

# Pengenalan Flutter

TI2137 - Pengembangan Aplikasi Mobil Front-End



# Team Teaching

- Sunaryo Winardi, S.Kom., M.Tl.
- Sio Jurnal Pipin, S.Kom., M.Kom.
- M. Taufiq Hidayat Pohan, S.Kom.
- Richy Rotuahta Saragih, S.Kom.



# Context

- Pengenalan Flutter
- Cara instalasi dan membuat proyek di Flutter
- Widget



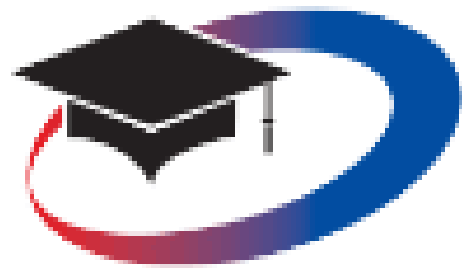
# Apa itu Flutter?



# Apa itu Flutter

- Flutter adalah *framework* yang berguna untuk membangun aplikasi *mobile, web, dan desktop*.
- Flutter dikembangkan oleh Google pada tahun 2015.
- Flutter menggunakan bahasa *Dart* sebagai bahasa dasar dari Flutter.
- Flutter banyak digunakan oleh beberapa perusahaan besar di dunia, seperti Google, Alibaba Group, hingga Grab.

# Kelebihan Flutter



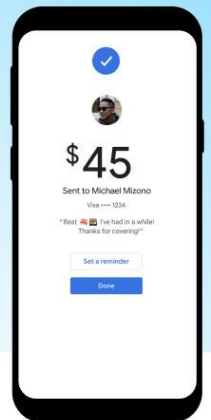
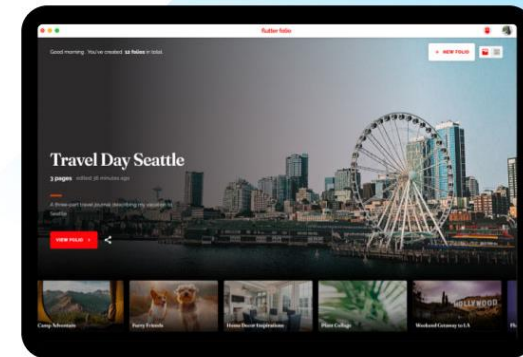
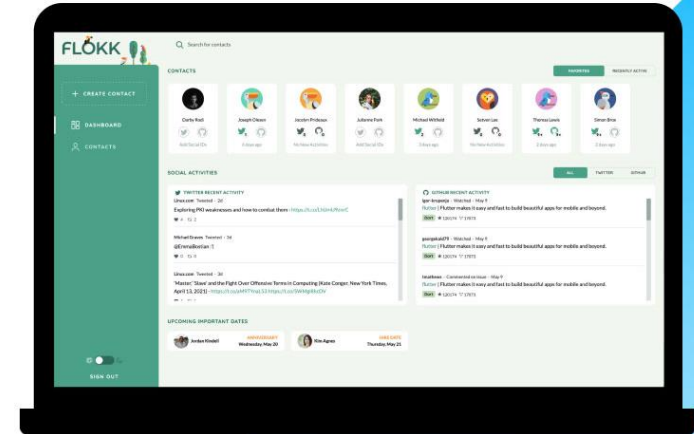
UNIVERSITAS  
**MIKROSKIL**

PRODI. TEKNIK INFORMATIKA (S-1)

# Multi-Platform

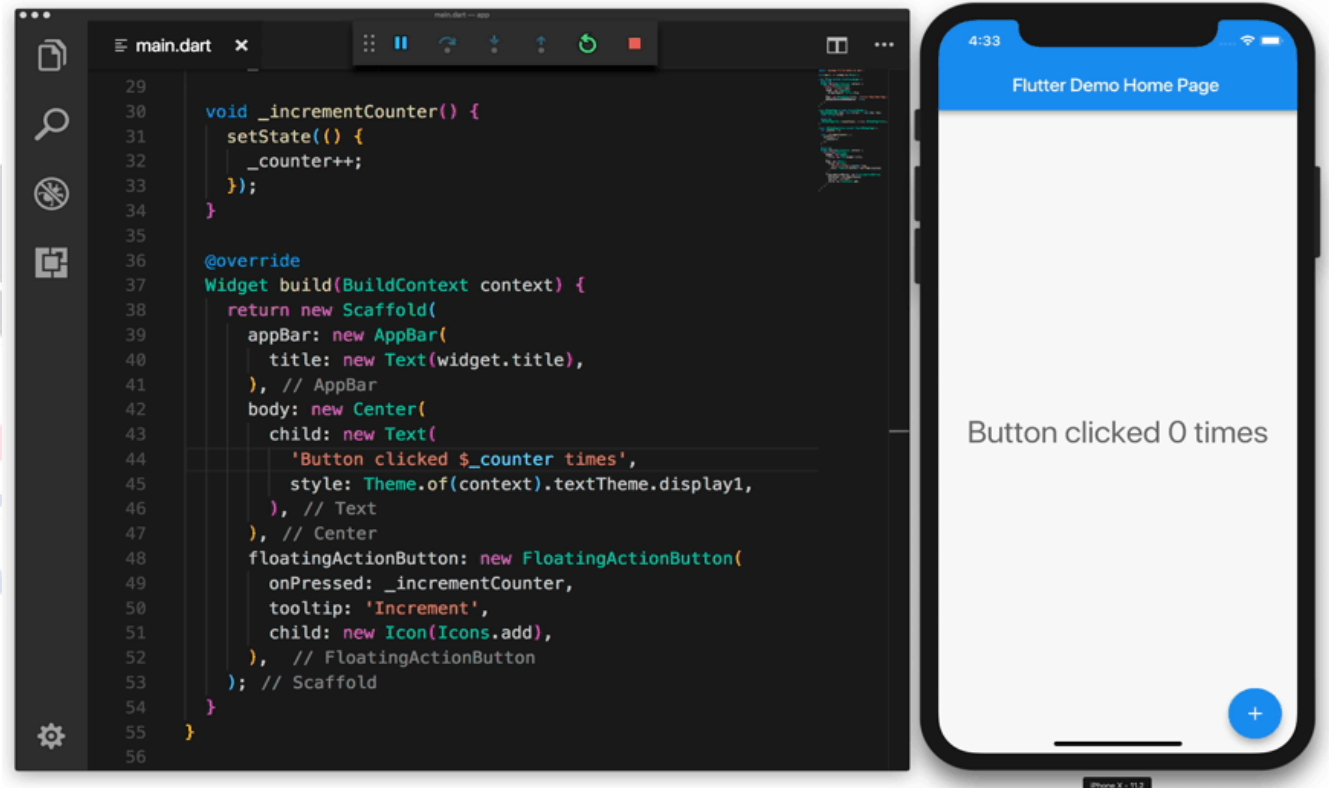
- Aplikasi yang dibuat pakai Flutter dapat dijalankan di Android dan iOS sekaligus.
- Selain di mobile, aplikasi dapat dijalankan di desktop dan web.

MIKROS



# Fast development (Hot reload)

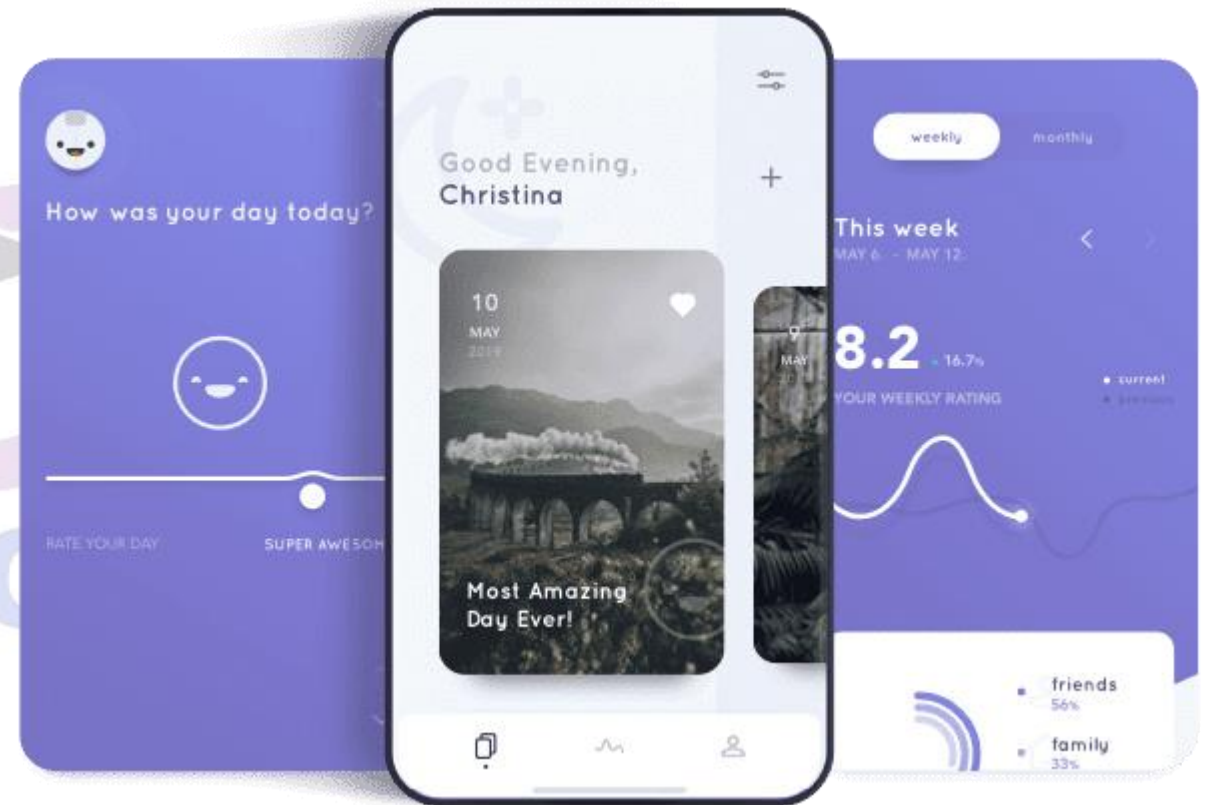
- Flutter memiliki sebuah fitur hot reload.
- Setiap perubahan pada kode, Flutter akan otomatis melakukan injeksi pada aplikasi sehingga perubahan dapat dilihat langsung tanpa perlu *compile* dari awal.



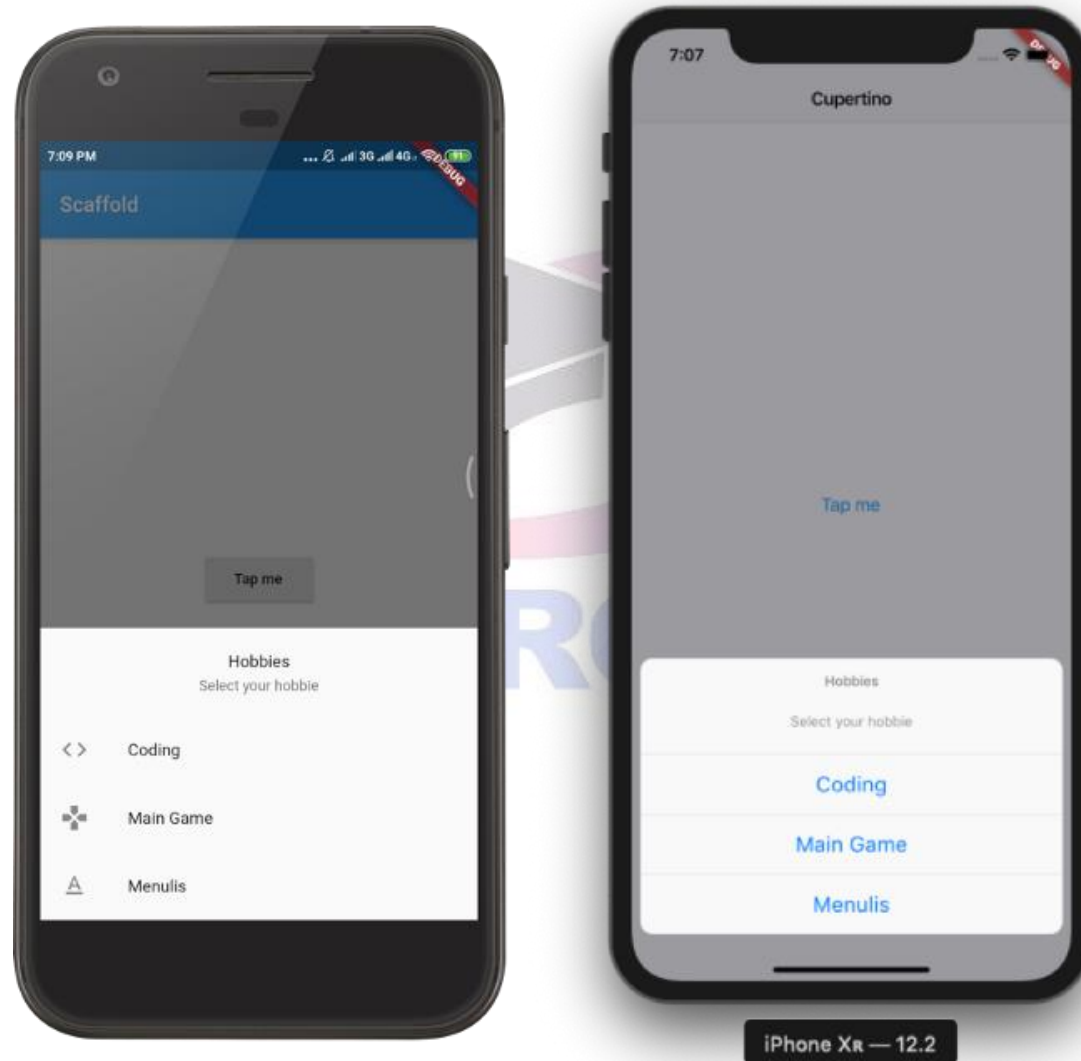


# Beautiful Design

- Keseluruhan UI pada Flutter dibangun menggunakan *widget*.
- *Widget* adalah komponen yang menggambarkan tampilan yang akan dibuat.
- Terdapat 2 set widget, Material Design (Android) dan Cupertino (iOS).

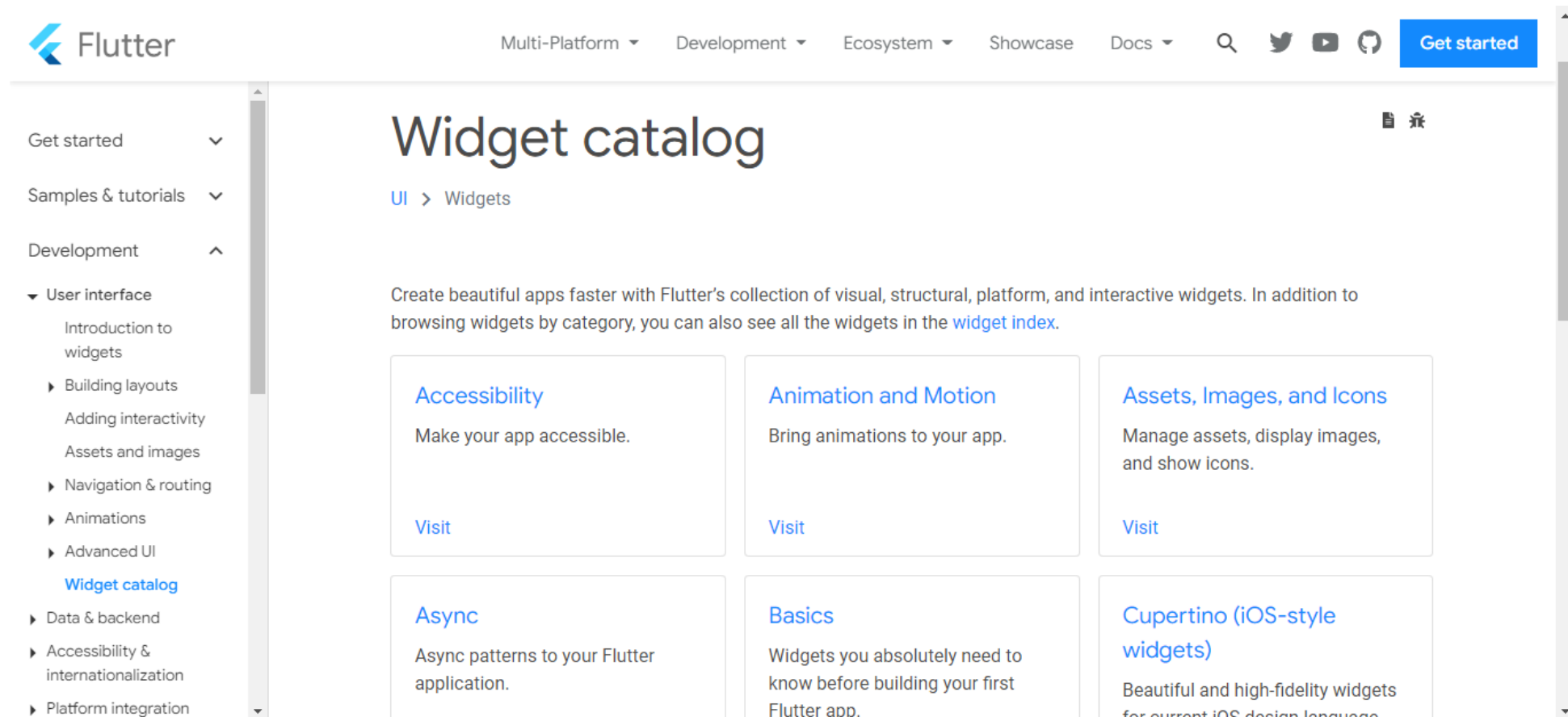


# Material and Cupertino



# Widget Catalog

<https://docs.flutter.dev/development/ui/widgets>



The screenshot shows the Flutter Widget Catalog page. The top navigation bar includes the Flutter logo, links for Multi-Platform, Development, Ecosystem, Showcase, Docs, a search icon, social media icons for Twitter, YouTube, and GitHub, and a 'Get started' button. The left sidebar contains a 'Development' section with a list of topics: User interface (Introduction to widgets, Building layouts, Adding interactivity, Assets and images, Navigation & routing, Animations, Advanced UI, and the highlighted 'Widget catalog'), Data & backend, Accessibility & internationalization, and Platform integration. The main content area is titled 'Widget catalog' with a breadcrumb 'UI > Widgets'. It features an introductory paragraph and a grid of six widget categories, each with a title, description, and a 'Visit' link.

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ 🔍 🐦 📺 🌐 [Get started](#)

Get started ▾  
Samples & tutorials ▾  
Development ▴  
▾ User interface  
    Introduction to widgets  
    ▶ Building layouts  
    Adding interactivity  
    Assets and images  
    ▶ Navigation & routing  
    ▶ Animations  
    ▶ Advanced UI  
    [Widget catalog](#)  
▶ Data & backend  
▶ Accessibility & internationalization  
▶ Platform integration ▾

## Widget catalog

[UI](#) > Widgets

Create beautiful apps faster with Flutter's collection of visual, structural, platform, and interactive widgets. In addition to browsing widgets by category, you can also see all the widgets in the [widget index](#).

[Accessibility](#)  
Make your app accessible.  
[Visit](#)

[Animation and Motion](#)  
Bring animations to your app.  
[Visit](#)

[Assets, Images, and Icons](#)  
Manage assets, display images, and show icons.  
[Visit](#)

[Async](#)  
Async patterns to your Flutter application.

[Basics](#)  
Widgets you absolutely need to know before building your first Flutter app.

[Cupertino \(iOS-style widgets\)](#)  
Beautiful and high-fidelity widgets for current iOS design language.

# Cara Instalasi Flutter

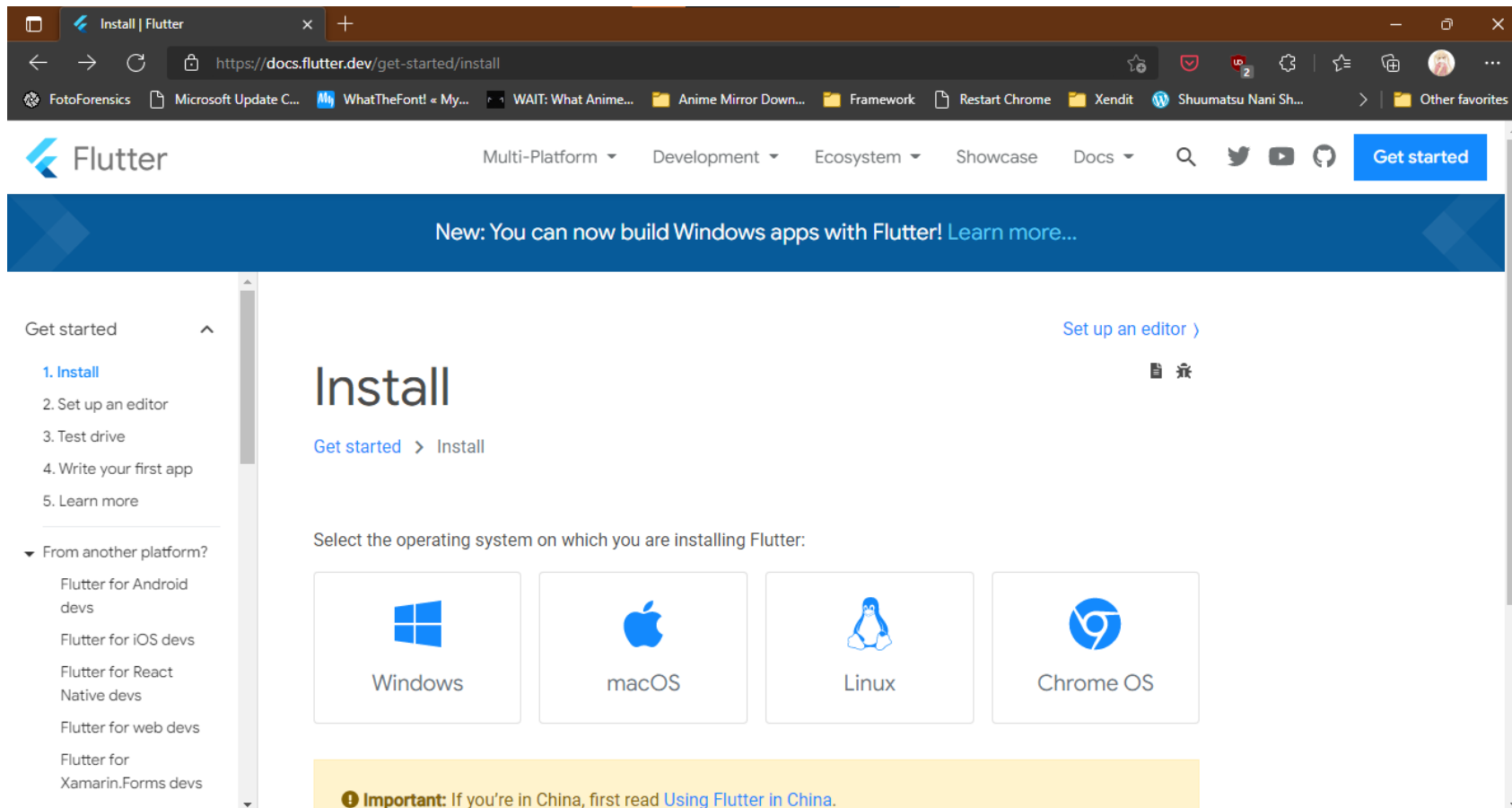


# Persyaratan Sistem dan Tools

- Windows 7 SP1 (atau terbaru) x64 / Linux x64 / macOS
- Disk Space 1.64GB (Windows), 600MB (Linux), 2.8GB (macOS)
- Tools
  - IDE (Android Studio / Visual Studio Code / Xcode)
  - Android SDK
  - Emulator (Genymotion / Android Virtual Device (AVD) / iOS Simulation (untuk iOS) atau bisa menggunakan Device langsung dengan mengaktifkan **USB Debugging** dan **Install via USB** pada menu Developer options.

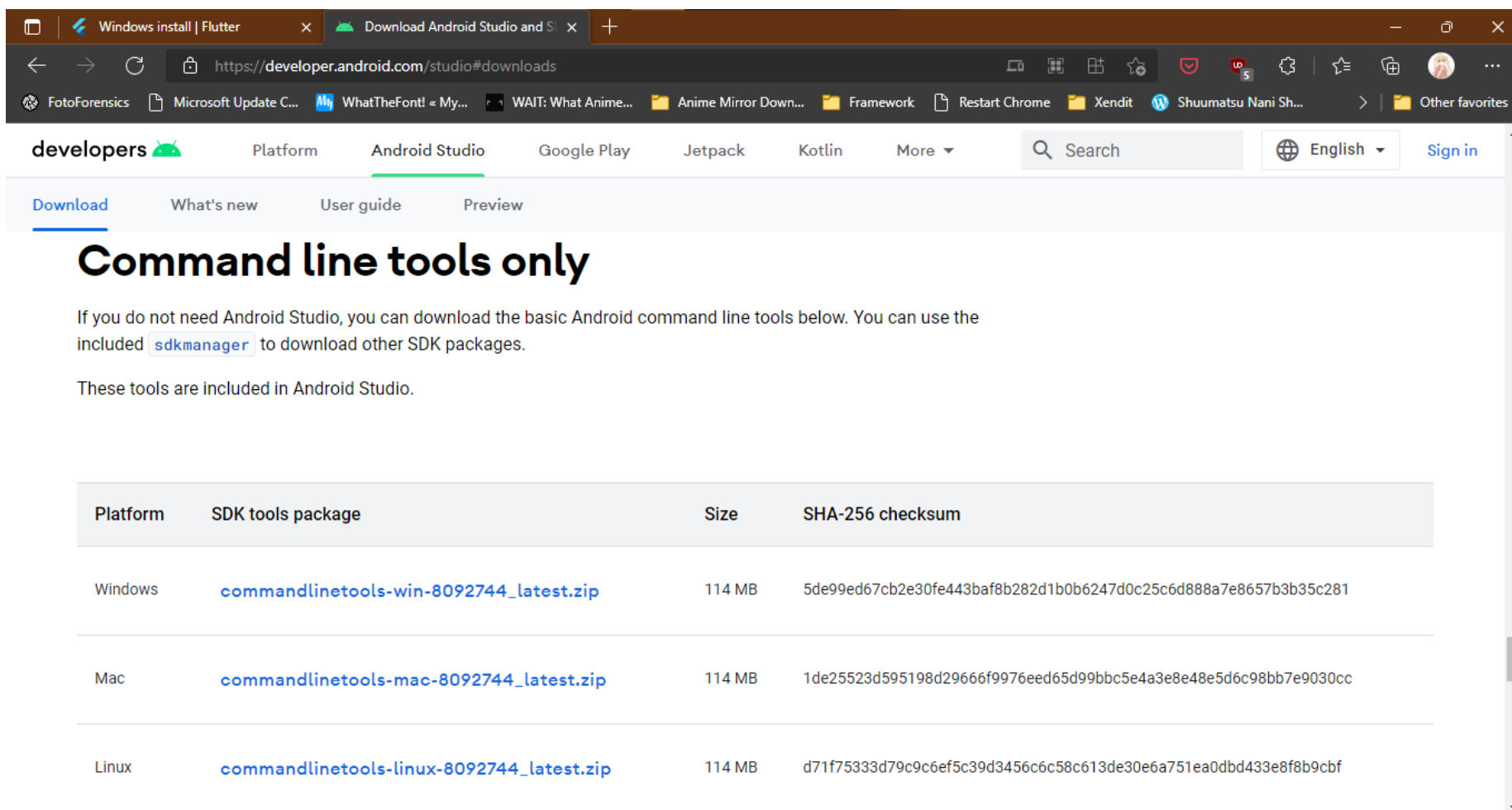
# Download Flutter SDK

<https://docs.flutter.dev/get-started/install>



# Download Android Command line tools only

<https://developer.android.com/studio#downloads>

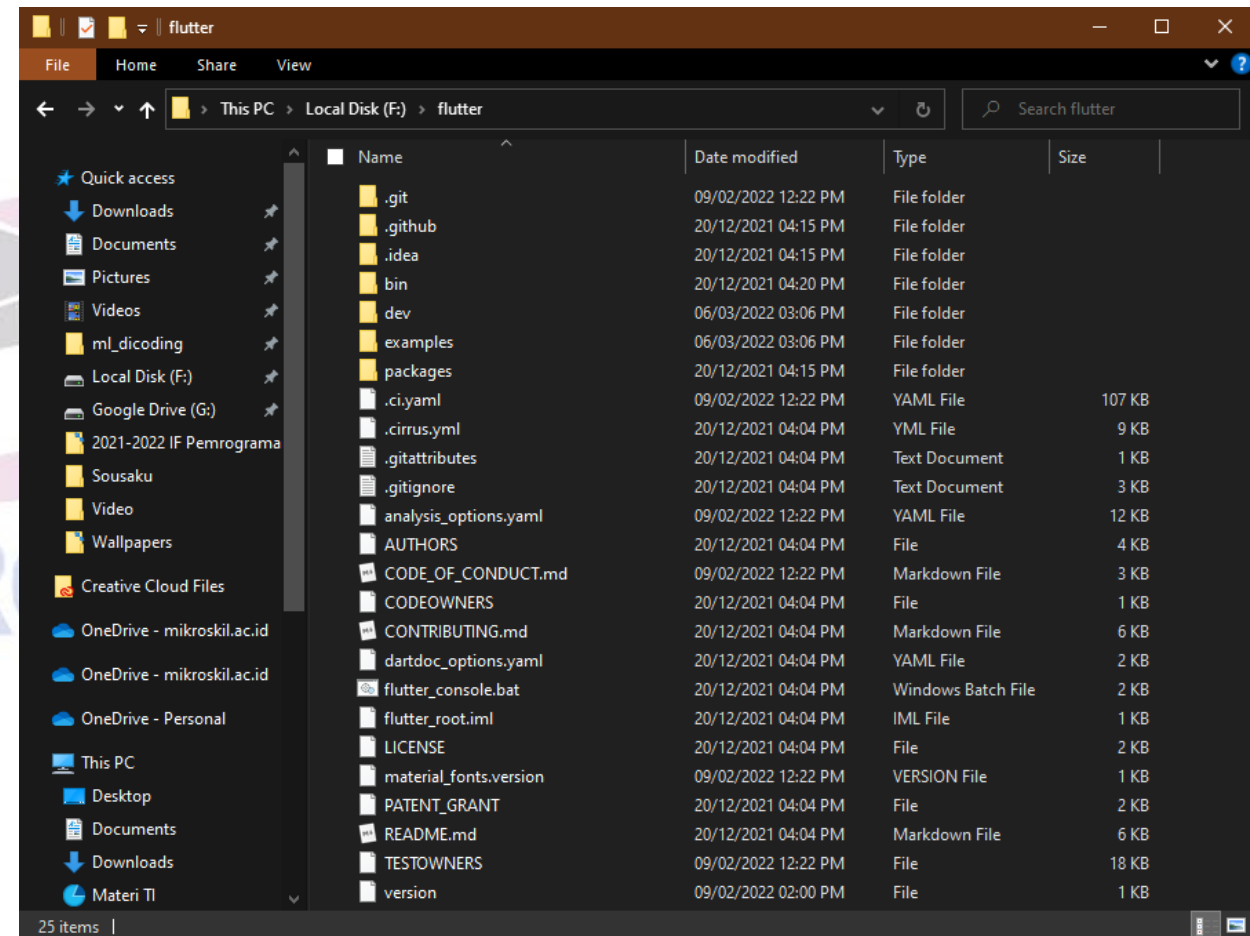


The screenshot shows a web browser window with the URL <https://developer.android.com/studio#downloads>. The page is titled "Command line tools only" and provides instructions on how to download the basic Android command line tools. It includes a table with the following data:

Platform	SDK tools package	Size	SHA-256 checksum
Windows	<a href="#">commandlinetools-win-8092744_latest.zip</a>	114 MB	5de99ed67cb2e30fe443baf8b282d1b0b6247d0c25c6d888a7e8657b3b35c281
Mac	<a href="#">commandlinetools-mac-8092744_latest.zip</a>	114 MB	1de25523d595198d29666f9976eed65d99bbc5e4a3e8e48e5d6c98bb7e9030cc
Linux	<a href="#">commandlinetools-linux-8092744_latest.zip</a>	114 MB	d71f75333d79c9c6ef5c39d3456c6c58c613de30e6a751ea0dbd433e8f8b9cbf

# Tutorial Instalasi

1. Buat folder **flutter** (bebas mau di taruh di drive manapun)
2. Extract file flutter SDK yang sudah di unzip ke folder **flutter** yang barusan dibuat.





# Tutorial Instalasi

3. Buat folder **android** (bebas mau di taruh di drive manapun)
4. Extract file **commandlinetools-win-8092744\_latest.zip** ke folder **android**
5. Di dalam folder **android** terdapat folder **cmdline-tools**, buat folder **latest** didalam folder **cmdline-tools**, kemudian pindahkan seluruh isi folder **cmdline-tools** ke folder **latest**.
6. Buka cmd, masuk ke directory **android/cmdline-tools/latest/bin**
7. Kemudian jalankan 3 perintah ini
  - `sdkmanager --update`
  - `sdkmanager "platforms;android-29" "build-tools;28.0.3" "extras;google;m2repository" "extras;android;m2repository" "platform-tools"`
  - `sdkmanager --licenses`

# Tutorial Instalasi

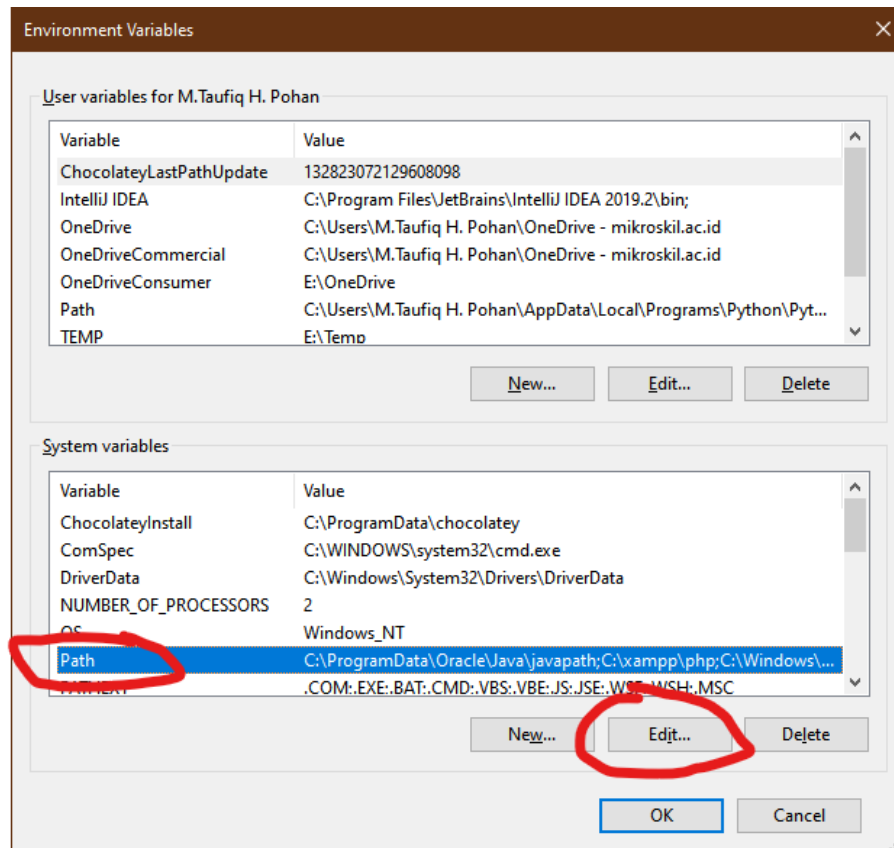
8. Update system path windows dengan menambahkan direktori folder flutter dan android dengan cara sebagai berikut:
- Klik Start Menu
  - Ketikkan **Environment Variable**
  - Pilih **Edit environment variable for your account**
  - Kemudian klik tombol **Environment variables**



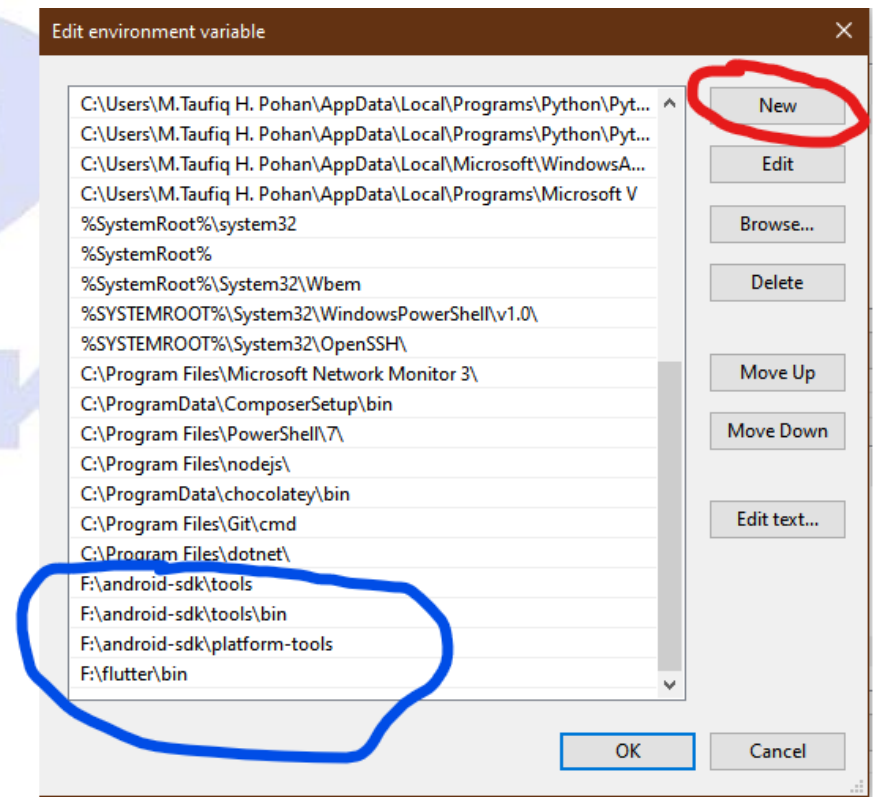
MIKROSKIL

# Tutorial Instalasi

- Di bagian **system variable**, pilih **path**, kemudian klik **Edit**

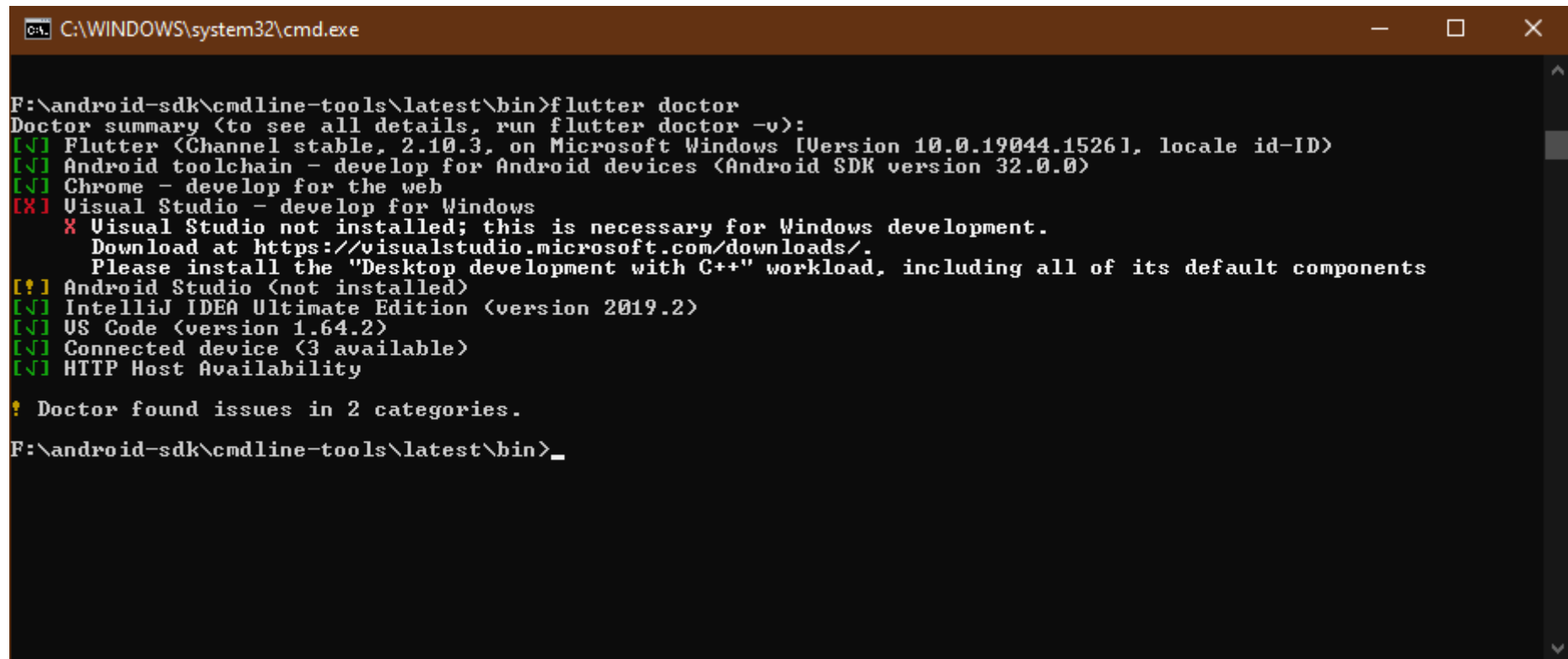


- Kemudian klik **New** dan masukkan lokasi folder **flutter** dan folder **android** berada



# Tutorial Instalasi

9. Buka cmd, kemudian jalankan **flutter doctor**. Jika hasilnya kurang lebih seperti di bawah ini, maka flutter berhasil di install



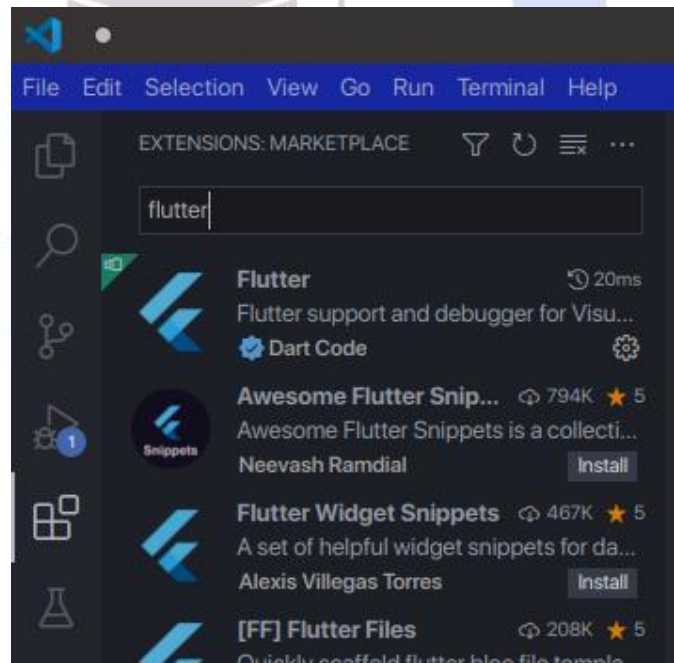
```
C:\WINDOWS\system32\cmd.exe

F:\android-sdk\cmdline-tools\latest\bin>flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 2.10.3, on Microsoft Windows [Version 10.0.19044.1526], locale id-ID)
[✓] Android toolchain - develop for Android devices (Android SDK version 32.0.0)
[✓] Chrome - develop for the web
[✗] Visual Studio - develop for Windows
    ✗ Visual Studio not installed; this is necessary for Windows development.
      Download at https://visualstudio.microsoft.com/downloads/.
      Please install the "Desktop development with C++" workload, including all of its default components
[!] Android Studio (not installed)
[✓] IntelliJ IDEA Ultimate Edition (version 2019.2)
[✓] VS Code (version 1.64.2)
[✓] Connected device (3 available)
[✓] HTTP Host Availability

! Doctor found issues in 2 categories.
F:\android-sdk\cmdline-tools\latest\bin>
```

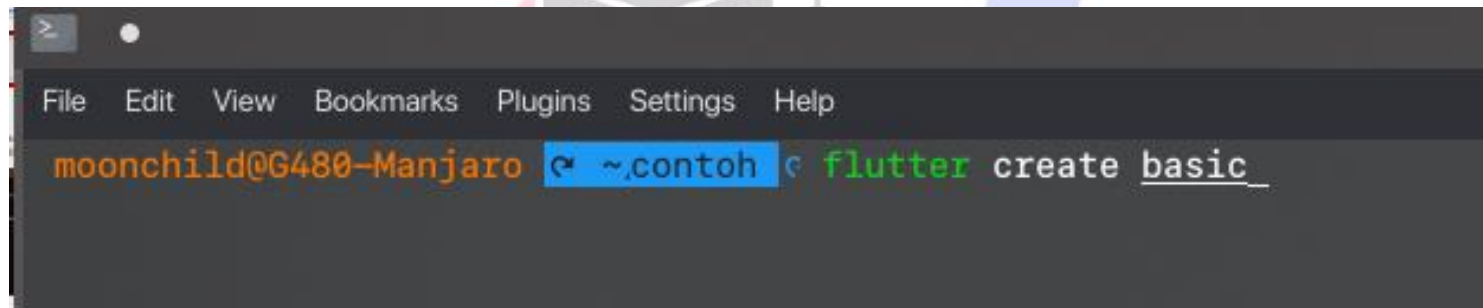
# Optional

- Untuk pengguna Visual Studio Code, disarankan untuk menginstall ekstensi Flutter agar mendapatkan beberapa fitur seperti autocompletion, debug, dll



# Membuat Proyek Baru Flutter

- Buka cmd atau terminal
- Jalankan perintah di bawah ini dan akan tercipta sebuah folder bernama **basic**

A screenshot of a terminal window with a dark background. The window has a menu bar at the top with options: File, Edit, View, Bookmarks, Plugins, Settings, and Help. Below the menu bar, the prompt 'moonchild@G480-Manjaro' is visible, followed by a blue cursor and the text '~.contoh'. The command 'flutter create basic\_' is being typed, with 'flutter' in green and 'create basic\_' in white. The terminal window is titled with a single dot icon and a close button.

- Atau untuk pengguna VS Code, anda bisa membuat proyek baru dengan menekan shortcut **Ctrl+Shift+P**, kemudian ketik **Flutter: New Project**

# Struktur Folder Flutter

```
--project_name
|--android
|--ios
|--lib
|   |--main.dart
|--test
|--web
|--windows
--.gitignore
--.metadata
--.gitignore
--.packages
--analysis_options.yaml
--project_name.iml
--pubspec.lock
--README.md
```



# Struktur Folder Flutter

- **Android:** Folder ini digunakan untuk proses build menjadi aplikasi native ke platform android.
- **IOS:** Memiliki fungsi yang sama seperti folder android yaitu diperlukan untuk menjalankan aplikasi ke platform IOS.
- **lib:** Letak dari file dart aplikasi flutter.

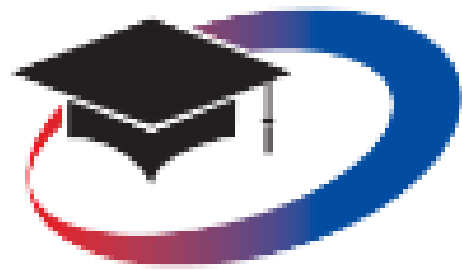
MIKROSKIL



# Struktur Folder Flutter

- **test:** Dapat digunakan untuk kebutuhan testing dari kode yang telah ditulis.
- **assets:** Secara default tidak tersedia. Ini bisa ditambahkan di dalam folder project yang nantinya dapat digunakan untuk menyimpan gambar, font dan file lainnya yang diperlukan dalam pengembangan aplikasi flutter.
- **pubspec.yaml:** Berisi konfigurasi dari proyek flutter yang terdiri dari nama, dekripsi dan versi flutter, dependencies yang digunakan dan inisialisasi assets yang digunakan.

# Widget



UNIVERSITAS  
**MIKROSKIL**

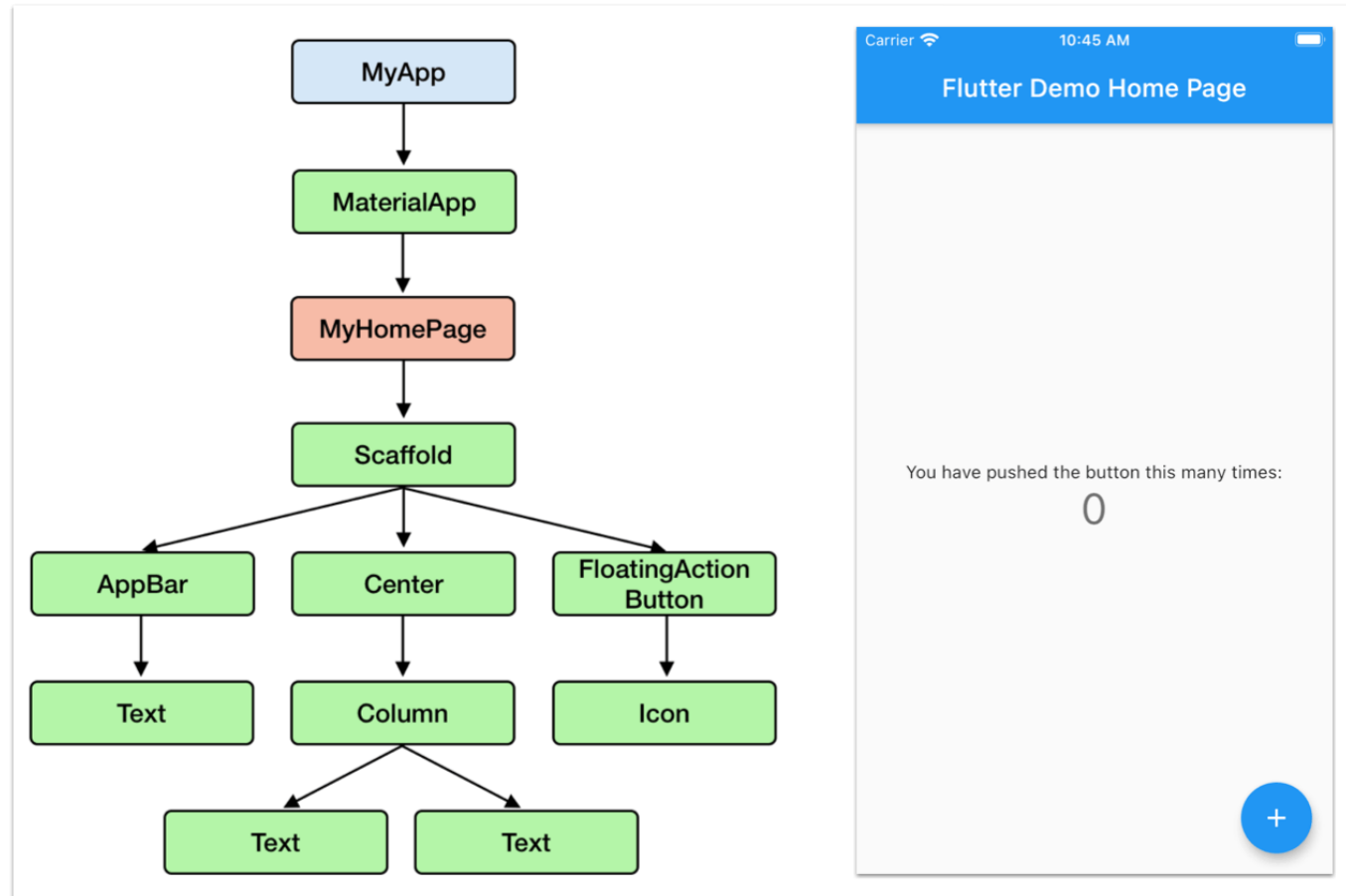
PRODI. TEKNIK INFORMATIKA (S-1)

# Apa itu Widget

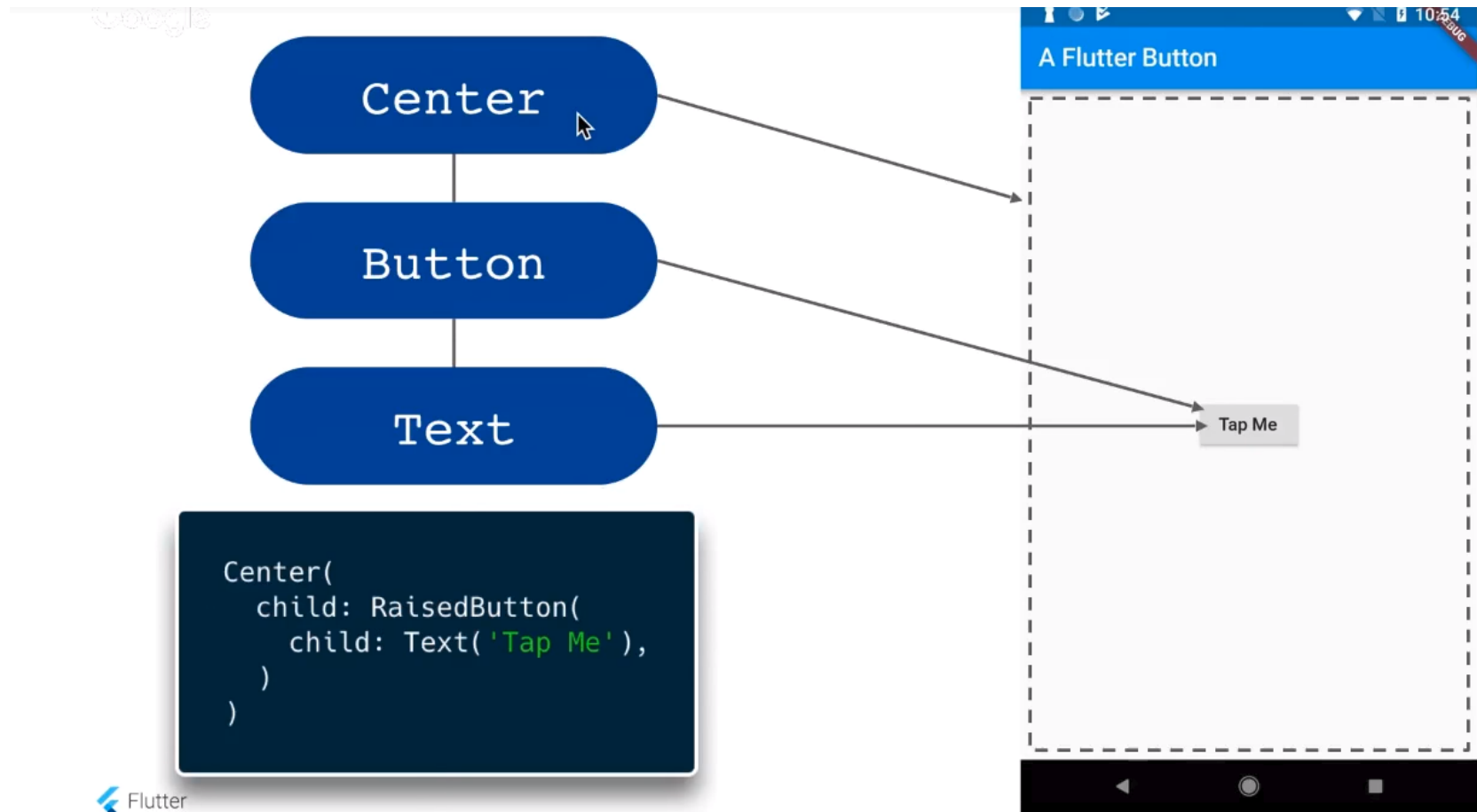
- Komponen User Interface (UI) yang digunakan untuk membuat user interface pada aplikasi
- Widget dapat memiliki 1 (satu) atau lebih children yang terdiri dari widget lainnya.



# Hieraki Widget



# Contoh Widget



# General Widget

- Scaffold
- Stateless
- Stateful
- Text
- Icon
- Image
- Layout
- Container



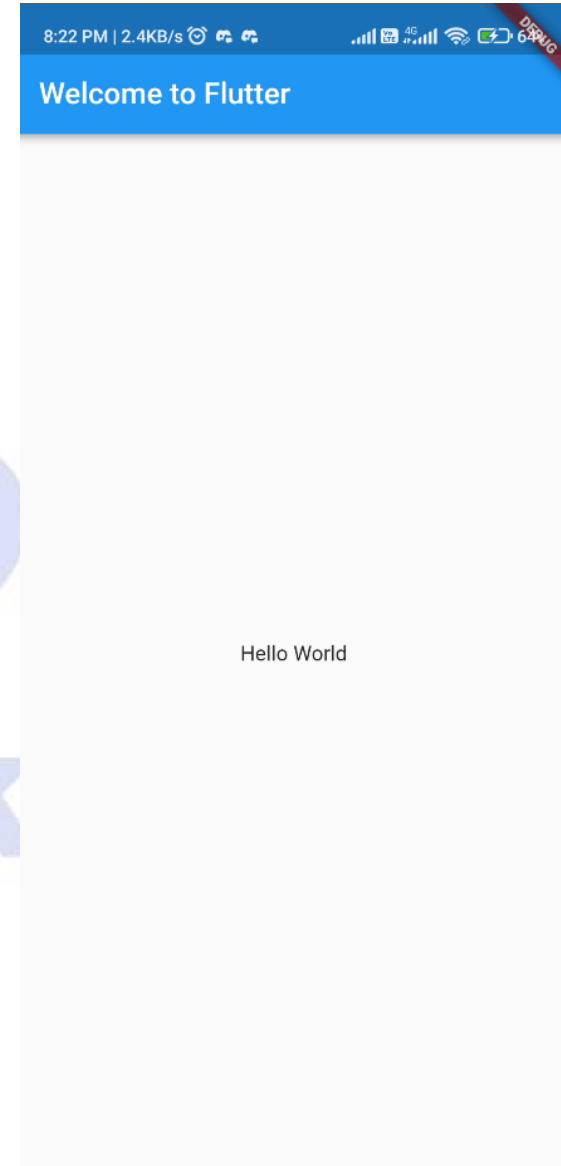
# Scaffold

- Widget utama untuk membuat sebuah halaman flutter
- Berisi beberapa parameter seperti appBar, body, atau floating action button ataupun mengganti warna background bodynya.



# Scaffold

```
home: Scaffold(  
  appBar: AppBar(  
    title: Text('Welcome to Flutter'),  
  ), // AppBar  
  body: Center(  
    child: Text('Hello World'),  
  ), // Center  
), // Scaffold
```





# Stateless

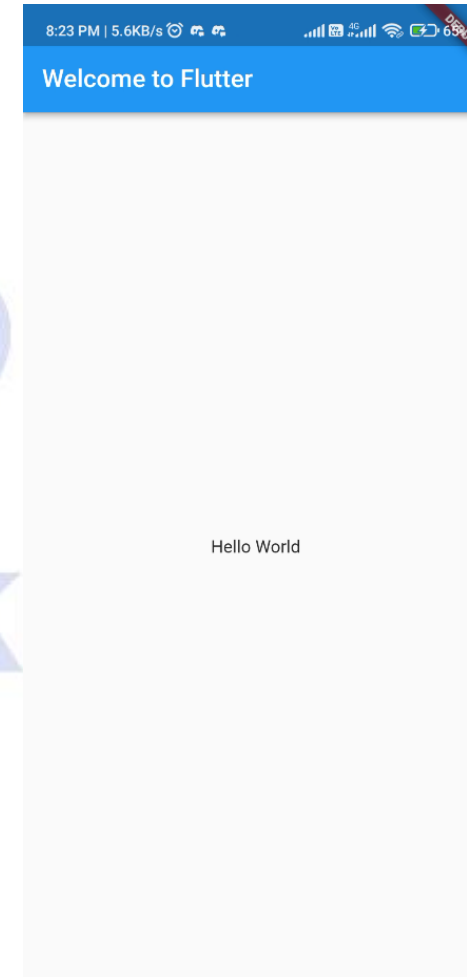
- Widget digunakan ketika halaman dari aplikasi tersebut tidak membutuhkan sesuatu yang dinamis atau nilainya yang berubah-ubah.
- Widget ini diletakkan di kelasnya dengan menggunakan keyword *extends StatelessWidget*



MIKROSKIL

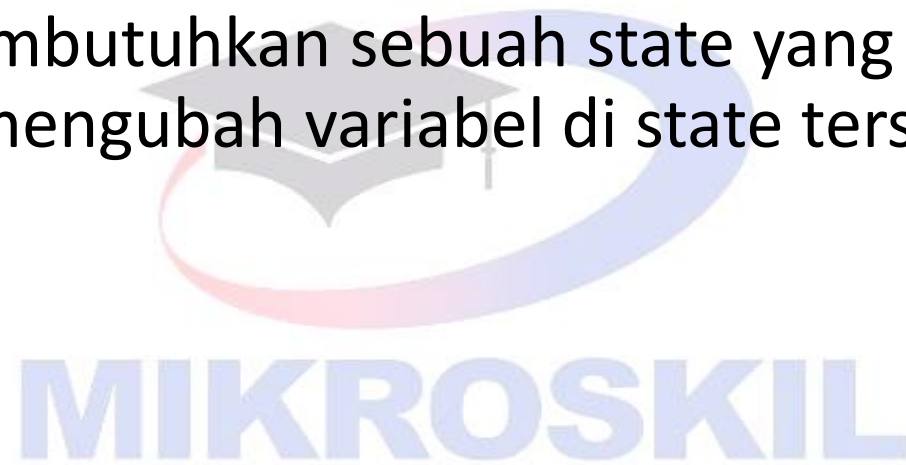
# Contoh Penerapan Stateless

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'First App',  
      home: Scaffold(  
        appBar: AppBar(  
          title: Text('Welcome to Flutter'),  
        ), // AppBar  
        body: Center(  
          child: Text('Hello World'),  
        ), // Center  
      ), // Scaffold  
    ); // MaterialApp  
  }  
}
```



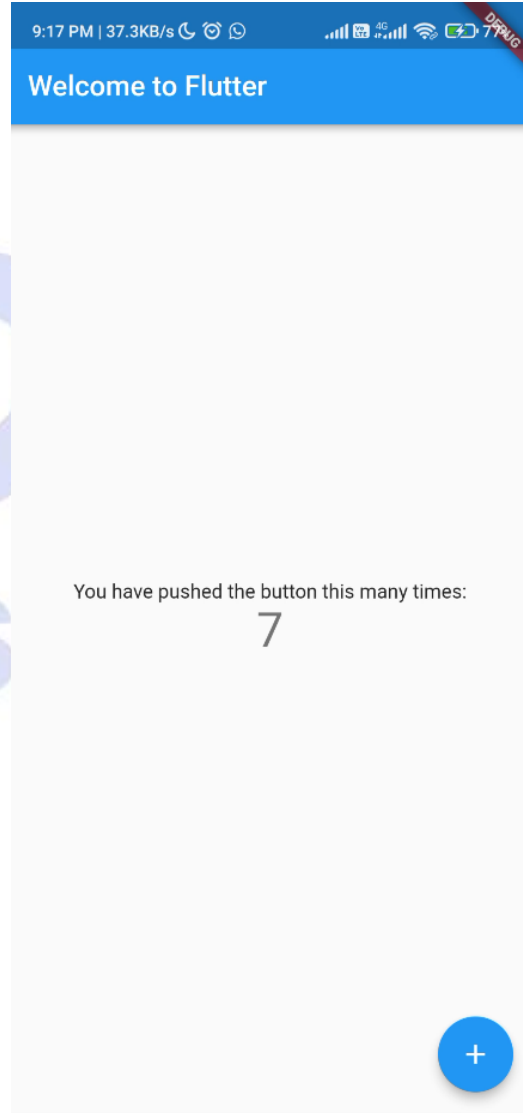
# Stateful

- Widget digunakan ketika halaman dari aplikasi tersebut interaktif dan dinamis.
- StatefulWidget membutuhkan sebuah state yang berisi data dan juga logic / fungsi yang mengubah variabel di state tersebut.



# Stateful

```
class MyHomePage extends StatefulWidget {  
  const MyHomePage({Key? key, required this.title}) : super(key: key);  
  final String title;  
  
  @override  
  // Fungsi untuk melakukan initialized state  
  State<MyHomePage> createState() => _MyHomePageState();  
}  
  
// Kelas State  
class _MyHomePageState extends State<MyHomePage> {  
  int _counter = 0;  
  
  void _incrementCounter() {  
    setState(() {  
      _counter++;  
    });  
  }  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: Text("Welcome to Flutter"),  
      ), // AppBar  
      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.headline4,  
            ) // Text  
          ], // <Widget>[]  
        ), // Column // Center  
        floatingActionButton: FloatingActionButton(  
          onPressed: _incrementCounter,  
          tooltip: 'Increment',  
          child: const Icon(Icons.add),  
        ), // FloatingActionButton  
      ); // Scaffold  
    )  
  }  
}
```



# Text

- Widget yang digunakan untuk menampilkan text
- Selain itu, widget ini dapat digunakan untuk mengatur style text



Info Detail : <https://api.flutter.dev/flutter/widgets/Text-class.html>

# Contoh Text

```
Text('Hello World'),
```

Hello World

MIKROS

# Properti pada Text

Properti	Fungsi
maxLines	Mengatur batas line/baris yang diizinkan
overflow	Mengatur tampilan teks ketika overflow (melewati maxline)
textAlign	Mengatur align pada teks
style	Mengatur tampilan teks seperti warna, font, size, dan sebagainya

MIKROSKIL

Info Detail : <https://api.flutter.dev/flutter/widgets/Text-class.html>

# TextStyle

- Digunakan untuk mengatur style dari text.
- Properti pada TextStyle adalah sebagai berikut

Properti	Fungsi
fontWeight	Menentukan ketebalan dari teks, seperti bold, dan sebagainya
fontStyle	Menentukan style dari teks, seperti italic.
color	Mengatur warna pada teks
fontSize	Mengatur ukuran teks
fontFamily	Mengatur jenis font pada teks

Info Detail : <https://api.flutter.dev/flutter/painting/TextStyle-class.html>



# Contoh Penggunaan TextStyle

```
body: Center(  
  child: Text(  
    'Hello World',  
    style: TextStyle(  
      fontWeight: FontWeight.bold,  
      fontSize: 50,  
      fontStyle: FontStyle.italic,  
      color: Colors.blue), // TextStyle  
  ), // Text  
) // Center
```

***Hello World***

# TextAlign

- Digunakan untuk mengatur posisi teks
- Properti yang tersedia pada TextAlign adalah sebagai berikut.



Properti	Fungsi
TextAlign.left	Teks rata kiri
TextAlign.center	Teks rata tengah
TextAlign.right	Text rata kanan
TextAlign.justify	Text rata kiri dan kanan

Info Detail : <https://api.flutter.dev/flutter/dart-ui/TextAlign.html>

# Contoh Penggunaan TextAlign

```
body: Center(  
  child: Text(  
    'The isolated Pacific island of Mikumo has  
    no internet nor cell phones,  
    not even any convenience stores or family  
    restaurants.',  
    style: TextStyle(fontSize: 20, color:  
      Colors.blue),  
    textAlign: TextAlign.right,  
  ), // Text  
), // Center // Scaffold
```

The isolated Pacific island of Mikumo has  
no internet nor cell phones, not even any  
convenience stores or family restaurants.

# Icon

- Tampilan suatu objek atau gambar yang merepresentasi dari yang dilambangkan tersebut.
- Flutter memiliki 2 jenis Icon yang include didalamnya, yaitu Material Icon (dengan memanggil kelas **Icons**) dan Cupertino (dengan memanggil kelas **CupertinoIcons**)

MIKROSKIL

Info Detail Widget Icon : <https://api.flutter.dev/flutter/widgets/Icon-class.html>

Info Detail Material Icon : <https://material.io/resources/icons/>

Info Detail Cupertino Icon : [https://flutter.github.io/cupertino\\_icons/](https://flutter.github.io/cupertino_icons/)

# Properti pada Icon

Properti	Fungsi
size	Menentukan size dari icon
color	Menentukan warna dari icon
semanticLabel	Berisi teks yang dipakai oleh TalkBack/VoiceOver pada fitur Accesibility dan tidak ditampilkan pada aplikasi

MIKROSKIL

# Contoh Penggunaan Icon

```
children: const <Widget>[  
  Icon(  
    Icons.favorite, // Icon dari Material  
    color: Colors.red,  
    size: 24,  
    semanticLabel: 'Tombol Like', // Tidak ditampilkan di app  
  ), // Icon  
  Icon(  
    CupertinoIcons.home, // Icon dari Cupertino  
    size: 24,  
    color: Colors.blue,  
  ) // Icon  
, // <Widget>[]  
], // <Widget>[]
```



# Images

- Sebuah widget yang berfungsi untuk menampilkan gambar
- Image memiliki 4 opsi berdasarkan sumber gambar, yaitu
  - **Image.network** (Untuk gambar yang berasal dari Internet)
  - **Image.asset** (Untuk gambar yang berasal dari folder asset yang sudah di daftarkan di file **pubspec.yaml**)
  - **Image.file** (Untuk gambar yang berasal dari local storage perangkat pengguna)
  - **Image.memory** (Untuk gambar yang berasal dari byte / blob)
- Pada slide ini akan membahas **Image.network** dan **Image.asset**

Info Detail :

<https://api.flutter.dev/flutter/widgets/Image-class.html>

<https://wisnuwiry.space/post/flutter-basic-images-and-icons/>

# Properti pada Image

Properti	Fungsi
width	Mengatur lebar dari gambar
height	Mengatur tinggi dari gambar
color	Jika tidak kosong maka akan memaksa/blend gambar tersebut menjadi satu warna parameter tersebut
fit	Untuk mengatur fit pada gambar tersebut misal cover, contain, fill, dll
alignment	Untuk mengatur alignment/posisi rata-rata pada gambar tersebut
filterQuality	Untuk mengatur kualitas gambar pada saat rendering widget



# Image.network

```
// Pastikan URL yang dipakai aktif
Image.network(
  'https://flutter.github.io/assets-for-api-docs/assets/widgets/owl-2.jpg',
  width: 300 // Set Width nya jadi 300
), // Image.network
```

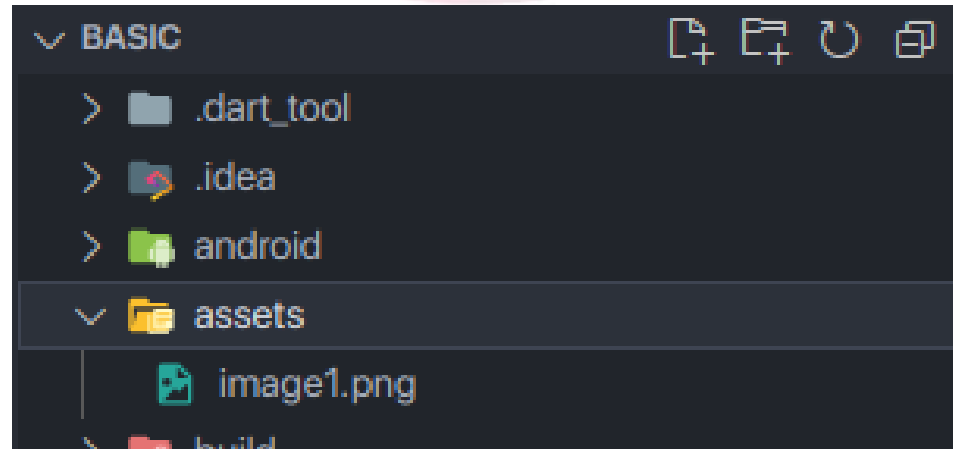
MIKROSKI

Welcome to Flutter



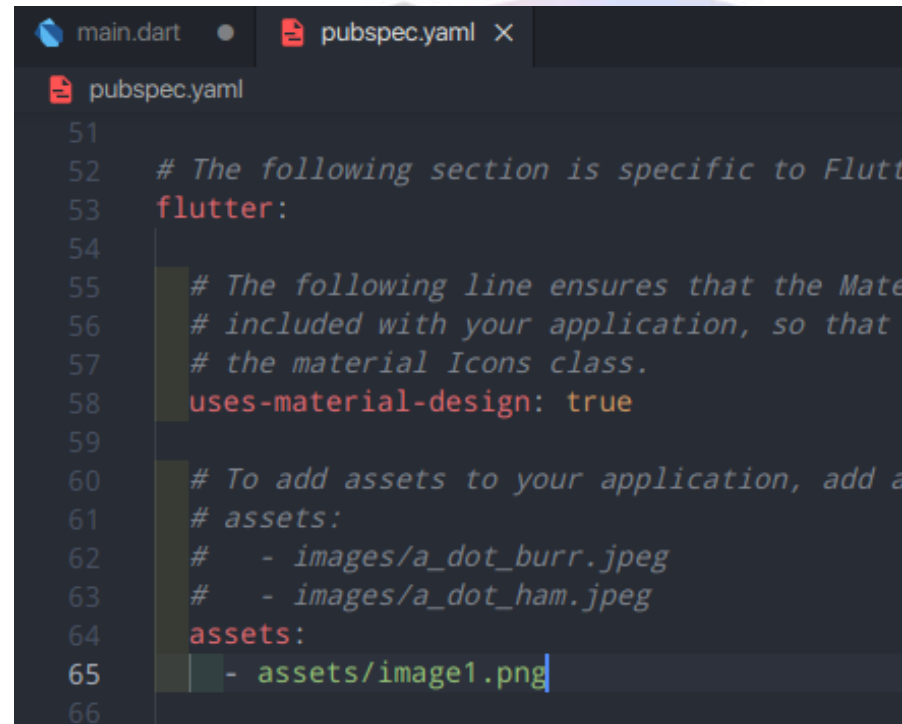
# Image.asset

- Image.asset cocok dipake jika image tersebut dipakai secara privat dan hanya ada di app tersebut.
- Hal yang pertama dilakukan adalah membuat folder **assets** di root project anda dan isikan dengan file gambar yang ingin di daftarkan ke app.



# Image.asset

- Kemudian daftarkan gambar tersebut ke file **pubspec.yaml**

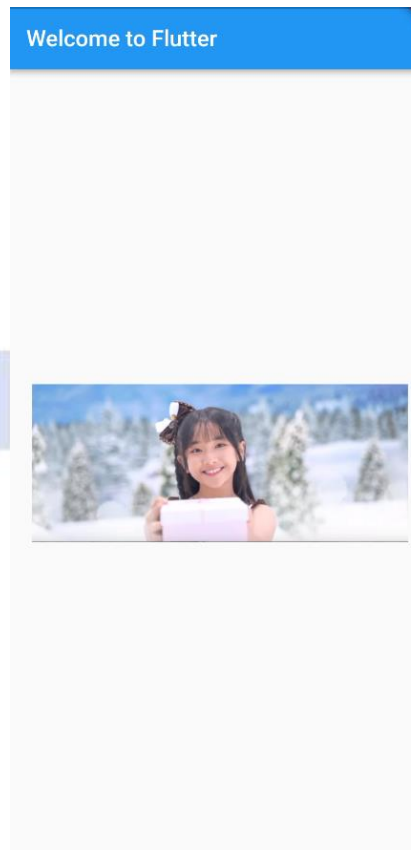


```
main.dart  pubspec.yaml X
pubspec.yaml
51
52 # The following section is specific to Flutter.
53 flutter:
54
55 # The following line ensures that the Material Icons
56 # are included with your application, so that
57 # you can use the Material Icons class.
58 uses-material-design: true
59
60 # To add assets to your application, add an assets
61 # section, like this:
62 # - images/a_dot_burr.jpeg
63 # - images/a_dot_ham.jpeg
64 assets:
65 - assets/image1.png
66
```

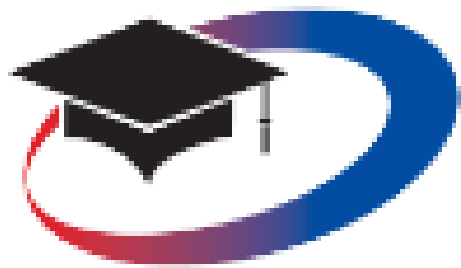
# Image.asset

- Kemudian tinggal implementkan pada file main.dartnya seperti di bawah ini.

```
// Pastikan URL yang dipakai aktif  
Image.asset('assets/image1.png',  
  width: 350 // Set Width nya jadi 350  
) // Image.asset
```



# Terima Kasih



UNIVERSITAS  
**MIKROSKIL**

PRODI. TEKNIK INFORMATIKA (S-1)