

---

## **PRAKTIKUM\_9**

-

## Daftar Isi

<b>1</b>	<b>WARUNG_AJIB</b>	<b>2</b>
1.1	lib/repositories/repositories.dart . . . . .	2
1.2	lib/repositories/user_repository.dart . . . . .	2
1.3	lib/repositories/product_repository.dart . . . . .	3
1.4	lib/models/product.dart . . . . .	3
1.5	lib/models/user.dart . . . . .	4
1.6	lib/models/models.dart . . . . .	5
1.7	lib/main.dart . . . . .	5
1.8	lib/splashscreen.dart . . . . .	6
1.9	lib/pages/register_page.dart . . . . .	7
1.10	lib/pages/user_update_page.dart . . . . .	9
1.11	lib/pages/sms_page.dart . . . . .	11
1.12	lib/pages/maps_page.dart . . . . .	12
1.13	lib/pages/login_page.dart . . . . .	14
1.14	lib/pages/payment_page.dart . . . . .	16
1.15	lib/pages/call_center_page.dart . . . . .	18
1.16	lib/pages/dashboard_page.dart . . . . .	19
1.17	lib/services/auth_services.dart . . . . .	24
1.18	lib/services/db_services.dart . . . . .	25
1.19	lib/services/services.dart . . . . .	26
1.20	screenshot . . . . .	26

## 1 WARUNG\_AJIB

### 1.1 lib/repositories/repositories.dart

```
1 export 'product_repository.dart';
2 export 'user_repository.dart';
```

### 1.2 lib/repositories/user\_repository.dart

```
1 import 'package:sqflite/sqflite.dart';
2 import 'package:warung_ajib/models/models.dart';
3 import 'package:warung_ajib/services/services.dart';
4
5 class UserRepository {
6   final tableName = 'users';
7
8   Future<User?> getUserByUsername(String username) async {
9     final db = await DBService.getDatabase();
10    final result = await db.query(
11      tableName,
12      columns: [
13        'id',
14        'username',
15        'password',
16      ],
17      where: 'username = ?',
18      whereArgs: [username],
19    );
20
21    db.close();
22
23    if (result.isNotEmpty) {
24      return User.fromMap(result.first);
25    } else {
26      return null;
27    }
28  }
29
30  Future<void> addUser(User newUser) async {
31    final db = await DBService.getDatabase();
32    await db.insert(
33      tableName,
34      newUser.toMap(),
35      conflictAlgorithm: ConflictAlgorithm.replace,
36    );
37    db.close();
38  }
39 }
```

### 1.3 lib/repositories/product\_repository.dart

```
1 import 'package:warung_ajib/models/models.dart';
2 import 'package:warung_ajib/services/services.dart';
3
4 class ProductRepository {
5   final tableName = 'products';
6
7   Future<List<Product>?> getAllProduct() async {
8     final db = await DBService.getDatabase();
9     final result = await db.query(tableName, columns: [
10       'id',
11       'name',
12       'description',
13       'price',
14       'image',
15     ]);
16     db.close();
17
18     if (result.isNotEmpty) {
19       return result.map((productMap) => Product.fromMap(productMap)).
20         toList();
21     } else {
22       return null;
23     }
24 }
```

### 1.4 lib/models/product.dart

```
1 class Product {
2   int? _id;
3   String name;
4   int price;
5   String description;
6   String image;
7
8   Product({
9     int? id,
10    required this.name,
11    required this.price,
12    required this.description,
13    required this.image,
14  }) : _id = id;
15 }
```

```
16   int? get id => _id;
17
18   set id(int? id) {
19     if (_id == null) {
20       _id = id;
21     } else {
22       throw Exception('ID sudah diatur dan tidak bisa dirubah lagi');
23     }
24   }
25
26   Map<String, dynamic> toMap() {
27     return {
28       'id': _id,
29       'name': name,
30       'price': price,
31       'description': description,
32       'image': image,
33     };
34   }
35
36   factory Product.fromMap(Map<String, dynamic> map) {
37     return Product(
38       name: map['name'],
39       price: map['price'],
40       description: map['description'],
41       image: map['image'],
42       id: map['id'],
43     );
44   }
45 }
```

### 1.5 lib/models/user.dart

```
1  class User {
2    int? _id;
3    String username;
4    String password;
5
6    User({
7      required this.username,
8      required this.password,
9      int? id,
10   }) : _id = id;
11
12   int? get id => _id;
13
14   set id(int? id) {
15     if (_id == null) {
16       _id = id;
```

```
17     } else {
18       throw Exception('ID sudah diatur dan tidak bisa diubah lagi.');
```

---

```
19     }
20   }
21
22   Map<String, dynamic> toMap() {
23     return {
24       'id': _id,
25       'username': username,
26       'password': password,
27     };
28   }
29
30   factory User.fromMap(Map<String, dynamic> map) {
31     final user = User(
32       username: map['username'],
33       password: map['password'],
34       id: map['id'],
35     );
36     return user;
37   }
38 }
```

## 1.6 lib/models/models.dart

```
1 export 'user.dart';
2 export 'product.dart';
```

## 1.7 lib/main.dart

```
1 import 'package:flutter/material.dart';
2 import 'package:warung_ajib/splashscreen.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: 'Login dan Registrasi',
15       theme: ThemeData(
16         primarySwatch: Colors.blue,
17     ),
```

```
18     home: const SplashScreen(),
19   );
20 }
21 }
```

## 1.8 lib/splashscreen.dart

```
1 import 'dart:async';
2 import 'package:flutter/material.dart';
3 import 'package:warung_ajib/pages/login_page.dart';
4
5 class SplashScreen extends StatefulWidget {
6   const SplashScreen({super.key});
7   @override
8   State<SplashScreen> createState() => _SplashScreenState();
9 }
10
11 class _SplashScreenState extends State<SplashScreen> {
12   startSplashScreen() async {
13     var duration = const Duration(seconds: 5);
14     return Timer(duration, () {
15       Navigator.pushReplacement(context,
16         MaterialPageRoute(builder: (context) => const LoginPage()));
17     });
18   }
19
20   @override
21   void initState() {
22     super.initState();
23     startSplashScreen();
24   }
25
26   @override
27   Widget build(BuildContext context) {
28     return Scaffold(
29       backgroundColor: Colors.blueGrey,
30       body: Column(
31         mainAxisAlignment: MainAxisAlignment.center,
32         crossAxisAlignment: CrossAxisAlignment.center,
33         children: [
34           Center(
35             child: Image.asset(
36               "assets/Estehajib-Jumbo.png",
37               width: 200.0,
38               height: 100.0,
39               fit: BoxFit.contain,
40             ),
41           ),
42           const SizedBox(
```

```
43         height: 24.0,
44       ),
45       const Text(
46         "Selamat Datang\ndi\nWarung Ajib",
47         style: TextStyle(
48           color: Colors.white,
49           fontWeight: FontWeight.bold,
50           fontSize: 35.0,
51         ),
52         textAlign: TextAlign.center,
53       ),
54     ],
55   ),
56 );
57 }
58 }
```

## 1.9 lib/pages/register\_page.dart

```
1  import 'package:flutter/material.dart';
2  import 'package:warung_ajib/services/auth_services.dart';
3
4  class RegisterPage extends StatefulWidget {
5    const RegisterPage({super.key});
6
7    @override
8    _RegisterPageState createState() => _RegisterPageState();
9  }
10
11  class _RegisterPageState extends State<RegisterPage> {
12    final _usernameController = TextEditingController();
13    final _passwordController = TextEditingController();
14    final _auth = AuthService();
15    bool _isLoading = false;
16
17    Future<void> _register() async {
18      final username = _usernameController.text.trim();
19      final password = _passwordController.text.trim();
20
21      if (username.isEmpty || password.isEmpty) {
22        if (mounted) {
23          ScaffoldMessenger.of(context).showSnackBar(
24            const SnackBar(content: Text('Username dan password harus
25                                   diisi!')),
26          );
27        }
28        return;
29      }
29    }
```



```
30     setState(() {
31       _isLoading = true;
32     });
33
34     try {
35       await _auth.register(username, password);
36       if (mounted) {
37         ScaffoldMessenger.of(context).showSnackBar(
38           const SnackBar(content: Text('Registrasi berhasil!')),
39         );
40         Navigator.of(context).pop(); // Kembali ke halaman login
41       }
42     } catch (e) {
43       if (mounted) {
44         ScaffoldMessenger.of(context).showSnackBar(
45           SnackBar(content: Text('Registrasi gagal: $e')),
46         );
47       }
48     } finally {
49       if (mounted) {
50         setState(() {
51           _isLoading = false;
52         });
53       }
54     }
55   }
56
57   @override
58   void dispose() {
59     _usernameController.dispose();
60     _passwordController.dispose();
61     super.dispose();
62   }
63
64   @override
65   Widget build(BuildContext context) {
66     return Scaffold(
67       appBar: AppBar(
68         title: const Text('Registrasi'),
69       ),
70       body: Padding(
71         padding: const EdgeInsets.all(16.0),
72         child: Column(
73           mainAxisAlignment: MainAxisAlignment.center,
74           children: [
75             TextField(
76               controller: _usernameController,
77               decoration: const InputDecoration(labelText: 'Username'),
78             ),
79             TextField(
80               controller: _passwordController,
```

```

81         decoration: const InputDecoration(labelText: 'Password'),
82         obscureText: true,
83     ),
84     const SizedBox(height: 20),
85     _isLoading
86     ? const CircularProgressIndicator()
87     : ElevatedButton(
88         onPressed: _register,
89         child: const Text('Registrasi'),
90     ),
91 ],
92 ),
93 ),
94 );
95 }
96 }

```

### 1.10 lib/pages/user\_update\_page.dart

```

1  import 'package:flutter/material.dart';
2  import 'package:shared_preferences/shared_preferences.dart';
3
4  class UserUpdatePage extends StatefulWidget {
5      const UserUpdatePage({super.key});
6
7      @override
8      _UserUpdatePageState createState() => _UserUpdatePageState();
9  }
10
11  class _UserUpdatePageState extends State<UserUpdatePage> {
12      final _nameController = TextEditingController();
13      final _currentPasswordController = TextEditingController();
14      final _newPasswordController = TextEditingController();
15
16      @override
17      void initState() {
18          super.initState();
19          _loadUserData();
20      }
21
22      Future<void> _loadUserData() async {
23          final prefs = await SharedPreferences.getInstance();
24          setState(() {
25              _nameController.text = prefs.getString('currentUser') ?? '';
26          });
27      }
28
29      Future<void> _updateUserInfoAndPassword() async {
30          final prefs = await SharedPreferences.getInstance();

```

```
31 String currentUser = _nameController.text;
32 String currentPassword = _currentPasswordController.text;
33 String newPassword = _newPasswordController.text;
34
35 // Validasi dan perbarui password
36 String? storedPassword = prefs.getString(currentUser);
37 if (storedPassword != null && storedPassword == currentPassword) {
38   await prefs.setString('userPassword', newPassword);
39   ScaffoldMessenger.of(context).showSnackBar(
40     const SnackBar(
41       content:
42         Text('Informasi pengguna dan password berhasil
43             diperbarui!')),
44   );
45 } else {
46   ScaffoldMessenger.of(context).showSnackBar(
47     const SnackBar(content: Text('Password sekarang salah!')),
48   );
49 }
50
51 @override
52 Widget build(BuildContext context) {
53   return Scaffold(
54     appBar: AppBar(
55       title: const Text('Update User & Password'),
56     ),
57     body: Padding(
58       padding: const EdgeInsets.all(16.0),
59       child: SingleChildScrollView(
60         child: Column(
61           children: [
62             TextField(
63               controller: _nameController,
64               decoration: const InputDecoration(labelText: 'Username'
65             ),
66             ),
67             const SizedBox(height: 20),
68             TextField(
69               controller: _currentPasswordController,
70               decoration:
71                 const InputDecoration(labelText: 'Password Sekarang
72             '),
73             ),
74             obscureText: true,
75             ),
76             TextField(
77               controller: _newPasswordController,
78               decoration: const InputDecoration(labelText: 'Password
79             Baru'),
80             ),
81             obscureText: true,
82             ),
83           ],
84         ),
85       ),
86     ),
87   );
```

```
78         const SizedBox(height: 20),
79         ElevatedButton(
80           onPressed: _updateUserInfoAndPassword,
81           child: const Text('Update'),
82         ),
83         const SizedBox(height: 10),
84         ElevatedButton(
85           onPressed: () => Navigator.pop(context),
86           child: const Text('Batal'),
87         ),
88       ],
89     ),
90   ),
91 ),
92 );
93 }
94 }
```

### 1.11 lib/pages/sms\_page.dart

```
1 import 'package:flutter/material.dart';
2 import 'package:url_launcher/url_launcher.dart';
3
4 class SmsPage extends StatelessWidget {
5   final String phoneNumber = '081234567890';
6
7   const SmsPage({super.key}); // Ganti dengan nomor telepon penjual
8
9   @override
10  Widget build(BuildContext context) {
11    return Scaffold(
12      appBar: AppBar(
13        title: const Text('SMS ke Penjual'),
14      ),
15      body: Padding(
16        padding: const EdgeInsets.all(16.0),
17        child: Column(
18          mainAxisAlignment: MainAxisAlignment.center,
19          children: [
20            const Text(
21              'Kirim SMS ke Penjual',
22              style: TextStyle(fontSize: 24, fontWeight: FontWeight.
23                bold),
24              textAlign: TextAlign.center,
25            ),
26            const SizedBox(height: 20),
27            const Text(
28              'Jika Anda ingin menghubungi penjual melalui SMS, silakan
29              tekan tombol di bawah ini:',
```

```
28         textAlign: TextAlign.center,
29         style: TextStyle(fontSize: 18),
30     ),
31     const SizedBox(height: 20),
32     Text(
33         phoneNumber,
34         style: const TextStyle(fontSize: 22, color: Colors.blue),
35     ),
36     const SizedBox(height: 20),
37     ElevatedButton(
38         onPressed: () => _sendSms(phoneNumber),
39         child: const Text('Kirim SMS'),
40     ),
41 ],
42 ),
43 ),
44 );
45 }
46
47 Future<void> _sendSms(String phoneNumber) async {
48     final Uri launchUri = Uri(
49         scheme: 'sms',
50         path: phoneNumber,
51     );
52     await launchUrl(launchUri);
53 }
54 }
```

## 1.12 lib/pages/maps\_page.dart

```
1 import 'package:flutter/material.dart';
2 import 'package:flutter_svg/flutter_svg.dart';
3 import 'package:map_launcher/map_launcher.dart';
4
5 class MapsPage extends StatelessWidget {
6     final String address =
7         'Jl. Pd. Majapahit I No.b.13, Bandungmulyo, Bandungrejo, Kec.
8         Mranggen, Kabupaten Demak, Jawa Tengah 59567'; // Ganti dengan
9         alamat penjual
10
11     const MapsPage({super.key});
12
13     @override
14     Widget build(BuildContext context) {
15         return Scaffold(
16             appBar: AppBar(
17                 title: const Text('Lokasi Penjual'),
18             ),
19             body: Padding(
```

```
18     padding: const EdgeInsets.all(16.0),
19     child: Column(
20       mainAxisAlignment: MainAxisAlignment.center,
21       children: [
22         const Text(
23           'Temukan Lokasi Kami',
24           style: TextStyle(fontSize: 24, fontWeight: FontWeight.
25             bold),
26           textAlign: TextAlign.center,
27         ),
28         const SizedBox(height: 20),
29         const Text(
30           'Jika Anda ingin mengunjungi kami, berikut adalah
31             alamatnya: ',
32           textAlign: TextAlign.center,
33           style: TextStyle(fontSize: 18),
34         ),
35         const SizedBox(height: 20),
36         Text(
37           address,
38           style: const TextStyle(fontSize: 22, color: Colors.blue),
39           textAlign: TextAlign.center,
40         ),
41         const SizedBox(height: 20),
42         ElevatedButton(
43           onPressed: () => _openMap(context),
44           child: const Text('Buka di Google Maps'),
45         ),
46       ],
47     ),
48   );
49 }
50
51 Future<void> _openMap(BuildContext context) async {
52   try {
53     final coords = Coords(-7.022162, 110.506912);
54     const title = "Warung Ajib";
55     final availableMaps = await MapLauncher.installedMaps;
56
57     showModalBottomSheet(
58       context: context,
59       builder: (BuildContext context) {
60         return SafeArea(
61           child: SingleChildScrollView(
62             child: Container(
63               child: Wrap(
64                 children: <Widget>[
65                   for (var map in availableMaps)
66                     ListTile(
67                       onTap: () => map.showMarker(
```

```

67         coords: coords,
68         title: title,
69       ),
70       title: Text(map.mapName),
71       leading: SvgPicture.asset(
72         map.icon,
73         height: 30.0,
74         width: 30.0,
75       ),
76     ),
77   ],
78 ),
79 ),
80 ),
81 );
82 },
83 );
84 } catch (e) {
85   print(e);
86 }
87 }
88 }

```

### 1.13 lib/pages/login\_page.dart

```

1  import 'package:flutter/material.dart';
2  import 'package:warung_ajib/pages/dashboard_page.dart';
3  import 'package:warung_ajib/pages/register_page.dart';
4  import 'package:warung_ajib/services/services.dart';
5
6  class LoginPage extends StatefulWidget {
7    const LoginPage({super.key});
8
9    @override
10   _LoginPageState createState() => _LoginPageState();
11 }
12
13 class _LoginPageState extends State<LoginPage> {
14   final _usernameController = TextEditingController();
15   final _passwordController = TextEditingController();
16   final _auth = AuthService();
17
18   Future<void> _login() async {
19     final username = _usernameController.text;
20     final password = _passwordController.text;
21     final loginResult = await _auth.login(username, password);
22
23     if (loginResult) {
24       // Login berhasil

```

```
25     ScaffoldMessenger.of(context).showSnackBar(  
26         const SnackBar(content: Text('Login berhasil!')),  
27     );  
28     Navigator.pushReplacement(  
29         context,  
30         MaterialPageRoute(builder: (context) => const DashboardPage()),  
31     );  
32 } else {  
33     // Gagal login  
34     ScaffoldMessenger.of(context).showSnackBar(  
35         const SnackBar(content: Text('Username atau password salah.')),  
36     );  
37 }  
38 }  
39  
40 @override  
41 Widget build(BuildContext context) {  
42     return Scaffold(  
43         appBar: AppBar(  
44             title: const Text('Login'),  
45         ),  
46         body: Padding(  
47             padding: const EdgeInsets.all(16.0),  
48             child: Column(  
49                 children: [  
50                     TextField(  
51                         controller: _usernameController,  
52                         decoration: const InputDecoration(labelText: 'Username'),  
53                     ),  
54                     TextField(  
55                         controller: _passwordController,  
56                         decoration: const InputDecoration(labelText: 'Password'),  
57                         obscureText: true,  
58                     ),  
59                     const SizedBox(height: 20),  
60                     ElevatedButton(  
61                         onPressed: _login,  
62                         child: const Text('Login'),  
63                     ),  
64                     TextButton(  
65                         onPressed: () {  
66                             Navigator.of(context).push(MaterialPageRoute(  
67                                 builder: (context) => const RegisterPage(),  
68                             ));  
69                         },  
70                         child: const Text('Belum punya akun? Registrasi di sini')  
71                     ),  
72                 ],  
73             ),  
74         ),  
75     );  
76 }
```



```
75     );  
76   }  
77 }
```

### 1.14 lib/pages/payment\_page.dart

```
1  import 'package:flutter/material.dart';  
2  import 'package:warung_ajib/models/product.dart';  
3  
4  class PaymentPage extends StatelessWidget {  
5    final int totalTransaction;  
6    final Map<Product, int> productQuantityMap;  
7    final VoidCallback onPaymentComplete;  
8  
9    const PaymentPage({  
10     super.key,  
11     required this.totalTransaction,  
12     required this.productQuantityMap,  
13     required this.onPaymentComplete,  
14   });  
15  
16   @override  
17   Widget build(BuildContext context) {  
18     return Scaffold(  
19       appBar: AppBar(  
20         title: const Text('Pembayaran'),  
21       ),  
22       body: Padding(  
23         padding: const EdgeInsets.all(16.0),  
24         child: Column(  
25           children: [  
26             const Center(  
27               child: Text(  
28                 'Daftar Produk:',  
29                 style: TextStyle(fontSize: 20, fontWeight: FontWeight.  
30                   bold),  
31             ),  
32             const SizedBox(height: 10),  
33             Expanded(  
34               child: ListView(  
35                 children: productQuantityMap.entries  
36                   .where((entry) => entry.value > 0)  
37                   .map((entry) => ListTile(  
38                     title: Text(entry.key.name),  
39                     subtitle: Text('Jumlah: ${entry.value}'),  
40                     trailing: Text('Rp. ${entry.key.price * entry  
41                       .value}'),  
42                   ))  
43             ),  
44           ],  
45         ),  
46       ),  
47     );  
48   }  
49 }
```

```
42         .toList(),
43     ),
44 ),
45     const SizedBox(height: 20),
46     const Center(
47       child: Text(
48         'Total Transaksi:',
49         style: TextStyle(fontSize: 20, fontWeight: FontWeight.
          bold),
50       ),
51     ),
52     Center(
53       child: Text(
54         'Rp. $totalTransaction',
55         style: const TextStyle(fontSize: 18),
56       ),
57     ),
58     const SizedBox(height: 20),
59     ElevatedButton(
60       onPressed: () async {
61         final result = await showDialog<bool>(
62           context: context,
63           builder: (context) => AlertDialog(
64             title: const Text('Konfirmasi Pembayaran'),
65             content: const Text('Apakah Anda yakin ingin
              melakukan pembayaran?'),
66             actions: [
67               TextButton(
68                 onPressed: () => Navigator.pop(context, false),
69                 child: const Text('Batal'),
70               ),
71               TextButton(
72                 onPressed: () => Navigator.pop(context, true),
73                 child: const Text('Ya'),
74             ),
75           ],
76         ),
77     );
78
79     if (result == true) {
80       onPaymentComplete();
81       Navigator.pop(context);
82     }
83   },
84   child: const Text('Bayar'),
85 ),
86 ],
87 ),
88 ),
89 );
90 }
```

```
91 }
```

### 1.15 lib/pages/call\_center\_page.dart

```
1 import 'package:flutter/material.dart';
2 import 'package:url_launcher/url_launcher.dart';
3
4 class CallCenterPage extends StatelessWidget {
5   final String phoneNumber = '081234567890';
6
7   const CallCenterPage({super.key}); // Ganti dengan nomor telepon
   penjual
8
9   @override
10  Widget build(BuildContext context) {
11    return Scaffold(
12      appBar: AppBar(
13        title: const Text('Call Center Penjual'),
14      ),
15      body: Padding(
16        padding: const EdgeInsets.all(16.0),
17        child: Column(
18          mainAxisAlignment: MainAxisAlignment.center,
19          children: [
20            const Text(
21              'Hubungi Call Center Kami',
22              style: TextStyle(fontSize: 24, fontWeight: FontWeight.
                bold),
23              textAlign: TextAlign.center,
24            ),
25            const SizedBox(height: 20),
26            const Text(
27              'Jika Anda memiliki pertanyaan atau masalah, silakan
                hubungi kami di:',
28              textAlign: TextAlign.center,
29              style: TextStyle(fontSize: 18),
30            ),
31            const SizedBox(height: 20),
32            Text(
33              phoneNumber,
34              style: const TextStyle(fontSize: 22, color: Colors.blue),
35            ),
36            const SizedBox(height: 20),
37            ElevatedButton(
38              onPressed: () => _makePhoneCall(phoneNumber),
39              child: const Text('Hubungi Sekarang'),
40            ),
41          ],
42        ),
43      ),
44    );
```

```
43     ),
44   );
45 }
46
47 Future<void> _makePhoneCall(String phoneNumber) async {
48   final Uri launchUri = Uri(
49     scheme: 'tel',
50     path: phoneNumber,
51   );
52   await launchUrl(launchUri);
53 }
54 }
```

### 1.16 lib/pages/dashboard\_page.dart

```
1 import 'package:flutter/material.dart';
2 import 'package:warung_ajib/models/product.dart';
3 import 'package:warung_ajib/pages/call_center_page.dart';
4 import 'package:warung_ajib/pages/login_page.dart';
5 import 'package:warung_ajib/pages/maps_page.dart';
6 import 'package:warung_ajib/pages/payment_page.dart';
7 import 'package:warung_ajib/pages/sms_page.dart';
8 import 'package:warung_ajib/pages/user_update_page.dart';
9 import 'package:warung_ajib/repositories/repositories.dart';
10
11 class DashboardPage extends StatefulWidget {
12   const DashboardPage({super.key});
13
14   @override
15   _DashboardPageState createState() => _DashboardPageState();
16 }
17
18 class _DashboardPageState extends State<DashboardPage> {
19   final _productRepository = ProductRepository();
20   final Map<Product, int> productQuantityMap = {};
21   int totalHarga = 0;
22
23   void _tambahKeTotal(Product product) {
24     setState(() {
25       totalHarga += product.price;
26       productQuantityMap[product] = (productQuantityMap[product] ?? 0)
27         + 1;
28     });
29   }
30
31   void _resetTransaction() {
32     setState(() {
33       totalHarga = 0;
34       productQuantityMap.clear();
35     });
36   }
37 }
```

```
34     });
35   }
36
37   void _showMenu(String choice) {
38     switch (choice) {
39       case 'Update User & Password':
40         Navigator.push(
41           context,
42           MaterialPageRoute(builder: (context) => const UserUpdatePage
43             ()),
44         );
45         break;
46       case 'Call Center Penjual':
47         Navigator.push(
48           context,
49           MaterialPageRoute(builder: (context) => const CallCenterPage
50             ()),
51         );
52         break;
53       case 'SMS ke Penjual':
54         Navigator.push(
55           context,
56           MaterialPageRoute(builder: (context) => const SmsPage()),
57         );
58         break;
59       case 'Maps Lokasi Agen':
60         Navigator.push(
61           context,
62           MaterialPageRoute(builder: (context) => const MapsPage()),
63         );
64         break;
65       case 'Pembayaran':
66         Navigator.push(
67           context,
68           MaterialPageRoute(
69             builder: (context) => PaymentPage(
70               totalTransaction: totalHarga,
71               productQuantityMap: productQuantityMap,
72               onPaymentComplete: _resetTransaction,
73             ),
74         );
75         break;
76       case 'Logout':
77         Navigator.pushReplacement(
78           context,
79           MaterialPageRoute(builder: (context) => const LoginPage()),
80         );
81         break;
82     }
```

```
83
84 @override
85 Widget build(BuildContext context) {
86   return Scaffold(
87     appBar: AppBar(
88       title: const Text('Warung Ajib'),
89       actions: [
90         PopupMenuButton<String>(
91           onSelected: _showMenu,
92           itemBuilder: (BuildContext context) {
93             return [
94               const PopupMenuItem(
95                 value: 'Update User & Password',
96                 child: Text('Update User & Password'),
97               ),
98               const PopupMenuItem<String>(
99                 value: 'Call Center Penjual',
100                 child: Text('Call Center Penjual'),
101               ),
102               const PopupMenuItem<String>(
103                 value: 'SMS ke Penjual',
104                 child: Text('SMS ke Penjual'),
105               ),
106               const PopupMenuItem<String>(
107                 value: 'Maps Lokasi Agen',
108                 child: Text('Maps Lokasi Agen'),
109               ),
110               const PopupMenuItem<String>(
111                 value: 'Pembayaran',
112                 child: Text('Pembayaran'),
113               ),
114               const PopupMenuItem<String>(
115                 value: 'Logout',
116                 child: Text('Logout'),
117               ),
118             ];
119           },
120         ),
121       ],
122     ),
123     body: FutureBuilder<List<Product>?>(
124       future: _productRepository.getAllProduct(),
125       builder: (context, snapshot) {
126         if (snapshot.connectionState == ConnectionState.waiting) {
127           return const Center(child: CircularProgressIndicator());
128         } else if (snapshot.hasError) {
129           return Center(
130             child: Text('Error: ${snapshot.error}'),
131           );
132         } else if (!snapshot.hasData || snapshot.data!.isEmpty) {
133           return const Center(
```

```

134         child: Text('Produk tidak tersedia'),
135       );
136     } else {
137       final products = snapshot.data!;
138       return SingleChildScrollView(
139         child: Column(
140           children: [
141             GridView.builder(
142               gridDelegate: const
143                 SliverGridDelegateWithFixedCrossAxisCount(
144                   crossAxisCount: 2,
145                   crossAxisSpacing: 8.0,
146                   mainAxisSpacing: 8.0,
147                   childAspectRatio: 0.7,
148                 ),
149               itemCount: products.length,
150               shrinkWrap: true,
151               physics: const NeverScrollableScrollPhysics(),
152               itemBuilder: (context, index) {
153                 final product = products[index];
154                 return _buildProductCard(context, product);
155               },
156             ),
157           ],
158         ),
159       );
160     },
161   ),
162   bottomNavigationBar: GestureDetector(
163     onTap: () {
164       _showMenu('Pembayaran');
165     },
166     child: BottomAppBar(
167       color: Colors.blue.shade100,
168       child: Padding(
169         padding: const EdgeInsets.all(16.0),
170         child: Text(
171           "Total: Rp. $totalHarga",
172           style: const TextStyle(fontSize: 18, fontWeight:
173             FontWeight.bold),
174           textAlign: TextAlign.center,
175         ),
176       ),
177     ),
178   );
179 }
180
181 Widget _buildProductCard(BuildContext context, Product product) {
182   return GestureDetector(

```

```
183     onTap: () {
184       _tambahKeTotal(product);
185     },
186     child: Card(
187       elevation: 4.0,
188       shape: RoundedRectangleBorder(borderRadius: BorderRadius.
189         circular(10)),
189       child: Column(
190         crossAxisAlignment: CrossAxisAlignment.stretch,
191         children: [
192           ClipRRect(
193             borderRadius:
194               const BorderRadius.vertical(top: Radius.circular(10))
195             ,
196             child: Image.asset(
197               'assets/images/${product.image}',
198               fit: BoxFit.cover,
199               height: 100,
200               width: double.infinity,
201             ),
202           Expanded(
203             child: Padding(
204               padding: const EdgeInsets.all(8.0),
205               child: Column(
206                 crossAxisAlignment: CrossAxisAlignment.start,
207                 children: [
208                   Text(
209                     product.name,
210                     maxLines: 1,
211                     style: const TextStyle(
212                       fontSize: 14, fontWeight: FontWeight.bold),
213                   ),
214                   const SizedBox(height: 4),
215                   Text(
216                     product.description,
217                     maxLines: 4,
218                     overflow: TextOverflow.ellipsis,
219                     style: const TextStyle(fontSize: 14),
220                   ),
221                   const Spacer(),
222                   Text(
223                     "Rp. ${product.price}",
224                     style: const TextStyle(
225                       fontSize: 16,
226                       fontWeight: FontWeight.bold,
227                   ),
228                 ),
229               ],
230             ),
231           ),
```



```
232         ),
233       ],
234     ),
235   ),
236 );
237 }
238 }
```

### 1.17 lib/services/auth\_services.dart

```
1 import 'dart:convert';
2 import 'package:crypto/crypto.dart';
3 import 'package:warung_ajib/repositories/repositories.dart';
4 import 'package:warung_ajib/models/models.dart';
5
6 class AuthService {
7   static final AuthService _instance = AuthService._internal();
8   factory AuthService() => _instance;
9   AuthService._internal();
10
11   final UserRepository _userRepository = UserRepository();
12
13   User? _currentUser;
14
15   User? get currentUser => _currentUser;
16
17   String _hashPassword(String password) {
18     final bytes = utf8.encode(password);
19     final digest = sha256.convert(bytes);
20     return digest.toString();
21   }
22
23   Future<bool> login(String username, String password) async {
24     final hashedPassword = _hashPassword(password);
25     final user = await _userRepository.getUserByUsername(username);
26
27     if (user != null && user.password == hashedPassword) {
28       _currentUser = user;
29       return true;
30     }
31
32     return false;
33   }
34
35   Future<bool> register(
36     String username,
37     String password,
38   ) async {
```

```
39     final existingUser = await _userRepository.getUserByUsername(
40         username);
41     existingUser ?? false;
42     final newUser = User(
43         username: username,
44         password: _hashPassword(password),
45     );
46
47     await _userRepository.addUser(newUser);
48     _currentUser = newUser;
49
50     return true;
51 }
52
53 bool logout() {
54     _currentUser = null;
55     return true;
56 }
57
58 bool isLoggedIn() {
59     return _currentUser != null;
60 }
61 }
```

### 1.18 lib/services/db\_services.dart

```
1 import 'dart:developer';
2
3 import 'package:sqflite/sqflite.dart';
4 import 'package:path/path.dart';
5
6 class DBService {
7     static Database? _database;
8
9     static Future<Database> getDatabase() async {
10         if (_database != null) return _database!;
11
12         _database = await _initDB();
13         return _database!;
14     }
15
16     static Future<Database> _initDB() async {
17         final dbPath = await getDatabasesPath();
18         final path = join(dbPath, 'app_database.db');
19
20         log('SQLite Location: $path');
21
22         return await openDatabase(
```

```
23     path,  
24     version: 1,  
25     onCreate: _onCreate,  
26   );  
27 }  
28  
29 static Future<void> _onCreate(Database db, int version) async {  
30   await db.execute('''  
31     CREATE TABLE products(  
32       id INTEGER PRIMARY KEY AUTOINCREMENT,  
33       name TEXT,  
34       price INTEGER,  
35       description TEXT,  
36       image TEXT  
37     )  
38   ''');  
39   await db.execute('''  
40     CREATE TABLE users(  
41       id INTEGER PRIMARY KEY AUTOINCREMENT,  
42       username TEXT,  
43       password TEXT  
44     )  
45   ''');  
46   db.close();  
47 }  
48  
49 static Future<void> close() async {  
50   final db = await getDatabase();  
51   db.close();  
52 }  
53 }
```

### 1.19 lib/services/services.dart

```
1 export 'db_services.dart';  
2 export 'auth_services.dart';
```

### 1.20 screenshot