TUGAS PENDAHULUAN
PEMROGRAMAN PERANGKAT BERGERAK

MODUL VII

NAVIGASI DAN NOTIFIKASI

Disusun Oleh :

Aditya prabu mukti / 2211104037

SE-06-02

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

TUGAS PENDAHULUAN

A. SOAL NOMOR 1

a) Buatlah satu project baru, yang mana di dalamnya terdapat navigasi untuk pindah dari satu halaman ke halaman lainnya.

Sourcecode:

1. **login_page.dart**

```
import 'package:flutter/material.dart';
import 'home_page.dart';

class LoginPage extends StatelessWidget {
  final TextEditingController _usernameController = TextEditingController();
  final TextEditingController _passwordController = TextEditingController();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.blueGrey[50],
      appBar: AppBar(
        title: Text('Login', style: TextStyle(fontWeight: FontWeight.bold)),
        backgroundColor: Colors.blueAccent,
        centerTitle: true,
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Center(
          child: SingleChildScrollView(
            child: Column(
              mainAxisAlignment: MainAxisAlignment.center,
              children: [
                Image.asset(
                  'assets/logo.png',
        height: 100,
                ),
                SizedBox(height: 20),
                Text(
                  'Welcome Back!',
                  style: TextStyle(
                    fontSize: 24,
                    fontWeight: FontWeight.bold,
                    color: Colors.blueAccent,
                  ),
                ),
                SizedBox(height: 16),
                Text(
                  'Please login to continue',
                  style: TextStyle(
                    fontSize: 16,
                    color: Colors.grey[600],
                  ),
                ),
                SizedBox(height: 32),
                TextField(
                  controller: _usernameController,
                  decoration: InputDecoration(
                    labelText: 'Username',
                    prefixIcon: Icon(Icons.person),
                    border: OutlineInputBorder(
                      borderRadius: BorderRadius.circular(12),
                    ),
                    filled: true,
                    fillColor: Colors.white,
                  ),
                ),
                SizedBox(height: 16),
                TextField(
                  controller: _passwordController,
                  decoration: InputDecoration(
                    labelText: 'Password',
                    prefixIcon: Icon(Icons.lock),
                    border: OutlineInputBorder(
                      borderRadius: BorderRadius.circular(12),
                    ),
                    filled: true,
                    fillColor: Colors.white,
                  ),
                  obscureText: true,
                ),
                SizedBox(height: 32),
                ElevatedButton(
                  onPressed: () {
                    final username = _usernameController.text;
                    Navigator.pushReplacement(
                      context,
                      MaterialPageRoute(
                          builder: (context) => HomePage(username:
username)),
                    );
                  },
                  style: ElevatedButton.styleFrom(
                    padding: EdgeInsets.symmetric(horizontal: 40, vertical:
15),
                    shape: RoundedRectangleBorder(
                      borderRadius: BorderRadius.circular(20),
                    ),

                    elevation: 5,
                  ),
                  child: Text(
                    'Login',
                    style: TextStyle(fontSize: 18, fontWeight:
FontWeight.bold),
                  ),
                ),
              ],
            ),
          ),
        ),
      ),
    );
  }
}
```

2. **home_page.dart**

```
import 'package:flutter/material.dart';
import 'login_page.dart';
import 'profile_page.dart';

class HomePage extends StatelessWidget {
  final String username;

  HomePage({required this.username});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Beranda'),
        actions: [
          IconButton(
            icon: Icon(Icons.logout),
            onPressed: () {
              // Aksi logout, kembali ke LoginPage
              Navigator.pushAndRemoveUntil(
                context,
                MaterialPageRoute(builder: (context) => LoginPage()),
                (Route<dynamic> route) => false,
              );
            },
          ),
          IconButton(
            icon: Icon(Icons.account_circle),
            onPressed: () {
              Navigator.push(
                context,
                MaterialPageRoute(
                    builder: (context) => ProfilePage(username: username)),
              );
            },
          ),
        ],
      ),
      body: Center(
        child: Text(
          'Selamat Datang, $username!',
          style: TextStyle(fontSize: 20),
        ),
      ),
    );
  }
}
```

3. profile_page.dart

```
import 'package:flutter/material.dart';

class ProfilePage extends StatelessWidget {
  final String username;

  ProfilePage({required this.username});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Profile'),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            Icon(Icons.account_circle, size: 100, color: Colors.blue),
            SizedBox(height: 20),
            Text(
              username,
              style: TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
            ),
            SizedBox(height: 10),
            Text(
              '$username@example.com',
              style: TextStyle(fontSize: 16),
            ),
          ],
        ),
      ),
    );
  }
}
```

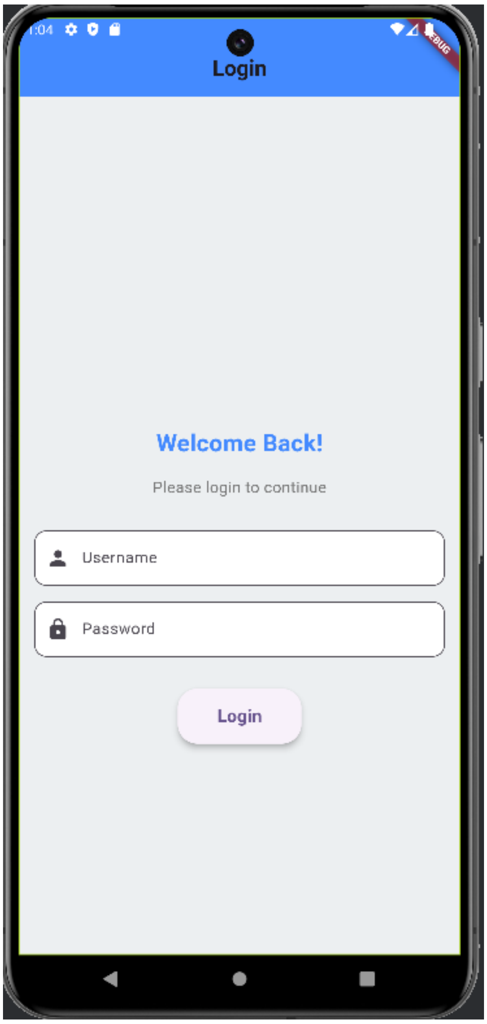
4. main.dart

```

import 'package:flutter/material.dart';
import 'login_page.dart';

```

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

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

  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        // This is the theme of your application.
        //
        // TRY THIS: Try running your application with "flutter run". You'll
see
        // the application has a purple toolbar. Then, without quitting the
app,
        // try changing the seedColor in the colorScheme below to
Colors.green
        // and then invoke "hot reload" (save your changes or press the "hot
        // reload" button in a Flutter-supported IDE, or press "r" if you
used
        // the command line to start the app).
        //
        // Notice that the counter didn't reset back to zero; the application
        // state is not lost during the reload. To reset the state, use hot
        // restart instead.
        //
        // This works for code too, not just values: Most code changes can be
        // tested with just a hot reload.
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
        useMaterial3: true,
      ),
      home: LoginPage(),
    );
  }
}

class MyHomePage extends StatefulWidget {
  const MyHomePage({super.key, required this.title});

  // This widget is the home page of your application. It is stateful,
meaning
  // that it has a State object (defined below) that contains fields that
affect
  // how it looks.

  // This class is the configuration for the state. It holds the values (in
this
  // case the title) provided by the parent (in this case the App widget) and
  // used by the build method of the State. Fields in a Widget subclass are
  // always marked "final".

  final String title;

  @override
  State<MyHomePage> createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  int _counter = 0;

  void _incrementCounter() {
    setState(() {
      // This call to setState tells the Flutter framework that something has
      // changed in this State, which causes it to rerun the build method
below
      // so that the display can reflect the updated values. If we changed
      // _counter without calling setState(), then the build method would not
be
      // called again, and so nothing would appear to happen.
      _counter++;
    });
  }

  @override
  Widget build(BuildContext context) {
    // This method is rerun every time setState is called, for instance as
done
    // by the _incrementCounter method above.
    //
    // The Flutter framework has been optimized to make rerunning build
methods
    // fast, so that you can just rebuild anything that needs updating rather
    // than having to individually change instances of widgets.
    return Scaffold(
      appBar: AppBar(
        // TRY THIS: Try changing the color here to a specific color (to
        // Colors.amber, perhaps?) and trigger a hot reload to see the AppBar
        // change color while the other colors stay the same.
        backgroundColor: Theme.of(context).colorScheme.inversePrimary,
        // Here we take the value from the MyHomePage object that was created
by
        // the App.build method, and use it to set our appbar title.
        title: Text(widget.title),
      ),
      body: Center(
        // Center is a layout widget. It takes a single child and positions
it
        // in the middle of the parent.
        child: Column(
          // Column is also a layout widget. It takes a list of children and
          // arranges them vertically. By default, it sizes itself to fit its
          // children horizontally, and tries to be as tall as its parent.
          //
          // Column has various properties to control how it sizes itself and
          // how it positions its children. Here we use mainAxisAlignment to
          // center the children vertically; the main axis here is the
vertical
          // axis because Columns are vertical (the cross axis would be
          // horizontal).
          //
          // TRY THIS: Invoke "debug painting" (choose the "Toggle Debug
Paint"
          // action in the IDE, or press "p" in the console), to see the
          // wireframe for each widget.
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            const Text(
              'You have pushed the button this many times:',
            ),
            Text(
              '$_counter',
              style: Theme.of(context).textTheme.headlineMedium,
            ),
          ],
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: _incrementCounter,
        tooltip: 'Increment',
        child: const Icon(Icons.add),
      ), // This trailing comma makes auto-formatting nicer for build
methods.
    );
  }
}
```

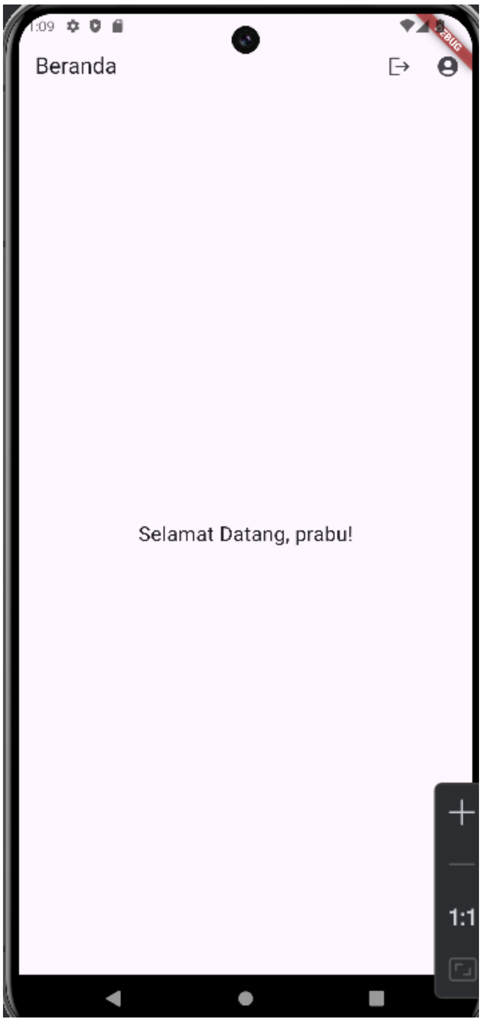
Screenshot Output:

Login:



1:04
DEBUG
Login
Welcome Back!
Please login to continue
Username
Password
Login

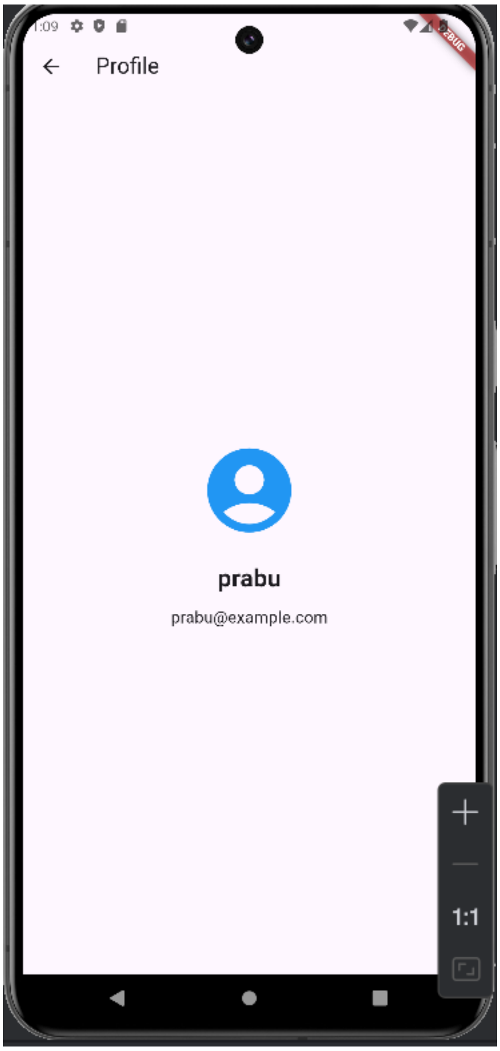
Home:



Beranda

Selamat Datang, prabu!

Profile:



prabu

prabu@example.com

Deskripsi Program

Program ini adalah aplikasi mobile berbasis Flutter yang terdiri dari tiga halaman utama: Login, Beranda, dan Profil. Program ini menggunakan algoritma navigasi antar-halaman dengan parameter yang memungkinkan transfer data antar-halaman. Pengguna memulai dengan memasukkan nama pengguna dan kata sandi pada halaman login; data nama pengguna yang dimasukkan akan dikirimkan ke halaman beranda sebagai parameter. Di halaman beranda, program menampilkan pesan selamat datang yang dipersonalisasi dengan nama pengguna, dan menyediakan opsi untuk membuka halaman profil atau melakukan logout. Pada halaman profil, pengguna dapat melihat nama akun yang sama, dan memiliki tombol logout untuk kembali ke halaman login. Output program adalah tampilan pesan selamat datang dengan nama pengguna di beranda, detail akun di profil, dan transisi kembali ke login saat pengguna logout.