

**LAPORAN PRAKTIKUM
PEMROGRAMAN PERANGKAT BERGERAK**

**MODUL VI
INTERAKSI PENGGUNA**



Disusun Oleh :

Dewi Atika Muthi / 2211104042

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

BAB I

PENDAHULUAN

A. DASAR TEORI

Package pada Flutter merupakan kumpulan code yang dapat digunakan ulang, terdapat dalam direktori yang berisi file pubspec.yaml, packages dapat diakses melalui pub.dev. Contoh penggunaan: HTTP requests, navigation handling, state management.

User Interaction Components

1) Widget Types

- Stateless Widget: Widget statis yang tidak memiliki state internal
- Stateful Widget: Widget dinamis yang dapat berubah berdasarkan interaksi atau data

2) Form Elements

- TextFormField: Input field untuk text
- Checkbox: Untuk multiple selection
- Radio: Untuk single selection
- Form widget: Container untuk form validation dan submission

3) Navigation

- TabBar: Horizontal navigation dengan tabs
- BottomNavigationBar: Navigation bar di bagian bawah layar

4) Button Types

- ElevatedButton: Button dengan elevation effect
- TextButton: Flat button dengan text
- DropdownButton: Button untuk menampilkan dropdown options

B. MAKSUD DAN TUJUAN

Maksud dan tujuan praktikum ini adalah antara lain:

- 1) Memahami dan mengimplementasikan konsep layout pada Flutter
- 2) Mengembangkan kemampuan dalam membuat user interface yang interaktif
- 3) Mempelajari penggunaan berbagai jenis widget untuk interaksi pengguna
- 4) Mengimplementasikan form dan validasi input dalam Flutter

BAB II IMPLEMENTASI

A. PRAKTIKUM

1) Implementasi Package

Soal Studi Case

Implementasi Google Fonts Package Langkah-langkah:

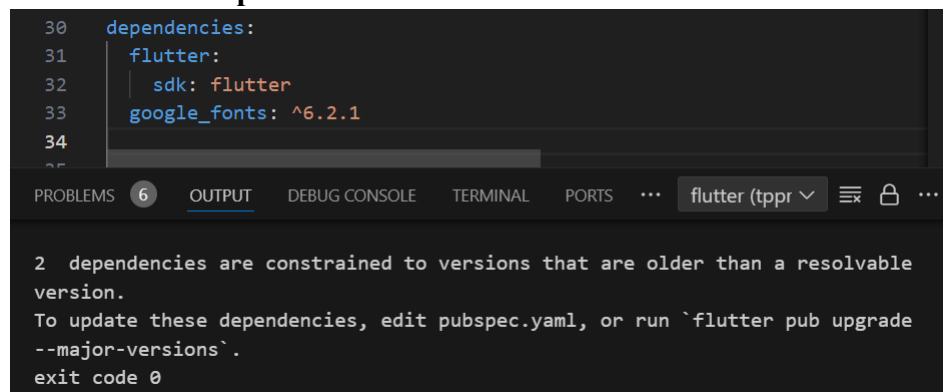
Membuka website pub.dev untuk mengecek versi dan instalasi:

https://pub.dev/packages/google_fonts/install

Sourcecode

```
# pubspec.yaml
dependencies:
  flutter:
    sdk: flutter
  google_fonts: ^6.1.0
```

Screenshoot Output



Deskripsi Program:

- Menambahkan dependency di pubspec.yaml (save)
- Melakukan pub get untuk mengunduh package
- Mengimport package dalam kode

2) Implementasi Package dan Form

Soal Studi Case

Halaman Login dengan Custom Styling

Sourcecode

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

class MyPackage extends StatefulWidget {
  const MyPackage({super.key});

  @override
```

```

    State<MyPackage> createState() => _MyPackageState();
  }

class _MyPackageState extends State<MyPackage> {
  String? selectedValue;

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(
          "belajar package",
          style: GoogleFonts.libreBaskerville(),
        ),
        backgroundColor: Colors.pink[200],
      ),
      body: Padding(
        padding: EdgeInsets.all(16.0),
        child: Column(
          children: [
            TextField(
              decoration: InputDecoration(
                // icon: Icon(Icons.person),
                prefixIcon: Icon(Icons.person),
                hintText: "masukan username",
                border: OutlineInputBorder(),
              ),
            ),
            SizedBox(
              height: 18,
            ),
            TextField(
              obscureText: true,
              obscuringCharacter: "*",
              decoration: InputDecoration(
                // icon: Icon(Icons.lock),
                prefixIcon: Icon(Icons.lock),
                hintText: "masukan password",
                border: OutlineInputBorder(),
              ),
            ),
            SizedBox(
              height: 15,
            ),
            ElevatedButton(
              onPressed: () {},
              child: Text(
                "ini adalah eleated botom",
                style: TextStyle(
                  color: Colors.white,
                  fontSize: 16,
                ),
              ),
              style:
                ElevatedButton.styleFrom(backgroundColor:
Colors.pink[400]),
            ),
            SizedBox(
              height: 15,
            ),
            TextButton(
              // style: ButtonStyle(
              //   backgroundColor: Colors.amberAccent,

```


```


        // ),
        onPressed: () {},
        child: Text(
            "ini adalah text button",
            style: TextStyle(fontSize: 20),
        ),
    ),
    SizedBox(
        height: 15,
    ),
    DropdownButton<String>(
        isExpanded: true,
        value: selectedValue,
        hint: Text('Pilih Opsi'),
        items: <String>['Opsi 1', 'Opsi 2', 'Opsi 3']
            .map<DropdownMenuItem<String>>((String value) {
                return DropdownMenuItem<String>(
                    value: value,
                    child: Text(value),
                );
            }).toList(),
        onChanged: (String? newValue) {
            setState(() {
                selectedValue = newValue;
            });
        },
    ),
],
),
),
);
}
}

```

Screenshoot Output

belajar package

 masukan username

 masukan password

ini adalah eleated botom

ini adalah text button

Pilih Opsi

belajar package

 tikature

 *****

ini adalah eleated botom

ini adalah text button

Opsi 1

Opsi 2

Opsi 3

Deskripsi Program:

Implementasi Google Fonts

- Menggunakan package google_fonts untuk custom font 'libreBaskerville'
- Diterapkan pada judul AppBar untuk styling yang unik

Form Elements

- TextField untuk username dengan icon person
- TextField untuk password dengan fitur obscureText
- Kedua field menggunakan OutlineInputBorder untuk konsistensi desain

Button Implementation

- ElevatedButton dengan custom styling (warna pink)
- TextButton sebagai alternatif style button
- Custom text styling pada buttons

Dropdown Implementation

- DropdownButton dengan 3 opsi
- Menggunakan state management untuk tracking selected value
- isExpanded: true untuk full width dropdown

Layout Management

- Penggunaan Padding untuk spacing konsisten
- SizedBox untuk memberikan jarak antar elements
- Column untuk vertical arrangement

3) Implementasi Tabbar

Soal Studi Case

Menu Aplikasi dengan Tab Navigation

Sourcecode

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

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

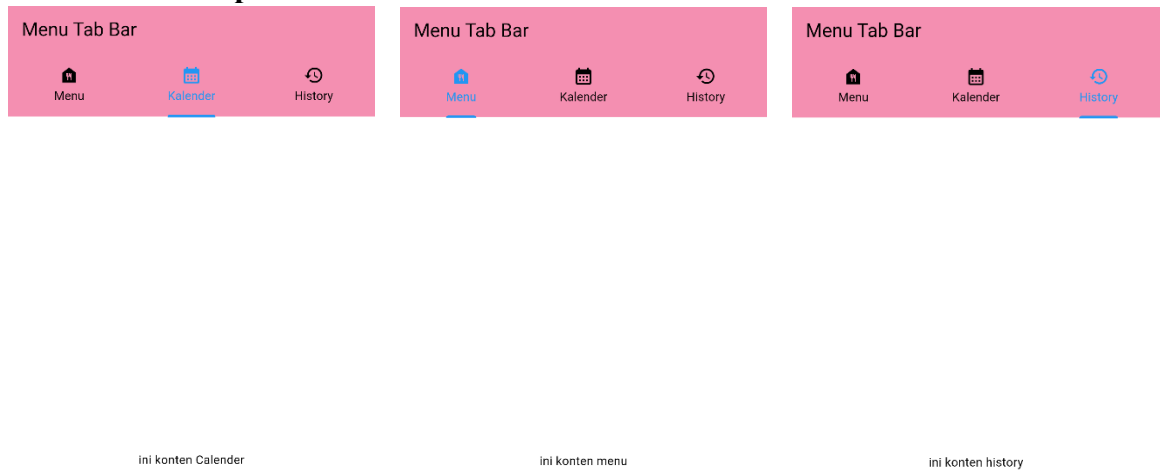
  @override
  Widget build(BuildContext context) {
    return DefaultTabController(
      length: 3,
      child: Scaffold(
        appBar: AppBar(
          title: const Text("Menu Tab Bar"),
          bottom: const TabBar(tabs: [
            Tab(
              icon: Icon(Icons.food_bank),
              text: "Menu",
            ),
            Tab(
              icon: Icon(Icons.calendar_month),
              text: "Kalender",
            ),
            Tab(
              icon: Icon(Icons.history),
              text: "History",
            ),
          ]),
        backgroundColor: Colors.pink[200],
      ),
    );
  }
}
```

```

    ),
    body: const TabBarView(children: [
      Center(child: Text("ini konten menu")),
      Center(child: Text("ini konten Calender")),
      Center(child: Text("ini konten history")),
    ]),
  ));
}
}

```

Screenshoot Output



Deskripsi Program

(deskripsikan program apa yang dibuat, memakai algoritma, dan cara kerja program sampai ke output yang dihasilkan dengan bahasa sendiri)

Tab Controller Setup

- DefaultTabController dengan 3 tab
- Implementasi StatelessWidget karena tidak memerlukan state management

Custom Tab Design

- Setiap tab memiliki icon dan text
- Icon yang relevan dengan konten (food_bank, calendar_month, history)
- Custom styling dengan warna pink pada AppBar

Tab Content

- TabBarView dengan 3 different contents
- Centered text untuk setiap konten tab
- Mudah dimodifikasi untuk konten yang lebih kompleks

Navigation Structure

- Intuitive navigation dengan visual indicators
- Consistent styling across tabs
- Responsive layout adjustment

4) Implementasi Bottom Navigation

Soal Studi Case

Multi-Page Navigation dengan Bottom Bar

Sourcecode

```
import 'package:flutter/material.dart';
import 'package:prak6/my_package.dart';
import 'package:prak6/my_tabbar.dart';

class MyNavbar extends StatefulWidget {
  const MyNavbar({super.key});

  @override
  State<MyNavbar> createState() => _MyNavbarState();
}

class _MyNavbarState extends State<MyNavbar> {
  int _selectedIndex = 0;

  void _onItemTapped(int index) {
    setState(() {
      _selectedIndex = index;
    });
  }

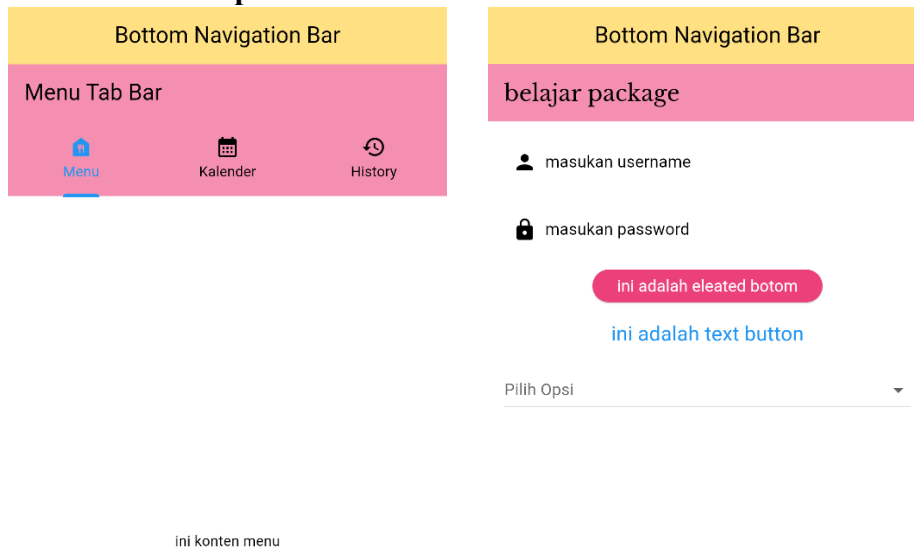
  List<Widget> _pages = <Widget>[
    MyTabbar(),
    MyPackage(),
    MyPackage(),
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Bottom Navigation Bar"),
        centerTitle: true,
        backgroundColor: Colors.amber[200],
      ),
      body: _pages[_selectedIndex],
      bottomNavigationBar: BottomNavigationBar(
        items: <BottomNavigationBarItem>[
          BottomNavigationBarItem(
            icon: Icon(Icons.home_max_rounded), label: "My
Tabbar"),
          BottomNavigationBarItem(
            icon: Icon(Icons.ios_share), label: "My package
1"),
          BottomNavigationBarItem(
            icon: Icon(Icons.local_activity_rounded), label:
"My package 2"),
        ],
        selectedItemColor: Colors.amber,
        currentIndex: _selectedIndex,
        onTap: _onItemTapped,
        selectedFontSize: 20,
      ),
    );
  }
}
```



```
}
```

Screenshoot Output



Deskripsi Program

State Management

- Menggunakan StatefulWidget untuk tracking selected index
- Implementation _onItemTapped untuk handling navigation
- State update menggunakan setState

Page Management

- List of widgets (_pages) untuk different screens
- Integration dengan MyTabbar dan MyPackage components
- Efficient page switching based on selected index

Bottom Navigation Design

- Custom icons untuk setiap navigation item
- Label text untuk clarity
- Selected item highlighting dengan warna amber

UI/UX Features

- Centered AppBar title
- Custom styling (backgroundColor, selectedFontSize)
- Responsive navigation feedback

Component Integration

- Seamless integration antara different pages
- Consistent styling across components
- Efficient state management between pages

Tab Controller Setup

- DefaultTabController dengan 3 tab
- Implementasi StatelessWidget karena tidak memerlukan state management

Custom Tab Design

- Setiap tab memiliki icon dan text
- Icon yang relevan dengan konten (food_bank, calendar_month, history)
- Custom styling dengan warna pink pada AppBar

Tab Content

- TabBarView dengan 3 different contents
- Centered text untuk setiap konten tab
- Mudah dimodifikasi untuk konten yang lebih kompleks

Navigation Structure

- Intuitive navigation dengan visual indicators
- Consistent styling across tabs
- Responsive layout adjustment

B. UNGUIDED

Modifikasi project TP 06 (Tugas Pendahuluan) Interaksi Pengguna, dengan mengimplementasikan penggunaan form dan button.

Sourcecode:

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

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

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Bottom Navigation Bar Demo',
      theme: ThemeData(
        primarySwatch: Colors.pink,
        appBarTheme: const AppBarTheme(
          backgroundColor: Colors.pink,
          foregroundColor: Colors.white,
        ),
      ),
      home: const MyHomePage(),
    );
  }
}

class MyHomePage extends StatefulWidget {
  const MyHomePage({Key? key}) : super(key: key);

  @override
```

```

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

  class _MyHomePageState extends State<MyHomePage> {
    int _selectedIndex = 0;
    final PageController _pageController = PageController();

    static const List<Widget> _pages = [
      HomePage(),
      MailPage(),
      ProfilePage(),
    ];

    void _onItemTapped(int index) {
      setState(() {
        _selectedIndex = index;
      });
      _pageController.animateToPage(
        index,
        duration: const Duration(milliseconds: 300),
        curve: Curves.ease,
      );
    }

    @override
    Widget build(BuildContext context) {
      return Scaffold(
        body: SafeArea(
          child: PageView(
            controller: _pageController,
            physics: const NeverScrollableScrollPhysics(),
            children: _pages,
          ),
        ),
        bottomNavigationBar: BottomNavigationBar(
          backgroundColor: Colors.pink,
          items: const <BottomNavigationBarItem>[
            BottomNavigationBarItem(
              icon: Icon(Icons.home),
              label: 'Home',
            ),
            BottomNavigationBarItem(
              icon: Icon(Icons.email),
              label: 'Email',
            ),
            BottomNavigationBarItem(
              icon: Icon(Icons.person),
              label: 'Profil',
            ),
          ],
          currentIndex: _selectedIndex,
          selectedItemColor: Colors.white,
          onTap: _onItemTapped,
        ),
      );
    }
  }

  class HomePage extends StatelessWidget {

```

```

const HomePage({Key? key}) : super(key: key);

@override
Widget build(BuildContext context) {
  return Column(
    crossAxisAlignment: CrossAxisAlignment.start,
    children: const [
      Padding(
        padding: EdgeInsets.only(top: 16.0, left: 16.0),
        child: Text(
          'Home Page',
          style: TextStyle(fontSize: 24, fontWeight:
FontWeight.bold),
        ),
      ),
      Expanded(
        child: Center(
          child: Text(
            'Ini adalah halaman Home',
            style: TextStyle(fontSize: 30, fontWeight:
FontWeight.bold),
          ),
        ),
      ),
    ],
  );
}

class MailPage extends StatelessWidget {
  const MailPage({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.all(16.0),
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          const Text(
            'Email Page',
            style: TextStyle(fontSize: 30, fontWeight:
FontWeight.bold),
          ),
          const SizedBox(height: 20),
          ElevatedButton(
            onPressed: () {
              ScaffoldMessenger.of(context).showSnackBar(
                const SnackBar(content: Text('Email button pressed!')),
              );
            },
            style: ElevatedButton.styleFrom(
              backgroundColor: Colors.pink,
              padding: const EdgeInsets.symmetric(horizontal: 50,
vertical: 20),
            ),
            child: const Text(
              'Check Emails',
              style: TextStyle(fontSize: 18),
            ),
          ),
        ],
      ),
    );
  }
}

```

```

    ),
  ),
],
),
);
}
}

class ProfilePage extends StatefulWidget {
  const ProfilePage({Key? key}) : super(key: key);

  @override
  State<ProfilePage> createState() => _ProfilePageState();
}

class _ProfilePageState extends State<ProfilePage> {
  final _formKey = GlobalKey<FormState>();
  final TextEditingController _nameController =
    TextEditingController();
  final TextEditingController _emailController =
    TextEditingController();
  final TextEditingController _phoneController =
    TextEditingController();

  @override
  void dispose() {
    _nameController.dispose();
    _emailController.dispose();
    _phoneController.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.all(16.0),
      child: Form(
        key: _formKey,
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.stretch,
          children: [
            const Text(
              'Profile Page',
              style: TextStyle(fontSize: 24, fontWeight:
                FontWeight.bold),
              textAlign: TextAlign.center,
            ),
            const SizedBox(height: 20),
            TextFormField(
              controller: _nameController,
              decoration: const InputDecoration(
                labelText: 'Name',
                border: OutlineInputBorder(),
                prefixIcon: Icon(Icons.person),
              ),
              validator: (value) {
                if (value == null || value.isEmpty) {
                  return 'Please enter your name';
                }
              }
            )
          ]
        )
      )
    );
  }
}

```

```

        return null;
      },
    ),
    const SizedBox(height: 16),
    TextFormField(
      controller: _emailController,
      decoration: const InputDecoration(
        labelText: 'Email',
        border: OutlineInputBorder(),
        prefixIcon: Icon(Icons.email),
      ),
      validator: (value) {
        if (value == null || value.isEmpty) {
          return 'Please enter your email';
        }
        if (!value.contains('@')) {
          return 'Please enter a valid email';
        }
        return null;
      },
    ),
    const SizedBox(height: 16),
    TextFormField(
      controller: _phoneController,
      decoration: const InputDecoration(
        labelText: 'Phone',
        border: OutlineInputBorder(),
        prefixIcon: Icon(Icons.phone),
      ),
      validator: (value) {
        if (value == null || value.isEmpty) {
          return 'Please enter your phone number';
        }
        return null;
      },
    ),
    const SizedBox(height: 24),
    ElevatedButton(
      onPressed: () {
        if (_formKey.currentState!.validate()) {
          // If the form is valid, display a snackbar
          ScaffoldMessenger.of(context).showSnackBar(
            SnackBar(
              content: Text(
                'Profile updated!\nName:
${_nameController.text}\nEmail: ${_emailController.text}\nPhone:
${_phoneController.text}',
              ),
              backgroundColor: Colors.green,
            ),
          );
        }
      },
      style: ElevatedButton.styleFrom(
        backgroundColor: Colors.pink,
        padding: const EdgeInsets.symmetric(vertical: 15),
      ),
      child: const Text(
        'Update Profile',

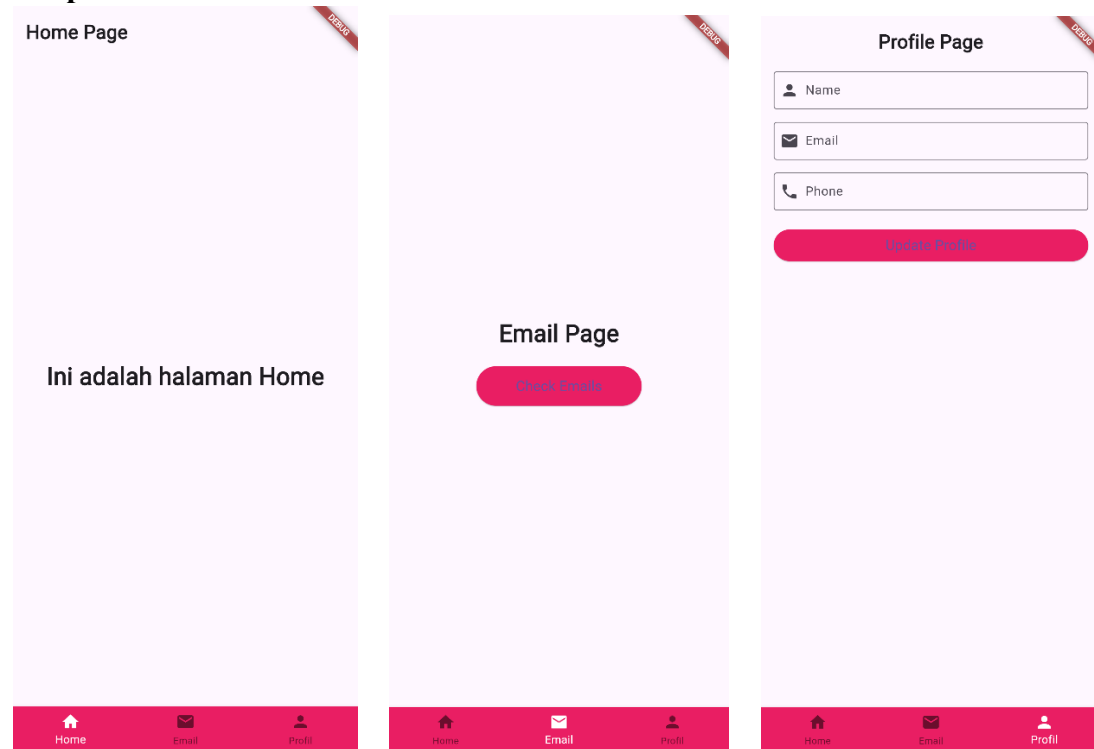
```

```

    style: TextStyle(fontSize: 18),
  ),
),
),
),
),
);
}

```

Output:



Deskripsi Program:

ProfilePage:

- Diubah menjadi StatefulWidget untuk mengelola form state
- Menambahkan form dengan 3 input field:
 - o Name (dengan validasi required)
 - o Email (dengan validasi email format)
 - o Phone (dengan validasi required)
- Menambahkan tombol "Update Profile" yang akan memvalidasi form
- Menampilkan Snackbar ketika form berhasil disubmit

MailPage:

- Ditambahkan button "Check Emails" dengan styling
- Menampilkan Snackbar ketika button ditekan

Styling:

- Menggunakan OutlineInputBorder untuk input field.
- Menambahkan icons pada input fields
- Konsisten menggunakan warna pink sesuai tema
- Spacing yang tepat antara element.

BAB III

KESIMPULAN DAN SARAN

A. KESIMPULAN

Dalam materi ini kita mempelajari beberapa hal. Diantaranya: flutter menyediakan berbagai widget yang memudahkan pembuatan UI interaktif, penggunaan, `statefulWidget` penting untuk mengelola state dalam aplikasi, form validation merupakan aspek penting dalam pengembangan aplikasi, kita juga mengetahui cara implementasi proper feedback ke user meningkatkan user experience.

B. REFERENSI

Flutter Official Documentation (flutter.dev)

Dart Packages (pub.dev)