

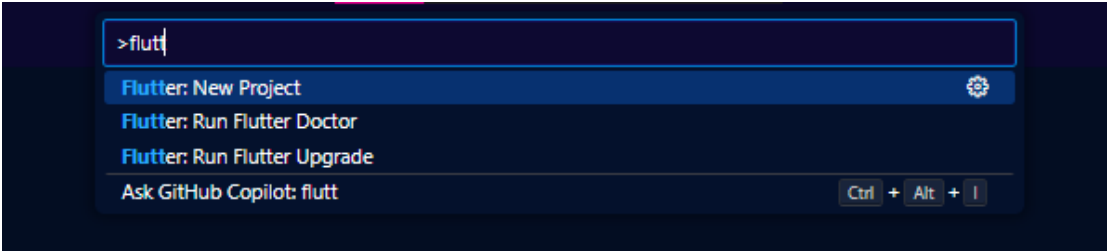
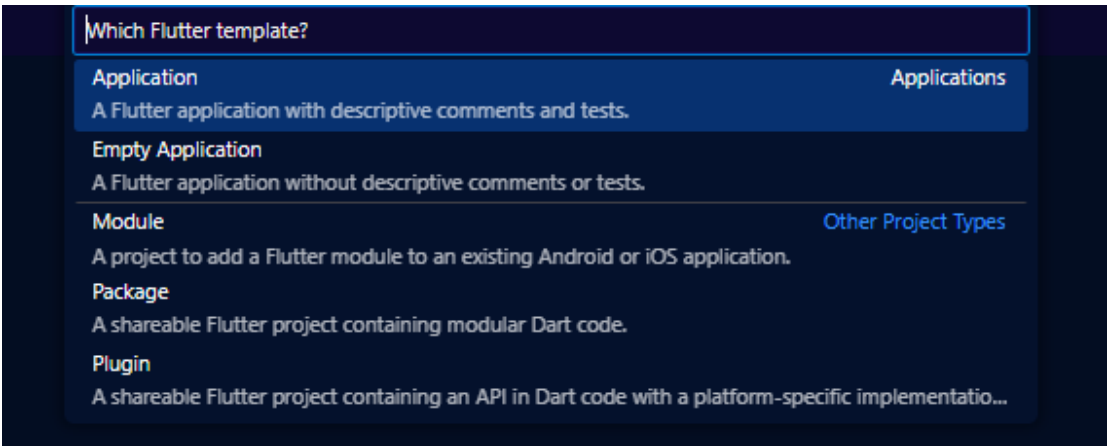


Mata Kuliah : Pemograman Mobile
Program Studi : D4 – Sistem Informasi Bisnis
Semester : 5

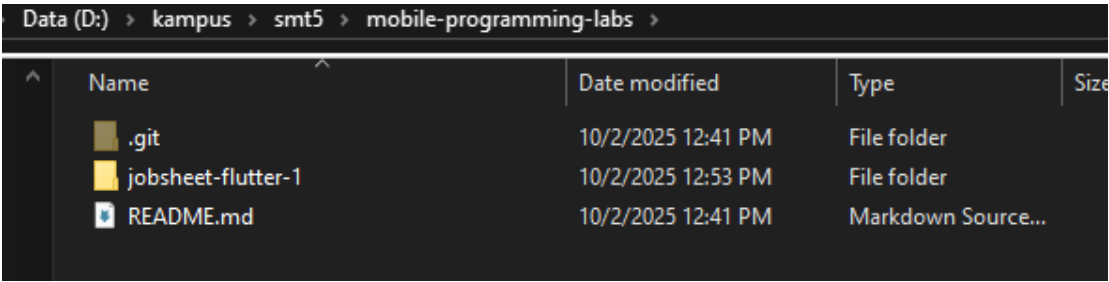
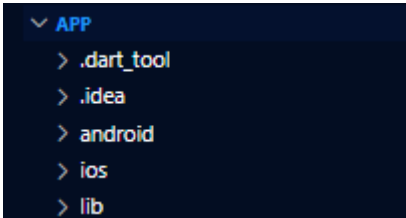
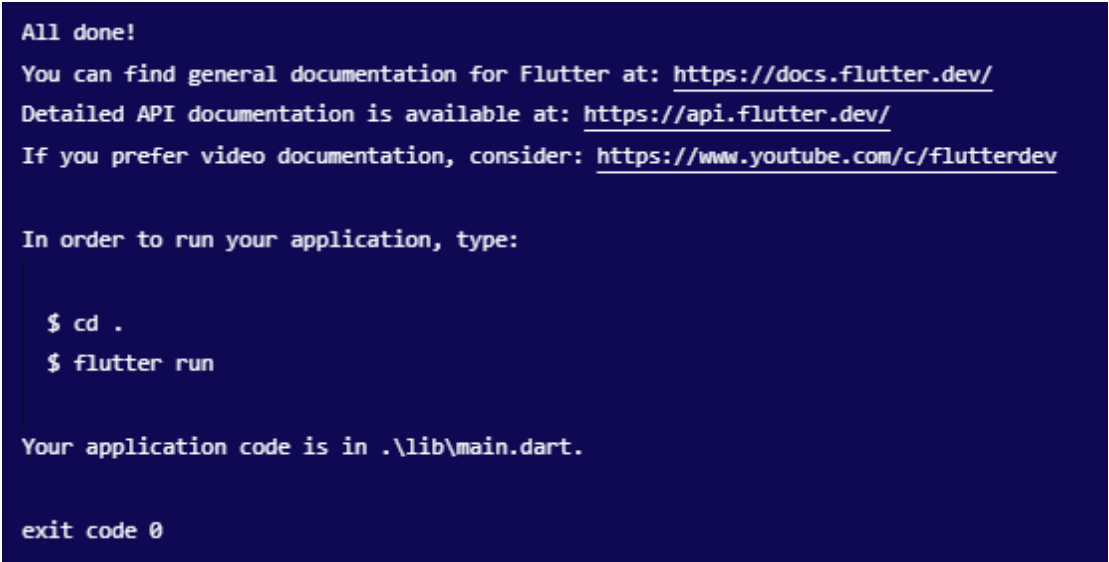
Kelas : SIB
NIM : 2341760056
Nama : Revani Nanda Putri
Jobsheet Ke- : 4

Laporan Jobsheet Flutter 1

Praktikum 1: Membuat Project Flutter Baru

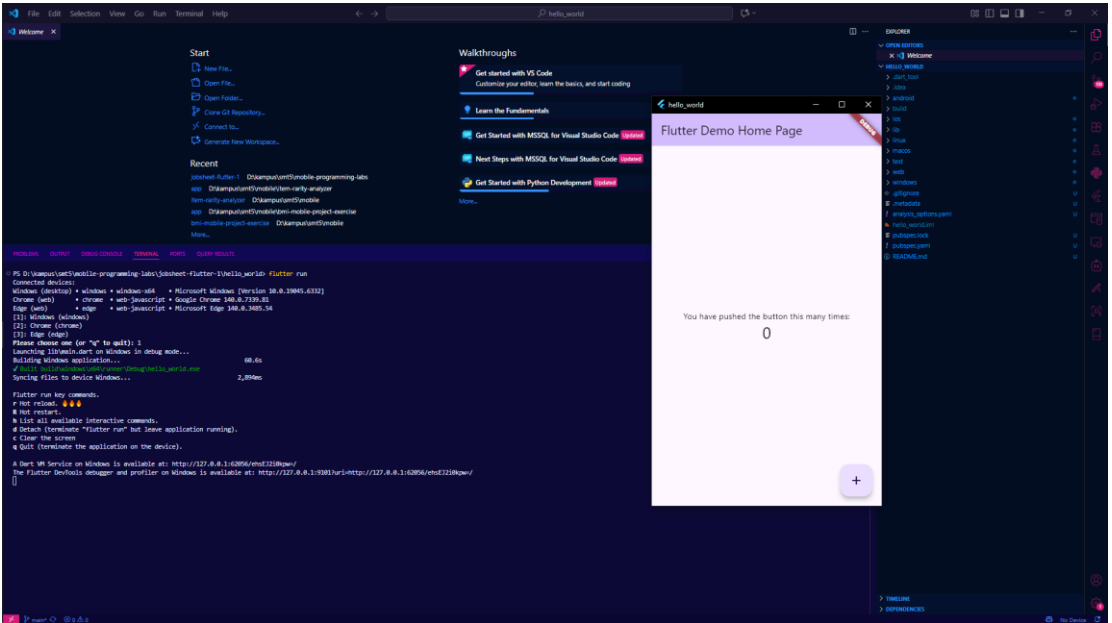
Langkah	Jawaban/Deskripsi
1	<p>Buka VS Code, lalu tekan tombol Ctrl + Shift + P maka akan tampil Command Palette, lalu ketik Flutter. Pilih New Application Project.</p>  
2	<p>Kemudian buat folder sesuai style laporan praktikum yang Anda pilih. Disarankan pada folder dokumen atau desktop atau alamat folder lain yang tidak terlalu dalam atau panjang. Lalu pilih Select a folder to create the project in.</p>



	
3	<p>Buat nama project flutter app seperti berikut, lalu tekan Enter. Tunggu hingga proses pembuatan project baru selesai.</p> 
4	<p>Jika sudah selesai proses pembuatan project baru, pastikan tampilan seperti berikut. Pesan akan tampil berupa "Your Flutter Project is ready!" artinya Anda telah berhasil membuat project Flutter baru.</p> 

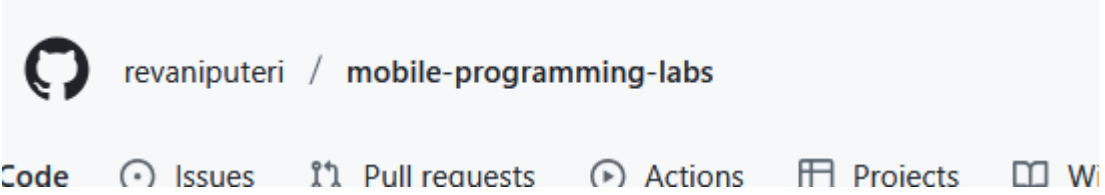

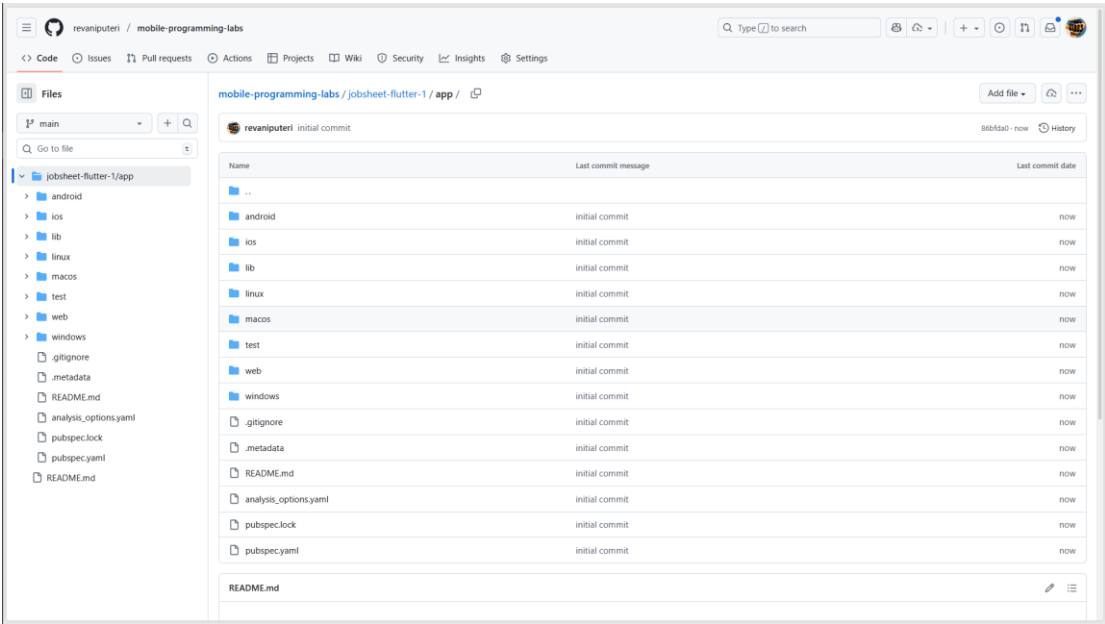


Praktikum 2: Menghubungkan Perangkat Android atau Emulator

Langkah	Jawaban/Deskripsi
1	<p>Melanjutkan dari praktikum 1, Anda diminta untuk menjalankan aplikasi ke perangkat fisik (device Android atau iOS). Silakan ikuti langkah-langkah pada codelab tautan berikut ini.</p> <p>https://developer.android.com/codelabs/basic-android-kotlin-compose-connect-device?hl=id#0</p>
2	<p>Hasil dengan flutter run di terminal</p>  <p>The screenshot shows the Visual Studio Code interface. The terminal window at the bottom displays the command <code>flutter run</code> and its output, including the Flutter version (3.0.0) and the device used (Windows). The output shows the application is running successfully. The main editor window displays the 'Flutter Demo Home Page' with a counter that says 'You have pushed the button this many times: 0'.</p>



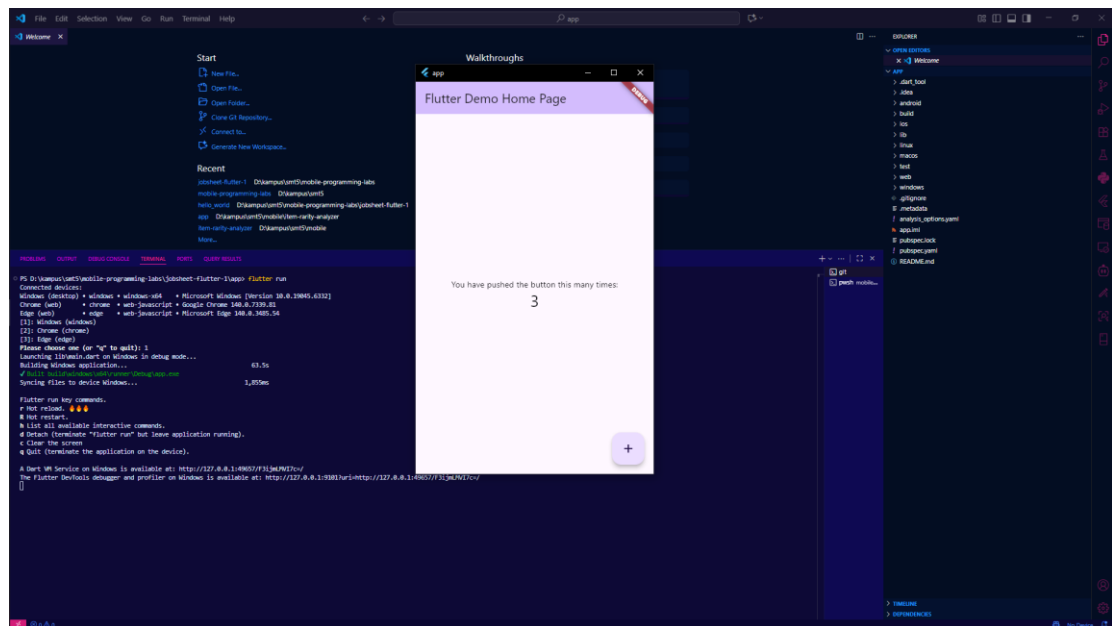
Praktikum 3: Membuat Repository GitHub dan Laporan Praktikum

Langkah	Jawaban/Deskripsi
1	Login ke akun github Anda, lalu buat repository baru dengan nama “mobile-programming-labs”
2	Lalu klik tombol "Create repository" lalu akan tampil seperti gambar berikut. 
3	Kembali ke VS code, project flutter app, buka terminal pada menu Terminal > New Terminal . Lalu ketik perintah berikut untuk inisialisasi git pada project Anda. 
4	Pilih menu Source Control di bagian kiri, lalu lakukan stages (+) pada file .gitignore untuk mengunggah file pertama ke repository GitHub. Beri pesan commit " tambah gitignore " lalu klik Commit (✓) . Lakukan push dengan klik bagian menu titik tiga > Push 



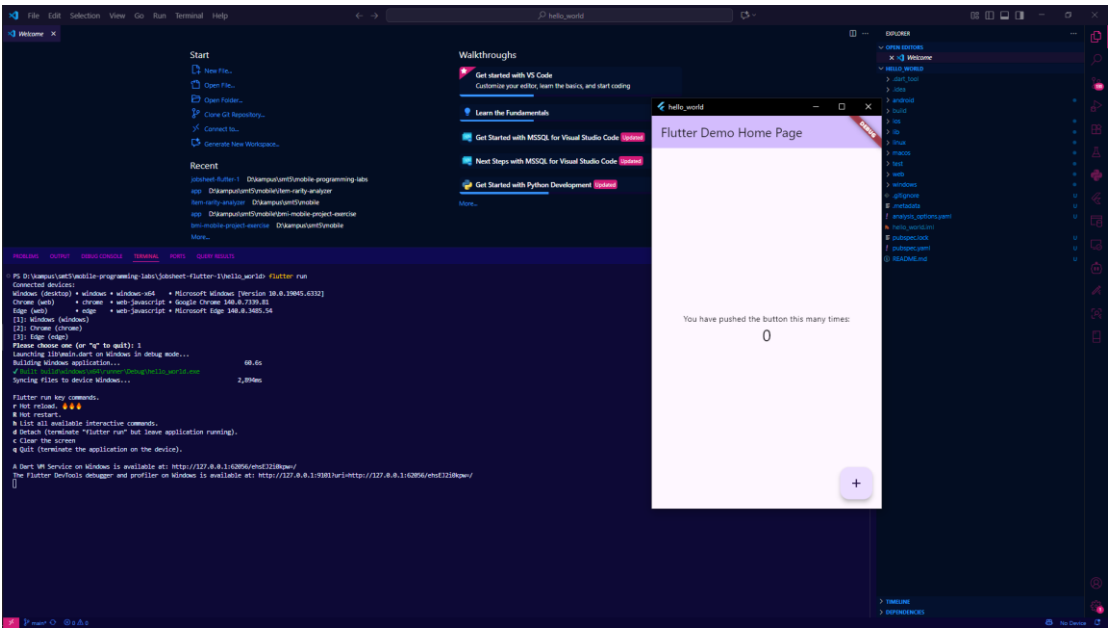
5

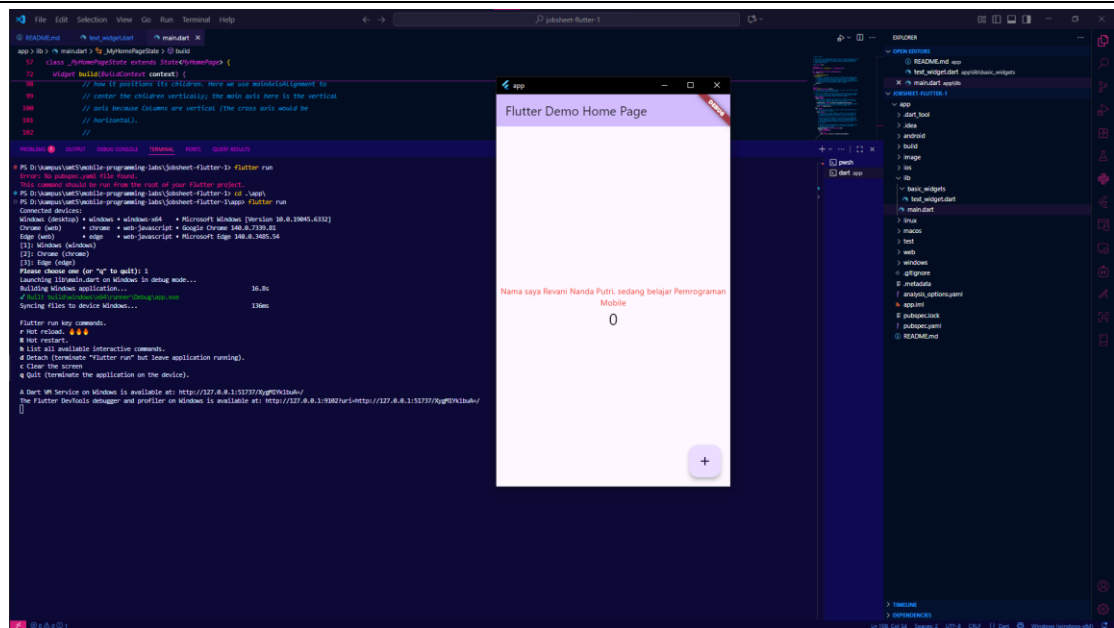
Kembali ke VS Code, ubah platform di pojok kanan bawah ke emulator atau device atau bisa juga menggunakan browser Chrome. Lalu coba running project **hello_world** dengan tekan **F5** atau **Run > Start Debugging**. Tunggu proses kompilasi hingga selesai, maka aplikasi flutter pertama Anda akan tampil seperti berikut.





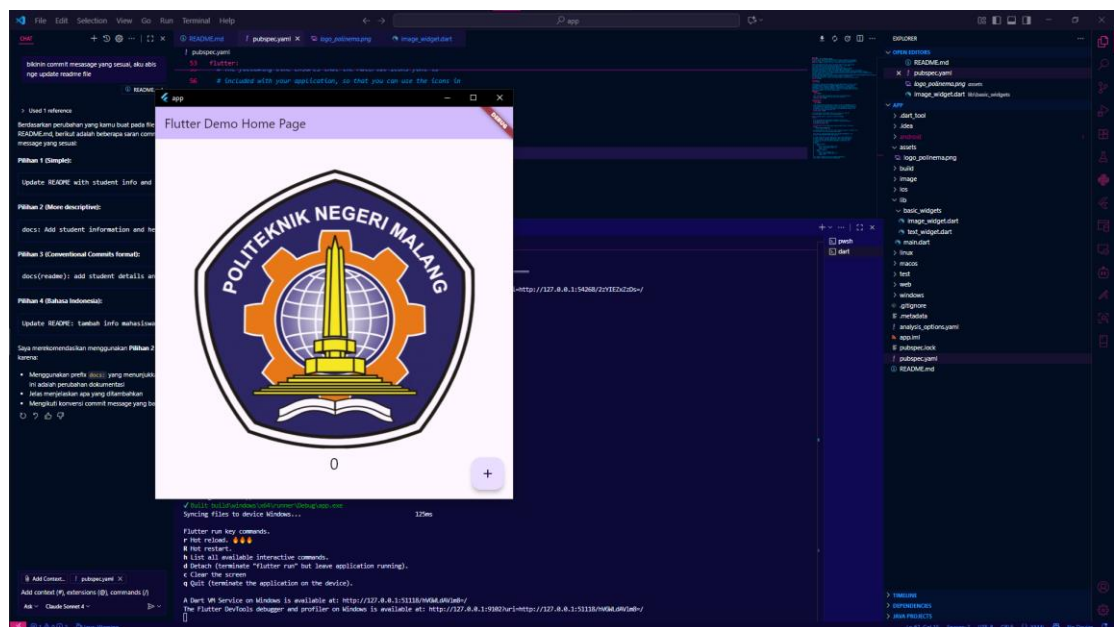
Praktikum 4: Menerapkan Widget Dasar

Langkah	Jawaban/Deskripsi
1	Buat folder baru basic_widgets di dalam folder lib . Kemudian buat file baru di dalam basic_widgets dengan nama text_widget.dart . Ketik atau salin kode program berikut ke project hello_world Anda pada file text_widget.dart .
2	Hasil dengan flutter run di terminal 
3	Lakukan import file text_widget.dart ke main.dart , lalu ganti bagian text widget dengan kode di atas. Maka hasilnya seperti gambar berikut. Screenshot hasil milik Anda, lalu dibuat laporan pada file README.md .




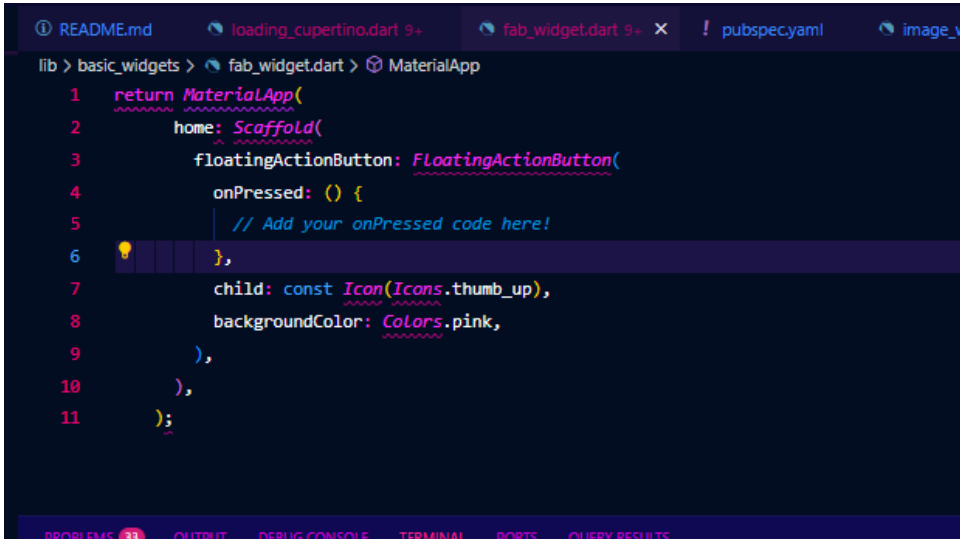
4

Buat sebuah file `image_widget.dart` di dalam folder `basic_widgets` dengan isi kode berikut. Lakukan penyesuaian asset pada file `pubspec.yaml` dan tambahkan file logo Anda di folder `assets` project `hello_world`. Jangan lupa sesuaikan kode dan import di file `main.dart` kemudian akan tampil gambar seperti berikut.





Praktikum 5: Menerapkan Widget Material Design dan iOS Cupertino

Langkah	Jawaban/Deskripsi
1	<p>Buat file di basic_widgets > loading_cupertino.dart. Import stateless widget dari material dan cupertino. Lalu isi kode di dalam method Widget build adalah sebagai berikut.</p>  <pre>lib > basic_widgets > loading_cupertino.dart > MaterialApp 1 return MaterialApp(2 home: Container(3 margin: const EdgeInsets.only(top: 30), 4 color: Colors.white, 5 child: Column(6 children: <Widget>[7 CupertinoButton(8 child: const Text("Contoh button"), 9 onPressed: () {}, 10), 11 const CupertinoActivityIndicator(), 12], // <Widget>[] 13), 14), 15);</pre>
2	<p>Button widget terdapat beberapa macam pada flutter yaitu FlatButton, DropdownButton, TextButton, FloatingActionButton, IconButton, OutlineButton, PopupMenuButton, dan ElevatedButton.</p> <p>Buat file di basic_widgets > fab_widget.dart. Import stateless widget dari material. Lalu isi kode di dalam method Widget build adalah sebagai berikut.</p>  <pre>lib > basic_widgets > fab_widget.dart > MaterialApp 1 return MaterialApp(2 home: Scaffold(3 floatingActionButton: FloatingActionButton(4 onPressed: () { 5 // Add your onPressed code here! 6 }, 7 child: const Icon(Icons.thumb_up), 8 backgroundColor: Colors.pink, 9), 10), 11);</pre>



```
import 'package:app/basic_widgets/fab_widget.dart';
import 'package:app/basic_widgets/loading_cupertino.dart';
import 'package:app/basic_widgets/image_widget.dart';
import 'package:flutter/material.dart';

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Flutter Demo',
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
      ),
      home: const FabWidget(),
    );
  }
}

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

  final String title;

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

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

  void _incrementCounter() {
    setState(() {
      _counter++;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        backgroundColor: Theme.of(context).colorScheme.inversePrimary,
        title: Text(widget.title),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            const MyImageWidget(),
            Text(
              '$_counter',
              style: Theme.of(context).textTheme.headlineMedium,
            ),
          ],
        ),
      ),
    );
  }
}
```



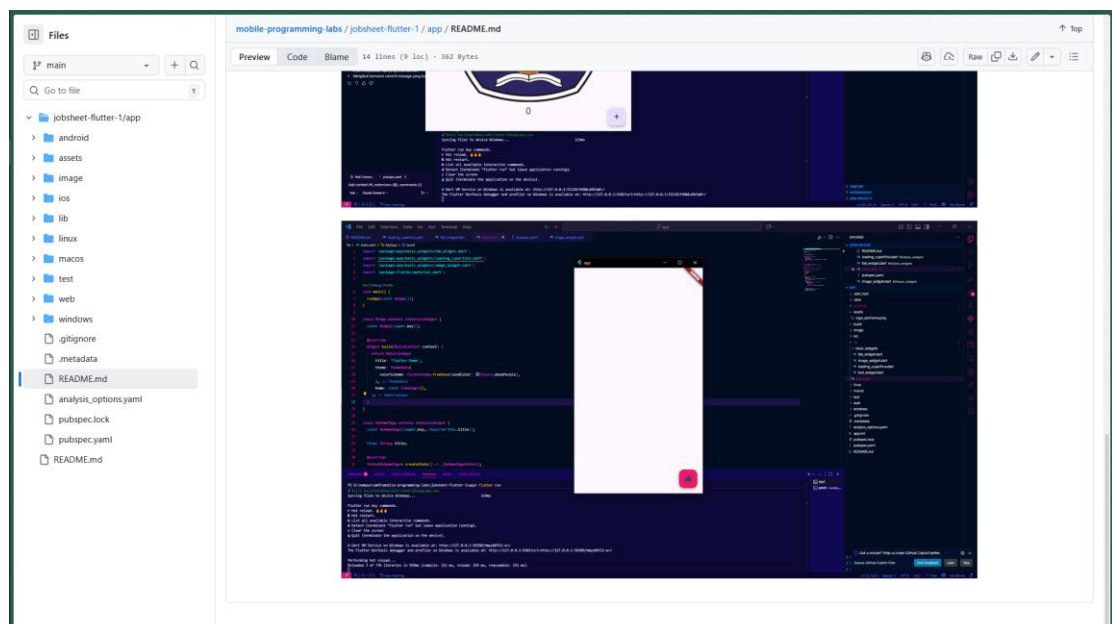
	<pre>floatingActionButton: <i>Flutter</i>FloatingActionButton(onPressed: _incrementCounter, tooltip: 'Increment', child: <i>const</i> <i>Icon</i>(<i>Icons</i>.add),),); }</pre>
3	<p>Scaffold widget digunakan untuk mengatur tata letak sesuai dengan material design.</p> <p>Ubah isi kode main.dart seperti berikut.</p> <pre>import 'package:flutter/material.dart'; void main() { runApp(<i>const</i> MyApp()); } class MyApp extends StatelessWidget { <i>const</i> MyApp({<i>Key?</i> key}) : <i>super</i>(key: key); // This widget is the root of your application. @override Widget build(BuildContext context) { return MaterialApp(title: 'Flutter Demo', theme: ThemeData(primarySwatch: <i>Colors</i>.red,), home: <i>const</i> MyHomePage(title: 'My Increment App'),); } } class MyHomePage extends StatefulWidget { <i>const</i> MyHomePage({<i>Key?</i> key, <i>required</i> this.title}) : <i>super</i>(key: key); final <i>String</i> title; @override <i>State</i><MyHomePage> createState() => _MyHomePageState(); } class _MyHomePageState extends <i>State</i><MyHomePage> { <i>int</i> _counter = 0; void _incrementCounter() { <i>setState</i>() { _counter++; }; } @override Widget build(BuildContext context) { return Scaffold(appBar: <i>AppBar</i>(title: <i>Text</i>(<i>widget</i>.title),),); } }</pre>



```
body: Center(  
  child: Column(  
    mainAxisAlignment: MainAxisAlignment.center,  
    children: <Widget>[  
      const Text(  
        'You have pushed the button this many times:',  
      ),  
      Text(  
        '$_counter',  
        style: Theme.of(context).textTheme.headlineMedium,  
      ),  
    ],  
  ),  
,  
bottomNavigationBar: BottomAppBar(  
  child: Container(  
    height: 50.0,  
  ),  
,  
floatingActionButton: FloatingActionButton(  
  onPressed: _incrementCounter,  
  tooltip: 'Increment Counter',  
  child: const Icon(Icons.add),  
,  
floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,  
)  
);  
}
```

4

Melakukan push dan commit, serta memasukkan screenshot hasil ke file ReadMe.md





5

Dialog widget pada flutter memiliki dua jenis dialog yaitu AlertDialog dan SimpleDialog.

Ubah isi kode main.dart seperti berikut.

```
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 const MaterialApp(
      home: Scaffold(
        body: MyLayout(),
      ),
    );
  }
}

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

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.all(8.0),
      child: ElevatedButton(
        child: const Text('Show alert'),
        onPressed: () {
          showAlertDialog(context);
        },
      ),
    );
  }
}

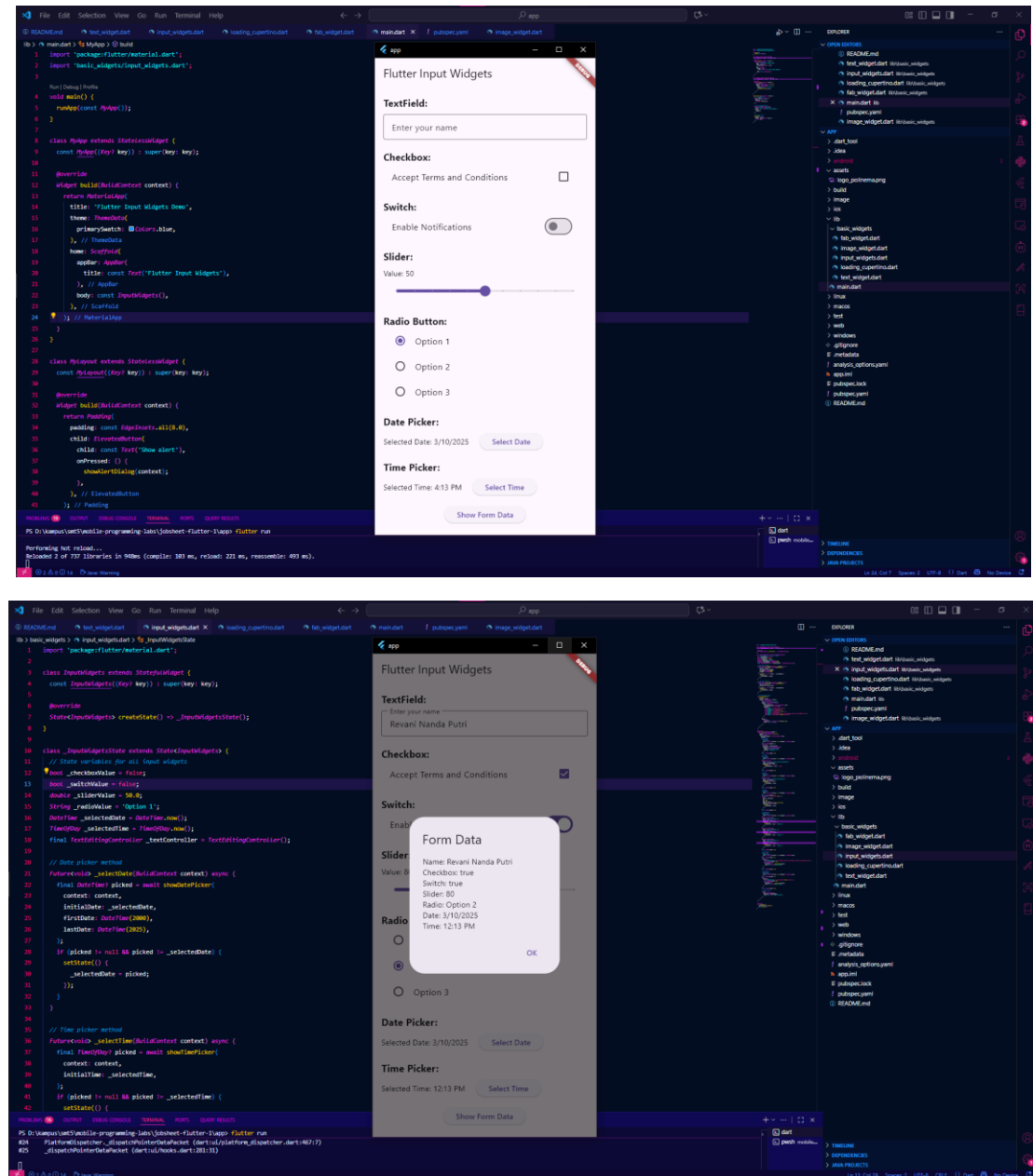
showAlertDialog(BuildContext context) {
  // set up the button
  Widget okButton = TextButton(
    child: const Text("OK"),
    onPressed: () {
      Navigator.pop(context);
    },
  );

  // set up the AlertDialog
  AlertDialog alert = AlertDialog(
    title: const Text("My title"),
    content: const Text("This is my message."),
    actions: [
      okButton,
    ],
  );

  // show the dialog
```

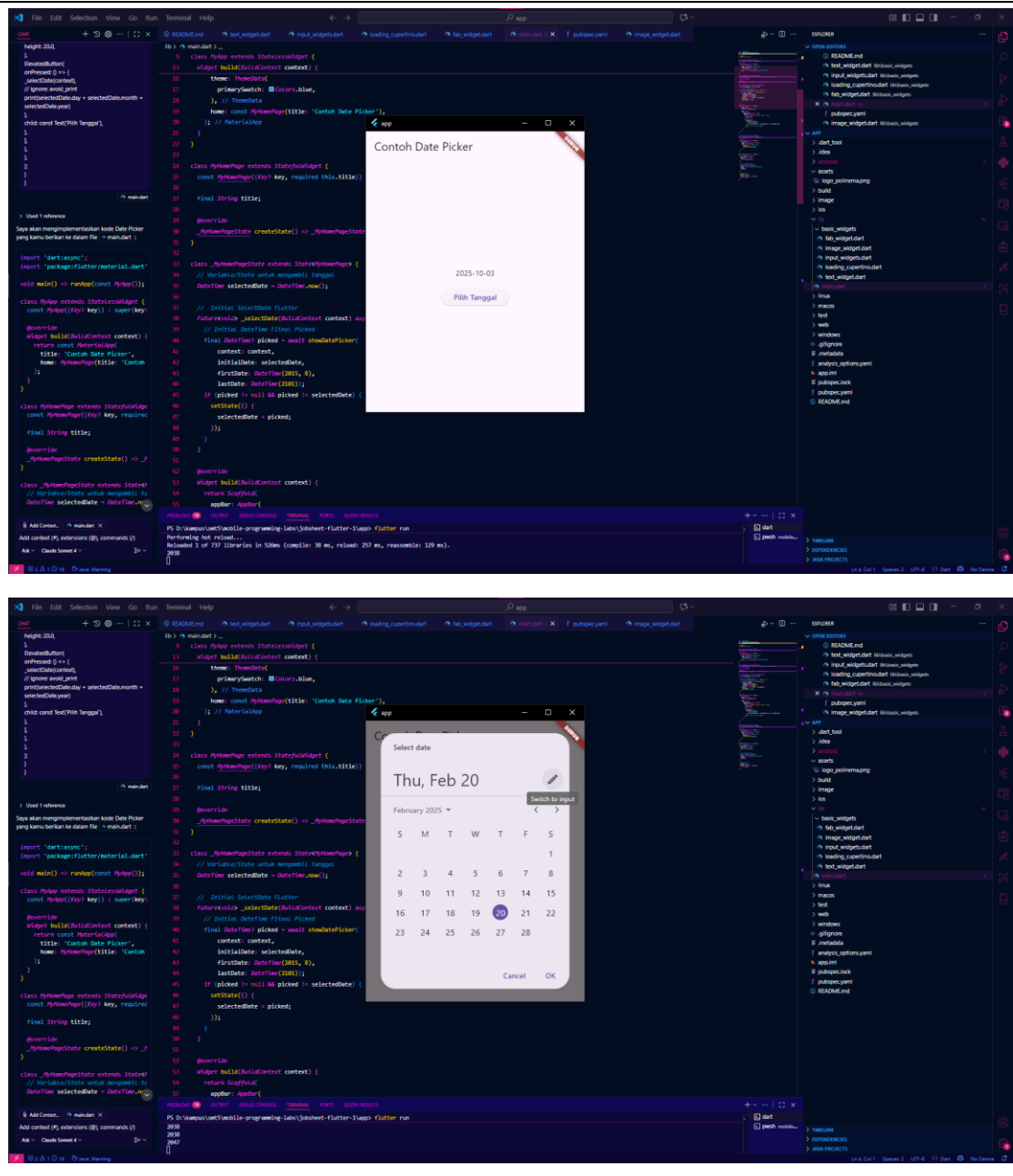


	<pre>showDialog(context: context, builder: (BuildContext context) { return alert; },); }</pre>
6	<p>Flutter menyediakan widget yang dapat menerima input dari pengguna aplikasi yaitu antara lain Checkbox, Date and Time Pickers, Radio Button, Slider, Switch, TextField.</p> <pre>lib > main.dart > main 1 import 'package:flutter/material.dart'; 2 import 'basic_widgets/input_widgets.dart'; 3 4 void main() { 5 runApp(const MyApp()); 6 } 7 8 class MyApp extends StatelessWidget { 9 const MyApp({Key? key}) : super(key: key); 10 11 @override 12 Widget build(BuildContext context) { 13 return MaterialApp(14 title: 'Flutter Input Widgets Demo', 15 theme: ThemeData(16 primarySwatch: Colors.blue, 17), // ThemeData 18 home: Scaffold(19 appBar: AppBar(20 title: const Text('Flutter Input Widgets'), 21), // AppBar 22 body: const InputWidgets(), 23), // Scaffold 24); // MaterialApp 25 } 26 }</pre> <pre>lib > basic_widgets > input_widgets.dart > _InputWidgetsState 1 import 'package:flutter/material.dart'; 2 3 class InputWidgets extends StatefulWidget { 4 const InputWidgets({Key? key}) : super(key: key); 5 6 @override 7 State<InputWidgets> createState() => _InputWidgetsState(); 8 } 9 10 class _InputWidgetsState extends State<InputWidgets> { 11 // State variables for all input widgets 12 bool _checkboxValue = false; 13 bool _switchValue = false; 14 double _sliderValue = 50.0; 15 String _radioValue = 'Option 1'; 16 DateTime _selectedDate = DateTime.now(); 17 TimeOfDay _selectedTime = TimeOfDay.now(); 18 final TextEditingController _textController = TextEditingController(); 19 }</pre>



7

Date and Time Pickers termasuk pada kategori input dan selection widget, berikut adalah contoh penggunaan Date and Time Pickers.





Praktikum 2: Menghubungkan Perangkat Android atau Emulator

Langkah	Jawaban/Deskripsi
1	
2	



KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI
POLITEKNIK NEGERI MALANG
JURUSAN TEKNOLOGI INFORMASI
Jl. Soekarno Hatta No. 9, Jatimulyo, Lowokwaru, Malang 65141
Telp. (0341) 404424 – 404425, Fax (0341) 404420
<https://jti.polinema.ac.id>
