

Tugas 6-1

Nama : Feri Kusuma Wardhana

NIM : 2113010226

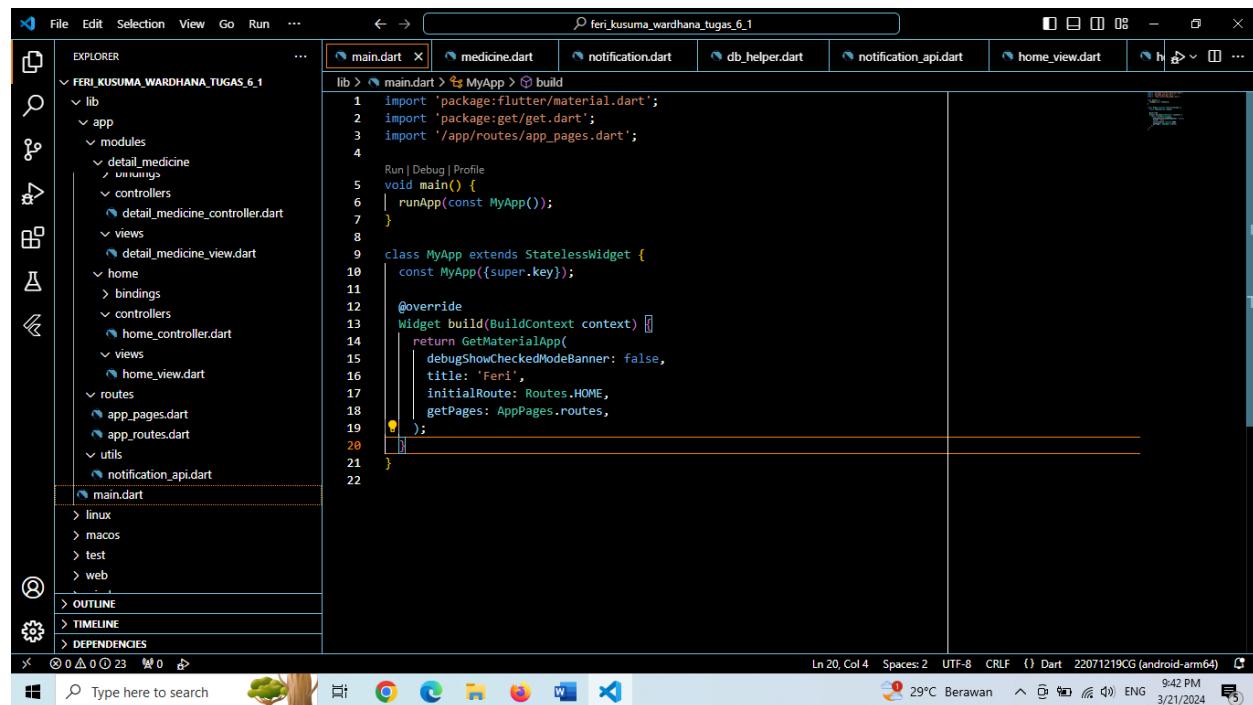
Grup : 1B

Saya telah mempersiapkan dengan kode ini ke git hub akan tetapi untuk folder build saya hapus karena file yang besar, berikut adalah linknya

Github untuk kode ini : <https://github.com/ferikw/Tugas-6-1-Feri-Kusuma-Wardhana.git>

Saya telah melakukan screenshot dari kode dan hasil tampilan dari kode yang telah saya buat , Saya telah membuat kode sesuai tugas yang ada di 6-1 , kodenya adalah sebagai berikut , saya juga sudah menambahkan google font didalam kodenya , saya menggunakan google font poppins , berikut adalah keseluruhan kodenya

1. Kode pada main.dart



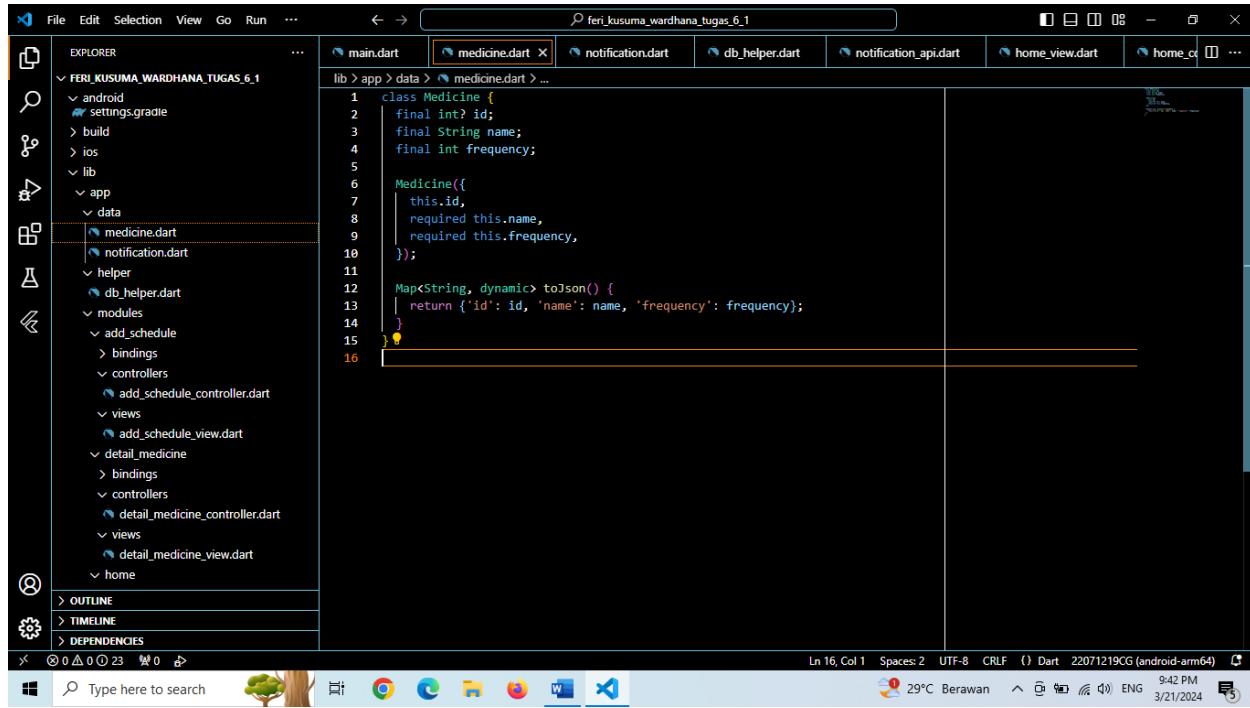
```
import 'package:flutter/material.dart';
import 'package:get/get.dart';
import '/app/routes/app_pages.dart';

void main() {
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

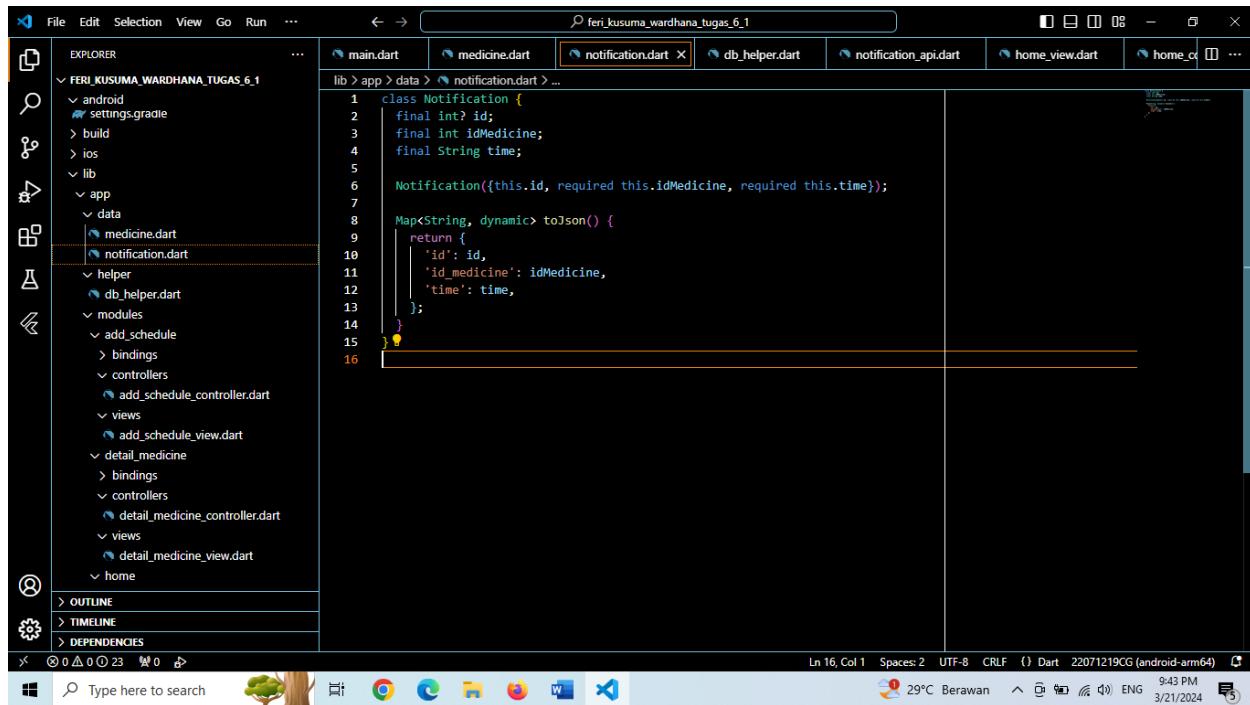
  @override
  Widget build(BuildContext context) {
    return GetMaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Feri',
      initialRoute: Routes.HOME,
      getPages: AppPages.routes,
    );
  }
}
```

2. Berikut adalah kode medicine.dart



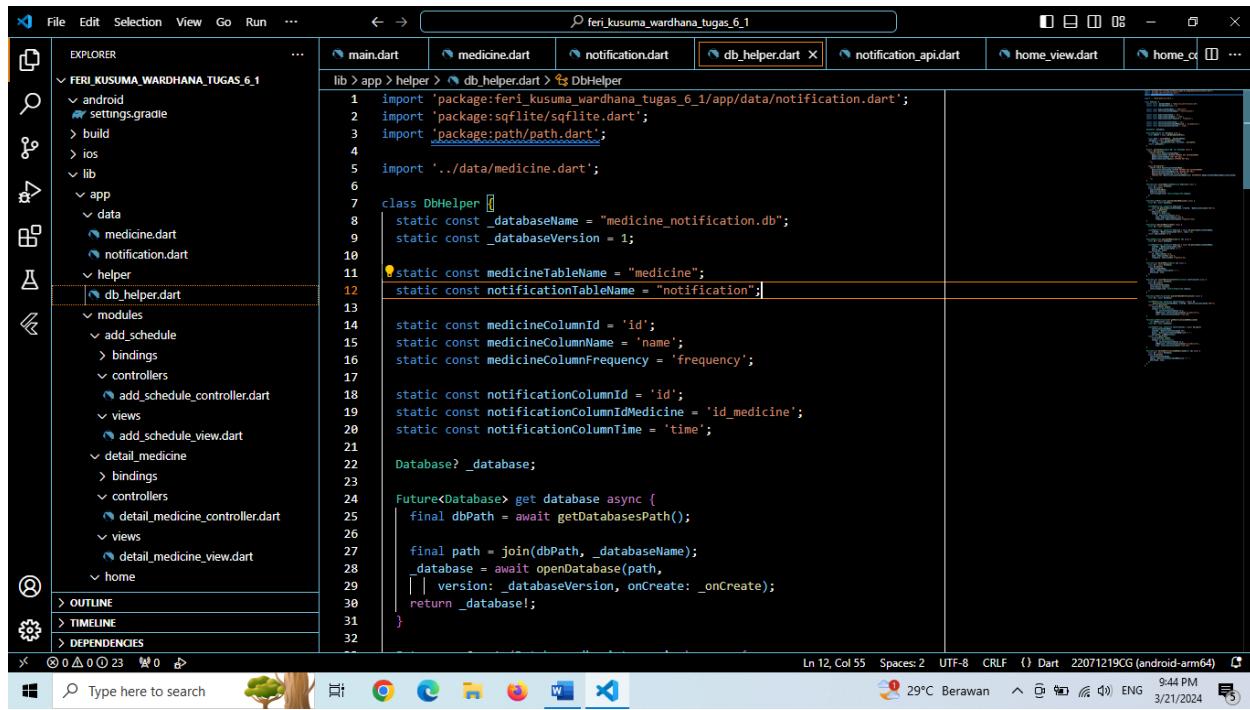
```
class Medicine {  
  final int? id;  
  final String name;  
  final int frequency;  
  
  Medicine({  
    required this.id,  
    required this.name,  
    required this.frequency,  
  });  
  
  Map<String, dynamic> toJson() {  
    return {'id': id, 'name': name, 'frequency': frequency};  
  }  
}
```

3. Berikut adalah kode dari notification.dart



```
class Notification {  
  final int? id;  
  final int idMedicine;  
  final String time;  
  
  Notification({this.id, required this.idMedicine, required this.time});  
  
  Map<String, dynamic> toJson() {  
    return {  
      'id': id,  
      'id_medicine': idMedicine,  
      'time': time,  
    };  
  }  
}
```

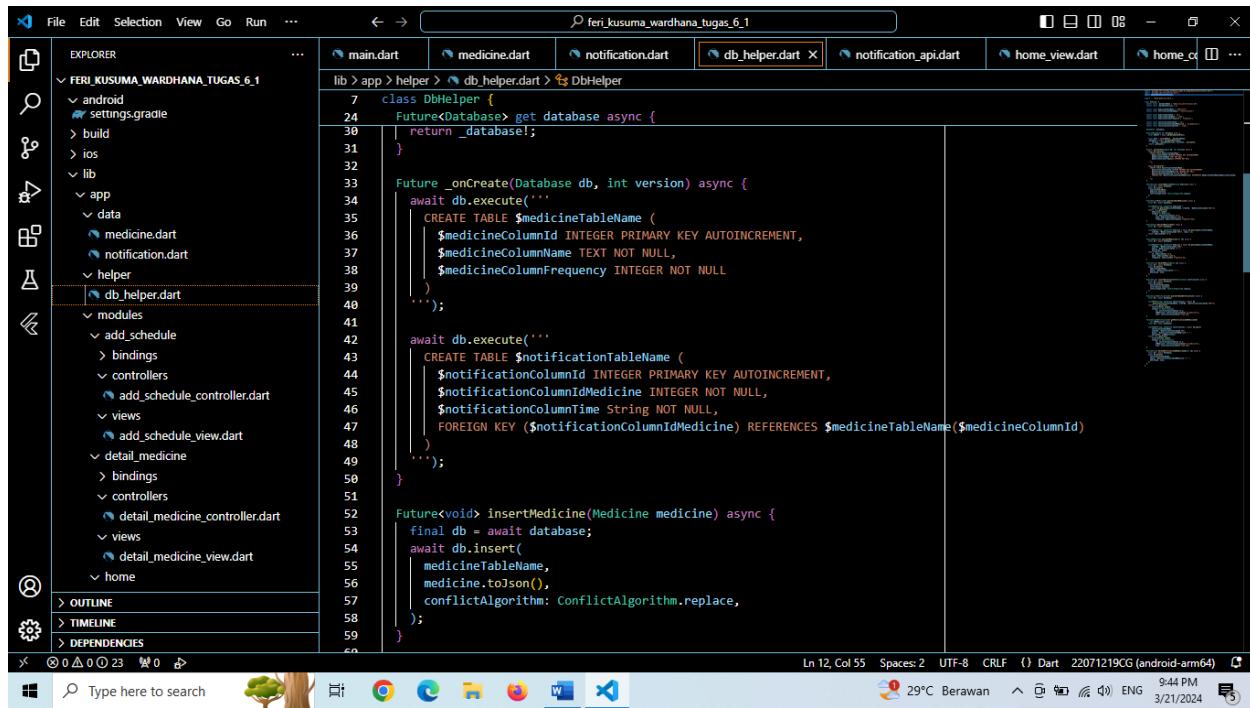
4. Berikut adalah kode dari db_helper



```
File Edit Selection View Go Run ... ⏪ ⏩ feri_kusuma_wardhana_tugas_6_1
EXPLORER lib > app > helper > db_helper.dart > DbHelper
1 import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/notification.dart';
2 import 'package:sqflite/sqflite.dart';
3 import 'package:path/path.dart';
4
5 import '../data/medicine.dart';
6
7 class DBHelper {
8     static const _databaseName = "medicine_notification.db";
9     static const _databaseVersion = 1;
10
11     static const medicineTableName = "medicine";
12     static const notificationTableName = "notification";
13
14     static const medicineColumnId = 'id';
15     static const medicineColumnName = 'name';
16     static const medicineColumnFrequency = 'frequency';
17
18     static const notificationColumnId = 'id';
19     static const notificationColumnIdMedicine = 'id_medicine';
20     static const notificationColumnTime = 'time';
21
22     Database? _database;
23
24     Future<Database> get database async {
25         final dbPath = await getDatabasesPath();
26
27         final path = join(dbPath, _databaseName);
28         _database = await openDatabase(path,
29             | version: databaseVersion, onCreate: _onCreate);
30         return _database!;
31     }
32 }
```

Ln 12, Col 55 Spaces: 2 UTF-8 CRLF () Dart 22071219CG (android-arm64) ⌂

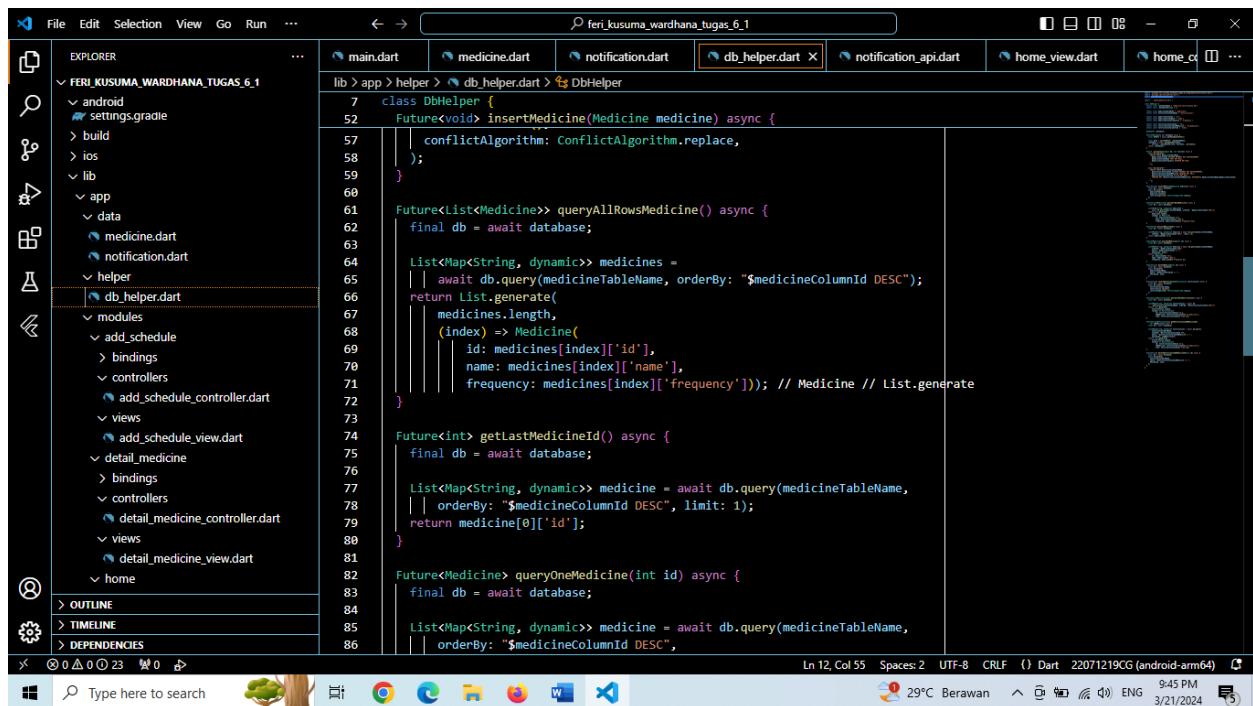
Type here to search 29°C Berawan 9:44 PM ENG 3/21/2024



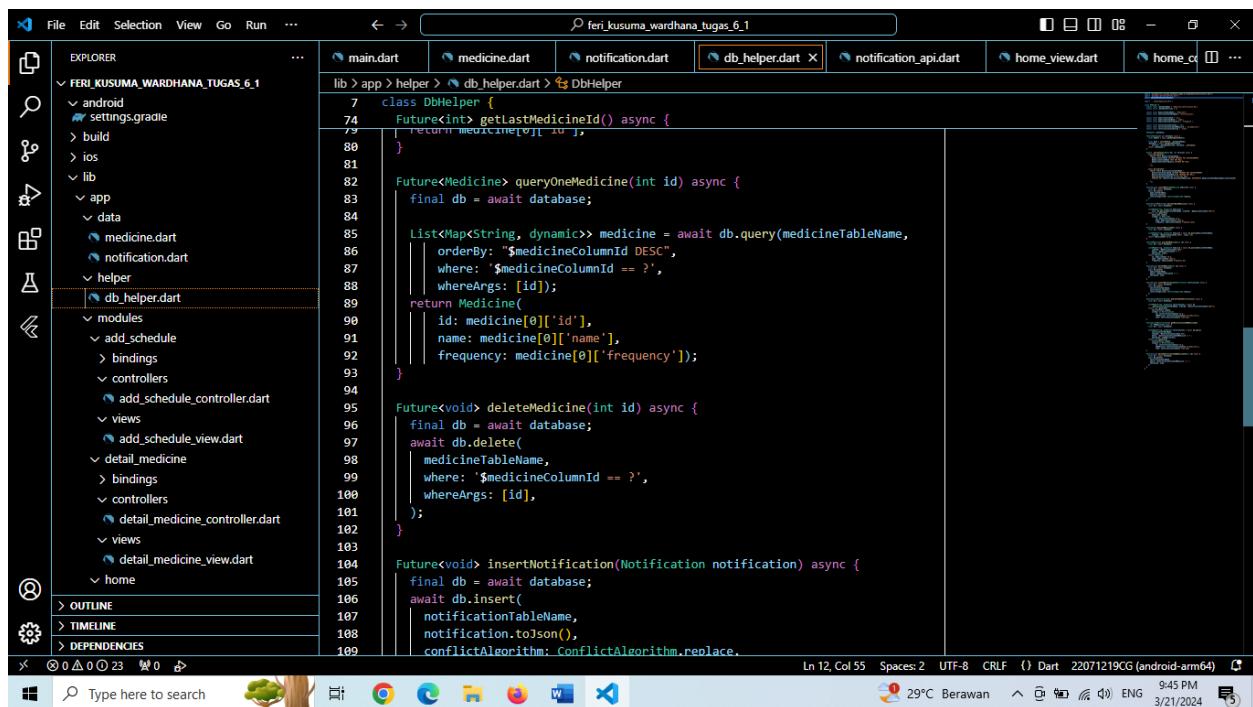
```
File Edit Selection View Go Run ... ⏪ ⏩ feri_kusuma_wardhana_tugas_6_1
EXPLORER lib > app > helper > db_helper.dart > DbHelper
7 class DBHelper {
24     Future<Database> get database async {
30     | return _database!;
31     }
32
33     Future _onCreate(Database db, int version) async {
34         await db.execute('''
35             CREATE TABLE $medicineTableName (
36                 $medicineColumnId INTEGER PRIMARY KEY AUTOINCREMENT,
37                 $medicineColumnName TEXT NOT NULL,
38                 $medicineColumnFrequency INTEGER NOT NULL
39             );
40         ''');
41
42         await db.execute('''
43             CREATE TABLE $notificationTableName (
44                 $notificationColumnId INTEGER PRIMARY KEY AUTOINCREMENT,
45                 $notificationColumnIdMedicine INTEGER NOT NULL,
46                 $notificationColumnTime String NOT NULL,
47                 FOREIGN KEY ($notificationColumnIdMedicine) REFERENCES $medicineTableName($medicineColumnId)
48             );
49         ''');
50
51     }
52
53     Future<void> insertMedicine(Medicine medicine) async {
54         final db = await database;
55         await db.insert(
56             medicineTableName,
57             medicine.toJson(),
58             conflictAlgorithm: ConflictAlgorithm.replace,
59         );
60     }
61 }
```

Ln 12, Col 55 Spaces: 2 UTF-8 CRLF () Dart 22071219CG (android-arm64) ⌂

Type here to search 29°C Berawan 9:44 PM ENG 3/21/2024



```
lib > app > helper > db_helper.dart > DBHelper
7   class DBHelper {
52     Future<void> insertMedicine(Medicine medicine) async {
57       conflictAlgorithm: ConflictAlgorithm.replace,
58     );
59   }
60
61   Future<List<Medicine>> queryAllRowsMedicine() async {
62     final db = await database;
63
64     List<Map<String, dynamic>> medicines =
65     || await db.query(medicineTableName, orderBy: "$medicineColumnId DESC");
66     return List.generate(
67       medicines.length,
68       (index) => Medicine(
69         id: medicines[index]['id'],
70         name: medicines[index]['name'],
71         frequency: medicines[index]['frequency']));
72   }
73
74   Future<int> getLastMedicineId() async {
75     final db = await database;
76
77     List<Map<String, dynamic>> medicine = await db.query(medicineTableName,
78     || orderBy: "$medicineColumnId DESC", limit: 1);
79     return medicine[0]['id'];
80   }
81
82   Future<Medicine> queryOneMedicine(int id) async {
83     final db = await database;
84
85     List<Map<String, dynamic>> medicine = await db.query(medicineTableName,
86     || orderBy: "$medicineColumnId DESC",
87     where: '$medicineColumnId == ?',
88     whereArgs: [id]);
89     return Medicine(
90       id: medicine[0]['id'],
91       name: medicine[0]['name'],
92       frequency: medicine[0]['frequency']);
93   }
94
95   Future<void> deleteMedicine(int id) async {
96     final db = await database;
97     await db.delete(
98       medicineTableName,
99       where: '$medicineColumnId == ?',
100      whereArgs: [id],
101    );
102  }
103
104  Future<void> insertNotification(Notification notification) async {
105    final db = await database;
106    await db.insert(
107      notificationTableName,
108      notification.toJson(),
109      conflictAlgorithm: ConflictAlgorithm.replace,
110    );
111  }
112}
```



```
lib > app > helper > db_helper.dart > DBHelper
7   class DBHelper {
74     Future<int> getLastMedicineId() async {
75       return medicine[0]['id'];
76     }
77
78     Future<Medicine> queryOneMedicine(int id) async {
79       final db = await database;
80
81       List<Map<String, dynamic>> medicine = await db.query(medicineTableName,
82       || orderBy: "$medicineColumnId DESC",
83       where: '$medicineColumnId == ?',
84       whereArgs: [id]);
85       return Medicine(
86         id: medicine[0]['id'],
87         name: medicine[0]['name'],
88         frequency: medicine[0]['frequency']);
89     }
90
91     Future<void> deleteMedicine(int id) async {
92       final db = await database;
93       await db.delete(
94         medicineTableName,
95         where: '$medicineColumnId == ?',
96         whereArgs: [id],
97       );
98     }
99
100    Future<void> insertNotification(Notification notification) async {
101      final db = await database;
102      await db.insert(
103        notificationTableName,
104        notification.toJson(),
105        conflictAlgorithm: ConflictAlgorithm.replace,
106      );
107    }
108  }
109
```

```
lib > app > helper > db_helper.dart > DBHelper
7   class DBHelper {
104     Future<void> insertNotification(Notification notification) async {
108       notification.toJson(),
109       conflictAlgorithm: ConflictAlgorithm.replace,
110     );
111   }
112
113   Future<List<Notification>> queryAllRowsNotification() async {
114     final db = await database;
115
116     List<Map<String, dynamic>> notifications = await db
117     | .query(notificationTableName, orderBy: "$notificationColumnId DESC");
118     return List.generate(
119       notifications.length,
120       (index) => Notification(
121         id: notifications[index]['id'],
122         idMedicine: notifications[index]['id_medicine'],
123         time: notifications[index]['time']));
124   }
125
126   Future<List<Notification>> getNotificationsByMedicineId(
127     int idMedicine) async {
128     final db = await database;
129
130     List<Map<String, dynamic>> notifications = await db.query(
131       notificationTableName,
132       orderBy: "$notificationColumnId ASC",
133       where: '$notificationColumnIdMedicine == ?',
134       whereArgs: [idMedicine]);
135
136     return List.generate(
137       notifications.length,
138       (index) => Notification(
139         id: notifications[index]['id'],
140         idMedicine: notifications[index]['id_medicine'],
141         time: notifications[index]['time']));
142   }
143
144   Future<void> deleteNotificationByMedicineId(int id) async {
145     final db = await database;
146     await db.delete(
147       notificationTableName,
148       where: '$notificationColumnIdMedicine == ?',
149       whereArgs: [id],
150     );
151   }
152 }
```

```
lib > app > helper > db_helper.dart > DBHelper
7   class DBHelper {
113     Future<List<Notification>> queryAllRowsNotification() async {
114     | | | | | time: notifications[index]['time']);
115   }
116
117   Future<List<Notification>> getNotificationsByMedicineId(
118     int idMedicine) async {
119     final db = await database;
120
121     List<Map<String, dynamic>> notifications = await db.query(
122       notificationTableName,
123       orderBy: "$notificationColumnId ASC",
124       where: '$notificationColumnIdMedicine == ?',
125       whereArgs: [idMedicine]);
126
127     return List.generate(
128       notifications.length,
129       (index) => Notification(
130         id: notifications[index]['id'],
131         idMedicine: notifications[index]['id_medicine'],
132         time: notifications[index]['time']));
133   }
134
135   Future<void> deleteNotificationByMedicineId(int id) async {
136     final db = await database;
137     await db.delete(
138       notificationTableName,
139       where: '$notificationColumnIdMedicine == ?',
140       whereArgs: [id],
141     );
142   }
143
144   Future<void> insertNotification(Notification notification) async {
145     notification.toJson(),
146     conflictAlgorithm: ConflictAlgorithm.replace,
147   );
148 }
```

5. Berikut adalah kode notification api

The screenshot shows the VS Code interface with the following details:

- File Explorer (Left):** Shows the project structure:
 - lib
 - app
 - modules
 - add_schedule
 - views
 - detail_medicine
 - bindings
 - controllers
 - detail_medicine_controller.dart
 - views
 - detail_medicine_view.dart
 - home
 - bindings
 - controllers
 - home_controller.dart
 - views
 - home_view.dart
 - routes
 - app_pages.dart
 - app_routes.dart
 - utils
 - notification_api.dart
 - linux
 - macos
 - tests
- Search Bar:** Type here to search
- Code Editor (Right):** Displays the content of notification_api.dart. The code handles local notifications using the flutter_local_notifications package.

```
1 import 'package:flutter_local_notifications/flutter_local_notifications.dart';
2 import 'package:timezone/data/latest.dart' as tzdata;
3 import 'package:timezone/timezone.dart' as tz;
4
5 class NotificationApi {
6     static final _notification = FlutterLocalNotificationsPlugin();
7
8     static Future cancelNotification(int id) => _notification.cancel(id);
9     static Future cancelAllNotification() => _notification.cancelAll();
10    static Future cancelNotificationByTag(String tag) =>
11        _notification.cancel(0, tag: tag);
12
13    static Future init({
14        bool initScheduled = false,
15    }) async {
16        const androidSettings =
17            AndroidInitializationSettings('@mipmap/ic_launcher');
18        const settings = InitializationSettings(android: androidSettings);
19        await _notification.initialize(settings);
20
21        tzdata.initializeTimeZones();
22    }
23
24    static Future showNotification({
25        required int id,
26        required String title,
27        required String body,
28        required String payload,
29    }) =>
30        _notification.show(
31            id,
32            title,
```

The screenshot shows a Flutter development environment with the following details:

- File Explorer:** On the left, it lists the project structure. The file `notification_api.dart` is currently selected.
- Code Editor:** The main area displays the `NotificationApi` class from `notification_api.dart`. It includes methods for showing a notification and scheduling a notification.
- Bottom Bar:** Shows the current file path (`feri_kusuma_wardhana.tugas_6_1`), line number (Ln 46), column number (Col 1), and other developer information.
- Bottom Icons:** Includes icons for search, refresh, and various developer tools like browser, terminal, and build status.

```
class NotificationApi { static Future showNotification({ int id, String title, String body, const NotificationDetails( android: AndroidNotificationDetails( 'channel id', 'channel name', channelDescription: 'channel description', importance: Importance.max, priority: Priority.high, showWhen: false, ), // AndroidNotificationDetails ), // NotificationDetails payload: payload, ); static Future scheduledNotification({ required int id, required String title, required String body, required String payload, required Time scheduledDate, }) async { print("enter scheduledNotification"); await _notification.zonedSchedule( id, title, body, _scheduledDaily(scheduledDate), ); } }
```

The screenshot shows a code editor interface with the title bar "feri_kusuma_wardhana_tugas_6_1". The left sidebar displays the project structure under "EXPLORER". The main pane shows the code for a file named "notification_api.dart".

```
lib > app > utils > notification_api.dart > NotificationApi
5 class NotificationApi {
47 static Future scheduledNotification({
58   body,
59   _scheduledDaily(_scheduledDate),
60   const NotificationDetails(
61     android: AndroidNotificationDetails(
62       'channel id',
63       'channel name',
64       channelDescription: 'channel description',
65       importance: Importance.max,
66       priority: Priority.high,
67       showWhen: true,
68     ), // AndroidNotificationDetails
69   ), // NotificationDetails
70   androidAllowWhileIdle: true,
71   uiLocalNotificationDateInterpretation:
72     | | UILocalNotificationDateInterpretation.absoluteTime,
73   payload: payload,
74   matchDateTimeComponents: DateTimeComponents.time,
75 );
76 print("finished scheduledNotification");
77 }
78
static tz.TZDateTime _scheduledDaily(Time time) {
79   print("enter _scheduledDaily");
80   final jakarta = tz.getLocation('Asia/Jakarta');
81   tz.setLocalLocation(jakarta);
82   final now = tz.TZDateTime.now(jakarta);
83   tz.TZDateTime scheduledDate = tz.TZDateTime(tz.local, now.year, now.month,
84   | | now.day, time.hour, time.minute, time.second);
85   print("before return _scheduledDaily");
86 }
```

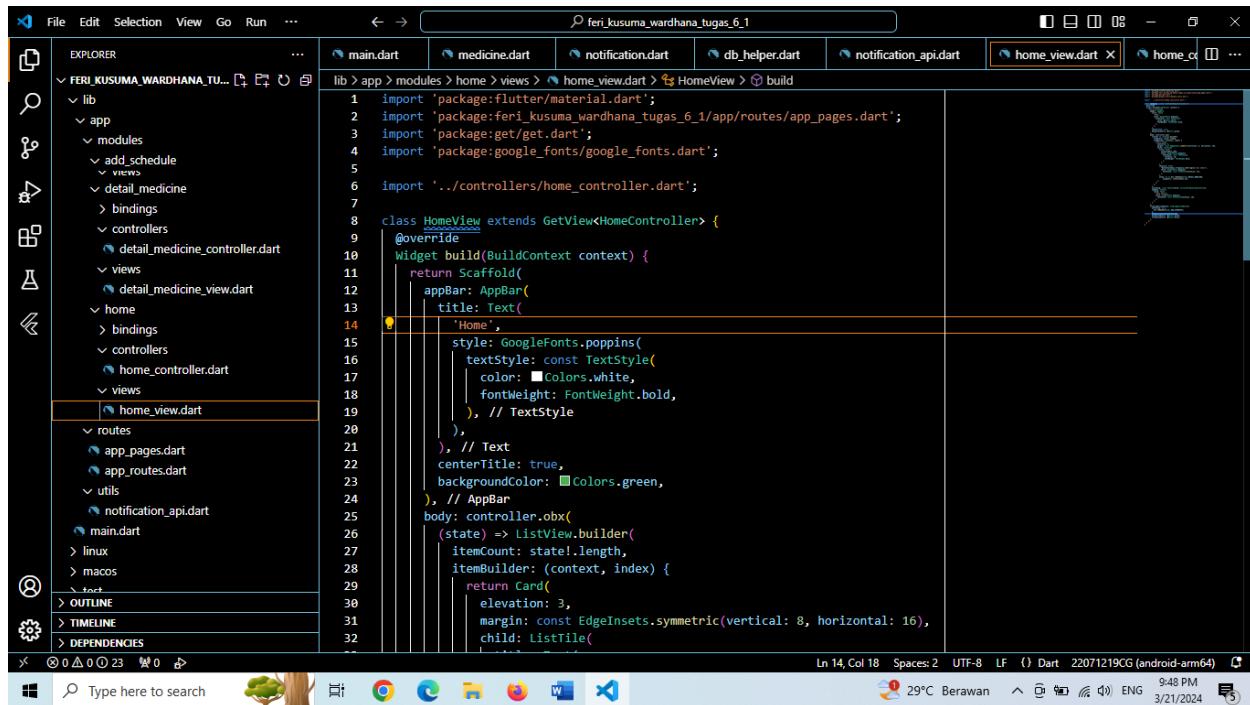
The status bar at the bottom indicates "Ln 46 Col 1 Spaces:2 UTF-8 CRLF {} Dart 22071219CG (android-arm64)" and the date "3/21/2024".

This screenshot is identical to the first one, showing the same code editor interface with the title bar "feri_kusuma_wardhana_tugas_6_1". The left sidebar shows the project structure, and the main pane shows the "notification_api.dart" file with the same Dart code.

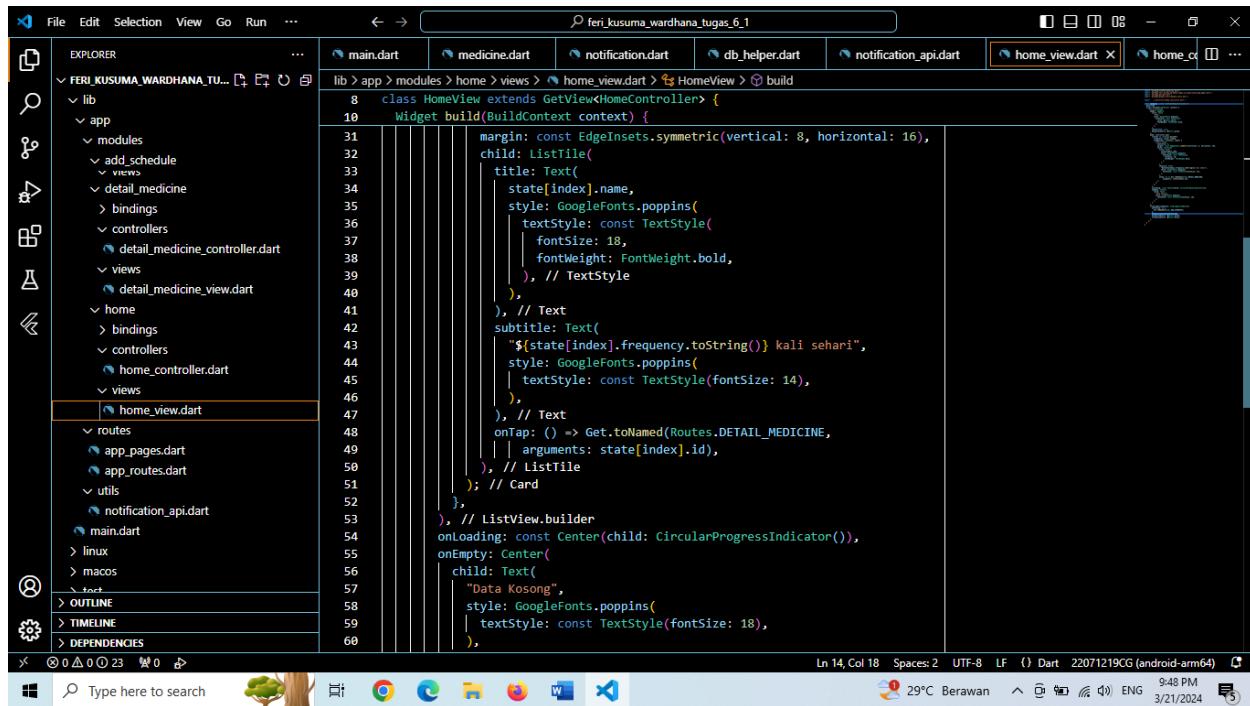
```
lib > app > utils > notification_api.dart > NotificationApi
5 class NotificationApi {
47 static Future scheduledNotification({
58   body,
59   _scheduledDaily(_scheduledDate),
60   const NotificationDetails(
61     android: AndroidNotificationDetails(
62       'channel id',
63       'channel name',
64       channelDescription: 'channel description',
65       importance: Importance.max,
66       priority: Priority.high,
67       showWhen: true,
68     ), // AndroidNotificationDetails
69   ), // NotificationDetails
70   androidAllowWhileIdle: true,
71   uiLocalNotificationDateInterpretation:
72     | | UILocalNotificationDateInterpretation.absoluteTime,
73   payload: payload,
74   matchDateTimeComponents: DateTimeComponents.time,
75 );
76 print("finished scheduledNotification");
77 }
78
static tz.TZDateTime _scheduledDaily(Time time) {
79   print("enter _scheduledDaily");
80   final jakarta = tz.getLocation('Asia/Jakarta');
81   tz.setLocalLocation(jakarta);
82   final now = tz.TZDateTime.now(jakarta);
83   tz.TZDateTime scheduledDate = tz.TZDateTime(tz.local, now.year, now.month,
84   | | now.day, time.hour, time.minute, time.second);
85   print("before return _scheduledDaily");
86
87   return scheduledDate.isBefore(now)
88     ? scheduledDate = scheduledDate.add(const Duration(days: 1))
89     : scheduledDate;
90 }
91 }
```

The status bar at the bottom indicates "Ln 46 Col 1 Spaces:2 UTF-8 CRLF {} Dart 22071219CG (android-arm64)" and the date "3/21/2024".

6. Berikut adalah kode dari home view



```
lib > app > modules > home > views > home_view.dart > HomeView > build
1 import 'package:flutter/material.dart';
2 import 'package:feri_kusuma_wardhana_tugas_6_1/app/routes/app_pages.dart';
3 import 'package:get/get.dart';
4 import 'package:google_fonts/google_fonts.dart';
5
6 import '../controllers/home_controller.dart';
7
8 class HomeView extends GetView<HomeController> {
9   @override
10  Widget build(BuildContext context) {
11    return Scaffold(
12      appBar: AppBar(
13        title: Text(
14          'Home',
15          style: GoogleFonts.poppins(
16            textStyle: const TextStyle(
17              color: Colors.white,
18              fontWeight: FontWeight.bold,
19            ), // TextStyle
20          ),
21        ), // Text
22        centerTitle: true,
23        backgroundColor: Colors.green,
24      ), // AppBar
25      body: controller.obx(
26        (state) => ListView.builder(
27          itemCount: state!.length,
28          itemBuilder: (context, index) {
29            return Card(
30              elevation: 3,
31              margin: const EdgeInsets.symmetric(vertical: 8, horizontal: 16),
32              child: ListTile(
33                margin: const EdgeInsets.symmetric(vertical: 8, horizontal: 16),
34                child: ListTile(
35                  title: Text(
36                    state[index].name,
37                    style: GoogleFonts.poppins(
38                      textStyle: const TextStyle(
39                        fontSize: 18,
40                        fontWeight: FontWeight.bold,
41                      ), // TextStyle
42                  ), // Text
43                  subtitle: Text(
44                    "${state[index].frequency.toString()} kali sehari",
45                    style: GoogleFonts.poppins(
46                      textStyle: const TextStyle(fontSize: 14),
47                    ),
48                  ), // Text
49                  onTap: () => Get.toNamed(Routes.DETAIL_MEDICINE,
50                  arguments: state[index].id),
51                ), // ListTile
52                ); // Card
53            ), // ListView.builder
54            onLoading: const Center(child: CircularProgressIndicator()),
55            onEmpty: Center(
56              child: Text(
57                "Data Kosong",
58                style: GoogleFonts.poppins(
59                  textStyle: const TextStyle(fontSize: 18),
60                ),
61            ),
62          ),
63        ),
64      ),
65    );
66  }
67}
```



```
lib > app > modules > home > views > home_view.dart > HomeView > build
8 class HomeView extends GetView<HomeController> {
9   @override
10  Widget build(BuildContext context) {
11    return Scaffold(
12      appBar: AppBar(
13        title: Text(
14          'Home',
15          style: GoogleFonts.poppins(
16            textStyle: const TextStyle(
17              color: Colors.white,
18              fontWeight: FontWeight.bold,
19            ), // TextStyle
20          ),
21        ), // Text
22        centerTitle: true,
23        backgroundColor: Colors.green,
24      ), // AppBar
25      body: controller.obx(
26        (state) => ListView.builder(
27          itemCount: state!.length,
28          itemBuilder: (context, index) {
29            return Card(
30              elevation: 3,
31              margin: const EdgeInsets.symmetric(vertical: 8, horizontal: 16),
32              child: ListTile(
33                margin: const EdgeInsets.symmetric(vertical: 8, horizontal: 16),
34                child: ListTile(
35                  title: Text(
36                    state[index].name,
37                    style: GoogleFonts.poppins(
38                      textStyle: const TextStyle(
39                        fontSize: 18,
40                        fontWeight: FontWeight.bold,
41                      ), // TextStyle
42                  ), // Text
43                  subtitle: Text(
44                    "${state[index].frequency.toString()} kali sehari",
45                    style: GoogleFonts.poppins(
46                      textStyle: const TextStyle(fontSize: 14),
47                    ),
48                  ), // Text
49                  onTap: () => Get.toNamed(Routes.DETAIL_MEDICINE,
50                  arguments: state[index].id),
51                ), // ListTile
52                ); // Card
53            ), // ListView.builder
54            onLoading: const Center(child: CircularProgressIndicator()),
55            onEmpty: Center(
56              child: Text(
57                "Data Kosong",
58                style: GoogleFonts.poppins(
59                  textStyle: const TextStyle(fontSize: 18),
60                ),
61            ),
62          ),
63        ),
64      ),
65    );
66  }
67}
```

The screenshot shows the Visual Studio Code interface with the file `home_view.dart` open. The code defines a `HomeView` class that extends `GetView<HomeController>`. It contains a `Widget build(BuildContext context)` method that builds a UI with a `ListView.builder`. If the list is empty, it displays a message "Data Kosong". Otherwise, it shows a list of medicines with a floating action button for adding new ones.

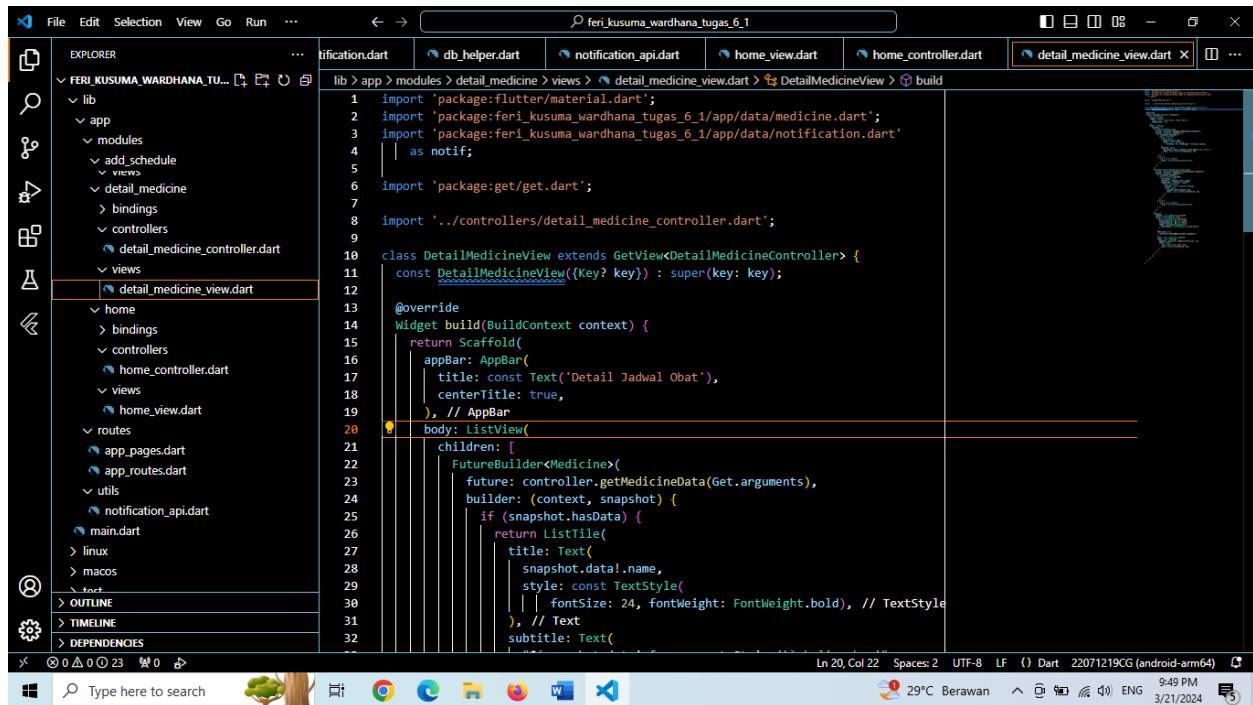
```
lib > app > modules > home > views > home_view.dart > HomeView > build
8 class HomeView extends GetView<HomeController> {
10    Widget build(BuildContext context) {
11        return ListView.builder(
12            itemCount: controller.listMedicines.length,
13            itemBuilder: (context, index) {
14                final medicine = controller.listMedicines[index];
15                return Card(
16                    child: Column(
17                        mainAxisAlignment: MainAxisAlignment.start,
18                        crossAxisAlignment: CrossAxisAlignment.start,
19                        children: [
20                            Text(medicine.name),
21                            Text(medicine.dosage),
22                            Text(medicine.instructions),
23                            Text(medicine.expiryDate),
24                            Text(medicine.notes),
25                        ],
26                    ),
27                );
28            },
29        );
30    }
31}
```

7. Berikut adalah kode dari home controller

The screenshot shows the Visual Studio Code interface with the file `home_controller.dart` open. The code defines a `HomeController` class that extends `GetxController` and implements `StateMixin<List<Medicine>>`. It has a `DbHelper db` field and a `listMedicines` observable list. The `onInit` method calls `getAllMedicineData()`. The `getAllMedicineData` method performs a database query to get all medicine data and updates the `listMedicines` observable.

```
lib > app > modules > home > controllers > home_controller.dart > ...
1 import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/medicine.dart';
2 import 'package:feri_kusuma_wardhana_tugas_6_1/app/helper/db_helper.dart';
3 import 'package:get/get.dart';
4
5 class HomeController extends GetxController with StateMixin<List<Medicine>> {
6     var db = DbHelper();
7     List<Medicine> listMedicines = <Medicine>[].obs;
8
9     @override
10    void onInit() {
11        // TODO: implement onInit
12        super.onInit();
13        getAllMedicineData();
14    }
15
16    Future getAllMedicineData() async {
17        change(null, status: RxStatus.loading());
18        listMedicines.clear();
19        final List<Medicine> medicines = await db.queryAllRowsMedicine();
20        listMedicines.addAll(medicines);
21        if (listMedicines.isEmpty) {
22            change(null, status: RxStatus.empty());
23        } else {
24            change(listMedicines, status: RxStatus.success());
25        }
26    }
27}
```

8. Berikut adalah kode dari detail_medicine view

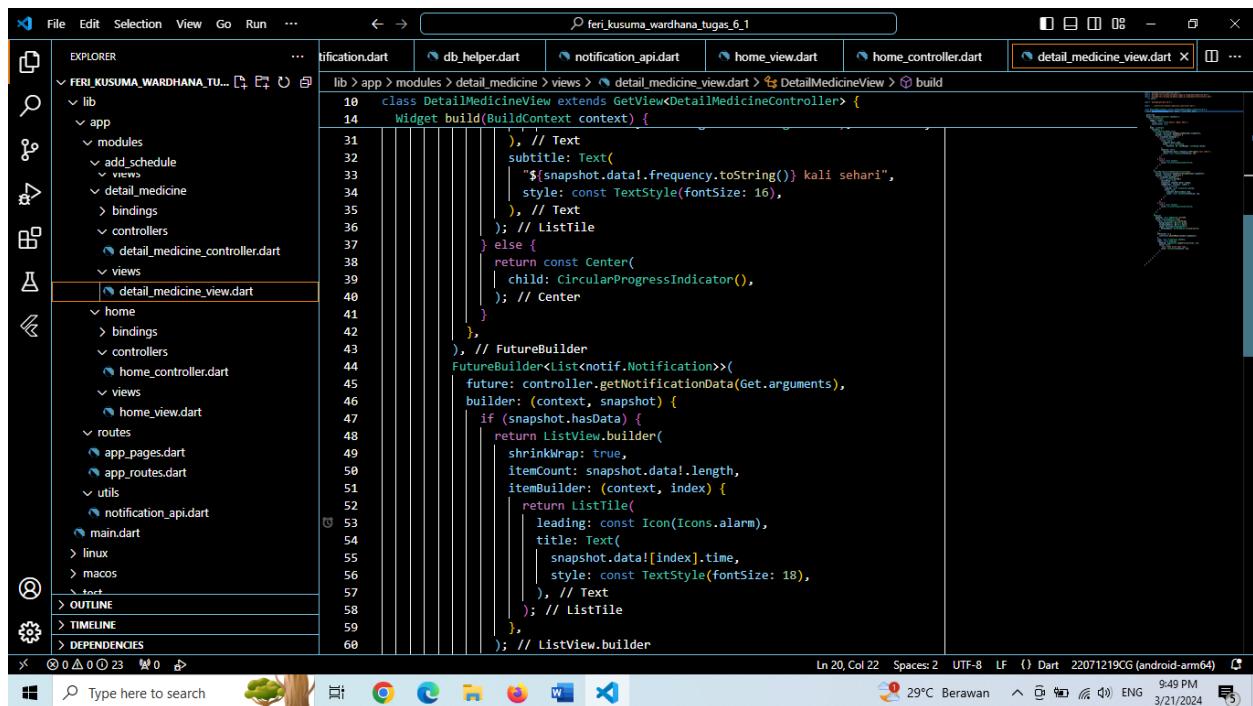


The screenshot shows the VS Code interface with the file 'detail_medicine_view.dart' open in the center editor tab. The code implements a ListView to display medicine data. It includes imports for flutter/material.dart, feri_kusuma_wardhana_tugas_6_1/app/data/medicine.dart, feri_kusuma_wardhana_tugas_6_1/app/data/notification.dart, and get/get.dart. The class DetailMedicineView extends GetView<DetailMedicineController>. It defines a scaffold with an AppBar titled 'Detail Jadwal Obat'. The body contains a ListView with children that are FutureBuilder<Medicine> widgets. Each FutureBuilder has a future of controller.getMedicineData(Get.arguments) and a builder that checks if snapshot.hasData to return a ListTile with title and subtitle text.

```
import 'package:flutter/material.dart';
import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/medicine.dart';
import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/notification.dart';
import 'package:get/get.dart';

class DetailMedicineView extends GetView<DetailMedicineController> {
  const DetailMedicineView({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Detail Jadwal Obat'),
        centerTitle: true,
      ), // AppBar
      body: ListView(
        children: [
          FutureBuilder<Medicine>(
            future: controller.getMedicineData(Get.arguments),
            builder: (context, snapshot) {
              if (snapshot.hasData) {
                return ListTile(
                  title: Text(
                    snapshot.data!.name,
                    style: const TextStyle(
                      fontSize: 24, fontWeight: FontWeight.bold), // TextStyle
                  ), // Text
                  subtitle: Text(
                    "${snapshot.data!.frequency.toString()} kali sehari",
                    style: const TextStyle(fontSize: 16),
                  ), // Text
                ); // ListTile
              } else {
                return const Center(
                  child: CircularProgressIndicator(),
                ); // Center
              }
            },
          ), // FutureBuilder
        ],
      ), // ListView
    );
  }
}
```



The screenshot shows the VS Code interface with the file 'detail_medicine_view.dart' open in the center editor tab. The code implements a ListView to display medicine data. It includes imports for flutter/material.dart, feri_kusuma_wardhana_tugas_6_1/app/data/medicine.dart, feri_kusuma_wardhana_tugas_6_1/app/data/notification.dart, and get/get.dart. The class DetailMedicineView extends GetView<DetailMedicineController>. It defines a scaffold with an AppBar titled 'Detail Jadwal Obat'. The body contains a FutureBuilder<List<notif.Notification>> widget. Its future is controller.getNotificationData(Get.arguments) and its builder is a Listview.builder that returns a list of ListTile items, each with a leading alarm icon and a title containing the notification time.

```
class DetailMedicineView extends GetView<DetailMedicineController> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Detail Jadwal Obat'),
        centerTitle: true,
      ), // AppBar
      body: FutureBuilder<List<notif.Notification>>(
        future: controller.getNotificationData(Get.arguments),
        builder: (context, snapshot) {
          if (snapshot.hasData) {
            return ListView.builder(
              shrinkWrap: true,
              itemCount: snapshot.data!.length,
              itemBuilder: (context, index) {
                return ListTile(
                  leading: const Icon(Icons.alarm),
                  title: Text(
                    snapshot.data![index].time,
                    style: const TextStyle(fontSize: 18),
                  ), // Text
                ); // ListTile
              },
            ); // ListView.builder
          } else {
            return const Center(
              child: CircularProgressIndicator(),
            ); // Center
          }
        },
      ), // FutureBuilder
    );
  }
}
```

The screenshot shows a code editor with the file `detail_medicine_view.dart` open. The code implements a `DetailMedicineView` class that extends `GetView<DetailMedicineController>`. The implementation includes a `Widget build(BuildContext context)` method which returns a `FutureBuilder` wrapped in a `Padding`. Inside the padding, there is an `ElevatedButton.icon` with a green background color, white foreground color, rounded rectangle border, and a circular border radius of 24.0. The button has an icon of a cancel icon, a label 'Saya tidak butuh obat lagi', and a text style of `fontSize: 18`. The code editor interface includes an Explorer sidebar, an Outline sidebar, and a Timeline sidebar.

```
lib > app > modules > detail_medicine > views > detail_medicine_view.dart > DetailMedicineView > build
10 class DetailMedicineView extends GetView<DetailMedicineController> {
11   @override
12   Widget build(BuildContext context) {
13     return FutureBuilder(
14       future: controller.getMedicine(),
15       builder: (context, AsyncSnapshot snapshot) {
16         if (snapshot.hasData) {
17           final medicine = snapshot.data;
18           return Scaffold(
19             appBar: AppBar(
20               title: Text(medicine.name),
21             ),
22             body: Padding(
23               padding: const EdgeInsets.all(20),
24               child: ElevatedButton.icon(
25                 onPressed: () {
26                   controller.deleteMedicine(medicine.id);
27                 },
28                 icon: const Icon(Icons.cancel),
29                 label: const Padding(
30                   padding: EdgeInsets.symmetric(vertical: 12),
31                   child: Text(
32                     'Saya tidak butuh obat lagi',
33                     style: TextStyle(fontSize: 18),
34                   ),
35                 ),
36               ),
37             ),
38           );
39         } else {
40           return Center(
41             child: CircularProgressIndicator(),
42           );
43         }
44       },
45     );
46   }
47 }
```

This screenshot is identical to the one above, showing the same code editor interface and the `DetailMedicineView` class implementation. The cursor is positioned at line 74 of the code, specifically on the closing brace of the `Widget build(BuildContext context)` method. The code editor's status bar indicates the current line is 74, column 47.

```
lib > app > modules > detail_medicine > views > detail_medicine_view.dart > DetailMedicineView > build
10 class DetailMedicineView extends GetView<DetailMedicineController> {
11   @override
12   Widget build(BuildContext context) {
13     return FutureBuilder(
14       future: controller.getMedicine(),
15       builder: (context, AsyncSnapshot snapshot) {
16         if (snapshot.hasData) {
17           final medicine = snapshot.data;
18           return Scaffold(
19             appBar: AppBar(
20               title: Text(medicine.name),
21             ),
22             body: Padding(
23               padding: const EdgeInsets.all(20),
24               child: ElevatedButton.icon(
25                 onPressed: () {
26                   controller.deleteMedicine(medicine.id);
27                 },
28                 icon: const Icon(Icons.cancel),
29                 label: const Padding(
30                   padding: EdgeInsets.symmetric(vertical: 12),
31                   child: Text(
32                     'Saya tidak butuh obat lagi',
33                     style: TextStyle(fontSize: 18),
34                   ),
35                 ),
36               ),
37             ),
38           );
39         } else {
40           return Center(
41             child: CircularProgressIndicator(),
42           );
43         }
44       },
45     );
46   }
47 }
```

9. Berikut adalah kode dari detail_medicine controller

The screenshot shows a code editor interface with the following details:

- File Path:** lib > app > modules > detail_medicine > controllers > DetailMedicineController.dart
- Code Content:**

```
1 import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/notification.dart'
2 ||| as notif;
3 import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/medicine.dart';
4 import 'package:feri_kusuma_wardhana_tugas_6_1/app/helper/db_helper.dart';
5 import 'package:feri_kusuma_wardhana_tugas_6_1/app/modules/home/controllers/home_controller.dart';
6 import 'package:feri_kusuma_wardhana_tugas_6_1/app/utils/notification_api.dart';
7 import 'package:get/get.dart';

8
9 class DetailMedicineController extends GetxController {
10   var db = DbHelper();
11   HomeController homeController = Get.put(HomeController());
12   late List<notif.Notification> listNotification;

13   Future<Medicine> getMedicineData(int id) async {
14     | return await db.queryOneMedicine(id);
15   }

16   Future<List<notif.Notification>> getNotificationData(int id) async {
17     | listNotification = await db.getNotificationsByMedicineId(id);
18     | return listNotification;
19   }

20   void deleteMedicine(int id) async {
21     | listNotification.forEach((element) {
22       | | NotificationApi.cancelNotification(element.id!);
23     });
24     | | await db.deleteMedicine(id);
25     | | await db.deleteNotificationByMedicineId(id);
26     | | homeController.getAllMedicineData();
27     | | Get.back();
28   }
29 }
30 }
```
- Environment:** 29°C Berawan, 9:51 PM, 3/21/2024

10. Berikut adalah add_medicine view

The screenshot shows a code editor interface with the following details:

- File Path:** lib > app > modules > add_schedule > views > AddScheduleView.dart
- Code Content:**

```
1 import 'dart:async';
2 import 'package:flutter/material.dart';
3 import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/medicine.dart';
4 import 'package:get/get.dart';
5 import '../controllers/add_schedule_controller.dart';

6 class AddScheduleView extends StatelessWidget {
7   final _timeOfDay = TimeOfDay.now();
8   final _formKey = GlobalKey<FormState>();
9   @override
10  Widget build(BuildContext context) {
11    return Scaffold(
12      appBar: AppBar(
13        title: const Text('Tambahkan Jadwal Obat'),
14        centerTitle: true,
15      ),
16      body: Form(
17        key: _formKey,
18        child: Center(
19          child: ListView(
20            children: [
21              Container(
22                alignment: Alignment.center,
23                padding: EdgeInsets.all(10),
24                child: const Text(
25                  'Tambah Jadwal Obat',
26                  style: TextStyle(
27                    color: Colors.green,
28                    fontWeight: FontWeight.w500,
29                    fontSize: 30), // TextStyle
30                ), // Text
31              ), // Container
32            ],
33          ),
34        ),
35      ),
36    );
37  }
38 }
```
- Environment:** 29°C Berawan, 9:52 PM, 3/21/2024

The screenshot shows a Flutter project structure in the left sidebar and the corresponding Dart code for the `add_schedule_view.dart` file in the main editor area.

Project Structure:

- EXPLORER
- FERI_KUSUMA_WARDHANA_TU...
- lib
- app
 - data
 - notification.dart
 - helper
 - db_helper.dart
 - modules
 - add_schedule
 - bindings
 - controllers
 - add_schedule_controller.dart
 - views
 - add_schedule_view.dart

Code Editor (add_schedule_view.dart):

```
lib/app/modules/add_schedule/views/add_schedule_view.dart
9 class AddScheduleView extends GetView<AddScheduleController> {
13   Widget build(BuildContext context) {
33     |   ), // Text
34     |   ), // Container
35     |   Container(
36     |     padding: EdgeInsets.all(10),
37     |     child: TextFormField(
38     |       controller: controller.nameController,
39     |       decoration: InputDecoration(
40     |         border: OutlineInputBorder(
41     |           borderRadius: BorderRadius.circular(20), // OutlineInputBorder
42     |           labelText: 'Nama',
43     |         ), // InputDecoration
44     |        textInputAction: TextInputAction.next,
45     |         validator: (value) {
46     |           if (value.isEmpty) {
47     |             return 'Ini tidak boleh kosong';
48     |           }
49     |           return null;
50     |         },
51     |       ), // TextFormField
52     |     ), // Container
53     |     Container(
54     |       padding: const EdgeInsets.fromLTRB(10, 10, 10, 20),
55     |       child: TextFormField(
56     |         controller: controller.frequencyController,
57     |         decoration: InputDecoration(
58     |           border: OutlineInputBorder(
59     |             borderRadius: BorderRadius.circular(20), // OutlineInputBorder
60     |             labelText: 'Berapa kali',
61     |           ), // InputDecoration
62     |          textInputAction: TextInputAction.next,
```

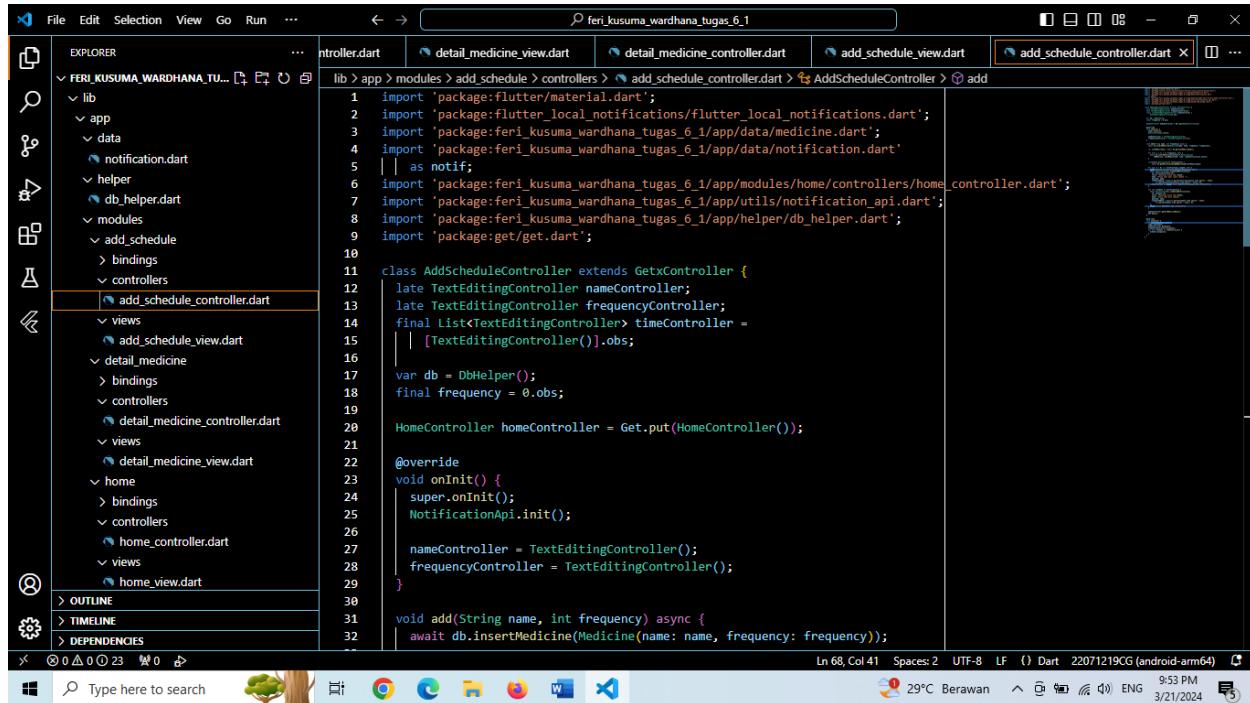
The screenshot shows a Dart development environment with the following details:

- File Explorer:** On the left, it lists the project structure. The file `add_schedule_view.dart` is currently selected.
- Code Editor:** The main area displays the source code for `AddScheduleView`. The code handles frequency input validation and lists items based on frequency values.
- Bottom Status Bar:** Shows the current line (Ln 72), column (Col 19), and encoding (UTF-8). It also indicates the file type (Dart) and the build target (22071219CG (android-arm64)).
- Bottom Taskbar:** Includes icons for File, Edit, Selection, View, Go, Run, and other common operations.
- Bottom Navigation:** A search bar with placeholder text "Type here to search" and several system status indicators (battery level, signal strength, etc.).

```
lib > app > modules > add_schedule > views > add_schedule_view.dart > AddScheduleView > build
9 class AddScheduleView extends GetView<AddScheduleController> {
13   Widget build(BuildContext context) {
14
15     final timeController = TextEditingController();
16
17     return Scaffold(
18       appBar: AppBar(
19         title: const Text('Add Schedule'),
20         actions: [
21           IconButton(
22             icon: const Icon(Icons.add),
23             onPressed: () {
24               Navigator.push(
25                 context,
26                 MaterialPageRoute(
27                   builder: (context) => AddScheduleView(),
28                 ),
29               );
30             },
31           ),
32         ],
33       ),
34       body: Padding(
35         padding: const EdgeInsets.all(16.0),
36         child: Form(
37           key: _formKey,
38           child: Column(
39             children: [
40               TextFormField(
41                 controller: nameController,
42                 decoration: InputDecoration(
43                   labelText: 'Name',
44                   hintText: 'Enter Name',
45                 ),
46                 validator: (value) {
47                   if (value!.isEmpty) {
48                     return 'Name cannot be empty';
49                   }
50                   return null;
51                 },
52               ),
53               TextFormField(
54                 controller: frequencyController,
55                 decoration: InputDecoration(
56                   labelText: 'Frequency',
57                   hintText: 'Enter Frequency',
58                 ),
59                 validator: (value) {
60                   if (value!.isEmpty) {
61                     return 'Frequency cannot be empty';
62                   }
63                   return null;
64                 },
65               ),
66               TextFormField(
67                 controller: timeController,
68                 decoration: InputDecoration(
69                   labelText: 'Time',
70                   hintText: 'Enter Time',
71                 ),
72                 validator: (value) {
73                   if (value!.isEmpty) {
74                     return 'Time cannot be empty';
75                   }
76                   return null;
77                 },
78               ),
79               ElevatedButton(
80                 onPressed: () {
81                   if (_formKey.currentState!.validate()) {
82                     AddScheduleController controller =
83                         Get.find<AddScheduleController>();
84                     controller.add(nameController.text);
85                     controller.frequencyController.text;
86                     controller.timeController.add(timeController.text);
87                     Navigator.pop(context);
88                   }
89                 },
90                 child: const Text('Add'),
91               ),
92             ],
93           ),
94         ),
95       ),
96     );
97   }
98 }
99 
```

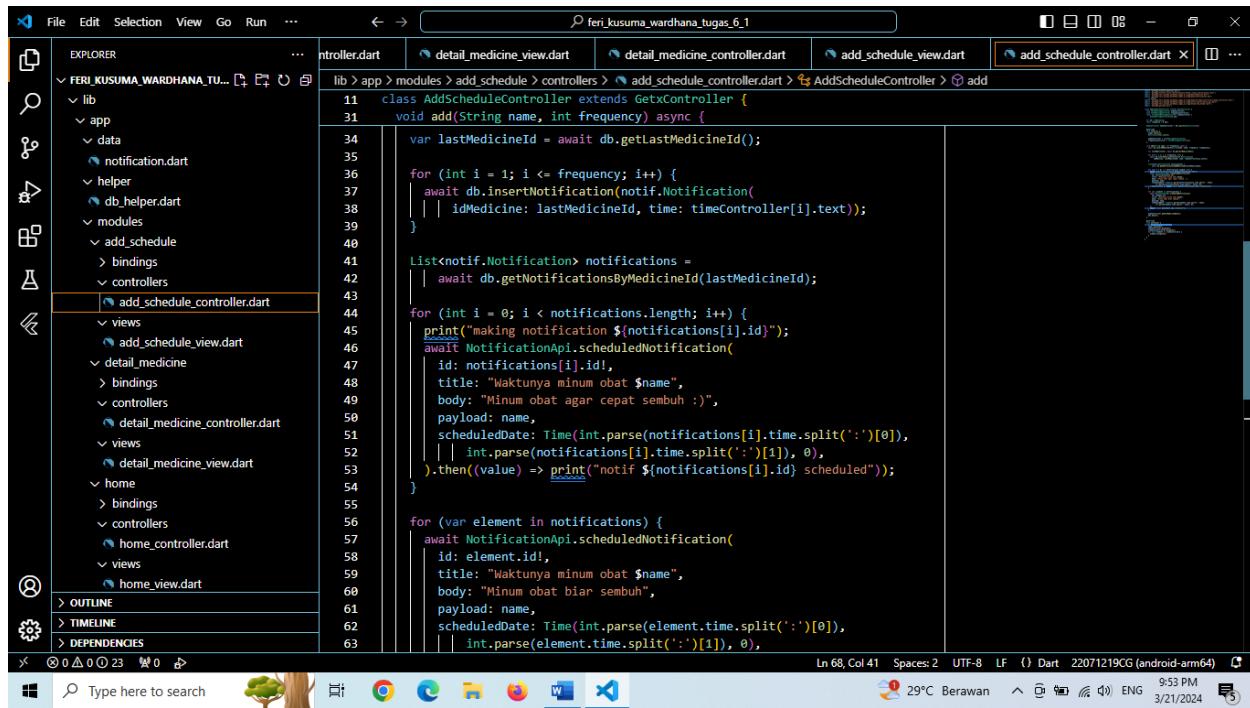
```
lib > app > modules > add_schedule > views > add_schedule_view.dart > AddScheduleView > build
9 class AddScheduleView extends GetView<AddScheduleController> {
13   Widget build(BuildContext context) {
14
15     final timeController = TextEditingController();
16
17     return Scaffold(
18       appBar: AppBar(
19         title: const Text('Add Schedule'),
20         actions: [
21           IconButton(
22             icon: const Icon(Icons.add),
23             onPressed: () {
24               Navigator.push(
25                 context,
26                 MaterialPageRoute(
27                   builder: (context) => AddScheduleView(),
28                 ),
29               );
30             },
31           ),
32         ],
33       ),
34       body: Padding(
35         padding: const EdgeInsets.all(16.0),
36         child: Form(
37           key: _formKey,
38           child: Column(
39             children: [
40               TextFormField(
41                 controller: nameController,
42                 decoration: InputDecoration(
43                   labelText: 'Name',
44                   hintText: 'Enter Name',
45                 ),
46                 validator: (value) {
47                   if (value!.isEmpty) {
48                     return 'Name cannot be empty';
49                   }
50                   return null;
51                 },
52               ),
53               TextFormField(
54                 controller: frequencyController,
55                 decoration: InputDecoration(
56                   labelText: 'Frequency',
57                   hintText: 'Enter Frequency',
58                 ),
59                 validator: (value) {
60                   if (value!.isEmpty) {
61                     return 'Frequency cannot be empty';
62                   }
63                   return null;
64                 },
65               ),
66               TextFormField(
67                 controller: timeController,
68                 decoration: InputDecoration(
69                   labelText: 'Time',
70                   hintText: 'Enter Time',
71                 ),
72                 validator: (value) {
73                   if (value!.isEmpty) {
74                     return 'Time cannot be empty';
75                   }
76                   return null;
77                 },
78               ),
79               ElevatedButton(
80                 onPressed: () {
81                   if (_formKey.currentState!.validate()) {
82                     AddScheduleController controller =
83                         Get.find<AddScheduleController>();
84                     controller.add(nameController.text);
85                     controller.frequencyController.text;
86                     controller.timeController.add(timeController.text);
87                     Navigator.pop(context);
88                   }
89                 },
90                 child: const Text('Add'),
91               ),
92             ],
93           ),
94         ),
95       ),
96     );
97   }
98 }
99 
```

11. Berikut adalah kode dari add_schedule controller



```
lib > app > modules > add_schedule > controllers > add_schedule_controller.dart > AddScheduleController > add
1 import 'package:flutter/material.dart';
2 import 'package:flutter_local_notifications/flutter_local_notifications.dart';
3 import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/medicine.dart';
4 import 'package:feri_kusuma_wardhana_tugas_6_1/app/data/notification.dart';
5 | | as notif;
6 import 'package:feri_kusuma_wardhana_tugas_6_1/app/modules/home/controllers/home_controller.dart';
7 import 'package:feri_kusuma_wardhana_tugas_6_1/app/utils/notification_api.dart';
8 import 'package:feri_kusuma_wardhana_tugas_6_1/app/helper/db_helper.dart';
9 import 'package:get/get.dart';

10 class AddScheduleController extends GetController {
11   late TextEditingController nameController;
12   late TextEditingController frequencyController;
13   final List<TextEditingController> timeController =
14     [TextEditingController().obs];
15
16   var db = Database();
17   final frequency = 0.obs;
18
19   HomeController homeController = Get.put(HomeController());
20
21   @override
22   void initState() {
23     super.initState();
24     NotificationApi.init();
25
26     nameController = TextEditingController();
27     frequencyController = TextEditingController();
28   }
29
30   void add(String name, int frequency) async {
31     await db.insertMedicine(Medicine(name: name, frequency: frequency));
32 }
```



```
lib > app > modules > add_schedule > controllers > add_schedule_controller.dart > AddScheduleController > add
11 class AddScheduleController extends GetController {
31   void add(String name, int frequency) async {
34     var lastMedicineId = await db.getLastMedicineId();
35
36     for (int i = 1; i <= frequency; i++) {
37       await db.insertNotification(notif.Notification(
38         id: lastMedicineId, time: timeController[i].text));
39     }
40
41     List<notif.Notification> notifications =
42       await db.getNotificationsByMedicineId(lastMedicineId);
43
44     for (int i = 0; i < notifications.length; i++) {
45       print("making notification ${notifications[i].id}");
46       await NotificationApi.scheduledNotification(
47         id: notifications[i].id,
48         title: "Makturnya minum obat $name",
49         body: "Minum obat agar cepat sembuh :)",
50         payload: name,
51         scheduledDate: Time(int.parse(notifications[i].time.split(':')[0]),
52           | | int.parse(notifications[i].time.split(':')[1]), 0),
53         ).then((value) => print("notif ${notifications[i].id} scheduled"));
54     }
55
56     for (var element in notifications) {
57       await NotificationApi.scheduledNotification(
58         id: element.id,
59         title: "Makturnya minum obat $name",
60         body: "Minum obat biar sembuh",
61         payload: name,
62         scheduledDate: Time(int.parse(element.time.split(':')[0]),
63           | | int.parse(element.time.split(':')[1]), 0),
64       );
65     }
66   }
67 }
```

```

lib > app > modules > add_schedule > controllers > add_schedule_controller.dart > AddScheduleController > add_schedule_controller.dart
11 class AddScheduleController extends GetxController {
12   void add(String name, int frequency) async {
13     for (var element in notifications) {
14       await NotificationApi.scheduledNotification(
15         id: element.id!,
16         title: "Waktunya minum obat $name",
17         body: "Minum obat biar sembuh",
18         payload: name,
19         scheduledDate: Time(int.parse(element.time.split(':')[0])),
20         | | int.parse(element.time.split(':')[1]), 0),
21       );
22       print("notif ${element.id} scheduled");
23     }
24     homeController.getAllMedicineData();
25     Get.back();
26   }
27   @override
28   void onClose() {
29     // TODO: implement onClose
30     super.onClose();
31     nameController.dispose();
32     frequencyController.dispose();
33     for (var element in timeController) {
34       element.dispose();
35     }
36   }
37 }

```

A. Tampilan Aplikasi

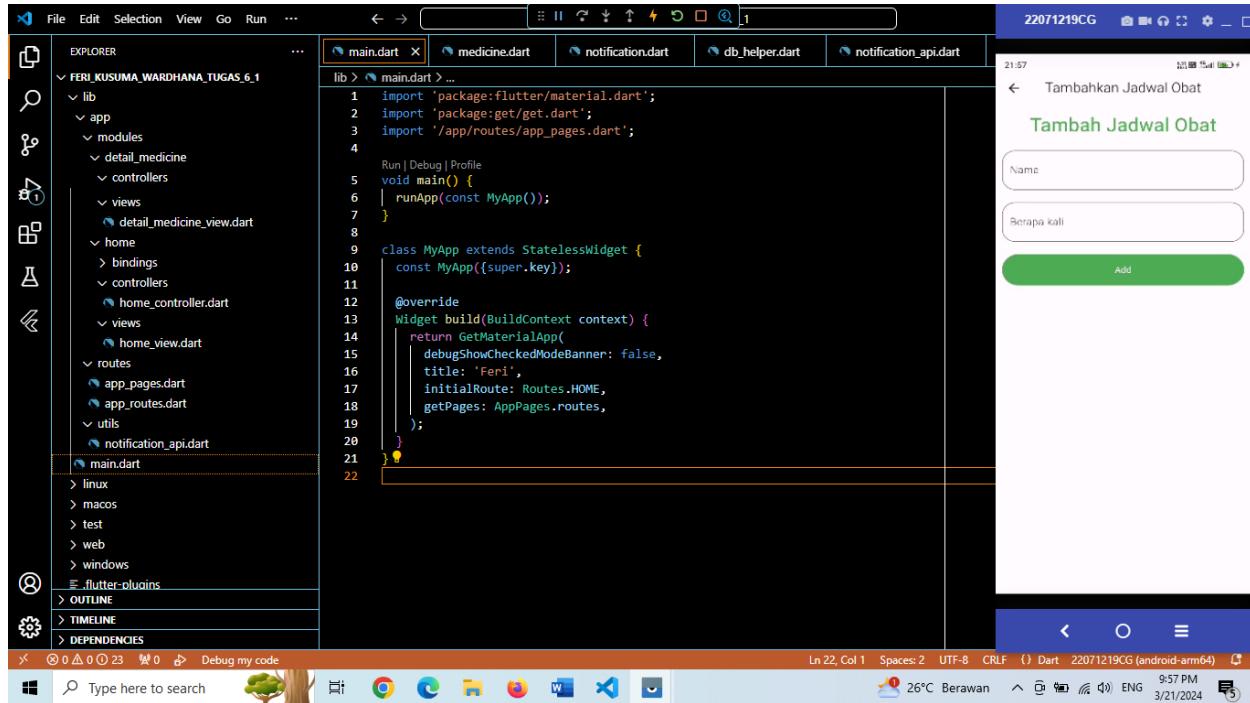
- Berikut adalah tampilan dari home kemudian jika ingin menambahkan bisa klik floating di kanan bawah

```

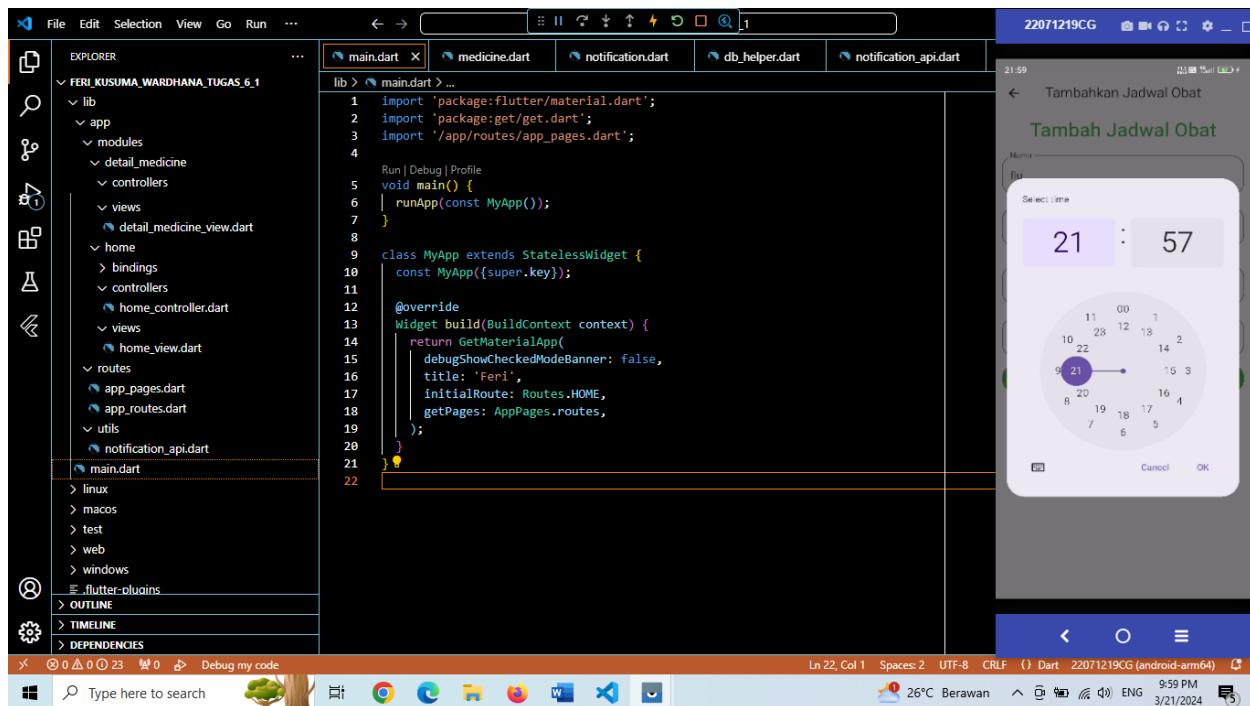
lib > main.dart > ...
1 import 'package:flutter/material.dart';
2 import 'package:get/get.dart';
3 import '/app/routes/app_pages.dart';
4
5 void main() {
6   runApp(const MyApp());
7 }
8
9 class MyApp extends StatelessWidget {
10   const MyApp({super.key});
11
12   @override
13   Widget build(BuildContext context) {
14     return GetMaterialApp(
15       debugShowCheckedModeBanner: false,
16       title: 'Feri',
17       initialRoute: Routes.HOME,
18       getPages: AppPages.routes,
19     );
20   }
21 }
22

```

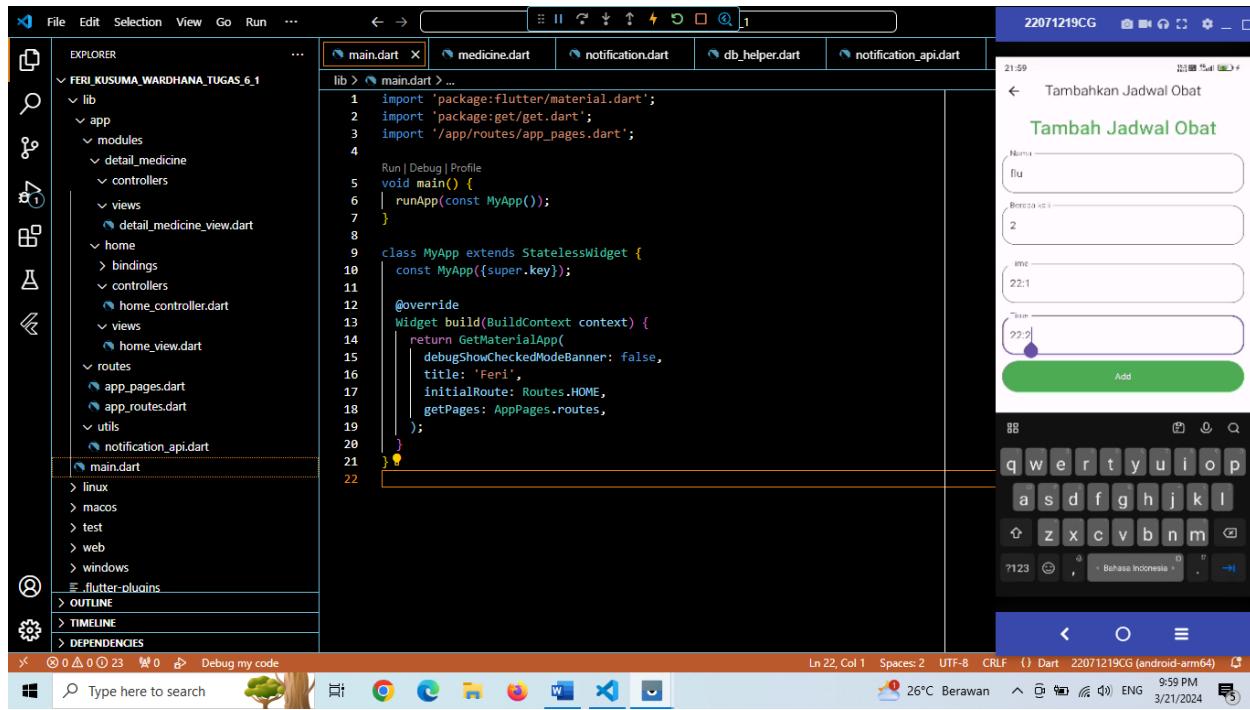
- 2) Berikut adalah tampilan dari halaman add_medicine dimana bisa menambahkan nama obat dan berapa kali minum obat dalam satu hari



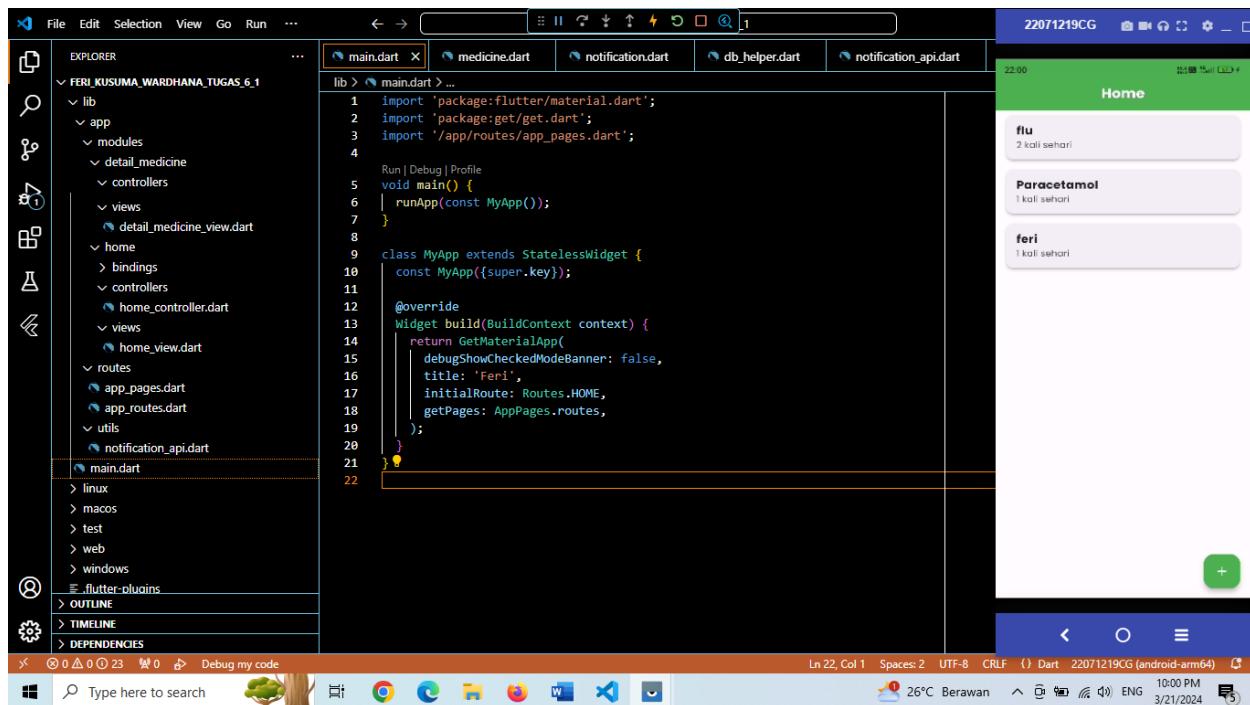
- 3) Berikut adalah saat melakukan pengambilan waktu



4) Tampilan akan menjadi seperti ini



5) Kemudian akan muncul dihalaman home



6) Berikut adalah saat ada notifikasi pengingat minum obat

The screenshot shows the Visual Studio Code interface with the following details:

- Explorer View:** Shows the project structure for "FERI KUSUMA WARDHANA TUGAS 6_1". The "lib/main.dart" file is currently selected.
- Main Editor:** Displays the code for "main.dart". The code initializes a Flutter application named "MyApp" with the initial route set to "Routes.HOME".
- Output Panel:** Shows the output "22071219CG" and the log message "Waktunya minum obat flu - sekarang *".
- Flutter Preview:** A floating window displays a notification card titled "Waktunya minum obat flu" with the message "Minum obat biar sembuh". It also shows a list of medications: "Paracetamol" and "feri", both with the note "1 kali sehari".
- System Tray:** Shows the date and time as "3/21/2024 10:01 PM".

7) Berikut adalah notifikasinya

The screenshot shows the Visual Studio Code interface with the following details:

- Explorer View:** Shows the project structure for "FERI_KUSUMA_WARDHANA_TUGAS_6_1". The "lib/main.dart" file is currently selected.
- Main Editor:** Displays the code for "main.dart". The code initializes a Flutter application named "MyApp" with the initial route set to "Routes.HOME".
- Output Panel:** Shows the output "22071219CG" and the log message "Waktunya minum obat flu".
- Flutter Preview:** A floating window displays a notification card titled "Waktunya minum obat flu" with the message "Minum obat biar sembuh". It also shows a list of medications: "Paracetamol" and "feri", both with the note "1 kali sehari".
- System Tray:** Shows the date and time as "3/21/2024 10:01 PM".

- 8) Berikut adalah halaman detail_medicine dimana ada nama obat kemudian waktu pengingat dan tombol jika pengingat sudah tidak diperlukan lagi

The screenshot shows the VS Code interface with the main.dart file open in the editor. The mobile application window displays a notification screen titled 'Detail Jadwal Obat'. It lists a notification for 'flu' with a timestamp of '2 kali sehari' at '22:1'. Below the list is a green button labeled 'Saya tidak butuh obat lagi'.

```

File Edit Selection View Go Run ...
main.dart X medicine.dart notification.dart db_helper.dart notification_api.dart
lib > main.dart > ...
1 import 'package:flutter/material.dart';
2 import 'package:get/get.dart';
3 import '/app/routes/app_pages.dart';
4
5 void main() {
6   runApp(const MyApp());
7 }
8
9 class MyApp extends StatelessWidget {
10   const MyApp({super.key});
11
12   @override
13   Widget build(BuildContext context) {
14     return GetMaterialApp(
15       debugShowCheckedModeBanner: false,
16       title: 'Feri',
17       initialRoute: Routes.HOME,
18       getPages: AppPages.routes,
19     );
20   }
21 }

```

- 9) Berikut jika klik tombol tidak butuh obat lagi maka pada halaman home , pengingat tersebut akan hilang

The screenshot shows the VS Code interface with the main.dart file open in the editor. The mobile application window displays a home screen with a list of medications. It shows 'Paracetamol' with a note '1 kali sehari' and 'feri' with a note '1 kali sehari'. At the bottom right is a green button with a plus sign.

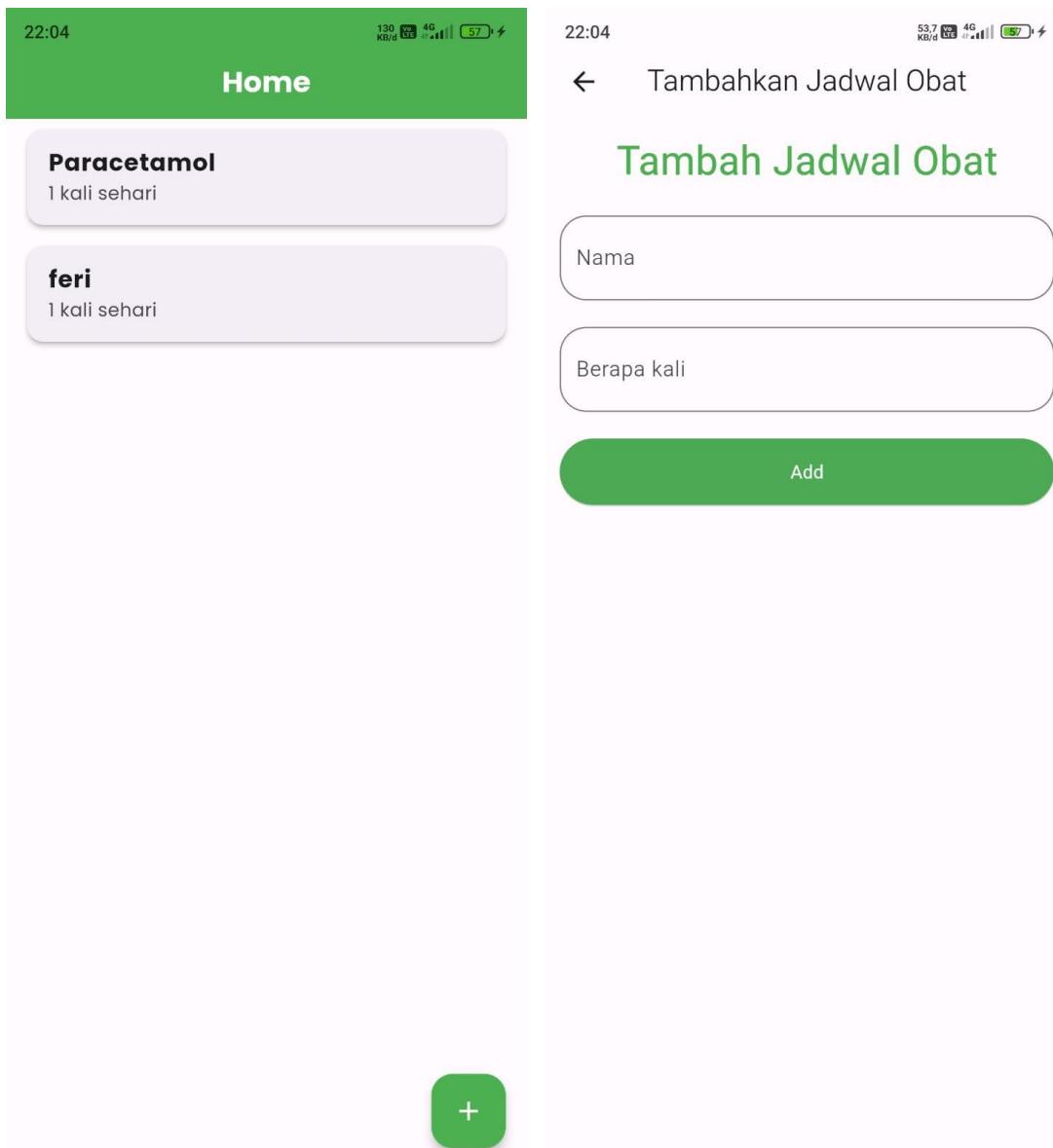
```

File Edit Selection View Go Run ...
main.dart X medicine.dart notification.dart db_helper.dart notification_api.dart
lib > main.dart > ...
1 import 'package:flutter/material.dart';
2 import 'package:get/get.dart';
3 import '/app/routes/app_pages.dart';
4
5 void main() {
6   runApp(const MyApp());
7 }
8
9 class MyApp extends StatelessWidget {
10   const MyApp({super.key});
11
12   @override
13   Widget build(BuildContext context) {
14     return GetMaterialApp(
15       debugShowCheckedModeBanner: false,
16       title: 'Feri',
17       initialRoute: Routes.HOME,
18       getPages: AppPages.routes,
19     );
20   }
21 }

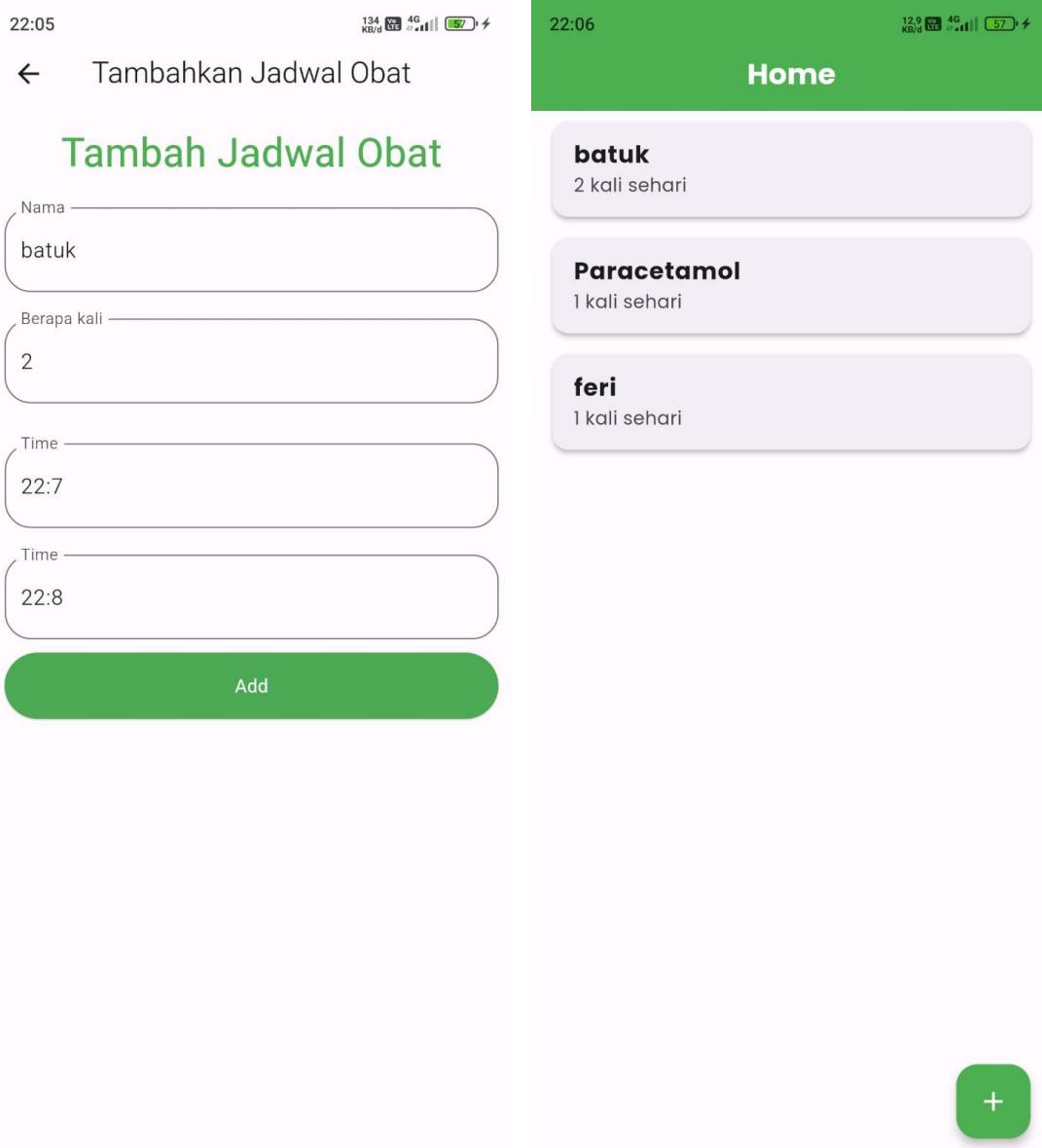
```

B. Berikut adalah tampilan dari hp

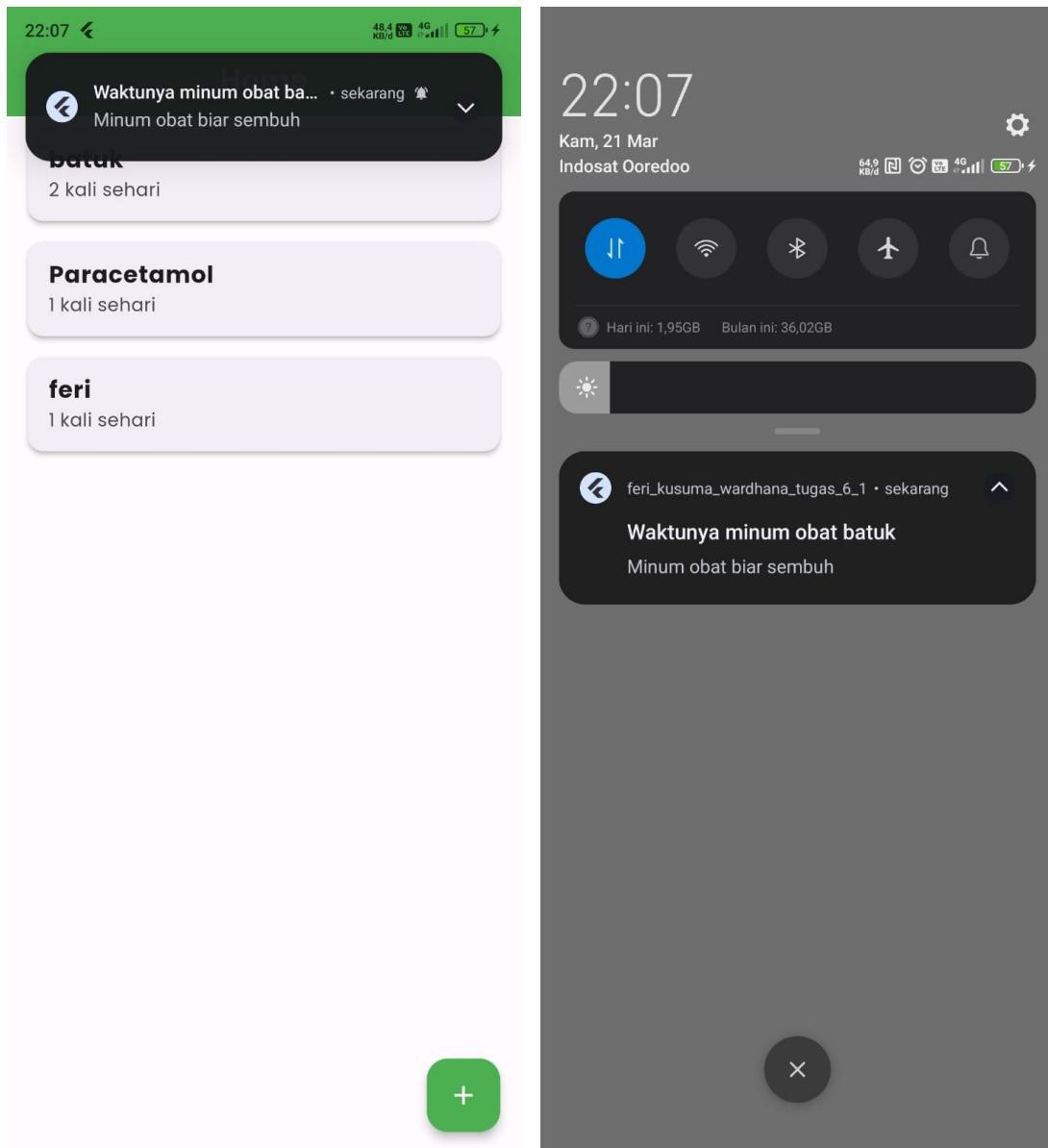
1. Berikut tampilan utama atau home akan seperti dibawah ini , dimana nanti bisa klik floating tambah yang ada di bawah kanan kemudian akan muncul tambah jadwal obat



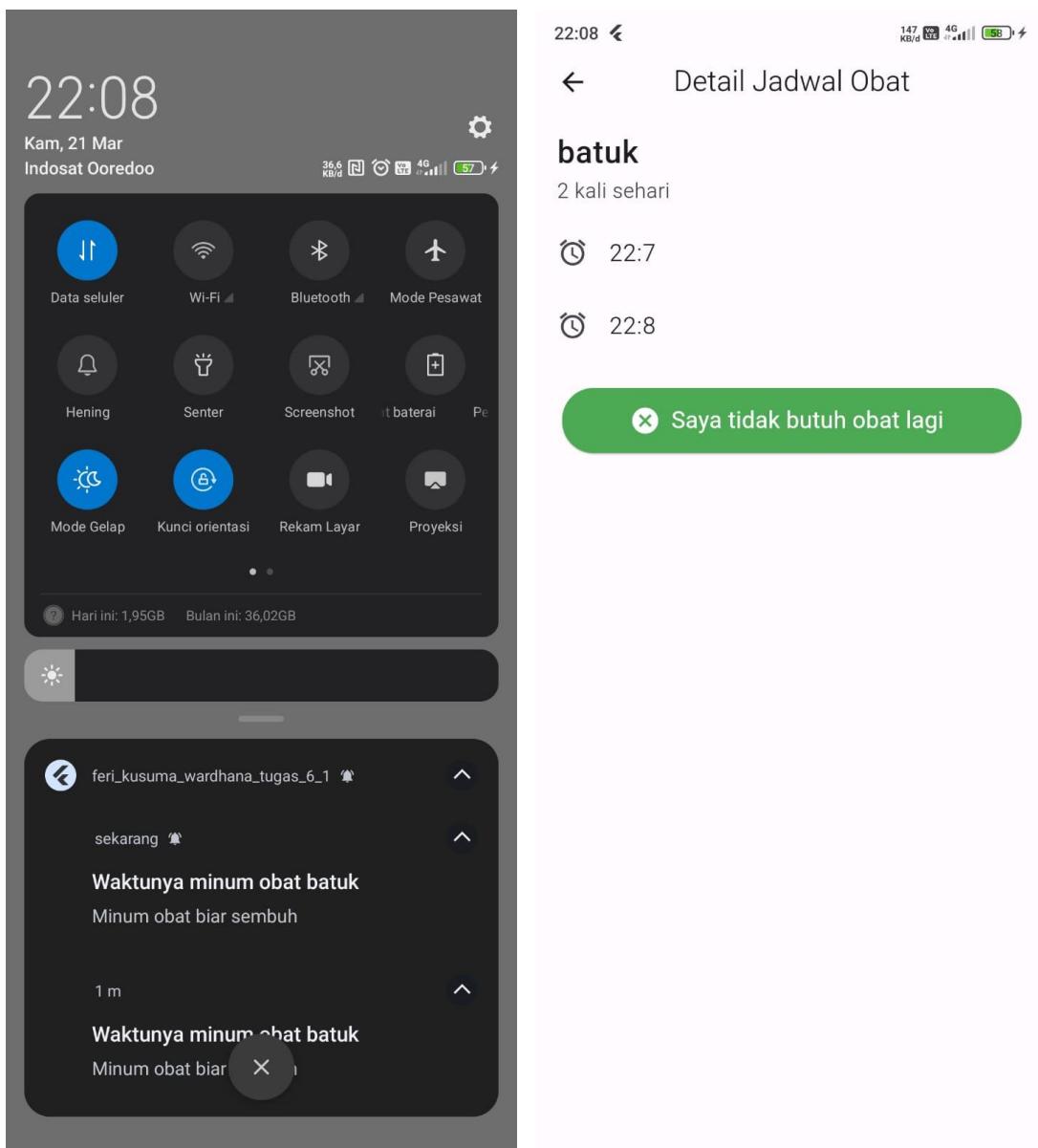
2. Berikut adalah jika sudah melakukan penambahan pengingat obat maka pada home juga akan bertambah



- 3) Kemudian pada waktu yang sudah ditentukan maka notifikasi akan muncul seperti dibawah ini



4) Kemudian berikut adalah detail dari jadwal obat



- 5) Kemudian jika diklik tombol saya tidak butuh obat lagi maka pengingat obat akan hilang di home

