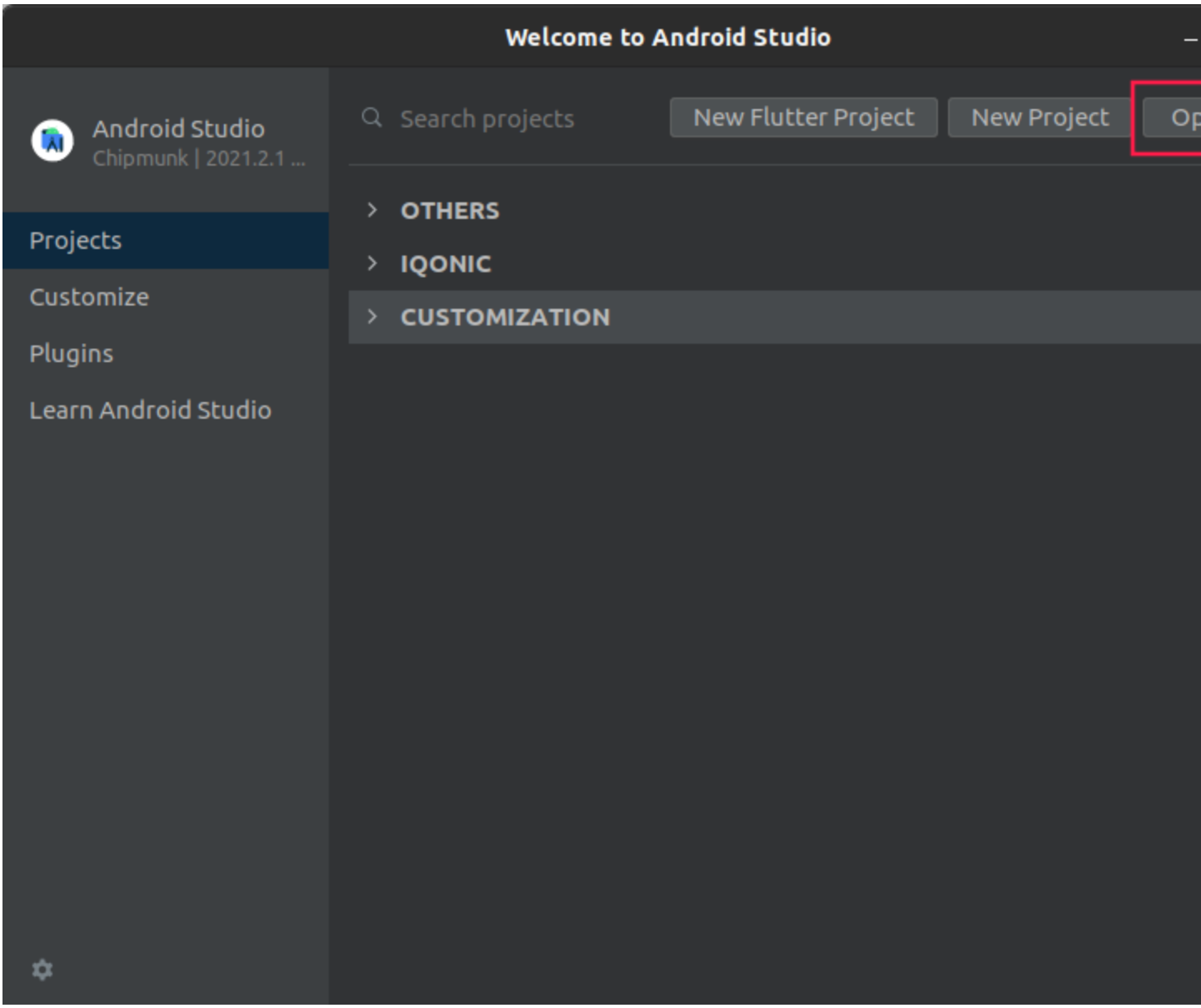
- Download the project from CodeCanayon and unzip it



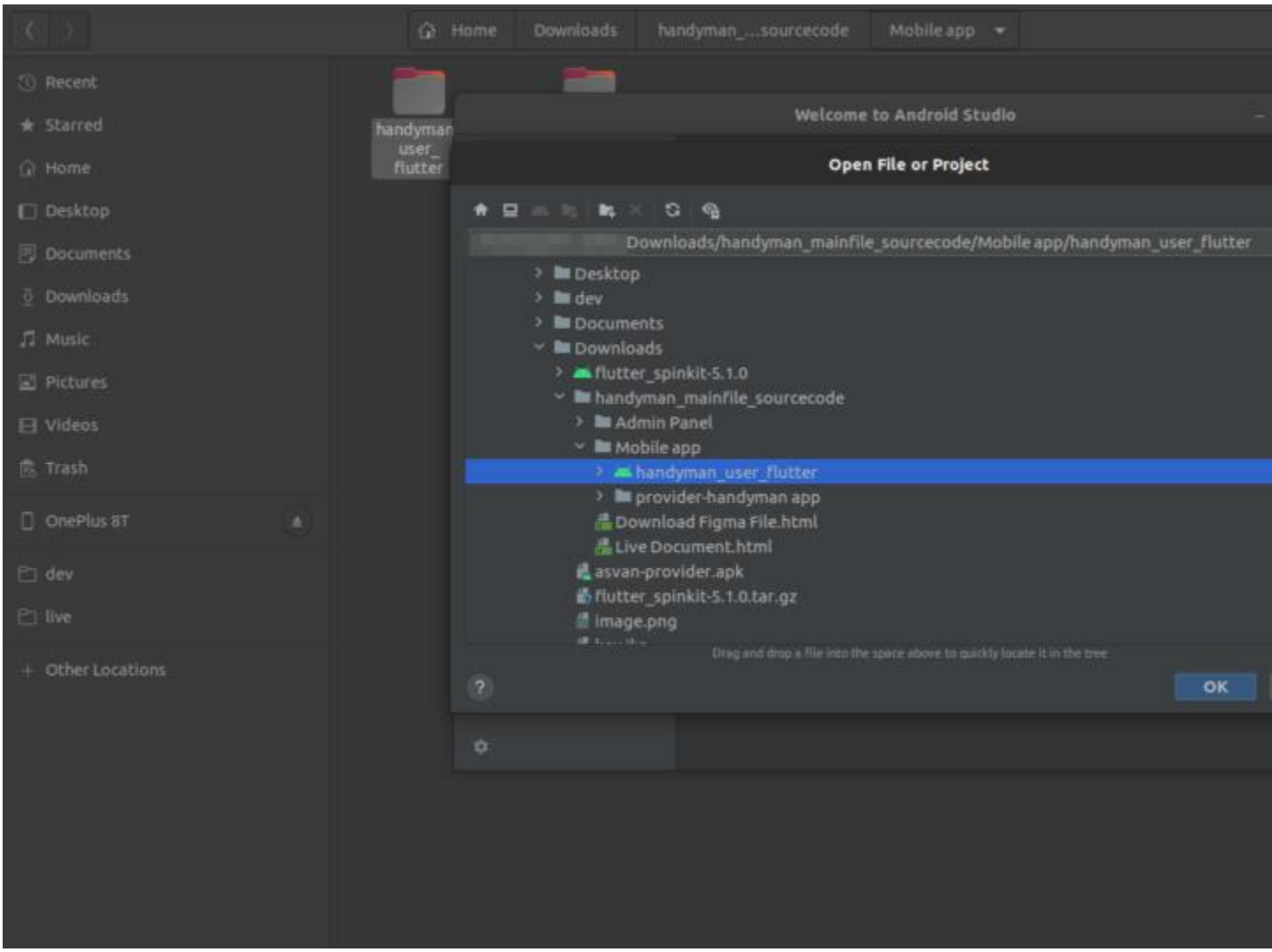
- After unzipping the file, locate the mobile apps folder and open it to find project



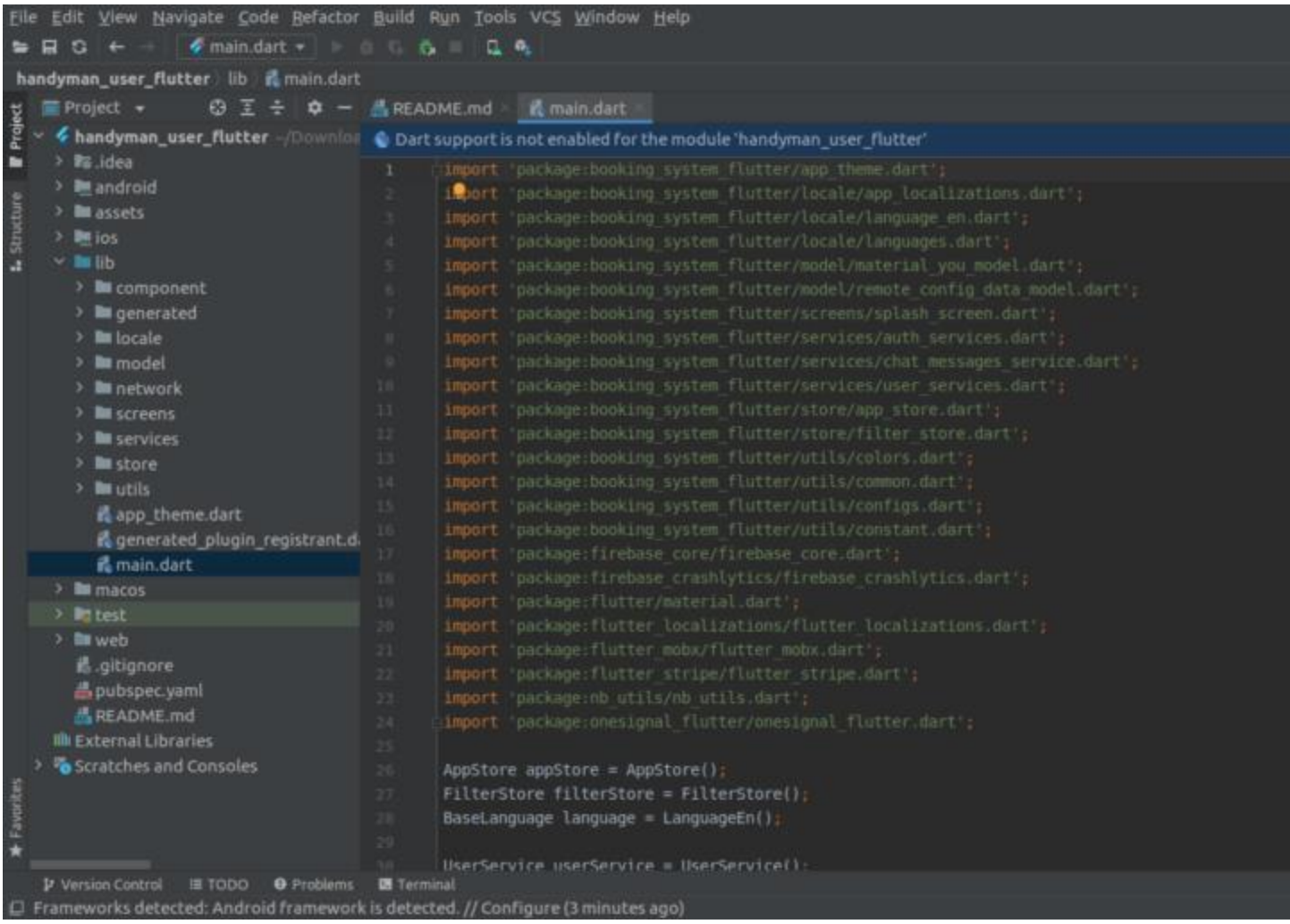
- If you want, you can copy both projects to a new location and then open any of them, in our case, we will be opening the user app
- Open Android Studio and click on the Open button

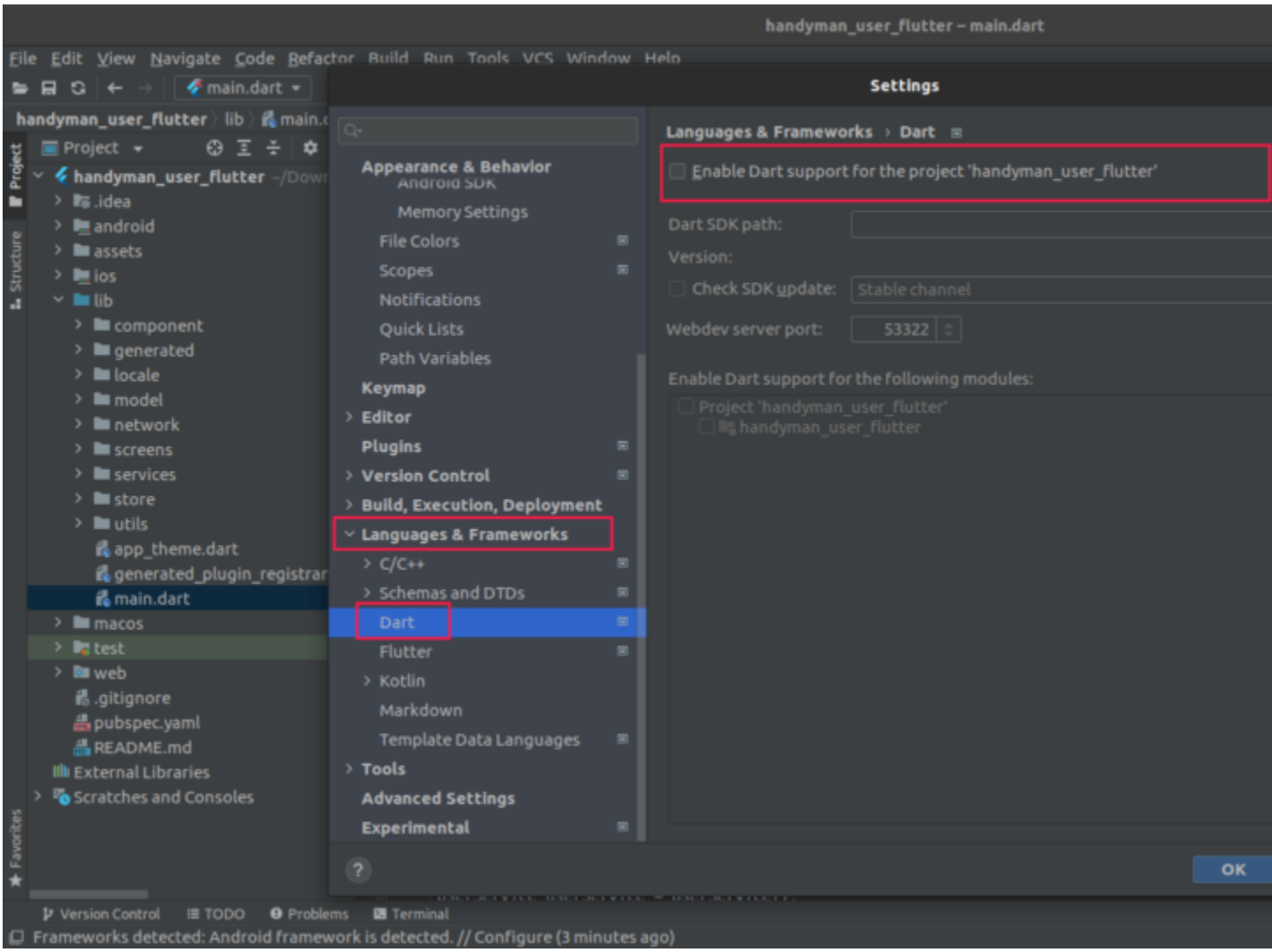


- Locate the project you wish to start with and open the project.

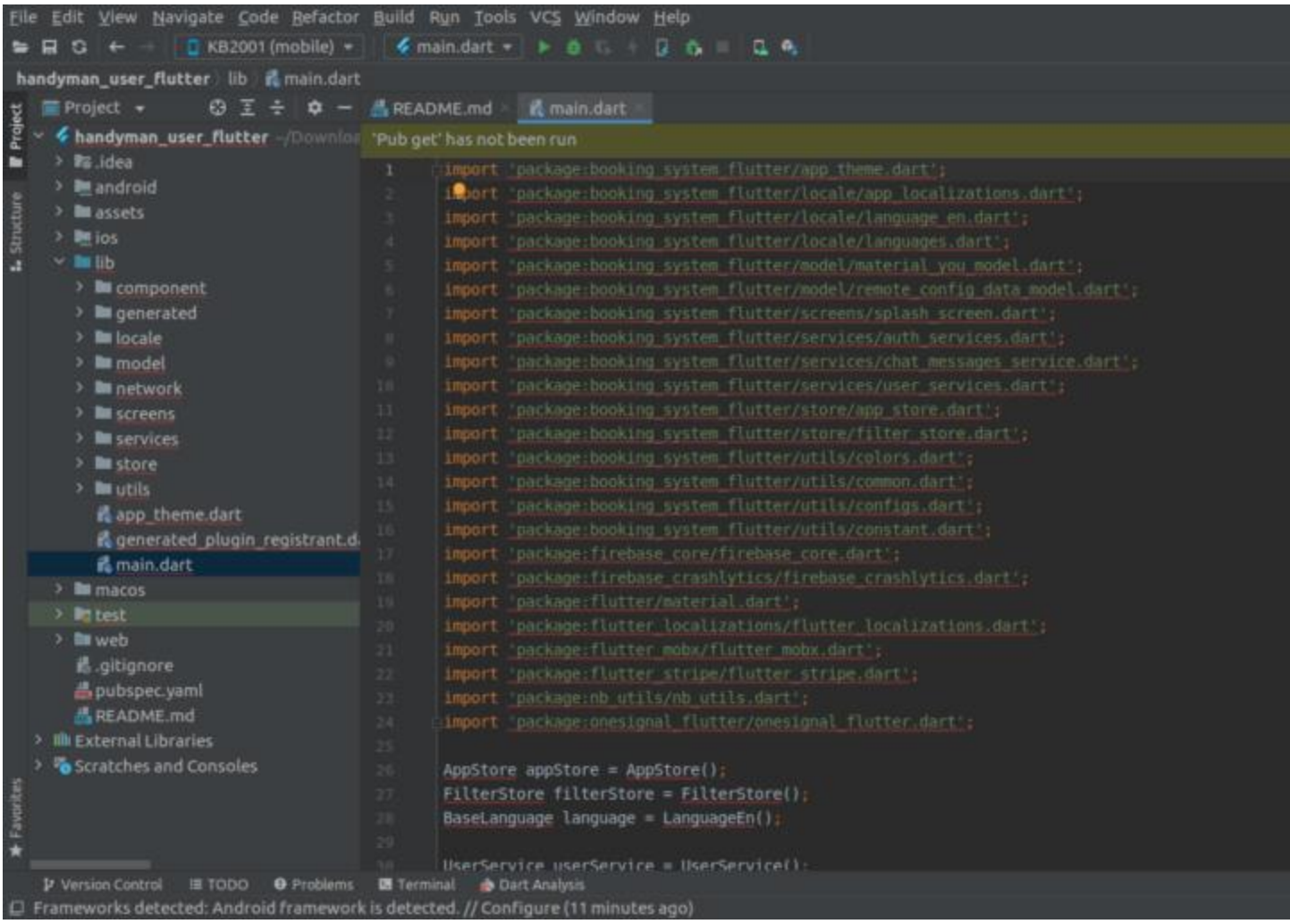


- First, enable dart support in the settings or by opening the main.dart file and looking for the "Enable Dart" option in the right corner. Click on that and dart will be enabled


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
main.dart
handyman_user_flutter lib main.dart
Project
handyman_user_flutter ~/Downloa
.idea
android
assets
ios
lib
component
generated
locale
model
network
screens
services
store
utils
app_theme.dart
generated_plugin_registrant.d
main.dart
macos
test
web
.gitignore
pubspec.yaml
README.md
External Libraries
Scratches and Consoles
README.md
main.dart
Dart support is not enabled for the module 'handyman_user_flutter'
import 'package:booking_system_flutter/app_theme.dart';
import 'package:booking_system_flutter/locale/app_localizations.dart';
import 'package:booking_system_flutter/locale/language_en.dart';
import 'package:booking_system_flutter/locale/languages.dart';
import 'package:booking_system_flutter/model/material_you_model.dart';
import 'package:booking_system_flutter/model/remote_config_data_model.dart';
import 'package:booking_system_flutter/screens/splash_screen.dart';
import 'package:booking_system_flutter/services/auth_services.dart';
import 'package:booking_system_flutter/services/chat_messages_service.dart';
import 'package:booking_system_flutter/services/user_services.dart';
import 'package:booking_system_flutter/store/app_store.dart';
import 'package:booking_system_flutter/store/filter_store.dart';
import 'package:booking_system_flutter/utils/colors.dart';
import 'package:booking_system_flutter/utils/common.dart';
import 'package:booking_system_flutter/utils/configs.dart';
import 'package:booking_system_flutter/utils/constant.dart';
import 'package:firebase_core/firebase_core.dart';
import 'package:firebase_crashlytics/firebase_crashlytics.dart';
import 'package:flutter/material.dart';
import 'package:flutter_localizations/flutter_localizations.dart';
import 'package:flutter_mobx/flutter_mobx.dart';
import 'package:flutter_stripe/flutter_stripe.dart';
import 'package:nb_utils/nb_utils.dart';
import 'package:onesignal_flutter/onesignal_flutter.dart';
AppStore appStore = AppStore();
FilterStore filterStore = FilterStore();
BaseLanguage language = LanguageEn();
UserService userService = UserService();
Version Control
TODO
Problems
Terminal
Frameworks detected: Android framework is detected. // Configure (3 minutes ago)



- After enabling, the Get Dependencies option will appear, click on it. (p.s. All files will be giving some error, after Get Dependencies, it will disappear automatically.)



TIPS

If the errors persist, you must try one of the following two steps.

1. Restart the **Android Studio**.
2. **Follow the following steps**
    - Select Tools -> Flutter -> Flutter Clean
    - Select Tools -> Flutter -> Flutter Pub Get
    - Select Tools -> Flutter -> Flutter Upgrade
    - Select File -&gt invalid Caches / Restart

- Connect the device to the computer or launch the emulator to run the project. Please run the project after connecting the device and wait for it to run in the device.

Table of contents

System requirements

Dart SDK Version 2.18.0 or greater.
Flutter SDK Version 3.3.0 or greater.

Check the UI of the entire app

Check the UI of all the app screens from a single place by setting up the 'initialRoute' to AppNavigation in the AppRoutes.dart file.

Application structure

After successful build, your application structure should look like this:

```
.
├── android                         - It contains files required to run the application on an Android platform.
├── assets                          - It contains all images and fonts of your application.
├── ios                             - It contains files required to run the application on an iOS platform.
├── lib                             - Most important folder in the application, used to write most of the Dart code..
    ├── main.dart                   - Starting point of the application
    ├── core
    │   ├── app_export.dart         - It contains commonly used file imports
    │   ├── constants               - It contains static constant class file
    │   └── utils                   - It contains common files and utilities of the application
    ├── presentation                - It contains widgets of the screens
    ├── routes                      - It contains all the routes of the application
```

```
    └── theme                       - It contains app theme and decoration
classes
    └── widgets                     - It contains all custom widget classes
```

How to format your code?

- if your code is not formatted then run following command in your terminal to format code
  ```
  dart format .
  ```

How you can improve code readability?

Resolve the errors and warnings that are shown in the application.

Libraries and tools used

- Provider - State management
  https://pub.dev/packages/provider
- cached_network_image - For storing internet image into cache
  https://pub.dev/packages/cached_network_image

Support

If you have any problems or questions, email@ uncutvynz@Gmail.com