

MODUL 1

CODELAB

PEMROGRAMAN MOBILE D

Dosen Pengampu:

Bashor Fauzan Muthohirin, S.Kom., M.Kom

Source Code: https://github.com/hisyam99/PRAK-PEMROGRAMAN-MOBILE-2024/tree/main/TUGAS/modul_1/codelab_01

Muhammad Hisyam Kamil
202210370311060

Daftar Isi

MODUL 1: CODELAB	3
I. Pendahuluan.....	3
II. Create project flutter	3
III. Instalasi State Management GetX.....	4
IV. Menerapkan Architecture Pattern GetX.....	6
V. Refactor Code dan Menambah Route Baru	7

MODUL 1: CODELAB

Source Code: https://github.com/hisyam99/mini-project-strukdat-2024/tree/main/MINI_PROJECT_2

CODELAB

TUGAS 1

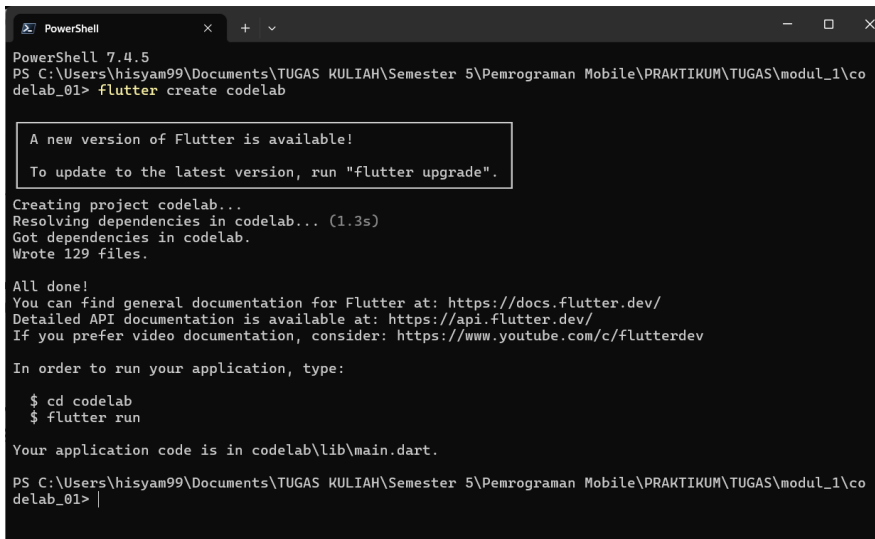
1. Lakukan instalasi State Management GetX pada project kalian
2. Lakukan refactoring folder project dengan menerapkan architecture pattern
3. Lakukan refactoring ke codingan kalian agar codingan lebih rapi dan mudah untuk dibaca.

I. Pendahuluan

Pada modul 1 codelab ini, kita diminta membuat sebuah aplikasi mobile yang sederhana menggunakan Flutter serta melakukan instalasi architecture pattern GetX menggunakan GET-CLI.

II. Create project flutter

1. Buat project dengan menggunakan command "flutter create nama_project"



```
PowerShell
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\codelab_01> flutter create codelab

A new version of Flutter is available!
To update to the latest version, run "flutter upgrade".

Creating project codelab...
Resolving dependencies in codelab... (1.3s)
Got dependencies in codelab.
Wrote 129 files.

All done!
You can find general documentation for Flutter at: https://docs.flutter.dev/
Detailed API documentation is available at: https://api.flutter.dev/
If you prefer video documentation, consider: https://www.youtube.com/c/flutterdev

In order to run your application, type:

$ cd codelab
$ flutter run

Your application code is in codelab\lib\main.dart.

PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\codelab_01>
```

2. Setelah project flutter sudah dibuat, ketik command "cd nama_project"

```
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\code
lab_01> cd codeLab
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\code
lab_01\codeLab> ls

    Directory: C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman
Mobile\PRAKTIKUM\TUGAS\modul_1\codeLab_01\codeLab

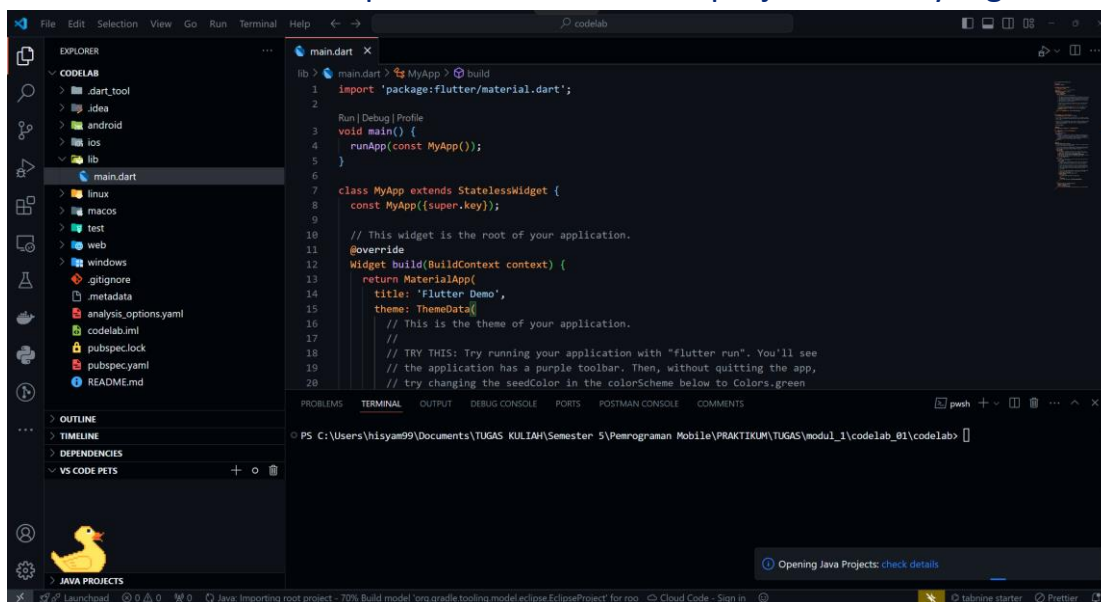
Mode                LastWriteTime         Length Name
----                -
d-----          9/28/2024   7:55 PM             dart_tool
d-----          9/28/2024   7:55 PM             .idea
d-----          9/28/2024   7:55 PM          android
d-----          9/28/2024   7:55 PM             ios
d-----          9/28/2024   7:55 PM             lib
d-----          9/28/2024   7:55 PM          linux
d-----          9/28/2024   7:55 PM          macos
d-----          9/28/2024   7:55 PM             test
d-----          9/28/2024   7:55 PM             web
d-----          9/28/2024   7:55 PM          windows
-a-----          9/28/2024   7:55 PM           734 .gitignore
-a-----          9/28/2024   7:55 PM          1706 .metadata
-a-----          9/28/2024   7:55 PM          1448 analysis_options.yaml
-a-----          9/28/2024   7:55 PM           859 codeLab.iml
-a-----          9/28/2024   7:55 PM          5272 pubspec.lock
-a-----          9/28/2024   7:55 PM          3962 pubspec.yaml
-a-----          9/28/2024   7:55 PM           566 README.md

PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\code
lab_01\codeLab> |
```

3. Jika menggunakan Visual Studio Code, bisa mengetikkan command "code ." untuk membuka project tersebut

```
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\code lab_01\code lab> code .\|
```

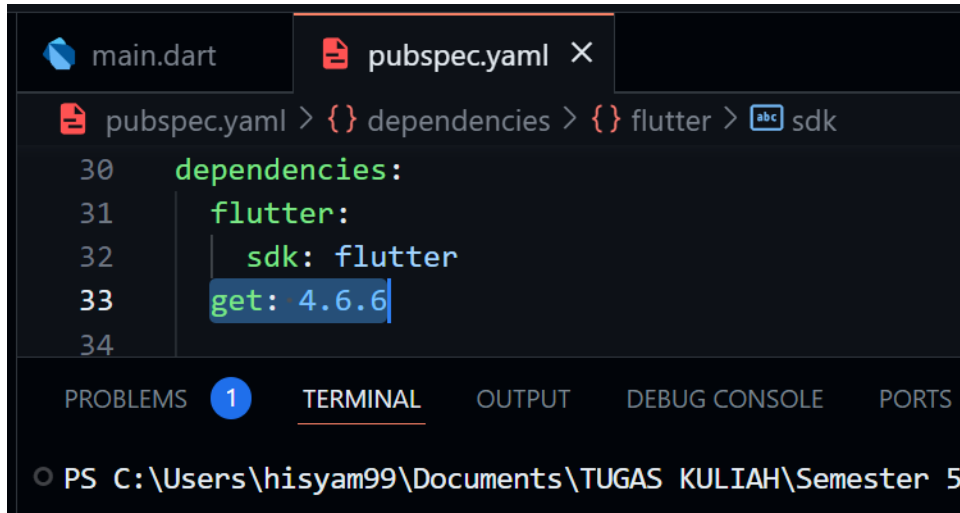
Dibawah ini adalah tampilan keseluruhan dari project flutter yang sudah dibuat



III. Instalasi State Management GetX

Agar memudahkan pengelolaan state, dependency injection, navigasi, dan lainnya, kita perlu menginstall state management getx. Langkah – langkahnya yaitu sebagai berikut:

1. Buka file "pubspec.yaml" pada project, lalu tambahkan dependensi GetX dan save:

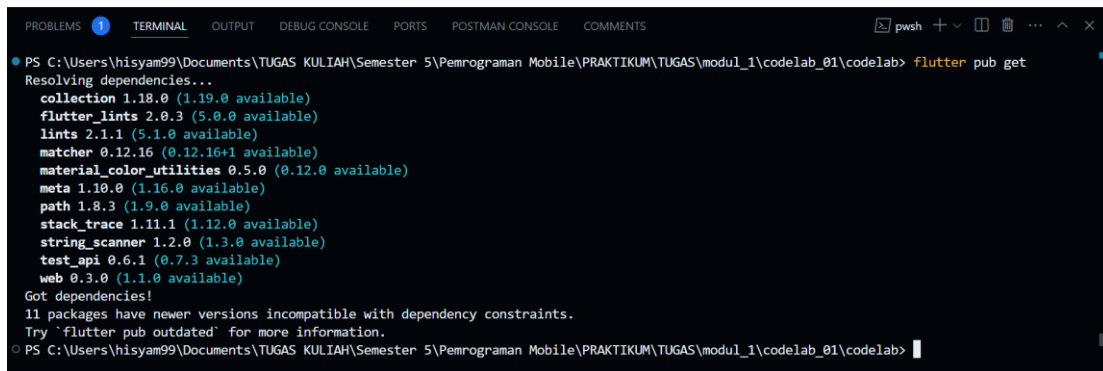


```
main.dart | pubspec.yaml X
pubspec.yaml > {} dependencies > {} flutter > abc sdk
30 dependencies:
31   flutter:
32     sdk: flutter
33   get: 4.6.6
34
```

PROBLEMS 1 | TERMINAL | OUTPUT | DEBUG CONSOLE | PORTS

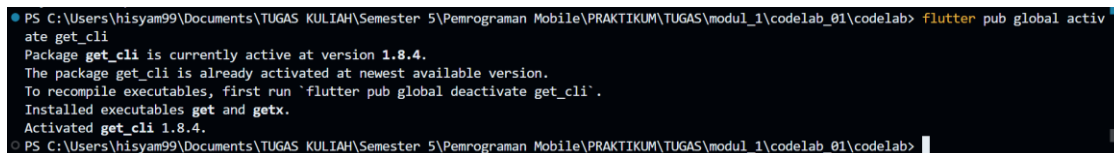
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\

2. Jalankan command "flutter pub get" pada terminal di dalam folder project tersebut untuk melakukan instalasi GetX



```
PROBLEMS 1 | TERMINAL | OUTPUT | DEBUG CONSOLE | PORTS | POSTMAN CONSOLE | COMMENTS
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\codelab_01\codelab> flutter pub get
Resolving dependencies...
collection 1.18.0 (1.19.0 available)
flutter_lints 2.0.3 (5.0.0 available)
lints 2.1.1 (5.1.0 available)
matcher 0.12.16 (0.12.16+1 available)
material_color_utilities 0.5.0 (0.12.0 available)
meta 1.10.0 (1.16.0 available)
path 1.8.3 (1.9.0 available)
stack_trace 1.11.1 (1.12.0 available)
string_scanner 1.2.0 (1.3.0 available)
test_api 0.6.1 (0.7.3 available)
web 0.3.0 (1.1.0 available)
Got dependencies!
11 packages have newer versions incompatible with dependency constraints.
Try 'flutter pub outdated' for more information.
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\codelab_01\codelab>
```

3. Agar memudahkan kita untuk melakukan instalasi GetX pada project flutter yang sudah ada, atau jika ingin membuat baru, kita bisa menginstall GET-CLI secara global pada OS kita. Yaitu dengan menjalankan command "flutter pub global activate get_cli"



```
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\codelab_01\codelab> flutter pub global activate get_cli
Package get_cli is currently active at version 1.8.4.
The package get_cli is already activated at newest available version.
To recompile executables, first run 'flutter pub global deactivate get_cli'.
Installed executables get and getx.
Activated get_cli 1.8.4.
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\codelab_01\codelab>
```

IV. Menerapkan Architecture Pattern GetX

1. Jalankan command "get init" untuk meng-inisialisasi GetX secara interaktif pada project flutter kita

```
PS C:\Users\hisyam99\Documents\TUGAS KULIAH\Semester 5\Pemrograman Mobile\PRAKTIKUM\TUGAS\modul_1\codelab_01\codelab> get init

1) GetX Pattern (by Kauê)
2) CLEAN (by Arktekko)
Which architecture do you want to use? [1]

1) Yes!
2) No
Your lib folder is not empty. Are you sure you want to overwrite your application?
WARNING: This action is irreversible [1]
✓ 'Package: get installed!
✓ File: main.dart created successfully at path: lib\main.dart
✓ File: home_controller.dart created successfully at path: lib\app\modules\home\controllers\home_controller.dart
✓ File: home_view.dart created successfully at path: lib\app\modules\home\views\home_view.dart
✓ File: home_binding.dart created successfully at path: lib\app\modules\home\bindings\home_binding.dart
✓ File: app_routes.dart created successfully at path: lib\app\routes\app_routes.dart
✓ File: app_pages.dart created successfully at path: lib\app\routes\app_pages.dart
✓ home route created successfully.
✓ Home page created successfully.
✓ GetX Pattern structure successfully generated.
```

Running `flutter pub get` ...

```
$ flutter pub get
```


● Resolving dependencies...

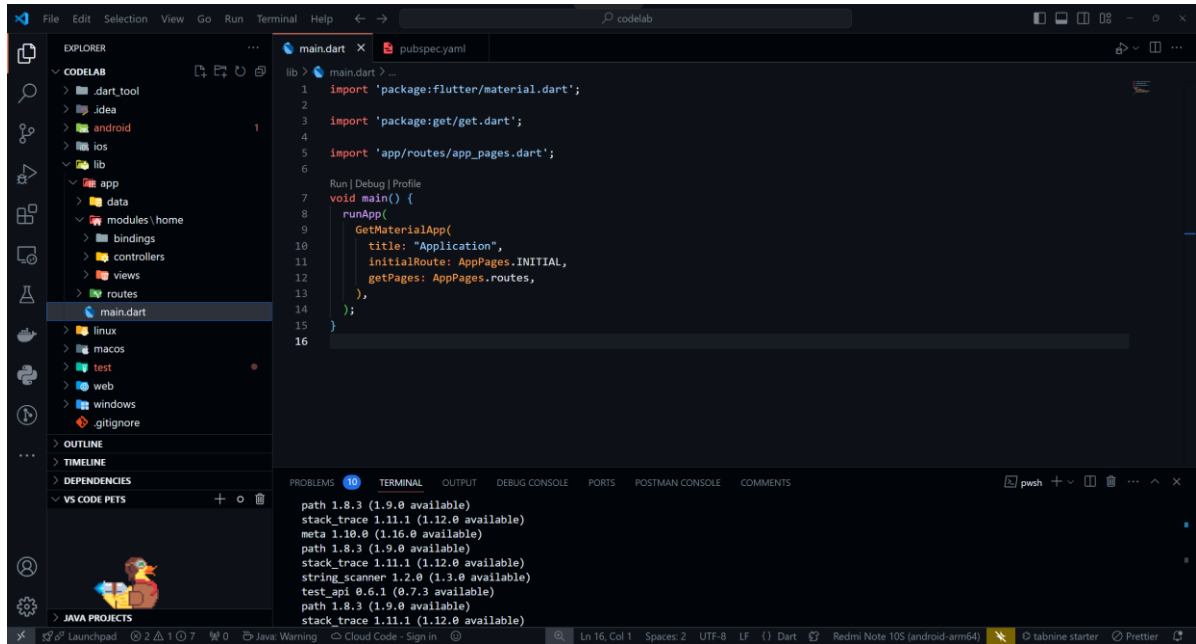
```
collection 1.18.0 (1.19.0 available)
flutter_lints 2.0.3 (5.0.0 available)
lints 2.1.1 (5.1.0 available)
matcher 0.12.16 (0.12.16+1 available)
material_color_utilities 0.5.0 (0.12.0 available)
meta 1.10.0 (1.16.0 available)
path 1.8.3 (1.9.0 available)
stack_trace 1.11.1 (1.12.0 available)
string_scanner 1.2.0 (1.3.0 available)
test_api 0.6.1 (0.7.3 available)
web 0.3.0 (1.1.0 available)
```

Got dependencies!

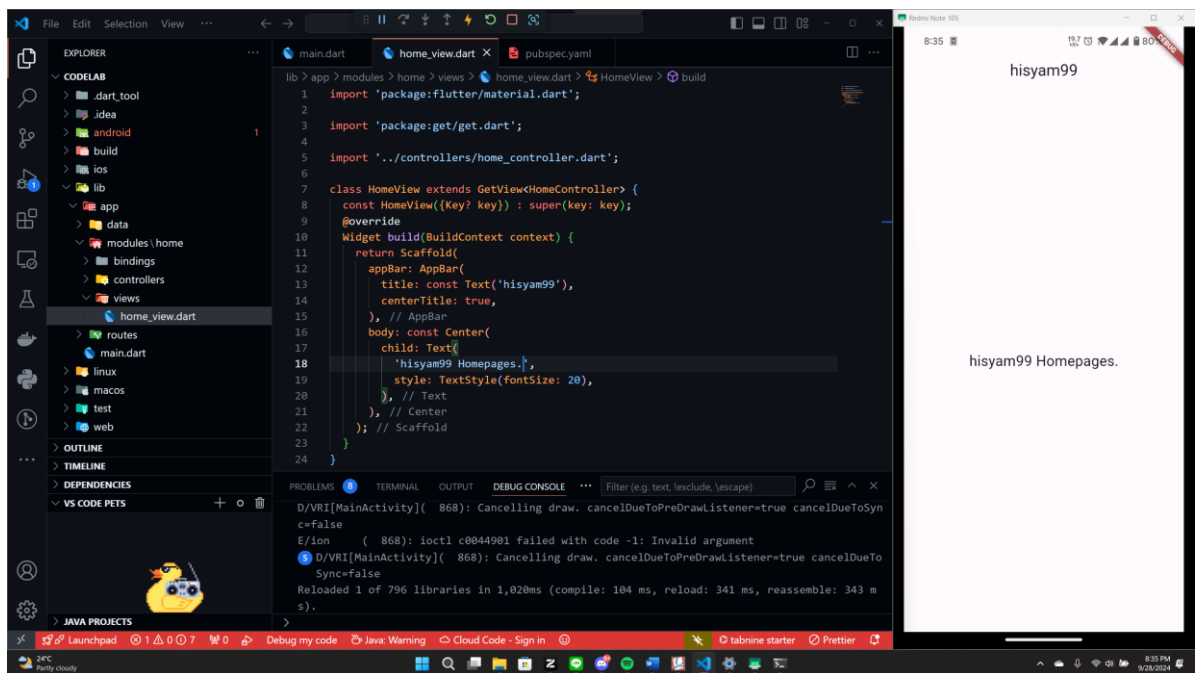
11 packages have newer versions incompatible with dependency constraints.

Try `flutter pub outdated` for more information.

2. Setelah melakukan "get init" maka struktur project flutter kita otomatis akan berubah, yang sebelumnya hanya ada file main.dart saja di folder /lib , sekarang ada folder baru, yaitu  app, yang berisi controller, bindings, views, routes, dan lain – lain. Pada main.dart juga akan berubah dan terlihat lebih simpel. Sekarang project flutter kita sudah siap untuk digunakan!.



Dibawah adalah tampilan ketika kita mencoba melakukan run debug pada project flutter kita:



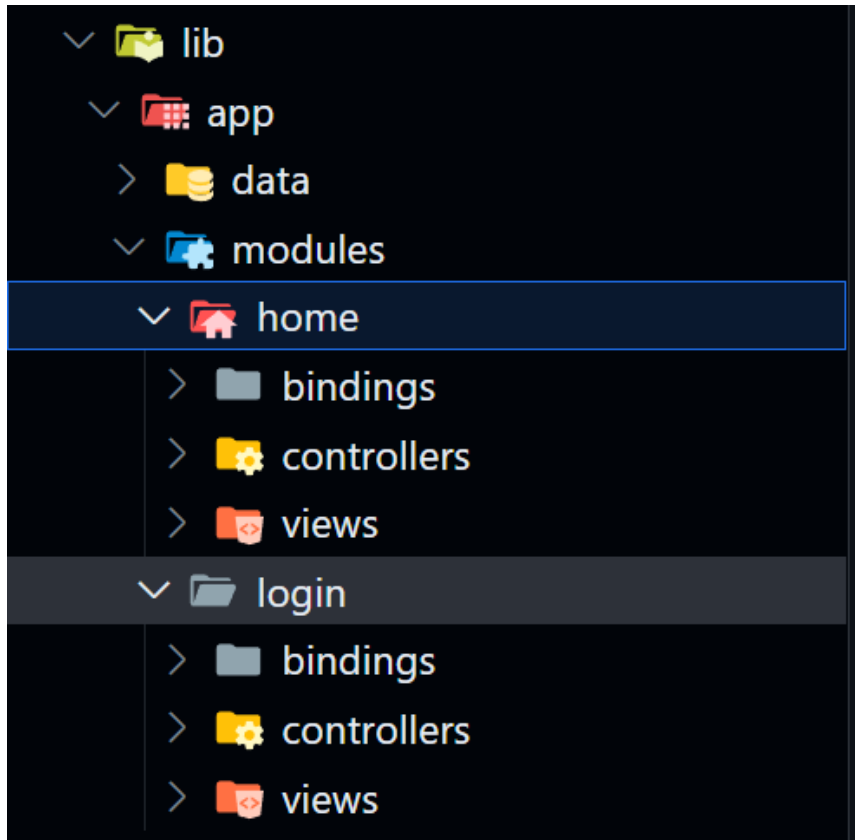
Saya menggunakan ADB Debugging untuk melakukan debugging pada project flutter tersebut.

V. Refactor Code dan Menambah Route Baru

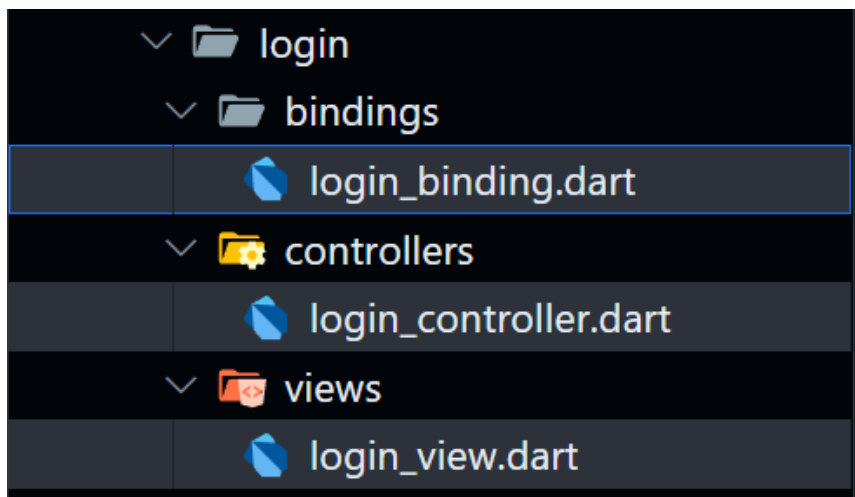
Pada project flutter tadi, kita sudah menerapkan Architecture Pattern GetX, jadi untuk penamaan variable, struktur folder dan lainnya sudah ditangani otomatis secara langsung

oleh GetX tersebut. Kali ini saya ingin membuat Login Page dan ada button untuk menuju Login Page pada Home page saya. Langkah – langkahnya yaitu sebagai berikut:

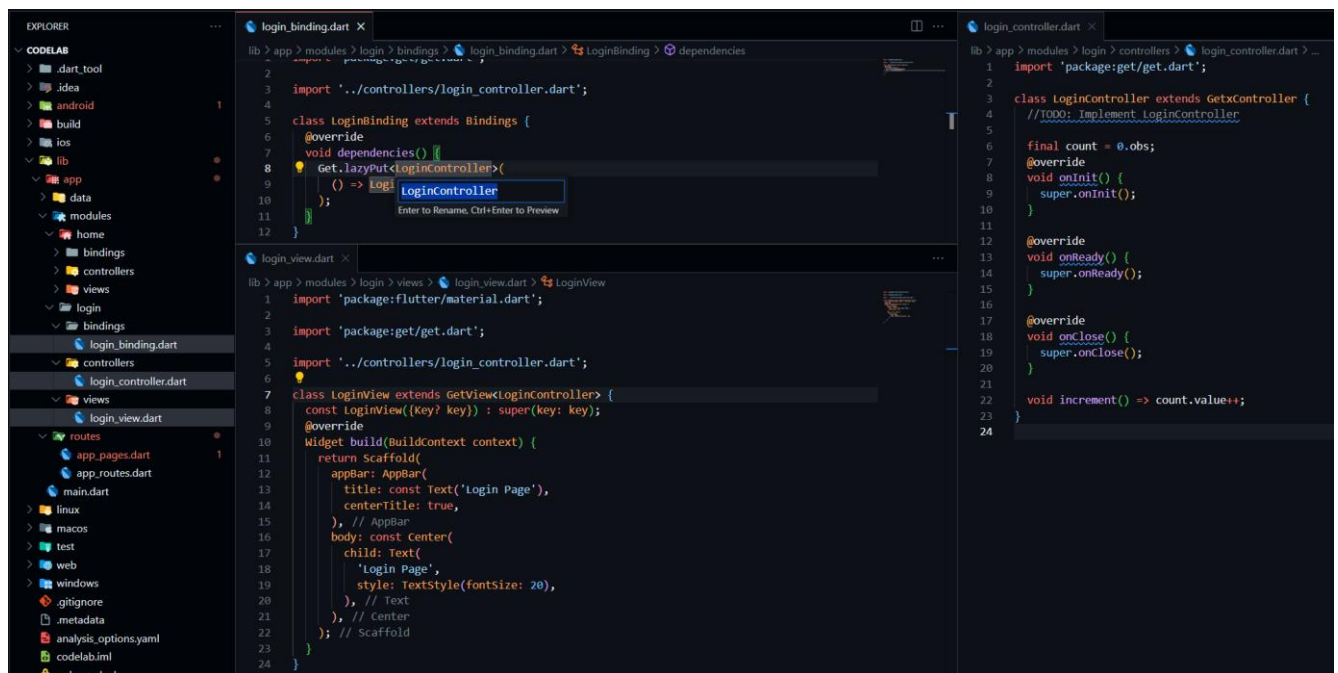
1. Duplikat folder “/lib/app/modules/home” dan rename menjadi “/lib/app/modules/login”, saya melakukan ini agar mempercepat proses pembuatan Login Pagenya



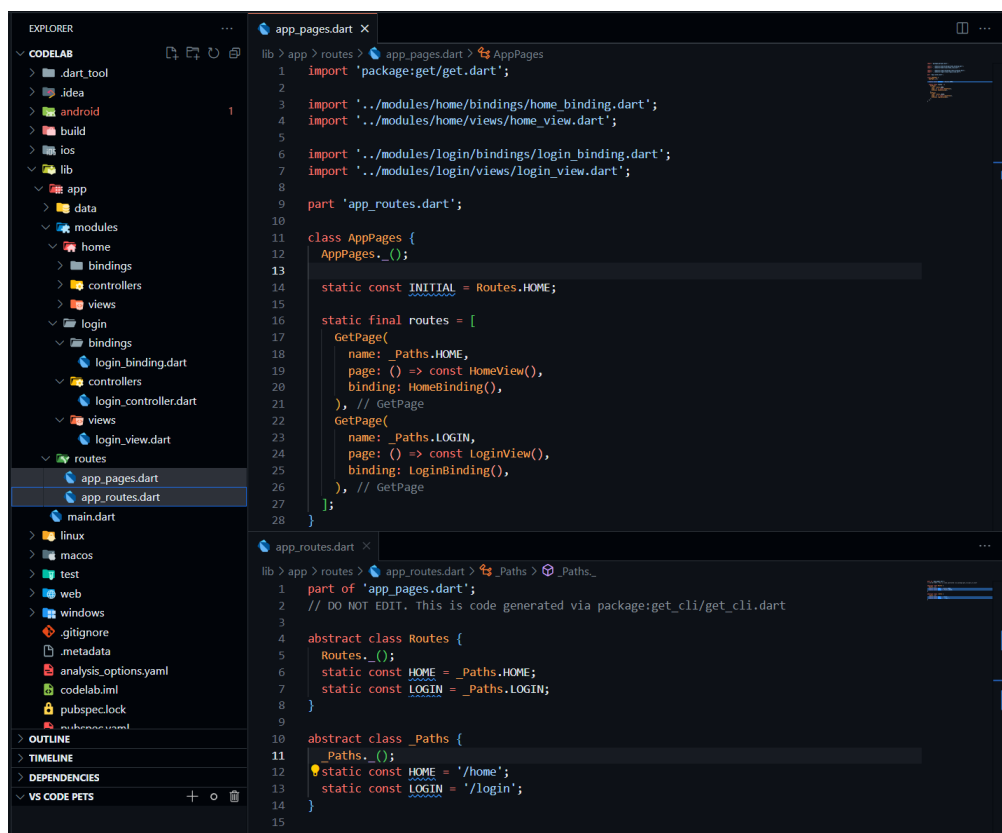
2. Lakukan refactoring pada subfolder “/lib/app/modules/login”, menjadi seperti yang ada pada gambar dibawah ini



3. Modifikasi beberapa kode pada file – file tersebut menjadi seperti dibawah ini:

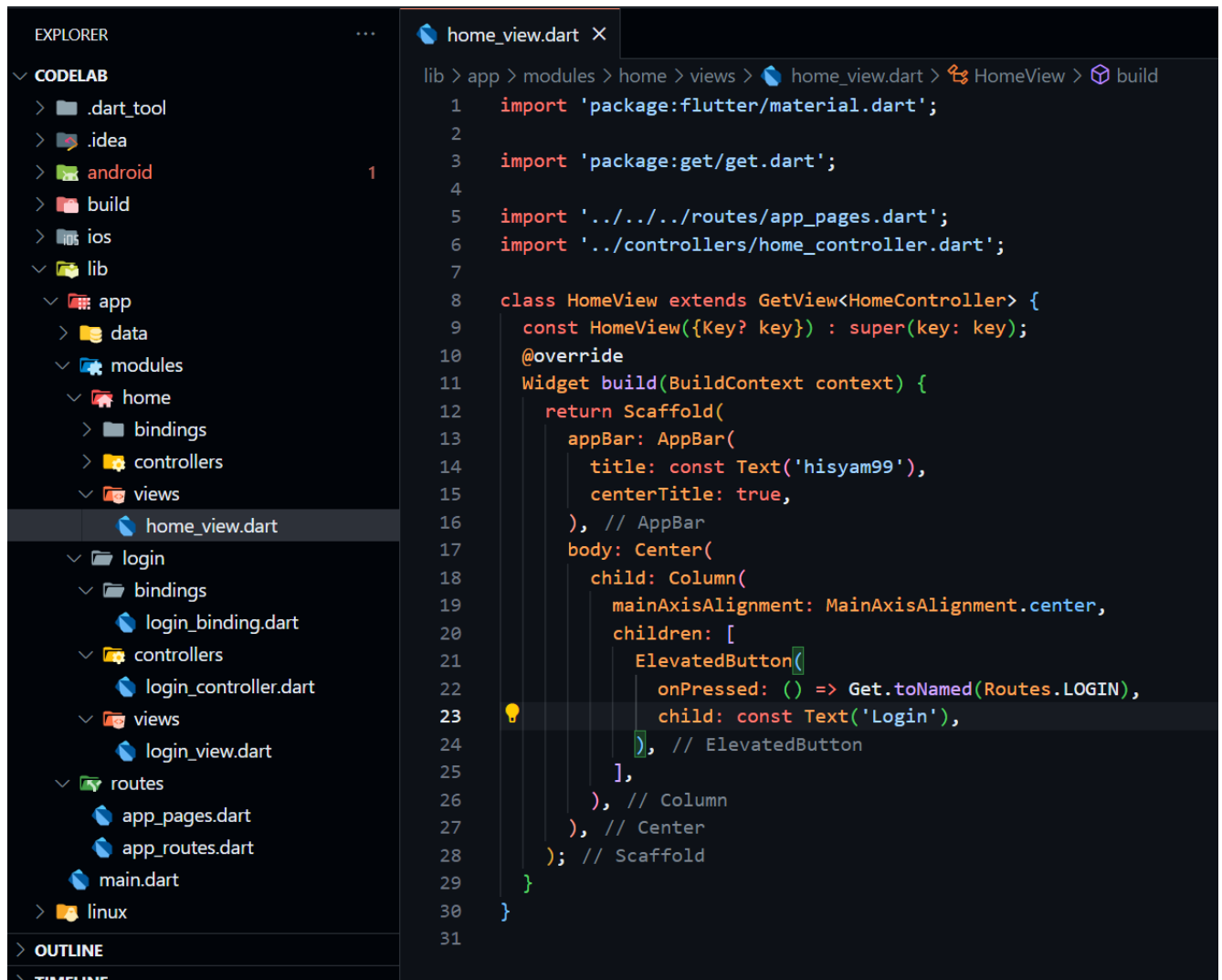


login_view.dart, login_binding.dart, login_controller.dart



app_pages.dart, app_routes.dart

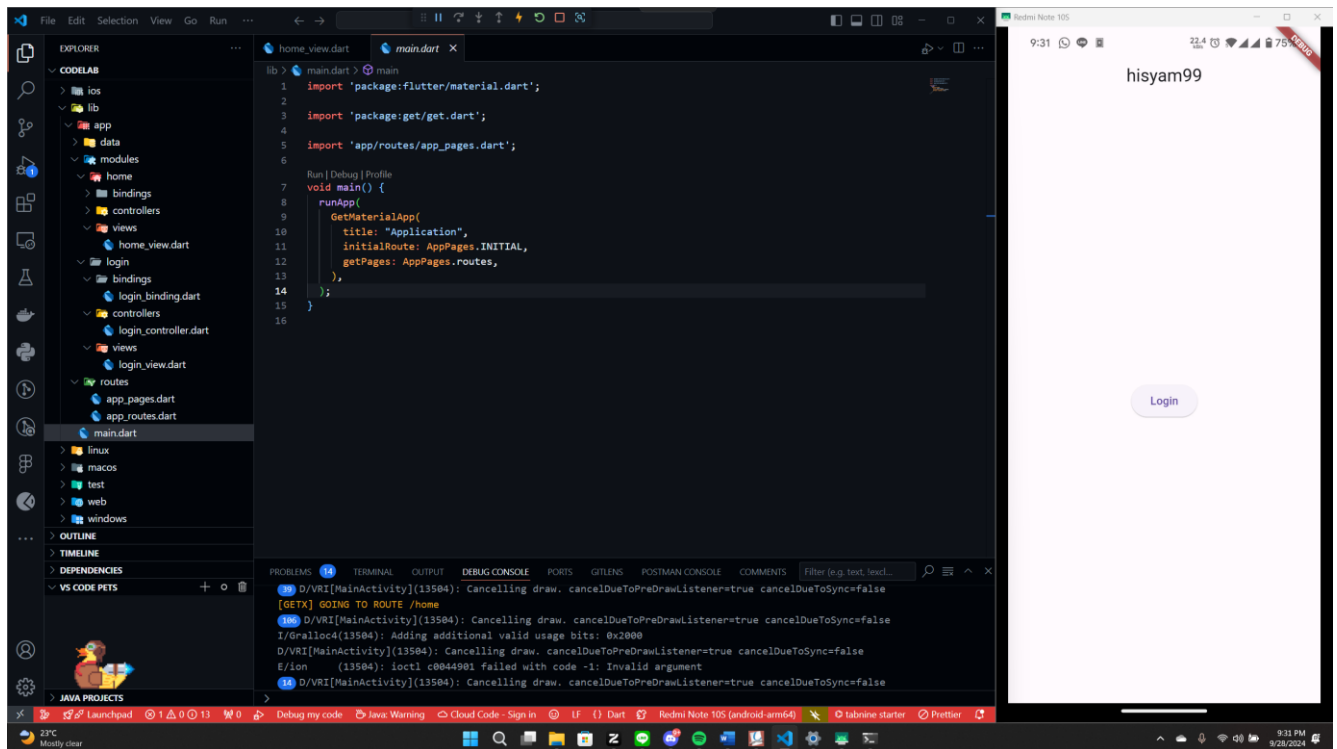
4. Rubah “home_view.dart” yaitu dengan menambahkan button Login yang dimana button tersebut ketika ditekan maka akan menuju ke Login Page yang sudah kita buat sebelumnya. Untuk kode yang ditambahkan yaitu sebagai berikut:



```
lib > app > modules > home > views > home_view.dart > HomeView > build
1  import 'package:flutter/material.dart';
2
3  import 'package:get/get.dart';
4
5  import '../routes/app_pages.dart';
6  import '../controllers/home_controller.dart';
7
8  class HomeView extends GetView<HomeController> {
9    const HomeView({Key? key}) : super(key: key);
10   @override
11   Widget build(BuildContext context) {
12     return Scaffold(
13       appBar: AppBar(
14         title: const Text('hisyam99'),
15         centerTitle: true,
16       ), // AppBar
17       body: Center(
18         child: Column(
19           mainAxisAlignment: MainAxisAlignment.center,
20           children: [
21             ElevatedButton(
22               onPressed: () => Get.toNamed(Routes.LOGIN),
23               child: const Text('Login'),
24             ), // ElevatedButton
25           ],
26         ), // Column
27       ), // Center
28     ); // Scaffold
29   }
30 }
31
```

/lib/app/modules/home/views/home_view.dart

5. Setelah kita menambahkan Login Page dan Button Login, saatnya kita tes semua kode yang sudah kita tambahkan tadi dengan kembali ke main.dart dan melakukan Run Debugging, jika berhasil maka menjadi seperti berikut ini:



Ketika tombol Login tersebut diklik, maka akan menuju ke Login Page seperti pada gambar dibawah ini

