

**LAPORAN PRAKTIKUM**  
**MATA KULIAH PEMOGRAMAN MOBILE**

**JOBSHEET 5:**  
**Aplikasi Pertama dan Widget Dasar Flutter**



**KAYLA RACHMAUDINA SATITI PUTRI**

**2341760103**

**D-IV SISTEM INFORMASI BISNIS**

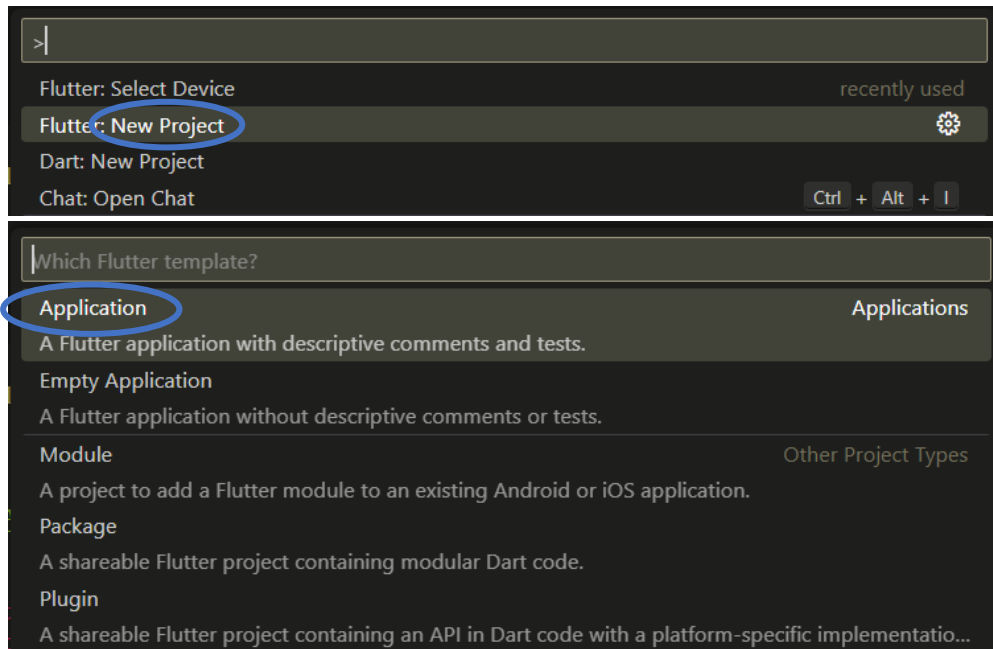
**JURUSAN TEKNOLOGI INFORMASI POLITEKNIK**  
**NEGERI MALANG**

**2025**

## Praktikum 1: Membuat Project Flutter Baru

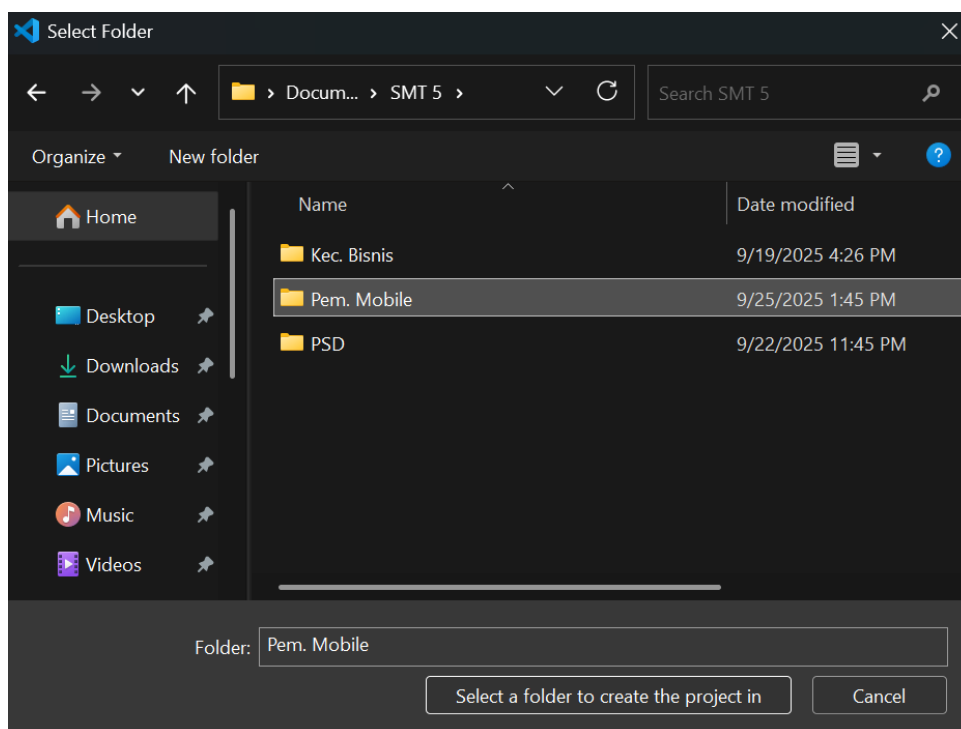
### Langkah 1:

Buka VS Code, lalu tekan tombol **Ctrl + Shift + P** maka akan tampil *Command Palette*, lalu ketik **Flutter**. Pilih **New Application Project**.



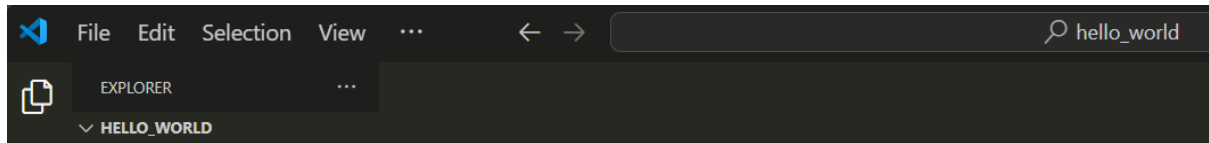
### Langkah 2:

Kemudian buat folder **sesuai style** laporan praktikum yang Anda pilih. Lalu pilih **Select a folder to create the project in**.



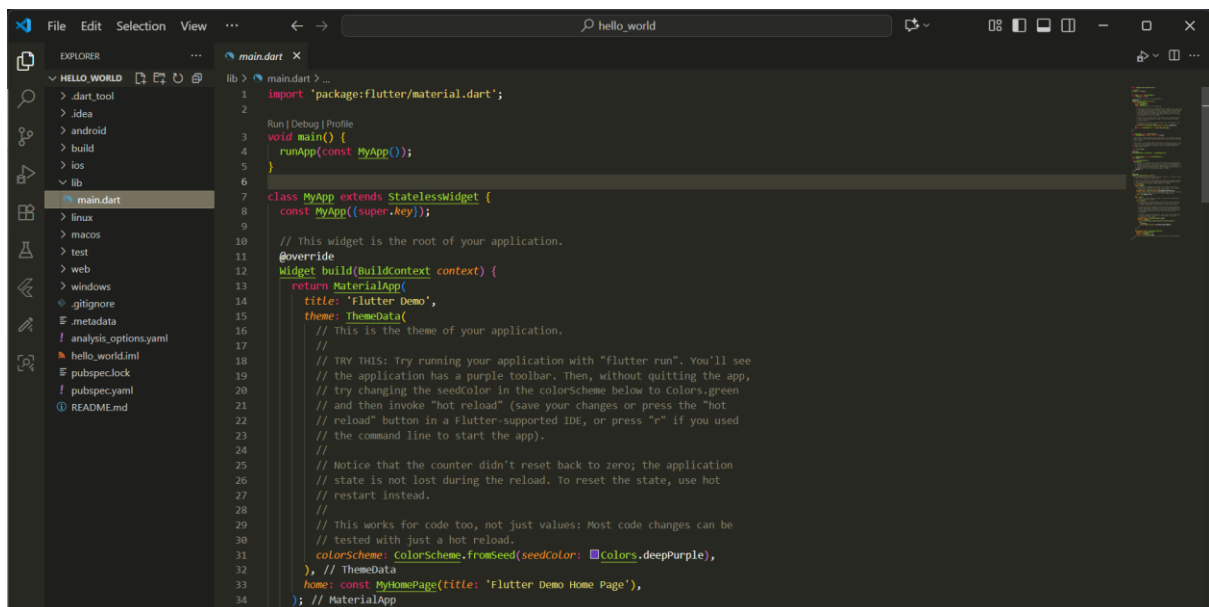
### Langkah 3:

Buat nama project flutter **hello\_world** seperti berikut, lalu tekan **Enter**. Tunggu hingga proses pembuatan project baru selesai.



### Langkah 4:

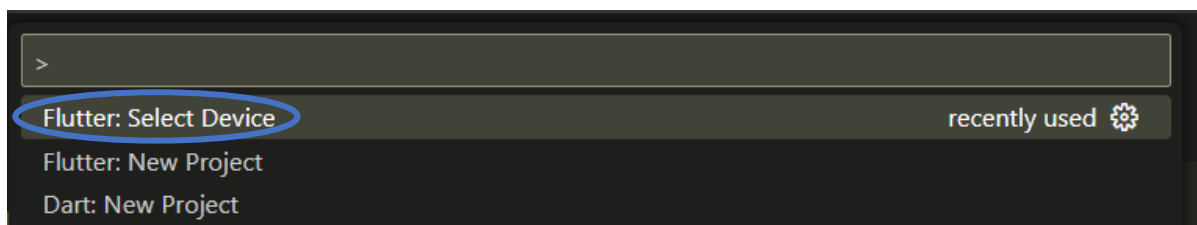
Jika sudah selesai proses pembuatan project baru, pastikan tampilan seperti berikut. Pesan akan tampil berupa **"Your Flutter Project is ready!"** artinya Anda telah berhasil membuat project Flutter baru.



## Praktikum 2: Menghubungkan Perangkat Android atau Emulator

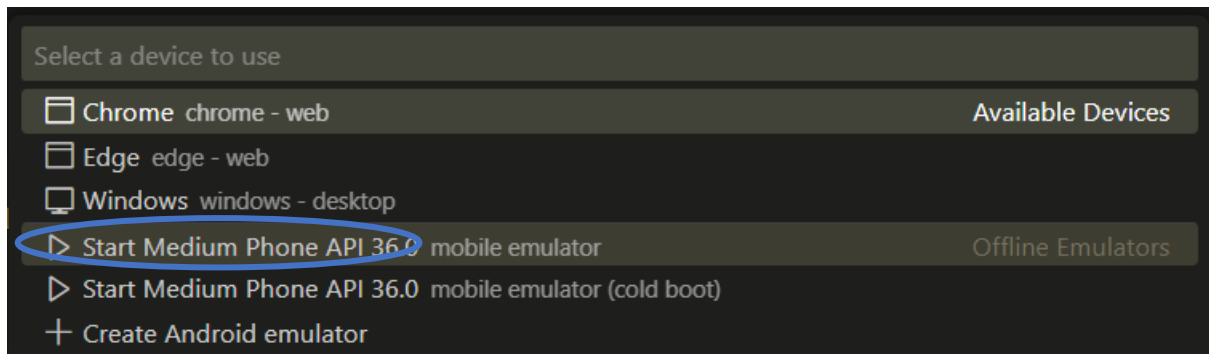
### Langkah 1:

Ketik "Flutter: Select Device"



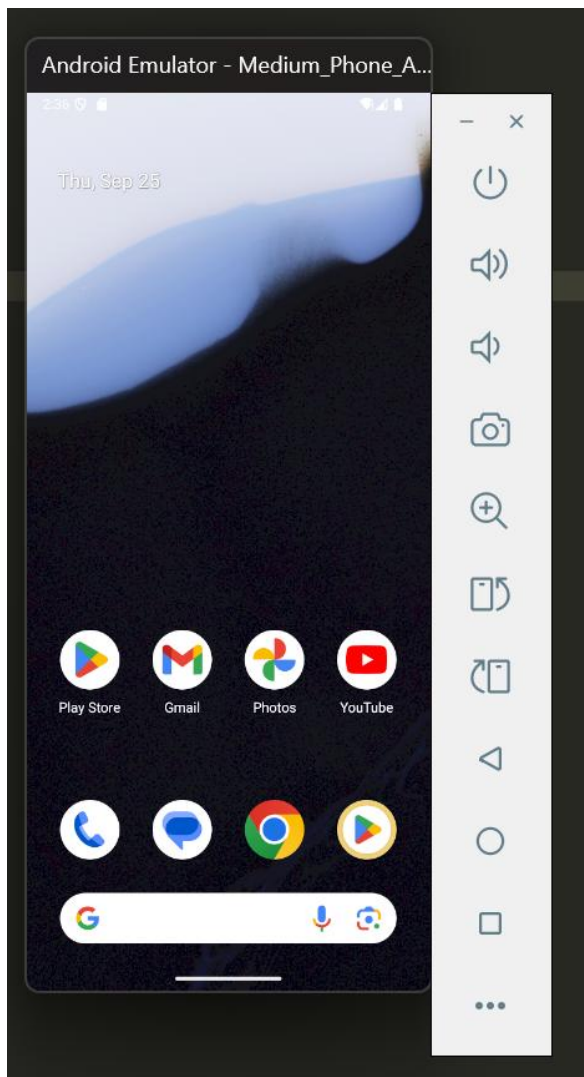
### Langkah 2:

Pilih emulator, disini saya memilih medium phone



### Langkah 3:

Berikut tampilan emulatoanya:



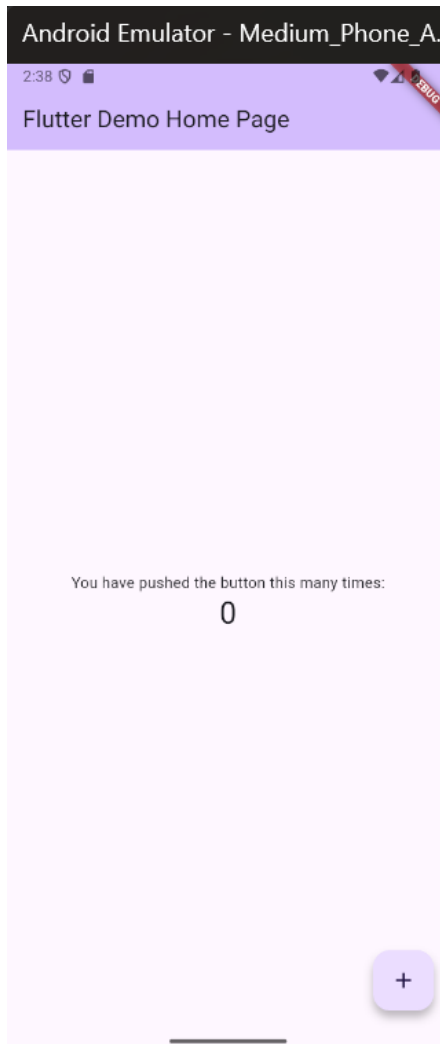
### Langkah 4:

Menjalankan flutter run

```
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\New folder\hello_world> flutter run
Launching lib\main.dart on sdk gphone64 x86 64 in debug mode...
Running Gradle task 'assembleDebug'... 2,560ms
✓ Built build\app\outputs\flutter-apk\app-debug.apk
Installing build\app\outputs\flutter-apk\app-debug.apk... 908ms
```

### Langkah 5:

Berikut tampilan ketika flutter sudah berhasil dijalankan



## Praktikum 3: Membuat Repository GitHub dan Laporan Praktikum


### Langkah 1:

Login ke akun [GitHub](https://github.com) Anda, lalu buat repository baru dengan nama "**flutter-fundamental-part1**"

1

General

Owner \*

 Kaylarsp

Repository name \*

flutter-fundamental-part1

✓ flutter-fundamental-part1 is available.

Great repository names are short and memorable. How about **psychic-octo-parakeet**?

Description

0 / 350 characters

2

Configuration

Choose visibility \*

Choose who can see and commit to this repository

Public

Add README

READMEs can be used as longer descriptions. [About READMEs](#)

Off ☐

Add .gitignore

.gitignore tells git which files not to track. [About ignoring files](#)

No .gitignore

Add license


Licenses explain how others can use your code. [About licenses](#)

No license

Create repository

## Langkah 2:

Lalu klik tombol "**Create repository**" lalu akan tampil seperti gambar berikut.

 flutter-fundamental-part1


Public

Pin


Watch 0

Fork 0

Star 0



**Start coding with Codespaces**  
Add a README file and start coding in a secure, configurable, and dedicated development environment.  
[Create a codespace](#)



**Add collaborators to this repository**  
Search for people using their GitHub username or email address.  
[Invite collaborators](#)

**Quick setup — if you've done this kind of thing before**

Set up in Desktop

 or 

HTTPS

SSH

https://github.com/kaylarsp/flutter-fundamental-part1.git

Get started by [creating a new file](#) or [uploading an existing file](#). We recommend every repository include a [README](#), [LICENSE](#), and [.gitignore](#).

**...or create a new repository on the command line**

```
echo "# flutter-fundamental-part1" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
```

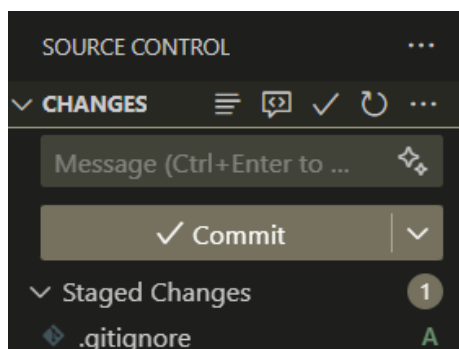
### Langkah 3:

Kembali ke VS code, project flutter hello\_world, buka terminal pada menu **Terminal > New Terminal**. Lalu ketik perintah berikut untuk inisialisasi git pada project Anda.

```
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git init
hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in C:/Users/ADVAN/OneDrive/Documents/SMT 5/Pem. Mobile/hello_world/.git/
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> |
```

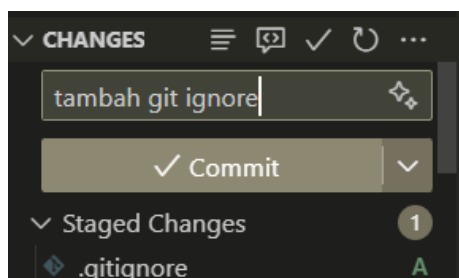
### Langkah 4:

Pilih menu **Source Control** di bagian kiri, lalu lakukan **stages (+)** pada file **.gitignore** untuk mengunggah file pertama ke repository GitHub.



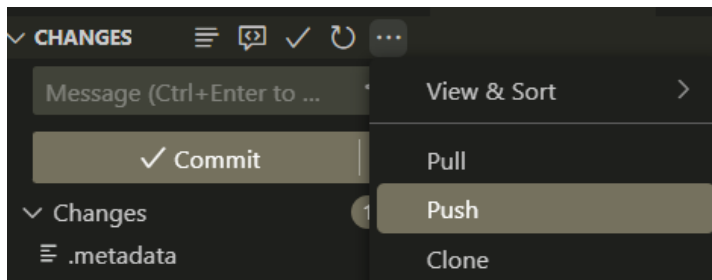
### Langkah 5:

Beri pesan commit "**tambah git ignore**" lalu klik **Commit (✓)**



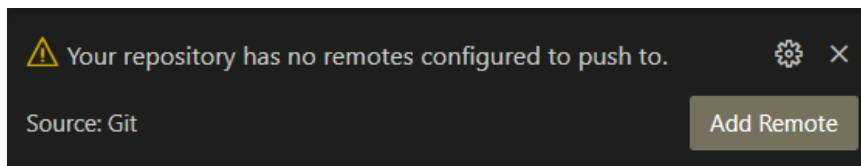
### Langkah 6:

Lakukan push dengan klik bagian menu titik tiga > **Push**



### Langkah 7:

Di pojok kanan bawah akan tampil seperti gambar berikut. Klik "**Add Remote**"



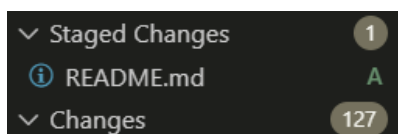
### Langkah 8:

Disini saya menggunakan terminal vs code untuk membuat remote

```
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello world> git remote add origin https://github.com/Kaylarsp/flutter-fundamental-part1.git
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 628 bytes | 628.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Kaylarsp/flutter-fundamental-part1.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

### Langkah 9:

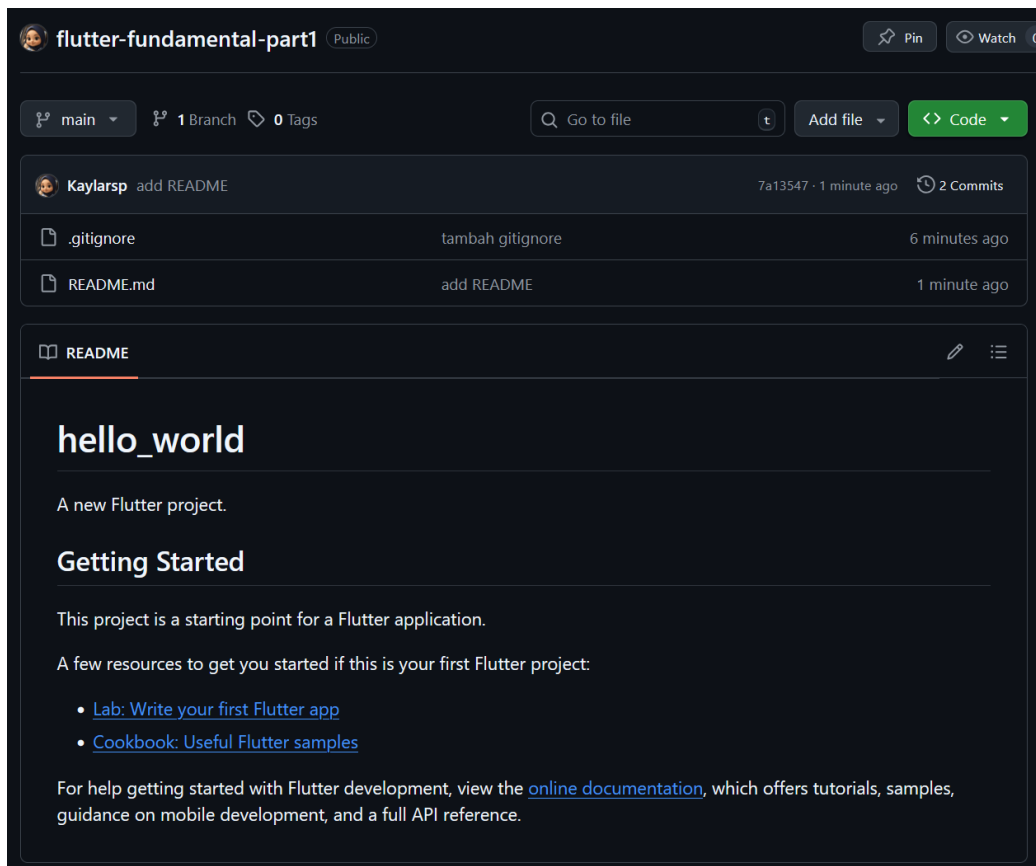
Lakukan hal yang sama pada file **README.md**



```
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git commit -m "add README"
[main 7a13547] add README
1 file changed, 16 insertions(+)
create mode 100644 README.md
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git push
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (3/3), 590 bytes | 118.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Kaylarsp/flutter-fundamental-part1.git
2ac6225..7a13547 main -> main
```



Berikut tampilannya di github



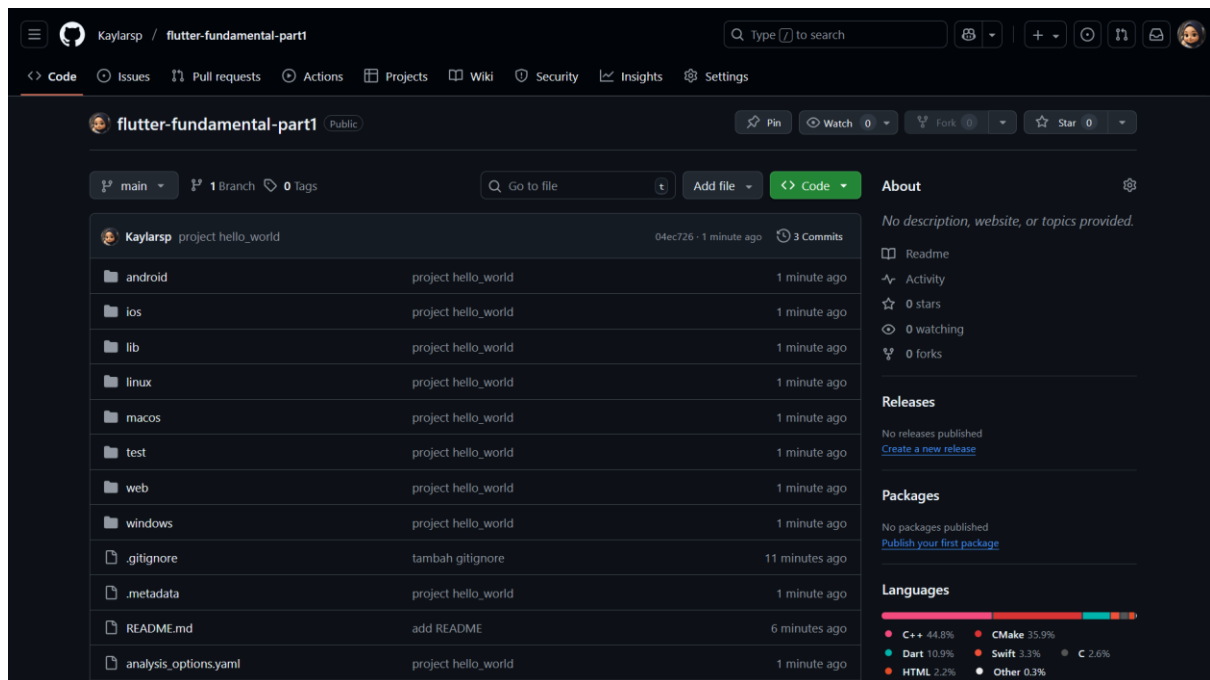
## Langkah 10:

Lakukan push juga untuk semua file lainnya dengan pilih **Stage All Changes**. Beri pesan commit "project hello\_world".

```
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git add .
warning: in the working copy of '.metadata', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'linux/flutter/generated_plugin_registrant.cc', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'linux/flutter/generated_plugin_registrant.h', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'linux/flutter/generated_plugins.cmake', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'macos/Flutter/GeneratedPluginRegistrant.swift', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'pubspec.lock', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'windows/flutter/generated_plugin_registrant.cc', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'windows/flutter/generated_plugin_registrant.h', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'windows/flutter/generated_plugins.cmake', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git commit -m "project hello_world"
[main 04ec726] project hello_world
127 files changed, 4888 insertions(+)
create mode 100644 .metadata
create mode 100644 analysis_options.yaml
create mode 100644 android/.gitignore
create mode 100644 android/app/build.gradle.kts
create mode 100644 android/app/src/debug/AndroidManifest.xml
```

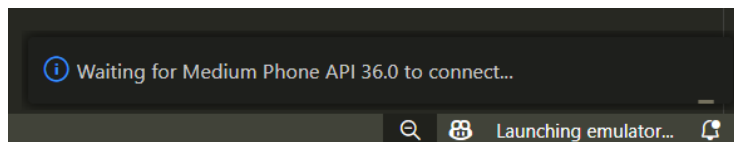
```
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git push
Enumerating objects: 178, done.
Counting objects: 100% (178/178), done.
Delta compression using up to 12 threads
Compressing objects: 100% (146/146), done.
Writing objects: 100% (177/177), 266.88 KiB | 4.30 MiB/s, done.
Total 177 (delta 19), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (19/19), done.
To https://github.com/Kaylarsp/flutter-fundamental-part1.git
7a13547..04ec726 main -> main
```

Berikut tampilannya di repository github



### Langkah 11:

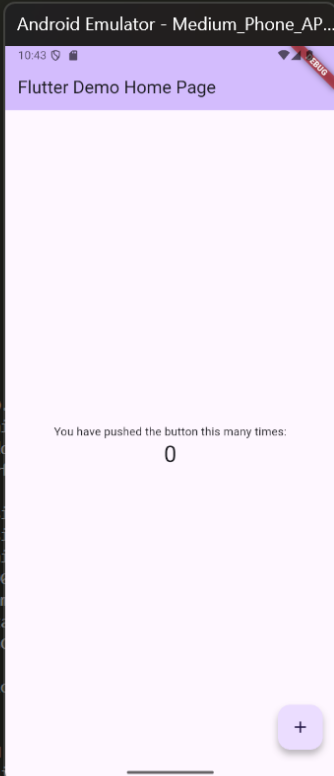
Kembali ke VS Code, ubah platform di pojok kanan bawah ke emulator atau device atau bisa juga menggunakan browser Chrome. Lalu coba running project **hello\_world** dengan tekan **F5** atau **Run > Start Debugging**. Tunggu proses kompilasi hingga selesai, maka aplikasi flutter pertama Anda akan tampil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



```
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> flutter run
Launching lib/main.dart on sdk gphone64 x86 64 in debug mode...
Running Gradle task 'assembleDebug'... 45.4s
✓ Built build/app/outputs/flutter-apk/app-debug.apk
Installing build/app/outputs/flutter-apk/app-debug.apk... 1,442ms
D/FlutterJNI( 6051): Beginning load of flutter...
D/FlutterJNI( 6051): flutter (null) was loaded normally!
I/flutter ( 6051): [IMPORTANT:flutter/shell/platform/android/android_context_gl_impeller.cc(104)]
Syncing files to device sdk gphone64 x86 64... 94ms

Flutter run key commands.
r Hot reload.
R Hot restart.
h List all available interactive commands.
d Detach (terminate "flutter run" but leave application running).
c Clear the screen
q Quit (terminate the application on the device).

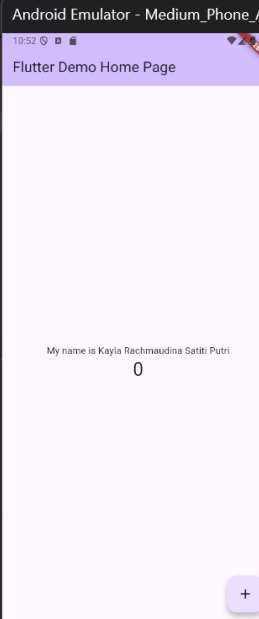
A Dart VM Service on sdk gphone64 x86 64 is available at: http://127.0.0.1:51341/XoI0Plkr1TA=/
The Flutter DevTools debugger and profiler on sdk gphone64 x86 64 is available at: http://127.0.0.1:51341/?url=
I/Choreographer( 6051): Skipped 193 frames! The application may be doing too much work on its main thread.
I/WindowExtensionsImpl( 6051): Initializing Window Extensions, vendor API level=9, activity embedder=0
I/ple.hello world( 6051): Compiler allocated 5042KB to compile void android.view.ViewRootImpl.performTraversals()
D/ProfileInstaller( 6051): Installing profile for com.example.hello_world
D/ActivityThread( 6051): Package [com.google.android.apps.restore] reported as REPLACED, but missing application info. Assuming REMOVED.
D/ActivityThread( 6051): Package [com.google.android.apps.restore] reported as REPLACED, but missing application info. Assuming REMOVED.
I/Choreographer( 6051): Skipped 136 frames! The application may be doing too much work on its main thread.
I/HWUI ( 6051): Davey! duration=2466ms; Flags=1, FrameTimelineVsyncId=45531, IntendedVsync=279000, AnimationStart=281370567100, PerformTraversalsStart=281372632000, DrawStart=281417837700, FrameInterval=16666666, SyncQueued=281459527200, SyncStart=281464904400, IssueDrawCommandsStart=281536584200, QueueBufferDuration=25401200, QueueBufferDuration=396500, GpuCompleted=281536584200, SwapBufferStart=281525580100,
D/WindowLayoutComponentImpl( 6051): Register WindowLayoutInfoListener on Context=com.example.hello_world
@74e2d28
D/InsetsController( 6051): hide(ime(), fromIme=false)
I/ImeTracker( 6051): com.example.hello_world:f85d9851: onCancelled at PHASE_CLIENT_ALREADY_HIDDEN
D/ActivityThread( 6051): Package [com.google.android.partnersetup] reported as REPLACED, but missing application info. Assuming REMOVED.
D/ActivityThread( 6051): Package [com.google.android.partnersetup] reported as REPLACED, but missing application info. Assuming REMOVED.
```

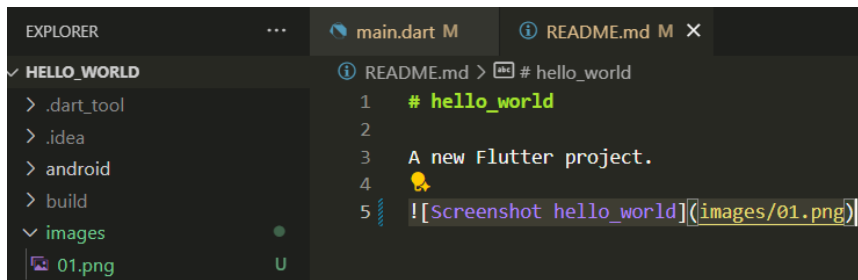


## Langkah 12:

Silakan screenshot seperti pada Langkah 11, namun teks yang ditampilkan dalam aplikasi berupa nama lengkap Anda. Simpan file screenshot dengan nama **01.png** pada folder **images** (buat folder baru jika belum ada) di project **hello\_world** Anda. Lalu ubah isi **README.md** seperti berikut, sehingga tampil hasil screenshot pada file **README.md**. Kemudian push ke repository Anda.

```
lib > main.dart > _MyHomePageState > build
56 class _MyHomePageState extends State<MyHomePage> {
71   Widget build(BuildContext context) {
103     // action in the IDE, or press "p" in the console), to see the
104     // wireframe for each widget.
105     mainAxisAlignment: MainAxisAlignment.center,
106     children: <Widget>[
107       const Text('My name is Kayla Rachmaudina Satiti Putri'),
108       Text(
109         '$_counter',
110         style: Theme.of(context).textTheme.headlineMedium,
111       ), // Text
112     ], // <Widget>[]
113   ], // column
114 ), // Center
115 floatingActionButton: FloatingActionButton(
116   onPressed: _incrementCounter,
117   tooltip: 'Increment',
118   child: const Icon(Icons.add),
119 ), // This trailing comma makes auto-formatting nicer for build methods.
120 ); // Scaffold
121 }
122 }
123 }
```





```
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git commit -m "add name to main.dart"
[main 3a14d35] add name to main.dart
1 file changed, 1 insertion(+), 1 deletion(-)
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 387 bytes | 387.00 KiB/s, done.
Total 4 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.

PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git commit -m "add screenshot image for readme"
[main 5d67f34] add screenshot image for readme
2 files changed, 1 insertion(+), 12 deletions(-)
create mode 100644 images/01.png
PS C:\Users\ADVAN\OneDrive\Documents\SMT 5\Pem. Mobile\hello_world> git push
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 12 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (5/5), 92.80 KiB | 15.47 MiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0
```

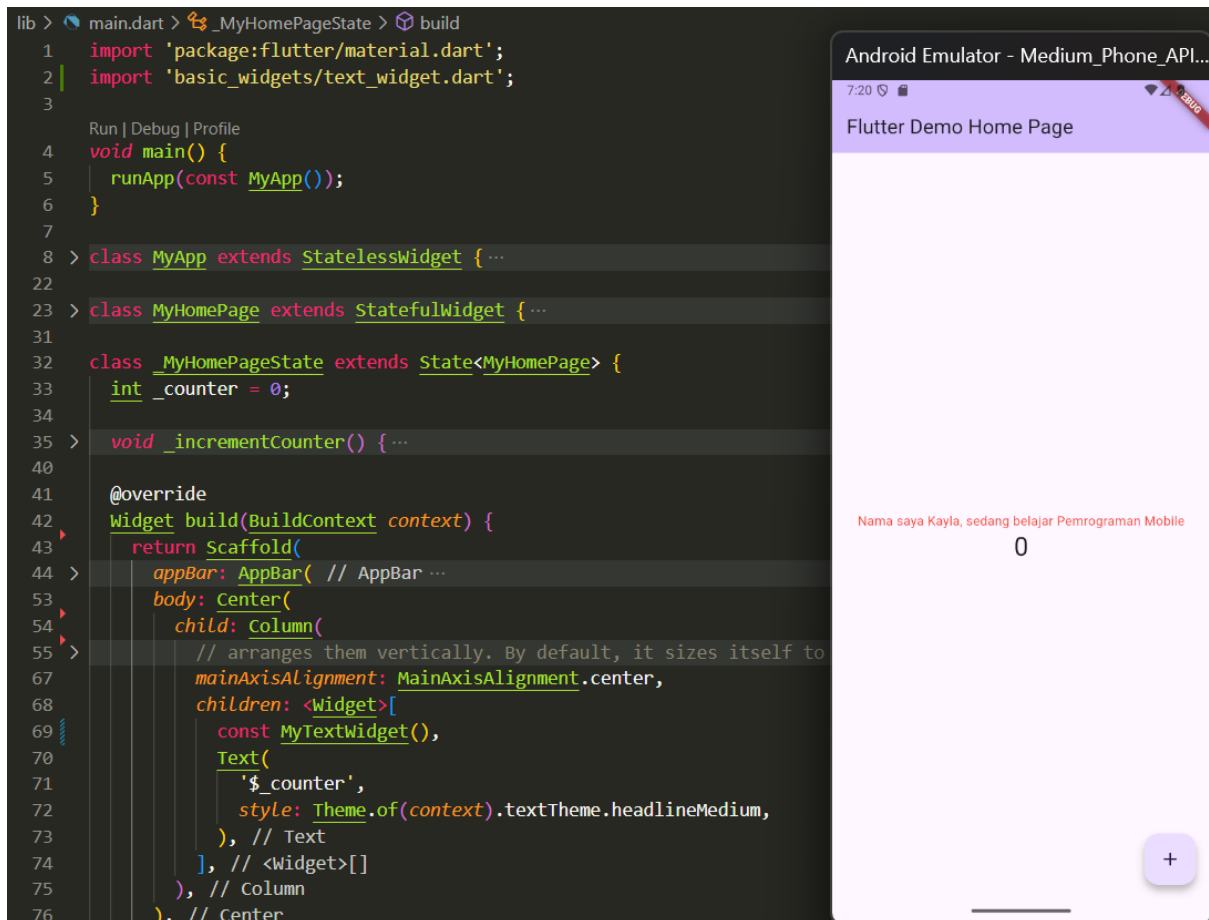
## Praktikum 4: Menerapkan Widget Dasar

### Langkah 1: Text Widget

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**.

```
lib > basic_widgets > text_widget.dart > MyTextWidget
1 import 'package:flutter/material.dart';
2
3 class MyTextWidget extends StatelessWidget {
4   const MyTextWidget({Key? key}) : super(key: key);
5
6   @override
7   Widget build(BuildContext context) {
8     return const Text(
9       "Nama saya Kayla, sedang belajar Pemrograman Mobile",
10      style: TextStyle(color: Colors.red, fontSize: 14),
11      textAlign: TextAlign.center); // Text
12   }
13 }
```

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**.

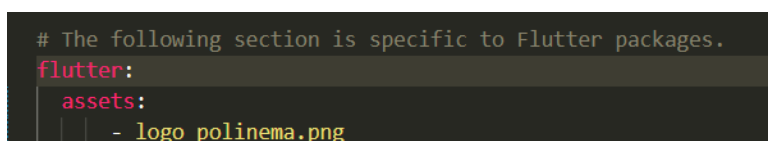


## Langkah 2: Image Widget

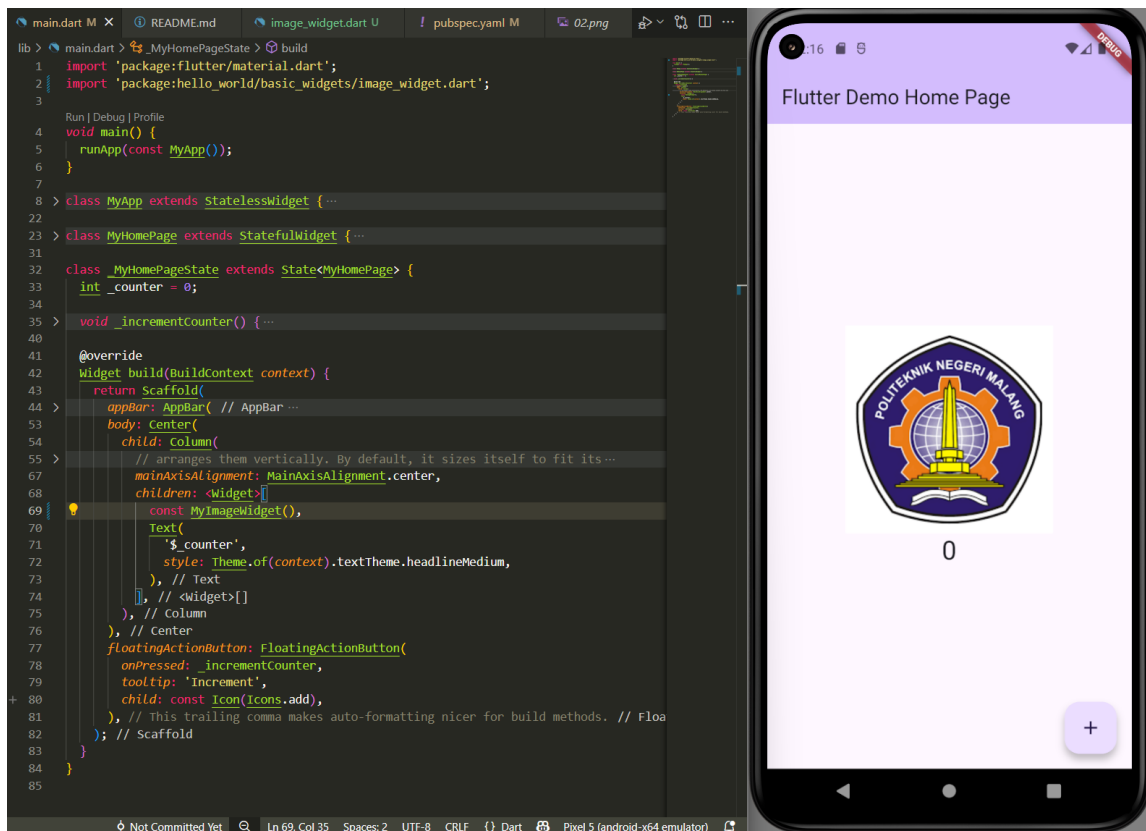
Buat sebuah file `image_widget.dart` di dalam folder `basic_widgets` dengan isi kode berikut.



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 1: Cupertino Button dan Loading Bar

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.



## Langkah 2: Floating Action Button (FAB)

Button widget terdapat beberapa macam pada flutter yaitu FlatButton, DropdownButton, TextButton, FloatingActionButton, IconButton, OutlineButton, PopupMenuButton, dan ElevatedButton.

Buat file di basic\_widgets > fab\_widget.dart. Import stateless widget dari material. Lalu isi kode di dalam method Widget build adalah sebagai berikut.

```
lib > basic_widgets > fab_widget.dart > FabWidget
1  import 'package:flutter/material.dart';
2
3  class FabWidget extends StatelessWidget {
4    const FabWidget({super.key});
5
6    @override
7    Widget build(BuildContext context) {
8      return MaterialApp(
9        home: Scaffold(
10         floatingActionButton: FloatingActionButton(
11           onPressed: () {
12             // Add your onPressed code here!
13           },
14           child: const Icon(Icons.thumb_up),
15           backgroundColor: Colors.pink,
16         ), // FloatingActionButton
17       ), // Scaffold
18     ); // MaterialApp
19   }
20 }
```

## Langkah 3: Scaffold Widget

Scaffold widget digunakan untuk mengatur tata letak sesuai dengan material design.

Ubah isi kode main.dart seperti berikut.

```
1  import 'package:flutter/material.dart';
2
3  void main() {
4    runApp(const MyApp());
5  }
6
7  class MyApp extends StatelessWidget {
8    const MyApp({Key? key}) : super(key: key);
9
10   // This widget is the root of your application.
11   @override
12   Widget build(BuildContext context) {
13     return MaterialApp(
14       title: 'Flutter Demo',
15       theme: ThemeData(
16         primarySwatch: Colors.red,
17       ),
18       home: const MyHomePage(title: 'My Increment App'),
19     );
20   }
21 }
22
23 class MyHomePage extends StatefulWidget {
24   const MyHomePage({Key? key, required this.title}) : super(key: key);
25
26   final String title;
27
28   @override
29   State<MyHomePage> createState() => _MyHomePageState();
30 }
```

```

32 class _MyHomePageState extends State<MyHomePage> {
33   int _counter = 0;
34
35   void _incrementCounter() {
36     setState(() {
37       _counter++;
38     });
39   }
40
41   @override
42   Widget build(BuildContext context) {
43     return Scaffold(
44       appBar: AppBar(
45         title: Text(widget.title),
46       ),
47       body: Center(
48         child: Column(
49           mainAxisAlignment: MainAxisAlignment.center,
50           children: <Widget>[
51             const Text(
52               'You have pushed the button this many times:',
53             ),
54             Text(
55               '$_counter',
56               style: Theme.of(context).textTheme.headlineMedium,
57             ),
58           ],
59         ),
60       ),
61       bottomNavigationBar: BottomAppBar(
62         child: Container(
63           height: 50.0,
64         ),
65       ),
66       floatingActionButton: FloatingActionButton(
67         onPressed: _incrementCounter,
68         tooltip: 'Increment Counter',
69         child: const Icon(Icons.add),
70       ),
71       floatingActionButtonLocation: FloatingActionButtonLocation.centerDocked,
72     );
73   }
74 }

```

## Langkah 4: Dialog Widget

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

Ubah isi kode main.dart seperti berikut.

```

1  import 'package:flutter/material.dart';
2
3  void main() {
4    runApp(const MyApp());
5  }
6
7  class MyApp extends StatelessWidget {
8    const MyApp({Key? key}) : super(key: key);
9
10   @override
11   Widget build(BuildContext context) {
12     return const MaterialApp(
13       home: Scaffold(
14         body: MyLayout(),
15       ),
16     );
17   }
18 }
19
20 class MyLayout extends StatelessWidget {
21   const MyLayout({Key? key}) : super(key: key);

```



```

23 @override
24 Widget build(BuildContext context) {
25   return Padding(
26     padding: const EdgeInsets.all(8.0),
27     child: ElevatedButton(
28       child: const Text('Show alert'),
29       onPressed: () {
30         showAlertDialog(context);
31       },
32     ),
33   );
34 }
35 }
36
37 showAlertDialog(BuildContext context) {
38   // set up the button
39   Widget okButton = TextButton(
40     child: const Text("OK"),
41     onPressed: () {
42       Navigator.pop(context);
43     },
44   );
45
46   // set up the AlertDialog
47   AlertDialog alert = AlertDialog(
48     title: const Text("My title"),
49     content: const Text("This is my message."),
50     actions: [
51       okButton,
52     ],
53   );
54
55   // show the dialog
56   showDialog(
57     context: context,
58     builder: (BuildContext context) {
59       return alert;
60     },
61   );
62 }

```

## Langkah 5: Input dan Selection Widget

Flutter menyediakan widget yang dapat menerima input dari pengguna aplikasi yaitu antara lain Checkbox, Date and Time Pickers, Radio Button, Slider, Switch, TextField.

Contoh penggunaan TextField widget adalah sebagai berikut:

```

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(
        appBar: AppBar(title: const Text("Contoh TextField")),
        body: const TextField(
          obscureText: false,
          decoration: InputDecoration(
            border: OutlineInputBorder(),
            labelText: 'Nama',
          ), // InputDecoration
        ), // TextField
      ), // Scaffold
    ); // MaterialApp
  }
}

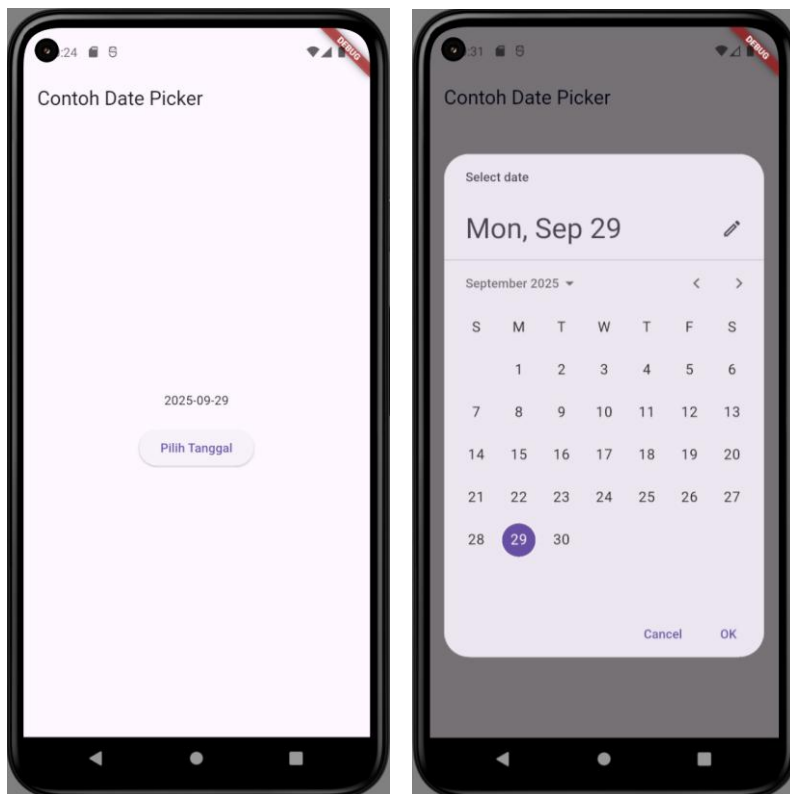
```

## Langkah 6: Date and Time Pickers

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

```
1  import 'dart:async';
2  import 'package:flutter/material.dart';
3
4  void main() => runApp(const MyApp());
5
6  class MyApp extends StatelessWidget {
7    const MyApp({Key? key}) : super(key: key);
8
9    @override
10   Widget build(BuildContext context) {
11     return const MaterialApp(
12       title: 'Contoh Date Picker',
13       home: MyHomePage(title: 'Contoh Date Picker'),
14     );
15   }
16 }
17
18 class MyHomePage extends StatefulWidget {
19   const MyHomePage({Key? key, required this.title}) : super(key: key);
20
21   final String title;
22
23   @override
24   _MyHomePageState createState() => _MyHomePageState();
25 }
26
27 class _MyHomePageState extends State<MyHomePage> {
28   // Variable/State untuk mengambil tanggal
29   DateTime selectedDate = DateTime.now();
30
31   // Initial SelectDate Flutter
32   Future<void> _selectDate(BuildContext context) async {
33     // Initial DateTime Final Picked
34     final DateTime? picked = await showDatePicker(
35       context: context,
36       initialDate: selectedDate,
37       firstDate: DateTime(2015, 8),
38       lastDate: DateTime(2101));
39     if (picked != null && picked != selectedDate) {
40       setState(() {
41         selectedDate = picked;
42       });
43     }
44   }
45
46   @override
47   Widget build(BuildContext context) {
48     return Scaffold(
49       appBar: AppBar(
50         title: Text(widget.title),
51       ),
52       body: Center(
53         child: Column(
54           mainAxisAlignment: MainAxisAlignment.min,
55           children: <Widget>[
56             Text("${selectedDate.toLocal()}".split(' ')[0]),
57             const SizedBox(
58               height: 20.0,
59             ),
60             ElevatedButton(
61               onPressed: () => {
62                 _selectDate(context),
63                 // ignore: avoid_print
64                 print(selectedDate.day + selectedDate.month + selectedDate.year)
65               },
66               child: const Text('Pilih Tanggal'),
67             ),
68           ],
69         ),
70       ),
71     );
72   }
73 }
```

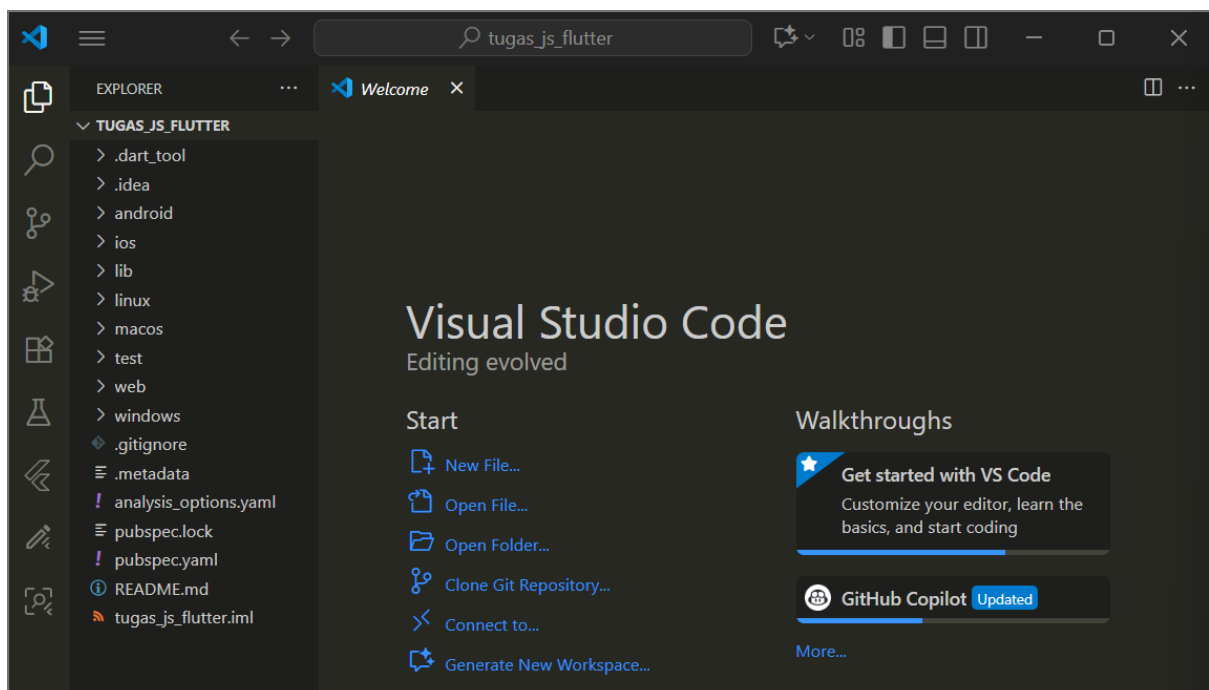
Berikut hasilnya:



## Tugas Praktikum

Selesaikan [Codelabs: Your first Flutter app](#), lalu buatlah laporan praktikumnya dan push ke repository GitHub Anda!

### Langkah 1: Membuat Project Flutter baru



## Langkah 2: Menyesuaikan pubspec.yaml

```
! pubspec.yaml
1  name: tugas_js_flutter
2  description: "A new Flutter project."
3  publish_to: "none"
4  version: 0.1.0
5
6  environment:
7    sdk: ^3.9.0
8
9  dependencies:
10   flutter:
11     sdk: flutter
12   english_words: ^4.0.0
13   provider: ^6.1.5
14
15  dev_dependencies:
16   flutter_test:
17     sdk: flutter
18   flutter_lints: ^6.0.0
19
20  flutter:
21    uses-material-design: true
```

## Langkah 3: Membuka dan mengganti analysis\_options.yaml

```
! analysis_options.yaml
1  include: package:flutter_lints/flutter.yaml
2
3  linter:
4    rules:
5      avoid_print: false
6      prefer_const_constructors_in_immutables: false
7      prefer_const_constructors: false
8      prefer_const_literals_to_create_immutables: false
9      prefer_final_fields: false
10     unnecessary_breaks: true
11     use_key_in_widget_constructors: false
```

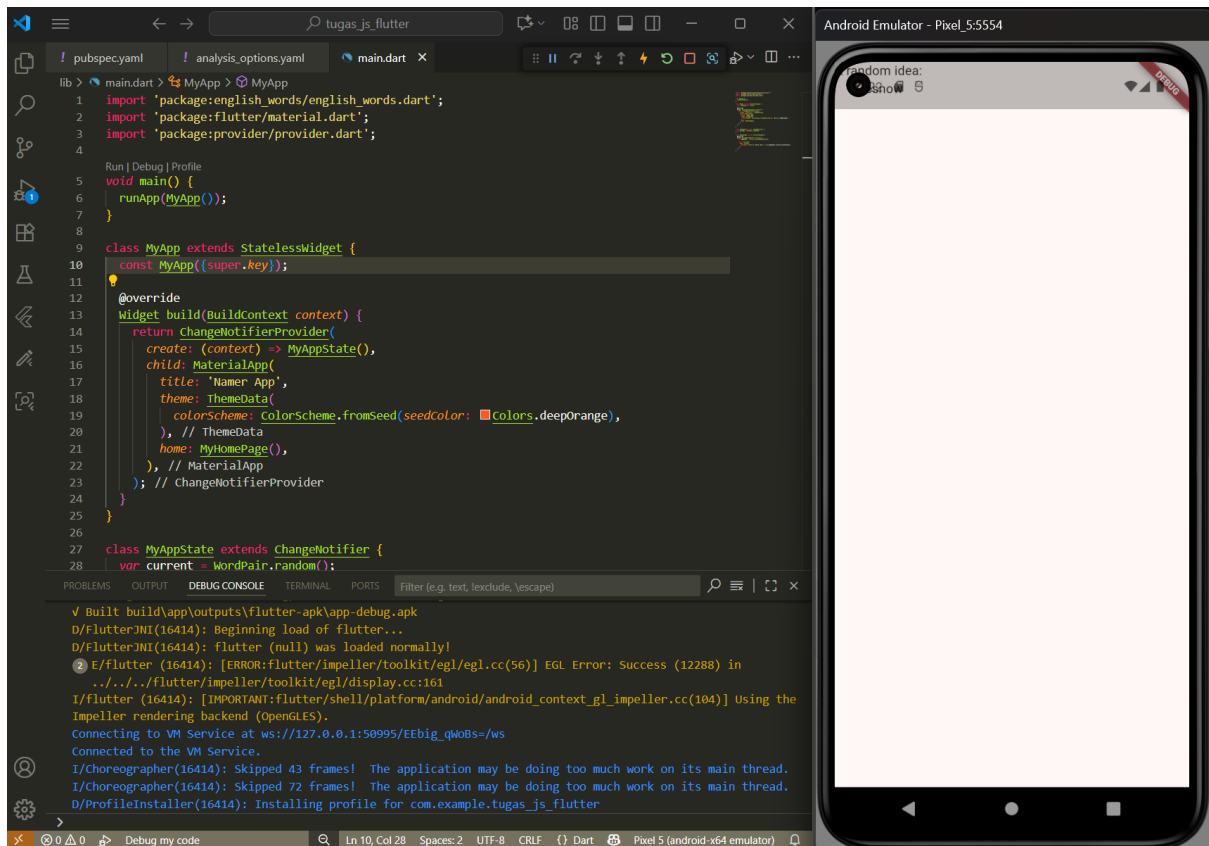
## Langkah 4: Membuka main.dart dan mengganti sesuai dengan kode di codelab

```

lib > main.dart > ...
1  import 'package:english_words/english_words.dart';
2  import 'package:flutter/material.dart';
3  import 'package:provider/provider.dart';
4
Run | Debug | Profile
5  void main() {
6    runApp(MyApp());
7  }
8
9  class MyApp extends StatelessWidget {
10   const MyApp({super.key});
11
12   @override
13   Widget build(BuildContext context) {
14     return ChangeNotifierProvider(
15       create: (context) => MyAppState(),
16       child: MaterialApp(
17         title: 'Namer App',
18         theme: ThemeData(
19           colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepOrange),
20         ), // ThemeData
21         home: MyHomePage(),
22       ), // MaterialApp
23     ); // ChangeNotifierProvider
24   }
25 }
26
27 class MyAppState extends ChangeNotifier {
28   var current = WordPair.random();
29 }
30
31 class MyHomePage extends StatelessWidget {
32   @override
33   Widget build(BuildContext context) {
34     var appState = context.watch<MyAppState>();
35
36     return Scaffold(
37       body: Column(
38         children: [Text('A random idea:'), Text(appState.current.asLowerCase)],
39       ), // Column
40     ); // Scaffold
41   }
42 }

```

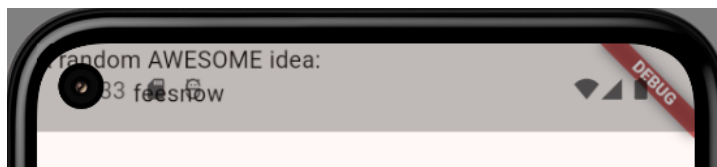
## Langkah 5: Melaunching app



Langkah 6: Melakukan hot reload setelah ada perubahan

```

return Scaffold(
  body: Column(
    children: [
      Text('A random AWESOME idea:'), // ← Example change.
      Text(appState.current.asLowerCase),
    ],
  ), // Column
); // Scaffold
  
```



Langkah 7: Menambah button

```
class MyHomePage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    var appState = context.watch<MyAppState>();

    return Scaffold(
      body: Column(
        children: [
          Text('A random AWESOME idea:'),
          Text(appState.current.asLowerCase),

          // ↓ Add this.
          ElevatedButton(
            onPressed: () {
              print('button pressed!');
            },
            child: Text('Next'),
          ), // ElevatedButton
        ],
      ), // Column
    ); // Scaffold
  }
}
```

A random AWESOME idea:

sweetguy

Next

DEBUG

## Langkah 8: Menambahkan method getNext()

```
class MyAppState extends ChangeNotifier {
  var current = WordPair.random();

  void getNext() {
    current = WordPair.random();
    notifyListeners();
  }
}
```

```
// ↓ Add this.
ElevatedButton(
  onPressed: () {
    appState.getNext(); // ← This instead of print().
  },
  child: Text('Next'),
), // ElevatedButton
```

## Langkah 9: Melaunching app

A random AWESOME idea:

bearmark

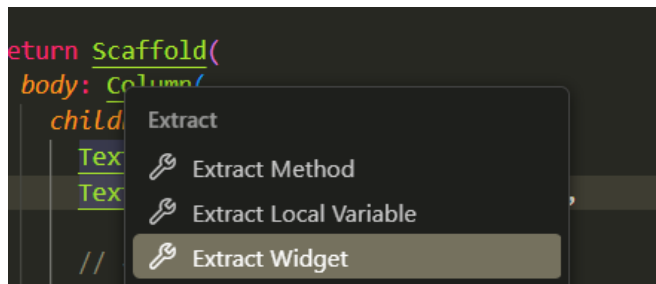
Next

ketika di pencet button next, akan langsung berganti.

### Langkah 10: Tulis Ulang MyHomePage Widget

```
class MyHomePage extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    var appState = context.watch<MyAppState>();  
    var pair = appState.current; // ← Add this.  
  
    return Scaffold(  
      body: Column(  
        children: [  
          Text('A random AWESOME idea:'),  
          Text(pair.asLowerCase), // ← Change to this.  
          ElevatedButton(  
            onPressed: () {  
              appState.getNext();  
            },  
            child: Text('Next'),  
          ), // ElevatedButton  
        ],  
      ), // Column  
    ); // Scaffold  
  }  
}
```

### Langkah 11: Membuat Widget Baru menggunakan refactor





```

@override
Widget build(BuildContext context) {
  var appState = context.watch<MyAppState>();

  return Scaffold(
    body: Column(
      children: [
        Text('A random AWESOME idea:'),
        BigCard(appState: appState),

        // ↓ Add this.
        ElevatedButton(
          onPressed: () {
            appState.getNext(); // ← This instead of print().
          },
          child: Text('Next'),
        ), // ElevatedButton
      ],
    ), // Column
  ); // Scaffold
}

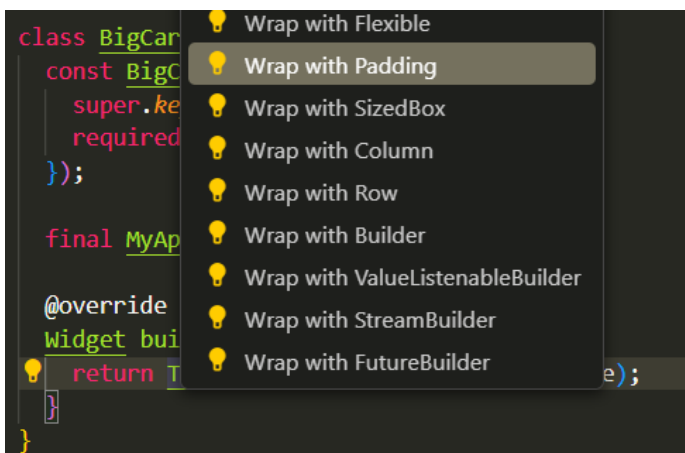
class BigCard extends StatelessWidget {
  const BigCard({
    super.key,
    required this.appState,
  });

  final MyAppState appState;

  @override
  Widget build(BuildContext context) {
    return Text(appState.current.asLowerCase);
  }
}

```

## Langkah 12: Membuka Refractor dan Memilih wrap with padding



```

class BigCard
const BigCard({
  super.key,
  required this.appState,
});

final MyAppState appState;

@override
Widget build(BuildContext context) {
  return Text(appState.current.asLowerCase);
}

```

- Wrap with Flexible
- Wrap with Padding
- Wrap with SizedBox
- Wrap with Column
- Wrap with Row
- Wrap with Builder
- Wrap with ValueListenableBuilder
- Wrap with StreamBuilder
- Wrap with FutureBuilder

```

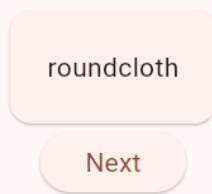
@override
Widget build(BuildContext context) {
  return Padding(
    padding: const EdgeInsets.all(8.0),
    child: Text(appState.current.asLowerCase),
  ); // Padding
}

```

### Langkah 13: Membuka Refractor dan memilih wrap with widget...

```
@override
Widget build(BuildContext context) {
  return Card(
    child: Padding(
      padding: const EdgeInsets.all(8.0),
      child: Text(appState.current.asLowerCase),
    ), // Padding
  ); // Card
}
```

A random AWESOME idea:



### Langkah 14: Menambahkan kode berikut ke build method di BigCard

```
@override
Widget build(BuildContext context) {
  final theme = Theme.of(context); // ← Add this.

  return Card(
    color: theme.colorScheme.primary, // ← And also this.
    child: Padding(
      padding: const EdgeInsets.all(20),
      child: Text(pair.asLowerCase),
    ), // Padding
  ); // Card
}
```

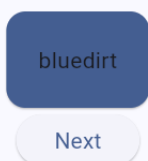
A random AWESOME idea:



Mencoba mengganti warna

```
theme: ThemeData(
  colorScheme: ColorScheme.fromSeed(seedColor: Colors.blueAccent),
```

A random AWESOME idea:



## Langkah 15: Menambahkan baris-baris berikut untuk mengubah ukuran teks

```
@override
Widget build(BuildContext context) {
  final theme = Theme.of(context);
  // ↓ Add this.
  final style = theme.textTheme.displayMedium!.copyWith(
    color: theme.colorScheme.onPrimary,
  );

  return Card(
    color: theme.colorScheme.primary,
    child: Padding(
      padding: const EdgeInsets.all(20),
      // ↓ Change this line.
      child: Text(pair.asLowerCase, style: style),
    ), // Padding
  ); // Card
}
```

A random AWESOME idea:

