

**TUGAS UNGUIDED
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL X
DATA STORAGE BAGIAN 1**



Disusun Oleh :

Althafia Defiyandrea Laskanadya Wibowo

2211104011 / SE06-01

Asisten Praktikum :

Muhammad Faza Zulian Gesit Al Barru

Aisyah Hasna Aulia

Dosen Pengampu :

Yudha Islami Sulistya, S.Kom., M.Cs.

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

Buatlah sebuah project aplikasi Flutter dengan SQLite untuk menyimpan data biodata mahasiswa yang terdiri dari nama, NIM, domisili, dan hobi. Data yang dimasukkan melalui form akan ditampilkan dalam daftar di halaman utama.

Alur Aplikasi:

- a. Form Input
Buat form input untuk menambahkan biodata mahasiswa, dengan kolom: Nama Nim Alamat Hobi
- b. Tampilkan Daftar Mahasiswa
Setelah data berhasil ditambahkan, tampilkan daftar semua data mahasiswa yang sudah disimpan di halaman utama.
- c. Implementasikan fitur Create (untuk menyimpan data mahasiswa) dan Read (untuk menampilkan daftar mahasiswa yang sudah disimpan).

Main.dart

```
1 import 'package:flutter/material.dart';
2 import 'package:praktikum_10/view/my_db_view.dart';
3
4 void main() {
5   runApp(MyApp());
6 }
7
8 class MyApp extends StatelessWidget {
9   @override
10  Widget build(BuildContext context) {
11    return MaterialApp(
12      debugShowCheckedModeBanner: false,
13      title: 'SQLite Biodata Mahasiswa',
14      theme: ThemeData(
15        primarySwatch: Colors.amber,
16      ),
17      home: MyDbView(),
18    );
19  }
20 }
```

Db_helper.dart

```
1  import 'package:sqflite/sqflite.dart';
2  import 'package:path/path.dart';
3
4  class DbHelper {
5    static final DbHelper _instance = DbHelper._init();
6    static Database? _database;
7
8    DbHelper._init();
9
10   factory DbHelper() => _instance;
11
12   Future<Database> get database async {
13     if (_database != null) return _database!;
14     _database = await _initDB('students.db');
15     return _database!;
16   }
17
18   Future<Database> _initDB(String filePath) async {
19     final dbPath = await getDatabasesPath();
20     final path = join(dbPath, filePath);
21     return await openDatabase(
22       path,
23       version: 1,
24       onCreate: _onCreate,
25     );
26   }
27
28   Future<void> _onCreate(Database db, int version) async {
29     await db.execute('''
30     CREATE TABLE students(
31       id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
32       name TEXT,
33       nim TEXT,
34       address TEXT,
35       hobby TEXT,
36       createdAt TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
37     )
38     ''');
39   }
40
41   Future<int> insert(Map<String, dynamic> data) async {
42     final db = await database;
43     return await db.insert('students', data);
44   }
45
46   Future<List<Map<String, dynamic>>> queryAllRows() async {
47     final db = await database;
48     return await db.query('students', orderBy: 'id DESC');
49   }
50
51   Future<int> delete(int id) async {
52     final db = await database;
53     return await db.delete('students', where: 'id = ?', whereArgs: [id]);
54   }
55 }
56
```

My_db_view.dart

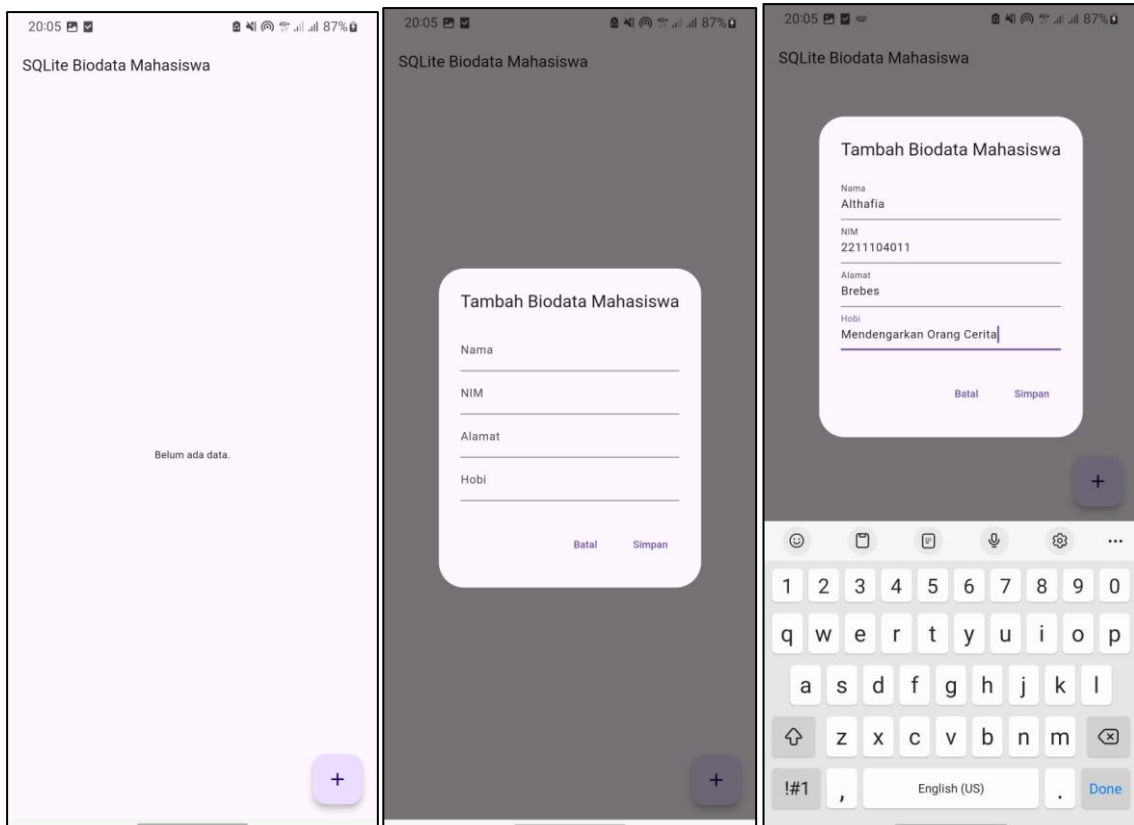
```
1 import 'package:flutter/material.dart';
2 import 'package:praktikum_10/helper/db_helper.dart';
3
4 class MyDbView extends StatefulWidget {
5   @override
6   _MyDbViewState createState() => _MyDbViewState();
7 }
8
9 class _MyDbViewState extends State<MyDbView> {
10   final DbHelper dbHelper = DbHelper();
11   final TextEditingController _nameController = TextEditingController();
12   final TextEditingController _nimController = TextEditingController();
13   final TextEditingController _addressController = TextEditingController();
14   final TextEditingController _hobbyController = TextEditingController();
15   List<Map<String, dynamic>> _dbData = [];
16
17   @override
18   void initState() {
19     super.initState();
20     _refreshData();
21   }
22
23   Future<void> _refreshData() async {
24     final data = await dbHelper.queryAllRows();
25     setState(() {
26       _dbData = data;
27     });
28   }
29
30   Future<void> _addData() async {
31     await dbHelper.insert({
32       'name': _nameController.text,
33       'nim': _nimController.text,
34       'address': _addressController.text,
35       'hobby': _hobbyController.text,
36     });
37     _nameController.clear();
38     _nimController.clear();
39     _addressController.clear();
40     _hobbyController.clear();
41     await _refreshData();
42   }
43
44   Future<void> _deleteData(int id) async {
45     await dbHelper.delete(id);
46     await _refreshData();
47   }
48
49   void _showAddDialog() {
50     _nameController.clear();
51     _nimController.clear();
52     _addressController.clear();
53     _hobbyController.clear();
54
55     showDialog(
56       context: context,
57       builder: (context) {
58         return AlertDialog(
59           title: Text('Tambah Biodata Mahasiswa'),
60           content: SingleChildScrollView(
61             child: Column(
62               mainAxisAlignment: MainAxisAlignment.min,
63               children: [
64                 TextField(
65                   controller: _nameController,
66                   decoration: InputDecoration(labelText: 'Nama'),
67                 ),
68                 TextField(
69                   controller: _nimController,
70                   decoration: InputDecoration(labelText: 'NIM'),
71                 ),
72                 TextField(
73                   controller: _addressController,
74                   decoration: InputDecoration(labelText: 'Alamat'),
75                 ),
76                 TextField(
77                   controller: _hobbyController,
78                   decoration: InputDecoration(labelText: 'Hobi'),
79                 ),
80               ],
81             ),
82           ),
83         ),
84       ),
85     );
86   }
87 }
```

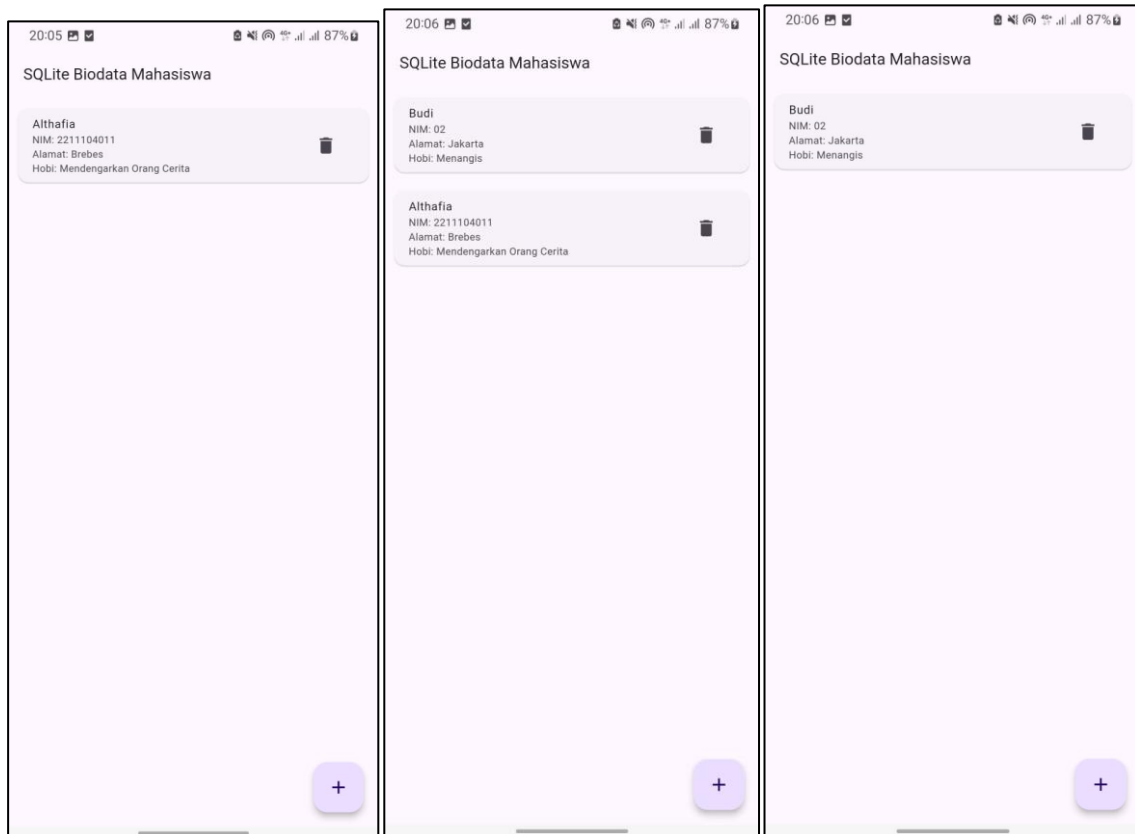
```

83     actions: [
84       TextButton(
85         onPressed: () {
86           Navigator.of(context).pop();
87         },
88         child: Text('Batal'),
89       ),
90       TextButton(
91         onPressed: () {
92           _addData();
93           Navigator.of(context).pop();
94         },
95         child: Text('Simpan'),
96       ),
97     ],
98   );
99 },
100 );
101 }
102
103 @override
104 Widget build(BuildContext context) {
105   return Scaffold(
106     appBar: AppBar(
107       title: Text('SQLite Biodata Mahasiswa'),
108     ),
109     body: _dbData.isEmpty
110       ? Center(child: Text('Belum ada data.'))
111       : ListView.builder(
112         itemCount: _dbData.length,
113         itemBuilder: (context, index) {
114           final item = _dbData[index];
115           return Card(
116             margin: EdgeInsets.all(10),
117             child: ListTile(
118               title: Text(item['name'] ?? 'No Name'),
119               subtitle: Text(
120                 'NIM: ${item['nim']} ?? ''\nAlamat: ${item['address']} ?? ''\nHobi: ${item['hobby']} ?? ''',
121               ),
122               trailing: IconButton(
123                 icon: Icon(Icons.delete),
124                 onPressed: () => _deleteData(item['id']),
125               ),
126             ),
127           );
128         },
129       ),
130     floatingActionButton: FloatingActionButton(
131       onPressed: _showAddDialog,
132       child: Icon(Icons.add),
133     ),
134   );
135 }
136 }
137

```

Output:





Aplikasi ini dirancang menggunakan Flutter dan SQLite untuk mengelola data biodata mahasiswa. Data yang disimpan mencakup Nama, NIM, Alamat, dan Hobi. Pengguna dapat memasukkan data melalui form input, yang kemudian ditampilkan dalam daftar biodata pada halaman utama aplikasi. Fitur utama aplikasi meliputi fungsi Create untuk menyimpan data baru dan Read untuk menampilkan data yang telah disimpan dalam database. Antarmuka aplikasi dirancang sederhana dengan navigasi yang mudah, memungkinkan pengguna untuk mengelola data mahasiswa secara efisien dan intuitif.