

## **Jobsheet 12 – Weather app using API integration in Flutter**

Disusun untuk memenuhi Tugas Mata Kuliah Pemrograman Mobile

Dosen Pembimbing : Bapak Ade Ismail, S.Kom., M.T.



Disusun oleh

Liya Novitasari 2241760006

Kelas 3E

**Jurusan Teknologi Informasi**

**Program Studi D-4 Sistem Informasi Bisnis**

**POLITEKNIK NEGERI MALANG**

**MALANG**

**2024**

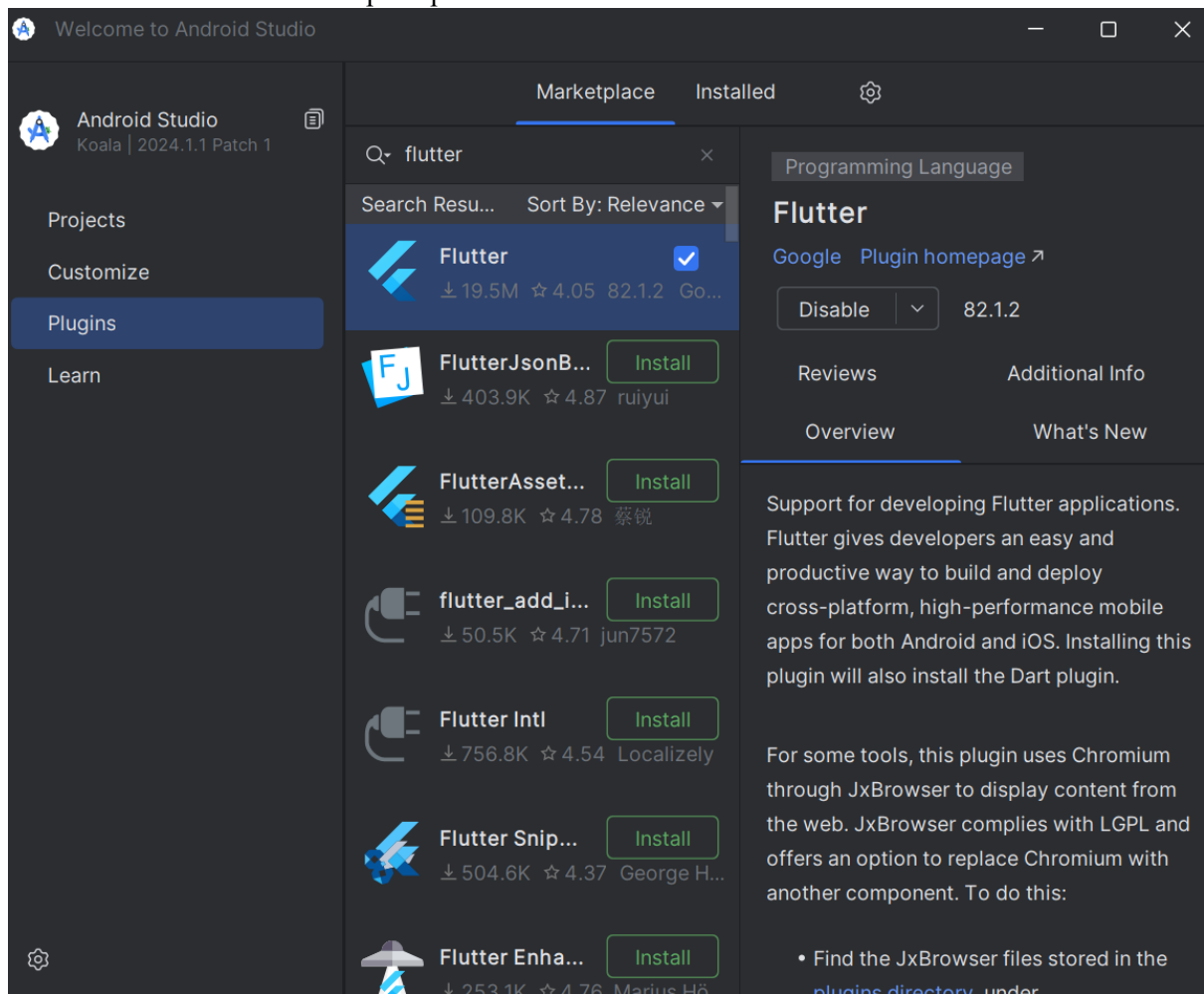
## Praktikum 1: Menerapkan Control (“if/else”)

Setiap Langkah-langkah praktikum berikut ini menggunakan DartPad di browser Anda.

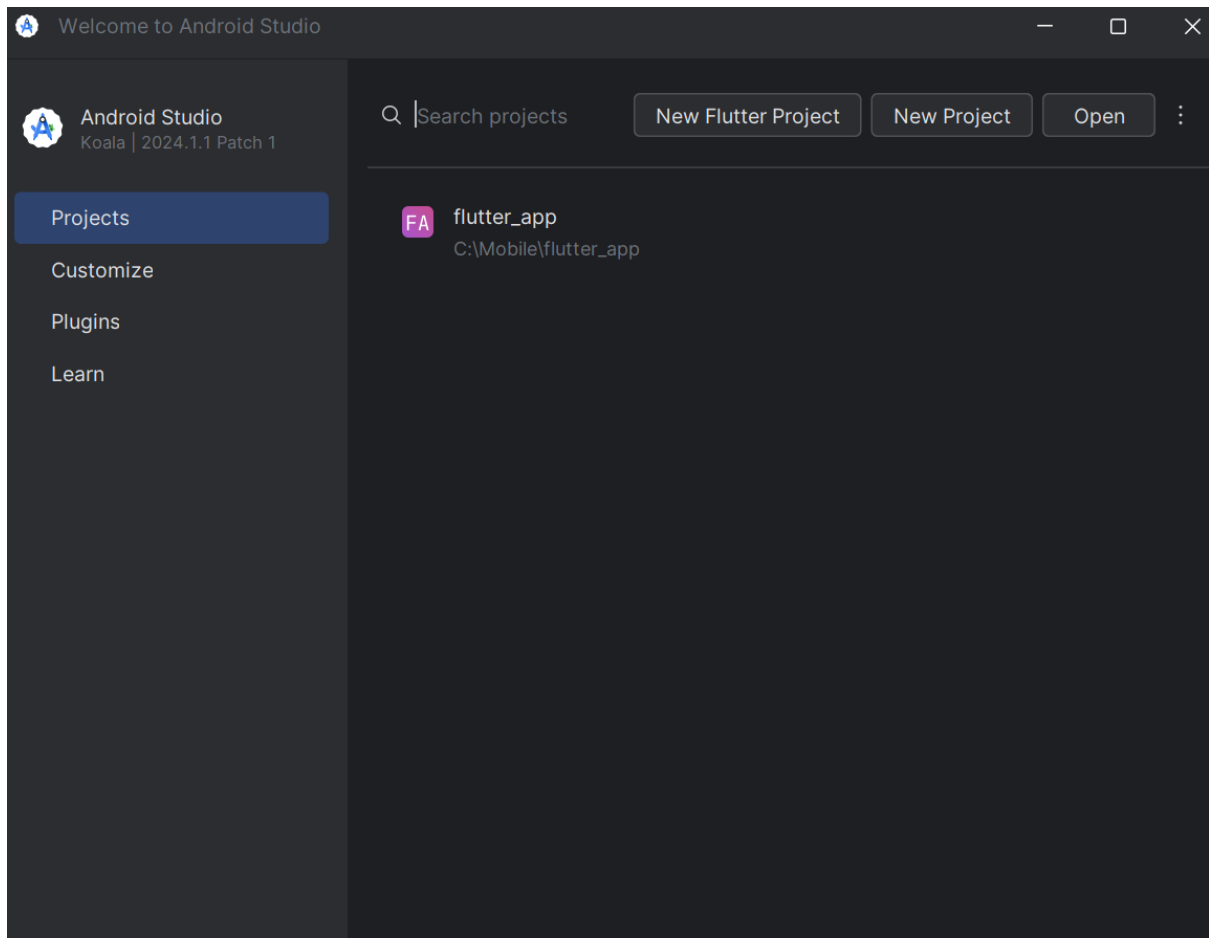
### Langkah 1:

Buka Android Studio sebagai IDE dan pilih “Start a new Flutter project”. Jika pada home project tidak ada menu tersebut maka dapat dipastikan bahwa “Plugin Fultter” sudah terinstal di Android Studio. Jika belum maka dapat dilakukan instalasi pada menu plugins seperti Berikut.

Saat menginstal plugin Flutter maka plugin Dart akan otomatis ikut terinstal. Setelah instalasi selesai, klik Restart IDE untuk menerapkan perubahan.

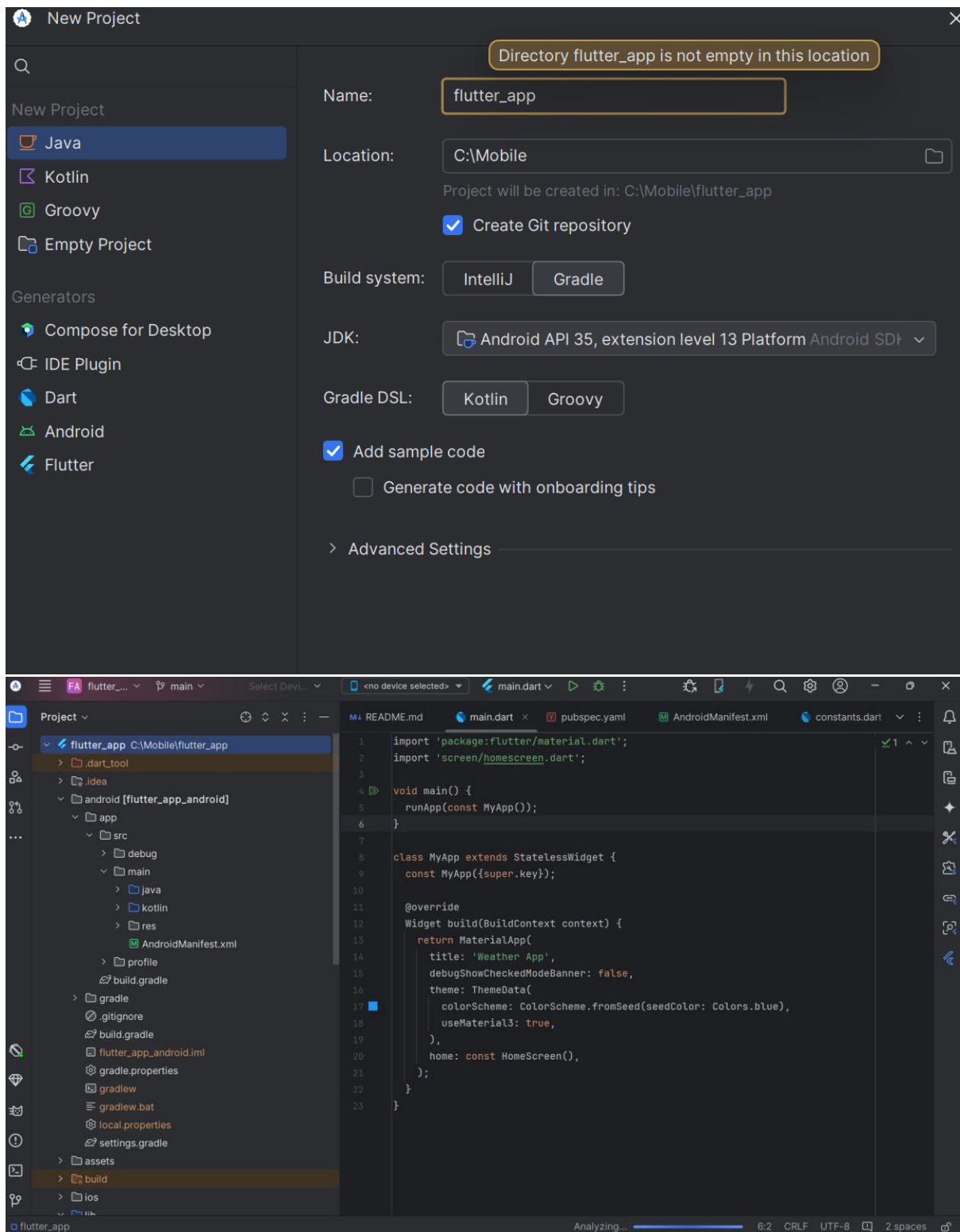


Maka setelah Android Studio terbuka kembali menu "New Flutter Project" akan muncul di halaman utama.

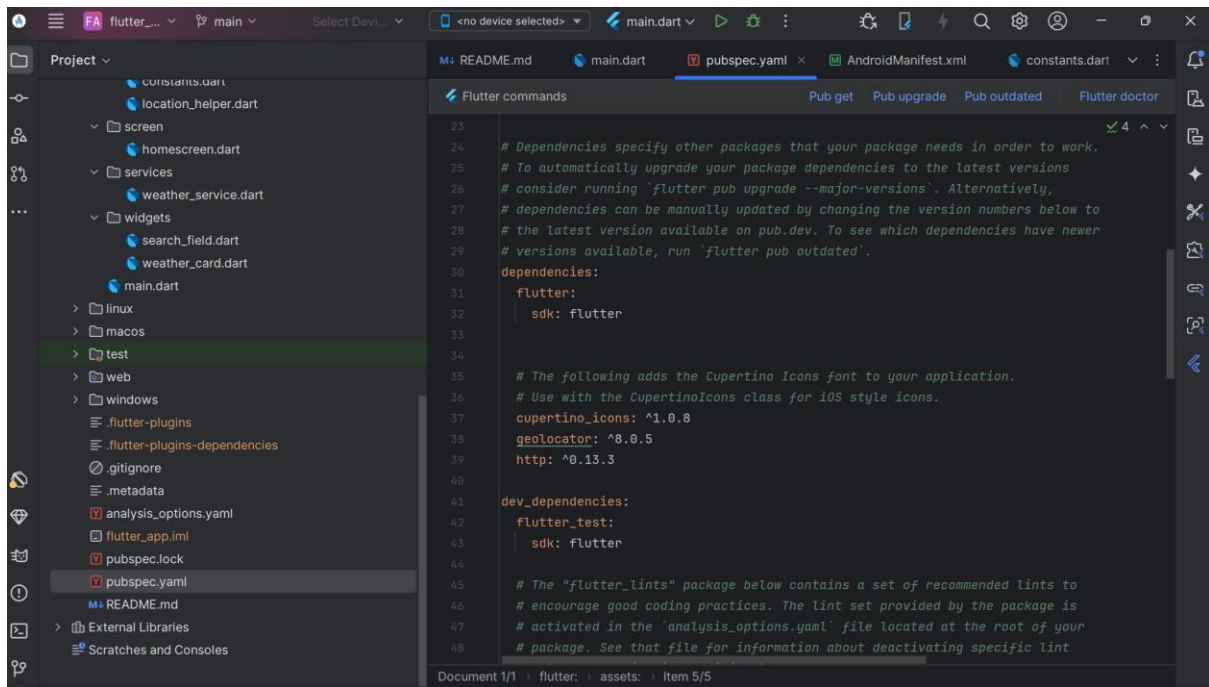


## Langkah 2:

Membuka nama project baru



Buka file pubspec.yaml dan tambahkan http untuk mengambil data dari API dan geolocator untuk mendapatkan lokasi.

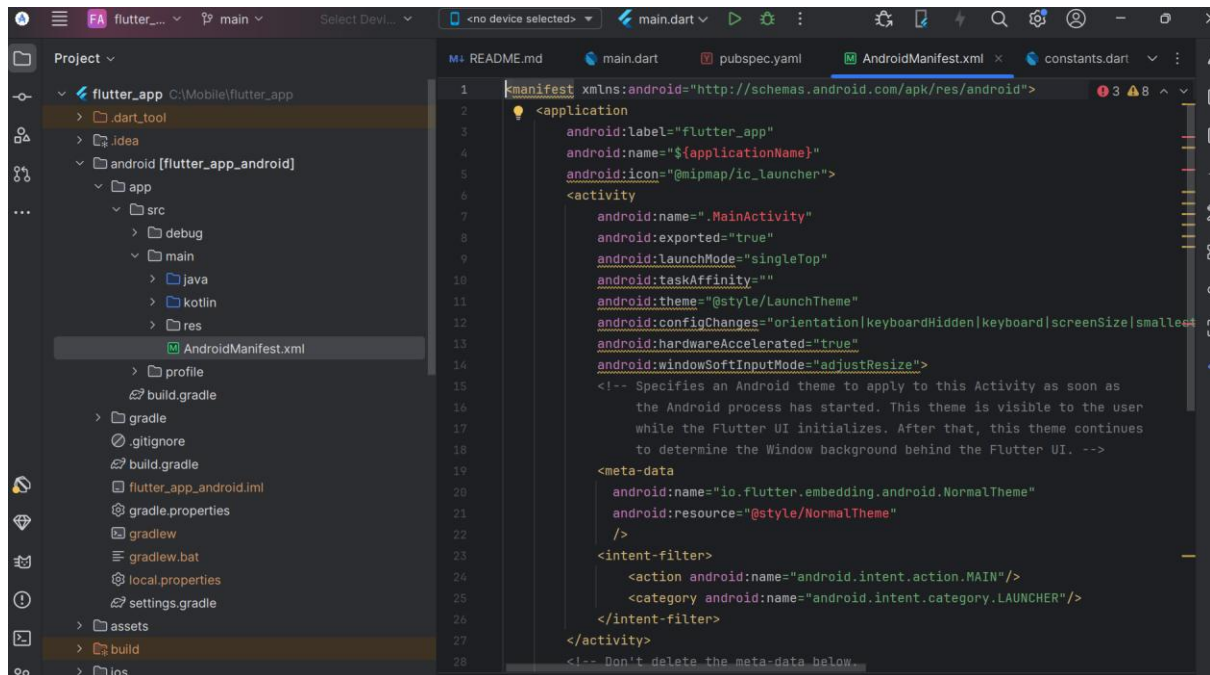


Jalankan perintah flutter pub get untuk mengunduh library.

```
PS C:\Mobile\flutter_app> flutter pub get
Resolving dependencies...
Downloading packages...
  async 2.11.0 (2.12.0 available)
  boolean_selector 2.1.1 (2.1.2 available)
  characters 1.3.0 (1.3.1 available)
  clock 1.1.1 (1.1.2 available)
  collection 1.18.0 (1.19.1 available)
  fake_async 1.3.1 (1.3.2 available)
  flutter_lints 4.0.0 (5.0.0 available)
  geolocator 8.2.1 (13.0.2 available)
  geolocator_android 3.2.1 (4.6.1 available)
  geolocator_web 2.2.1 (4.1.1 available)
  geolocator_windows 0.1.3 (0.2.3 available)
  http 0.13.6 (1.2.2 available)
  http_parser 4.0.2 (4.1.1 available)
  leak_tracker 10.0.5 (10.0.8 available)
  leak_tracker_flutter_testing 3.0.5 (3.0.9 available)
  lints 4.0.0 (5.1.0 available)
  matcher 0.12.16+1 (0.12.17 available)
  material_color_utilities 0.11.1 (0.12.0 available)
  meta 1.15.0 (1.16.0 available)
  path 1.9.0 (1.9.1 available)
  stack_trace 1.11.1 (1.12.0 available)
```

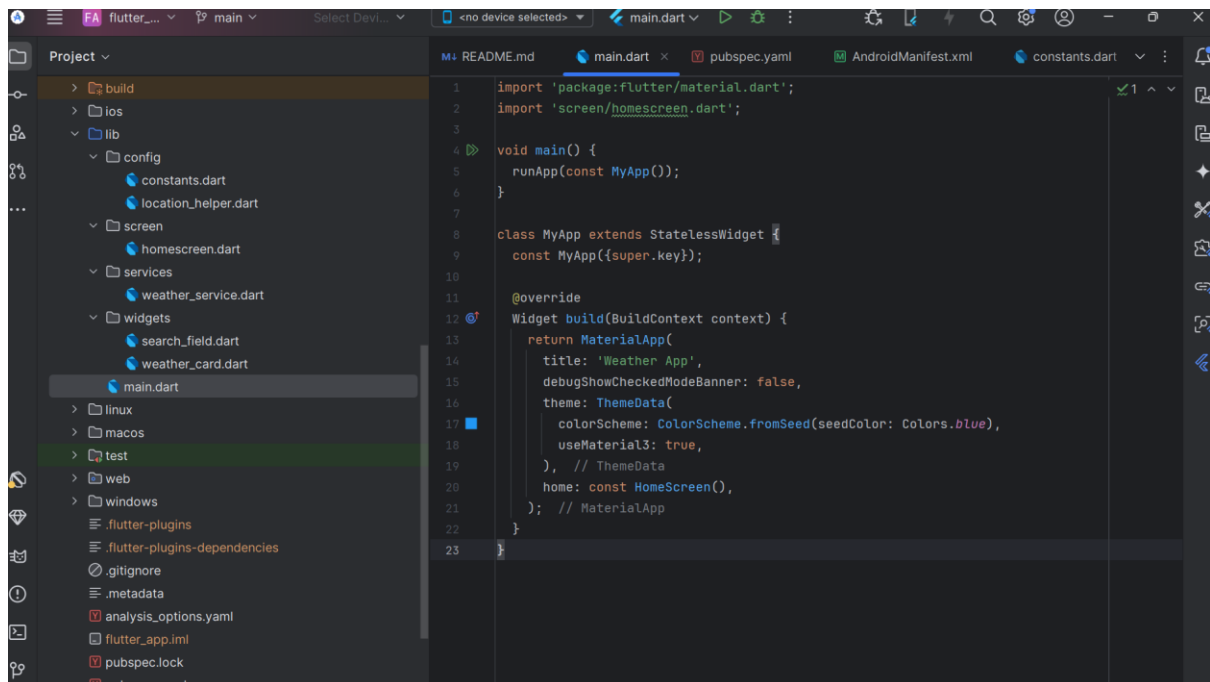
### Langkah 3:

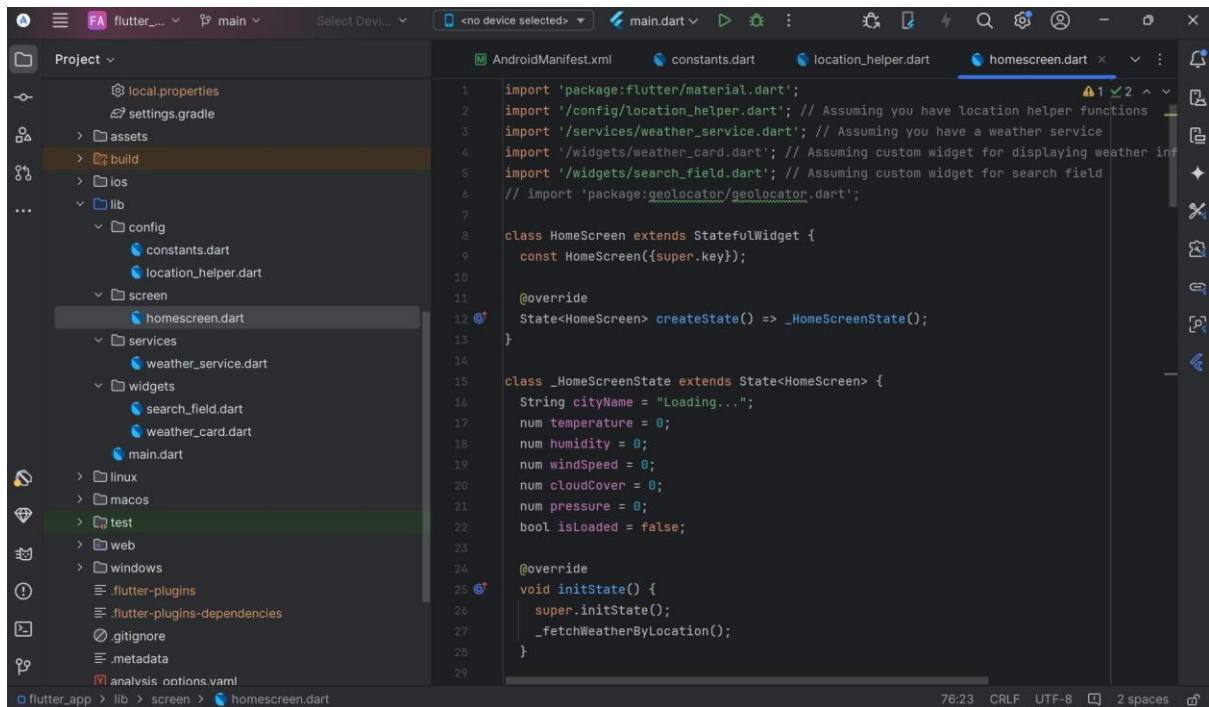
Konfigurasi Geolocator dengan menambahkan izin lokasi di file AndroidManifest.xml seperti berikut



### Langkah 4:

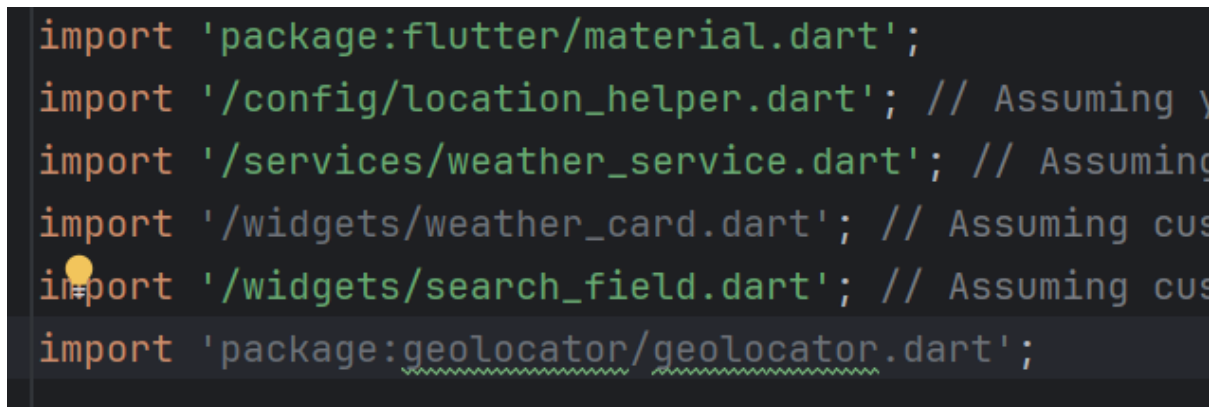
Buat Struktur Utama dengan Membuat file yang dibutuhkan seperti main.dart sebagai titik masuk aplikasi dan homescreen.dart sebagai halaman utama, serta mengubah widget MyApp menjadi MaterialApp di main.dart.





## Langkah 5:

Di `homescreen.dart` import `geolocator` lalu buat fungsi untuk mendapatkan koordinat lokasi seperti berikut.



```

@Override
void initState() {
  super.initState();
  _fetchWeatherByLocation();
}

void _fetchWeatherByLocation() async {
  final position = await getCurrentLocation(); // Assuming you have a function to get location
  if (position != null) {
    final weatherData = await fetchWeatherByLocation(position); // Assuming function to fetch weather by location
    if (weatherData != null) {
      setState(() {
        cityName = weatherData['name'];
        temperature = weatherData['main']['temp'] - 273.15; // Convert Kelvin to Celsius
        humidity = weatherData['main']['humidity'];
        windSpeed = weatherData['wind']['speed'];
        cloudCover = weatherData['clouds']['all'];
        pressure = weatherData['main']['pressure'];
        isLoading = true;
      });
    }
  }
}

```

## Langkah 6:

Dapatkan API Key di OpenWeatherMap <https://openweathermap.org> pilih paket gratis (Free Plan) lalu daftar akun bila belum memiliki akun dengan mengklik tombol “Sign In”

Setelah mendaftar maka akan ada perintah membuka gmail untuk verifikasi akun yang didaftarkan kemudian masuk ke akun yang baru dibuat.

Sign In To Your Account

Enter email

Password

☐ Remember me

Submit

Not registered? [Create an Account.](#)

Lost your password? [Click here to recover.](#)

Product Collections

- Current and Forecast APIs
- Historical Weather Data

Subscription

- How to start
- Pricing

Company

OpenWeather is a team of IT experts and data scientists that has been practising deep weather data science since 2014.




OpenWeather

Weather in your city

Guide API Dashboard Marketplace Pricing Maps Our Initiatives Partners Blog For Business liyano... Support

We have sent the confirmation link to [novitaliya03@gmail.com](mailto:novitaliya03@gmail.com). Please check your email.

New Products Services **API keys** Billing plans Payments Block logs My orders My profile Ask a question



### Historical weather for any location

Our new technology, Time Machine, has allowed us to enhance the data in the **Historical Weather Collection**.


- Historical weather data available for **ANY** coordinate
- The depth of historical data have been extended to **45 YEARS**

You can download data from **Personal account** or **contact us** to order it.

[Learn more](#) [Go to purchase](#)

### OpenWeather Dashboard

The **OpenWeather Dashboard** is a comprehensive solution that allows you to get up-to-date weather data collected in one place. We've put together our best technology to give you 24/7



Setelah login, buka dashboard akun. Di dashboard, pilih API keys pada menu sebelah kiri akan terdapat API Key default yang disediakan oleh OpenWeatherMap. API key ini sudah aktif dan bisa digunakan untuk membuat aplikasi.

OpenWeather



Weather in your city

Guide API Dashboard Marketplace Pricing Maps Our Initiatives Partners Blog For Business liyano... Support

We have sent the confirmation link to [novitaliya03@gmail.com](mailto:novitaliya03@gmail.com). Please check your email.

New Products Services **API keys** Billing plans Payments Block logs My orders My profile Ask a question

You can generate as many API keys as needed for your subscription. We accumulate the total load from all of them.

Key	Name	Status	Actions	Create key
c79a58443d3d3362dd1be0fc1e5e3e97	Default	Active	 	<input type="text" value="API key name"/> <input type="button" value="Generate"/>

**Product Collections**

- Current and Forecast APIs
- Historical Weather Data
- Weather Maps
- Weather Dashboard

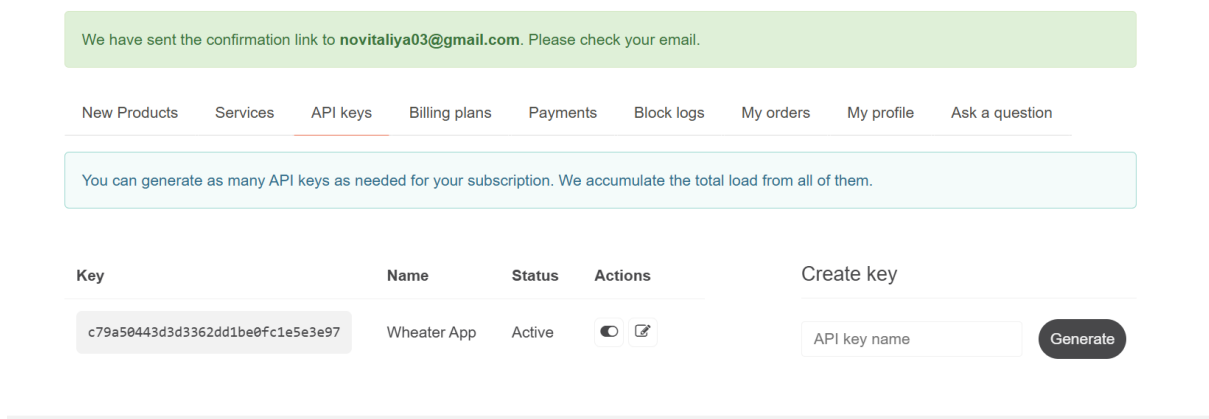
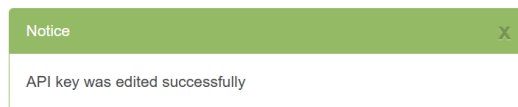
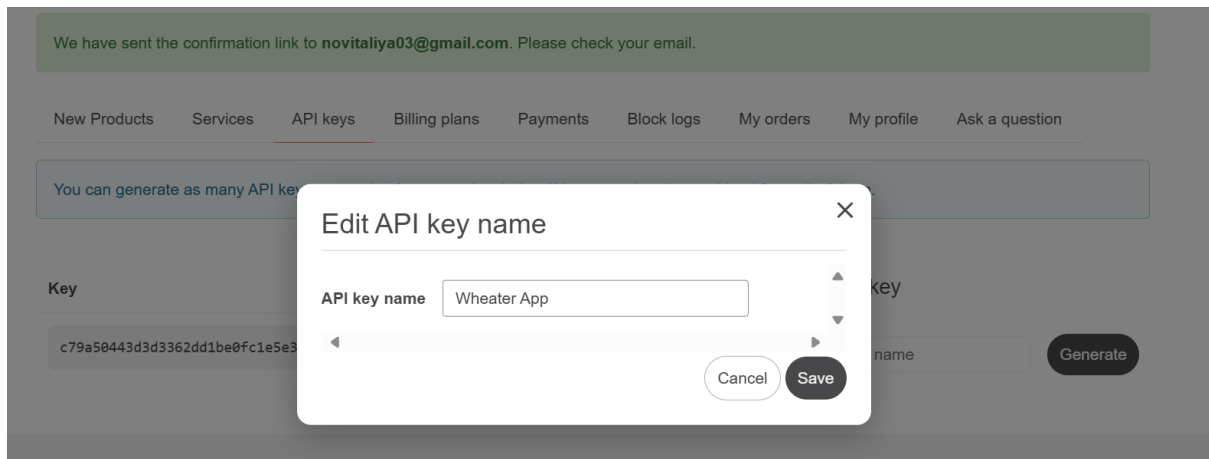
**Subscription**

- How to start
- Pricing
- Subscribe for free
- FAQ

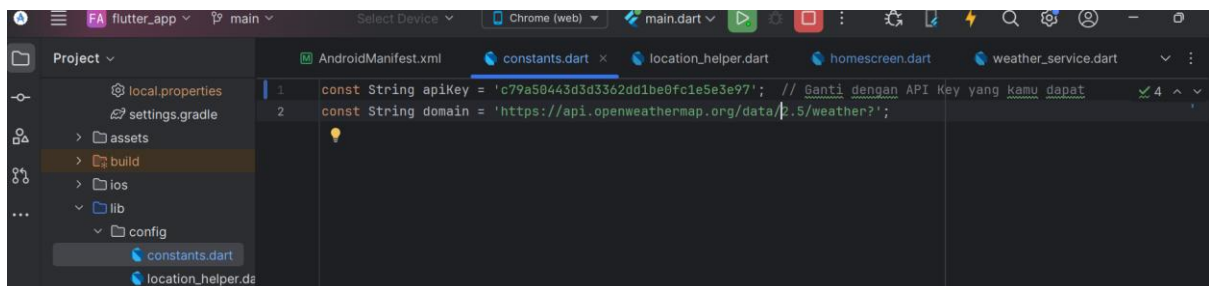
**Company**

OpenWeather is a team of IT experts and data scientists that has been practising deep weather data science since 2014. For each point on the globe, OpenWeather provides historical, current and forecasted weather data via light-speed APIs.

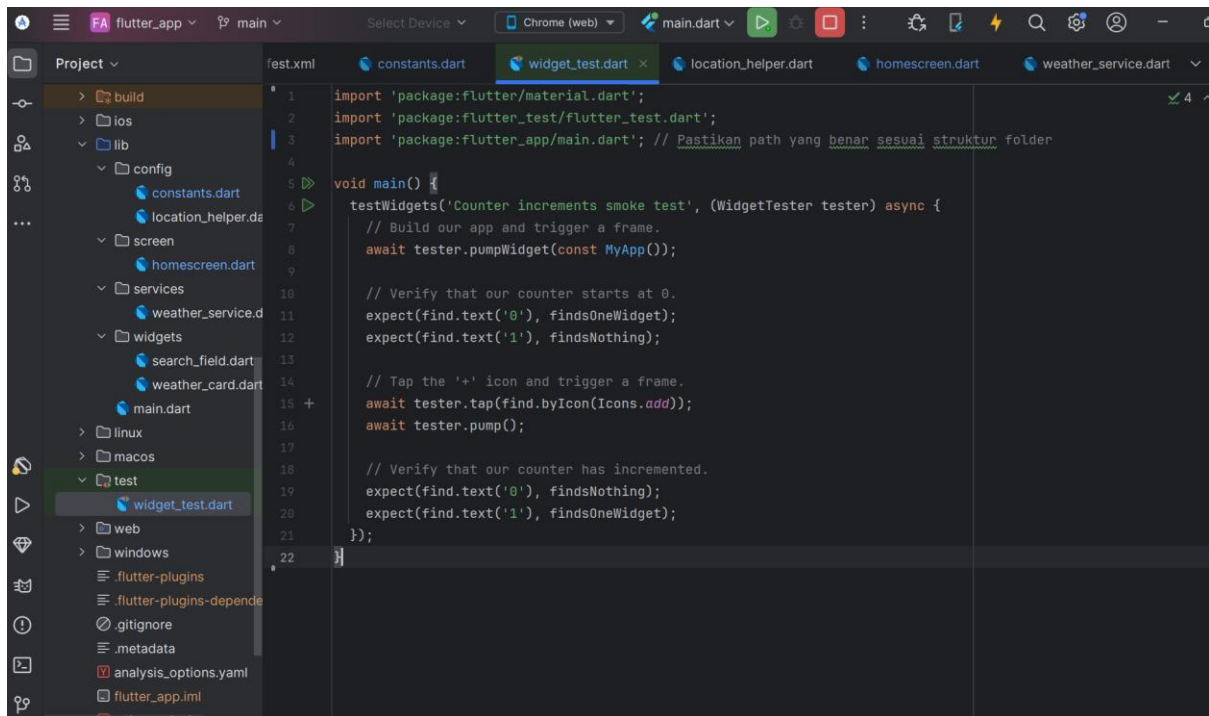
Disini saya ganti nama Default nya menjadi mengikuti nama aplikasinya yaitu Weather App.



Kemudian masukan API Key ke Aplikasi dengan membuat folder baru bernama config yang berisi file constants.dart yang mana dalam file tersebut buat konstanta apiKey seperti berikut



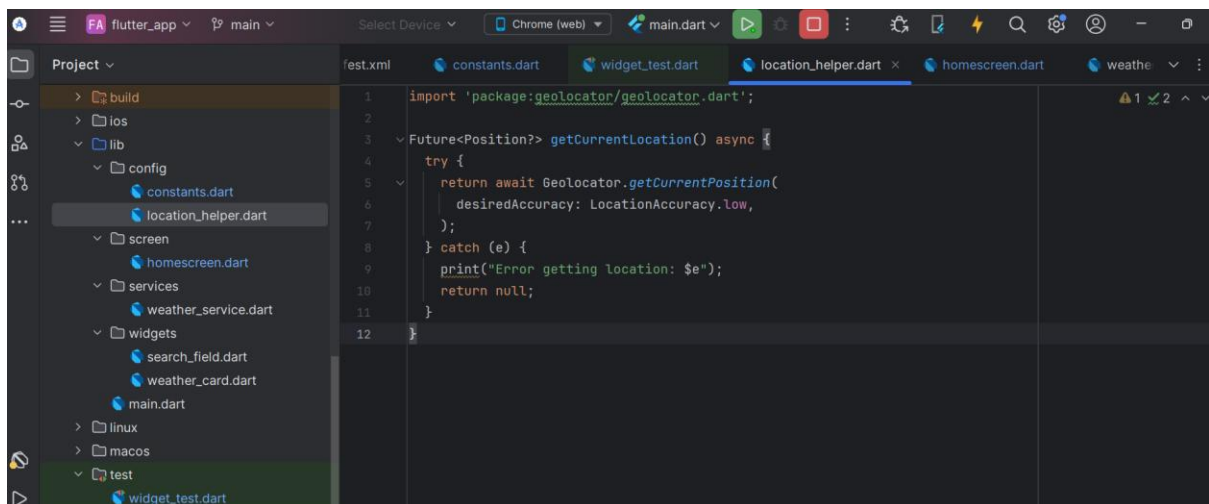
Dan perbaiki file widget\_test.dart di folder test seperti berikut



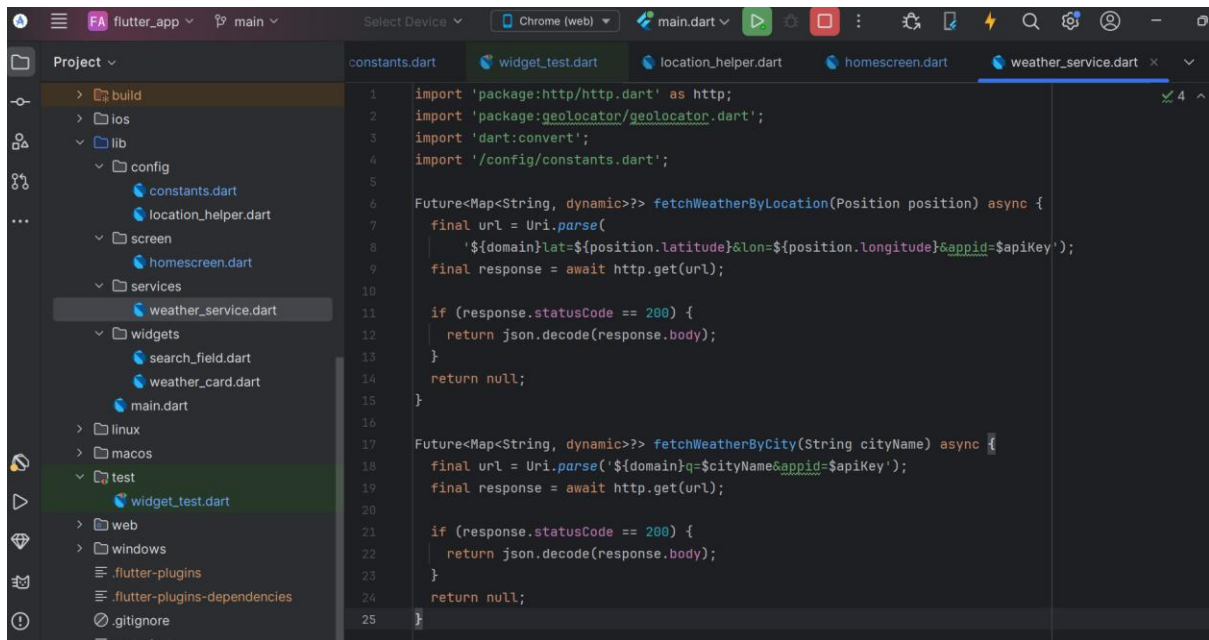
## Langkah 7:

Ambil Data Cuaca gunakan library http untuk mengambil data dari API berdasarkan latitude/longitude, dan berdasarkan nama kota serta decode respons JSON menggunakan `dart:convert`.

Disini saya membuat file baru dalam folder config bernama `location_helper.dart` yang isinya seperti berikut

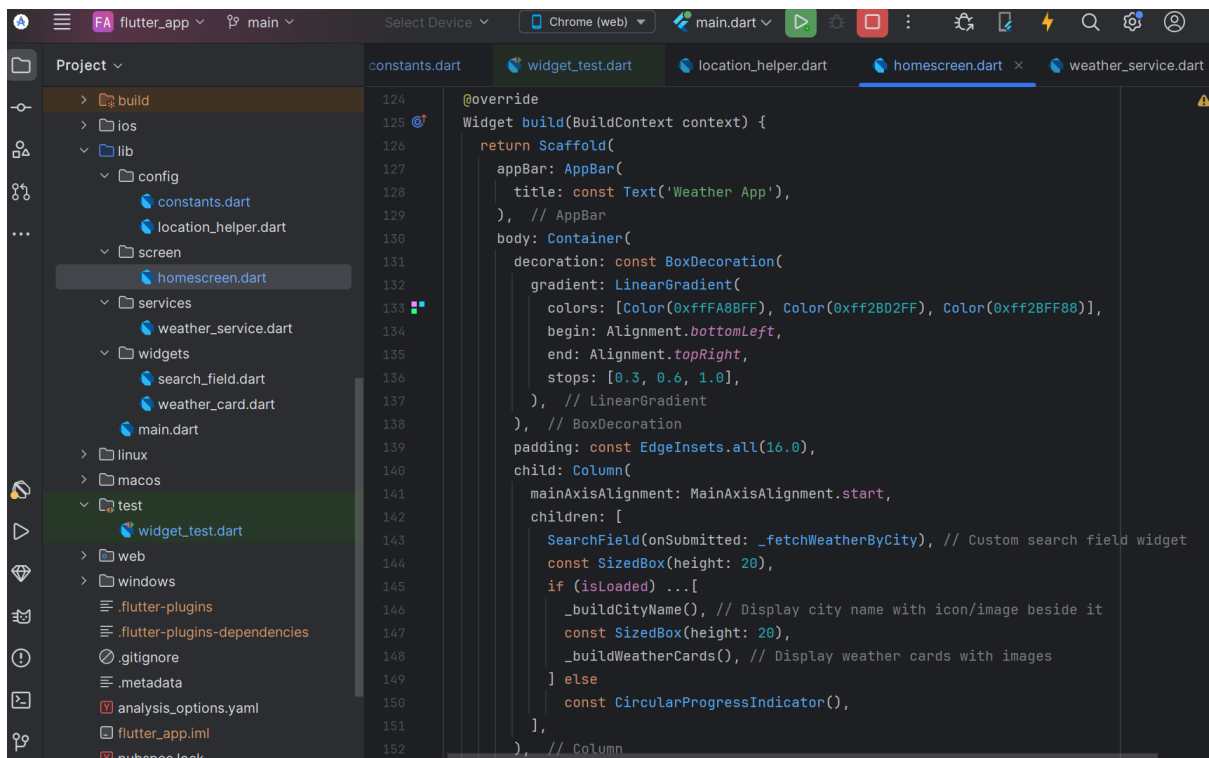


Kemudian saya juga membuat folder services yang berisi file `weather_service.dart` untuk mengambil data cuaca berdasarkan lokasi yang inputannya seperti berikut.



## Langkah 8:

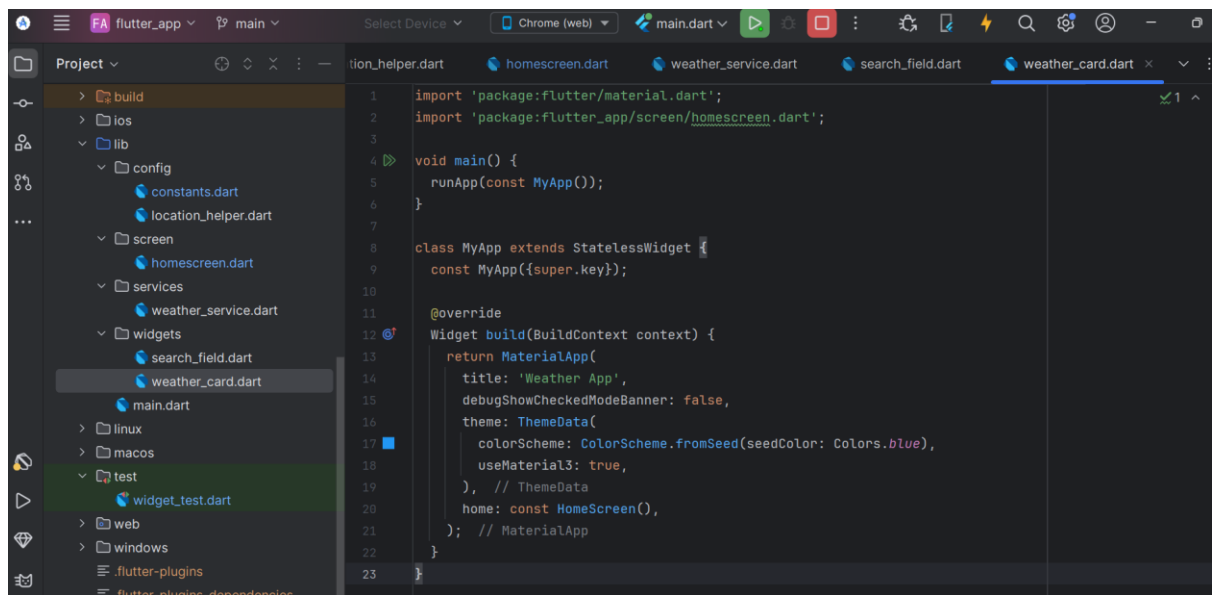
Bangun tampilan UI dengan menambahkan background gradas di homescreen.dart pada folder screen



## Langkah 9:

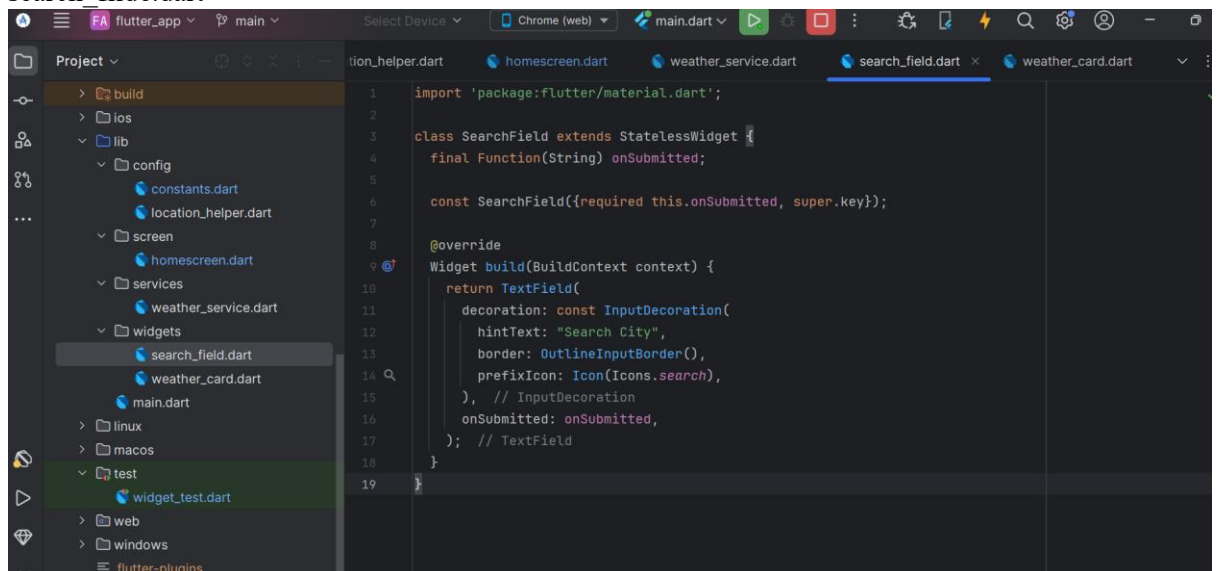
Tambahkan data cuaca menggunakan Widget Text jadi disini saya membuat folder baru di dalam lib yaitu folder widgets yang berisi file weather\_card.dart yang berisi data cuaca dan file search\_filde.dart yang digunakan widget nya untuk mencari atau menambah data lokasi. Berikut isi dari masing-masing file tersebut

- weather\_card.dart



```
1 import 'package:flutter/material.dart';
2 import 'package:flutter_app/screen/homescreen.dart';
3
4 void main() {
5   runApp(const MyApp());
6 }
7
8 class MyApp extends StatelessWidget {
9   const MyApp({super.key});
10
11   @override
12   Widget build(BuildContext context) {
13     return MaterialApp(
14       title: 'Weather App',
15       debugShowCheckedModeBanner: false,
16       theme: ThemeData(
17         colorScheme: ColorScheme.fromSeed(seedColor: Colors.blue),
18         useMaterial3: true,
19       ), // ThemeData
20       home: const HomeScreen(),
21     ); // MaterialApp
22   }
23 }
```

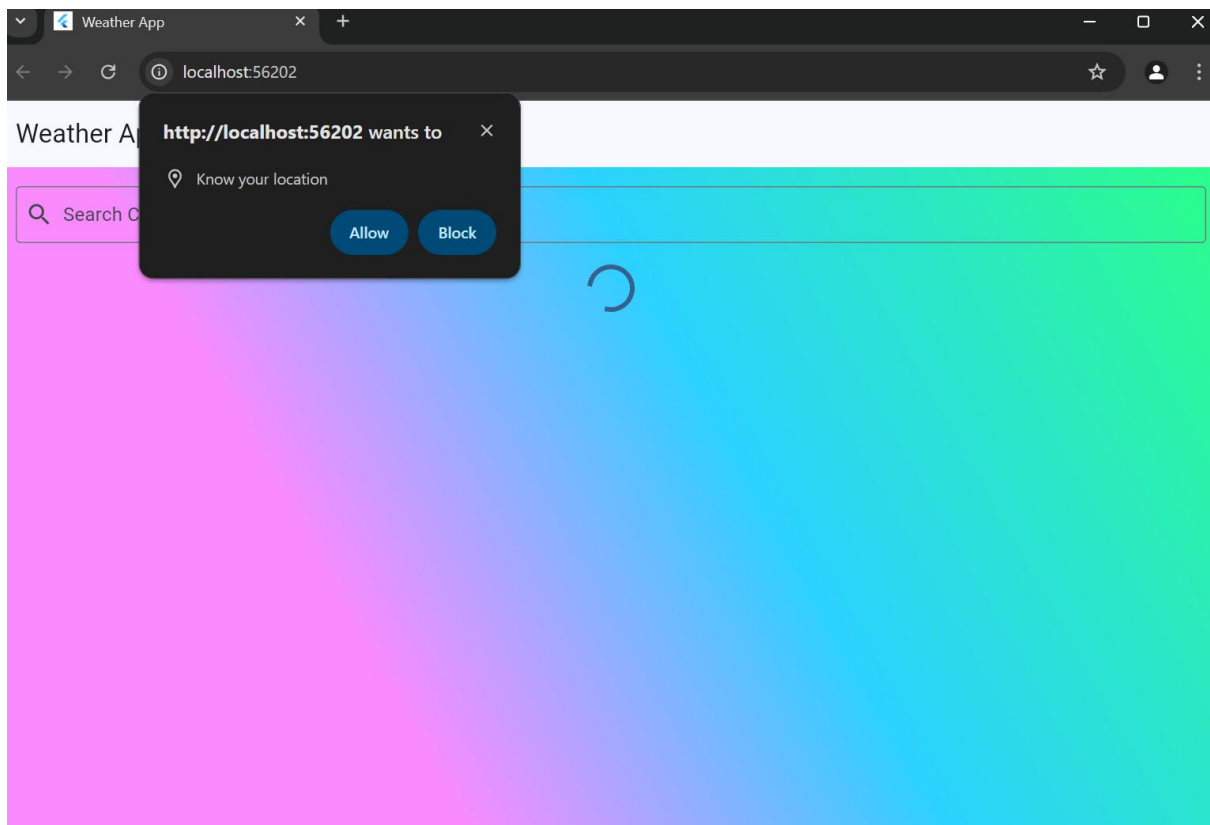
- search\_filde.dart

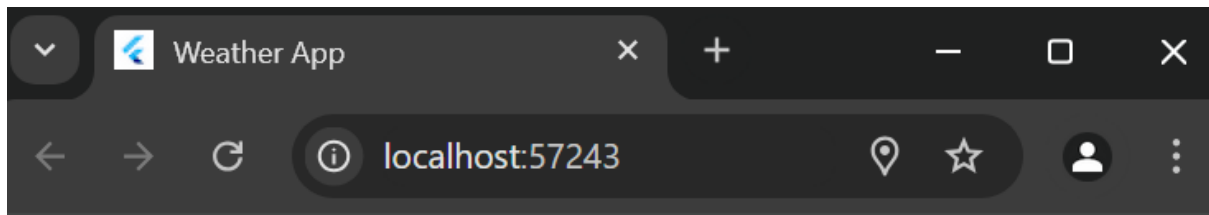


```
1 import 'package:flutter/material.dart';
2
3 class SearchField extends StatelessWidget {
4   final Function(String) onSubmitted;
5
6   const SearchField({required this.onSubmitted, super.key});
7
8   @override
9   Widget build(BuildContext context) {
10     return TextField(
11       decoration: const InputDecoration(
12         hintText: "Search City",
13         border: OutlineInputBorder(),
14         prefixIcon: Icon(Icons.search),
15       ), // InputDecoration
16       onSubmitted: onSubmitted,
17     ); // TextField
18   }
19 }
```

## Langkah 10:

Uji dan Finalisasi. Berikut output yang dihasilkan dari aplikasi yang telah saya buat di atas yang menampilkan cuaca berdasarkan nama kota dan hasil pengujian menunjukan bahwa seluruh fungsi sudah berjalan dengan baik.

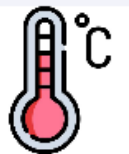




# Weather App

🔍 Search City

📍 Malang



**Temperature**  
29.77 °C



**Pressure**  
1010 hPa



**Humidity**  
61 %



**Cloud Cover**  
98 %

Berikut hasil proses berserta hasil pencarian cuaca berdasarkan lokasi.

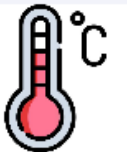
Weather App

localhost:57243

Weather App


Tuban

Tuban




Temperature

31.30 °C




Pressure

1008 hPa



Humidity

60 %



Cloud Cover

100 %