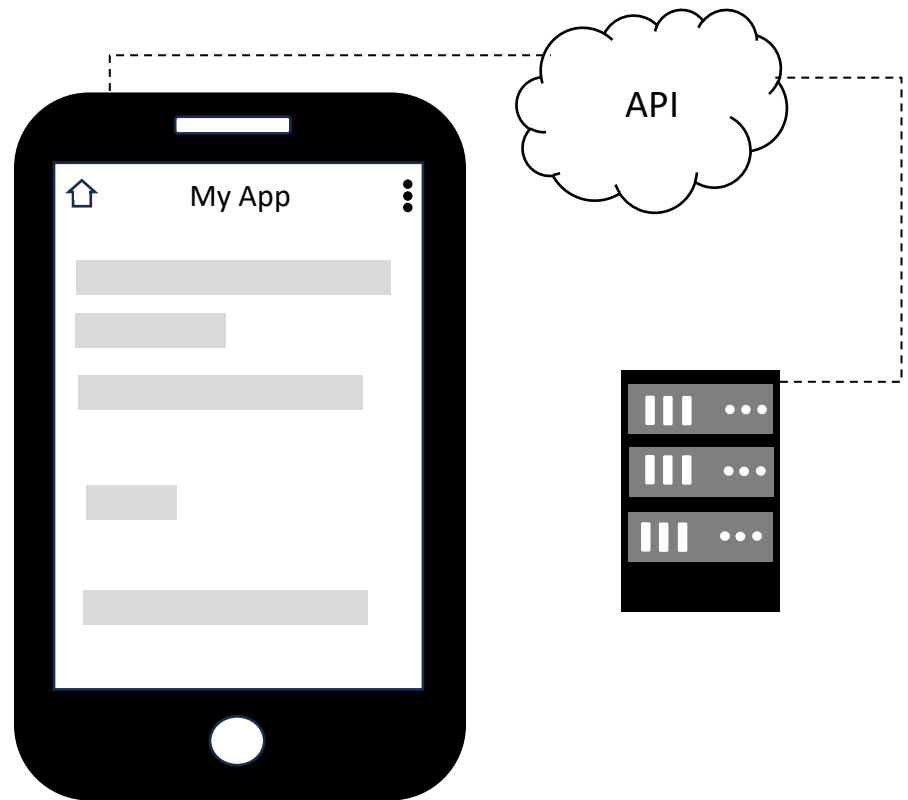


MOBILE & WEB SERVICES : WIDGET



Nama : Nur Fairus Fatin Fadhilah

NPM : 5220411344

WIDGET

MATERIAL APP

Sebuah aplikasi yang menggunakan Desain Material. Widget kemudahan yang merangkum sejumlah widget yang biasa dibutuhkan untuk aplikasi Desain Material. Itu dibangun di atas WidgetsApp dengan menambahkan fungsionalitas khusus desain material, seperti AnimatedTheme dan GridPaper.

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

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp();
  }
}
```

Hasil



SCAFFOLD

Mengimplementasikan struktur tata letak visual Desain Material dasar. Kelas ini menyediakan API untuk menampilkan drawers dan BottomSheet. Untuk menampilkan BottomSheet yang persisten, dapatkan ScaffoldState pada BuildContext saat ini melalui Scaffold.of dan gunakan fungsi ScaffoldState.showBottomSheet.

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

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      home: Scaffold(),
    );
  }
}
```

Hasil



MENGHILANGKAN TULISAN DEBUG

Mengaktifkan spanduk kecil "DEBUG" dalam mode debug untuk menunjukkan bahwa aplikasi berada dalam mode debug. Ini aktif secara default (dalam mode debug), untuk mematikannya, setel argumen konstruktor ke false. Dalam mode rilis, ini tidak berpengaruh. Untuk mendapatkan spanduk ini di aplikasi Anda jika Anda tidak menggunakan WidgetsApp, sertakan widget CheckedModeBanner di aplikasi Anda. Spanduk ini dimaksudkan untuk mencegah orang mengeluh bahwa aplikasi Anda lambat saat berada dalam mode debug. Dalam mode debug, Flutter mengaktifkan sejumlah besar diagnostik mahal untuk membantu pengembangan, sehingga performa dalam mode debug tidak mewakili apa yang akan terjadi dalam mode rilis.

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

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(),
    );
  }
}
```

Hasil



APP BAR AND APP BAR TITLE

Bilah aplikasi Desain Material. Bilah aplikasi terdiri dari toolbar dan kemungkinan widget lainnya, seperti TabBar dan FleksibelSpaceBar. Bilah aplikasi biasanya menampilkan satu atau beberapa tindakan umum dengan IconButton yang secara opsional diikuti oleh PopupMenuButton untuk operasi yang kurang umum (terkadang disebut "menu tambahan"). Bilah aplikasi biasanya digunakan di properti Scaffold.appBar, yang menempatkan bilah aplikasi sebagai widget dengan tinggi tetap di bagian atas layar. Untuk bilah aplikasi yang dapat digulir, lihat SliverAppBar, yang menyematkan AppBar dalam sliver untuk digunakan dalam CustomScrollView. AppBar menampilkan widget toolbar, awalan, judul, dan tindakan, di atas bagian bawah (jika ada). Bagian bawah biasanya digunakan untuk TabBar. Jika widget FleksibelSpace ditentukan maka widget tersebut ditumpuk di belakang toolbar dan widget bawah. Diagram berikut menunjukkan lokasi masing-masing slot ini muncul di toolbar ketika bahasa penulisannya dari kiri ke kanan (misalnya bahasa Inggris):

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

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: Text('App Title'),
        ),
      ),
    );
  }
}
```

Hasil :



BODY

Konten utama perancah. Ditampilkan di bawah appBar, di atas bagian bawah MediaQueryData.viewInsets ambient MediaQuery, dan di belakang FloatingActionButton dan laci. Jika resizeModeAvoidBottomInset salah maka ukuran badan tidak diubah saat keyboard di layar muncul, yaitu tidak disisipkan oleh viewInsets.bottom. Widget di badan scaffold diposisikan di kiri atas ruang yang tersedia antara bilah aplikasi dan bagian bawah scaffold. Untuk memusatkan widget ini, pertimbangkan untuk meletakkannya di widget Tengah dan menjadikannya sebagai badan. Untuk memperluas widget ini, pertimbangkan untuk meletakkannya di SizedBox.expand. Jika Anda memiliki kolom widget yang biasanya muat di layar, namun mungkin meluap dan dalam kasus seperti itu perlu digulir, pertimbangkan untuk menggunakan ListView sebagai badan perancah. Ini juga merupakan pilihan yang baik jika isi Anda adalah daftar yang dapat digulir.

Kode :

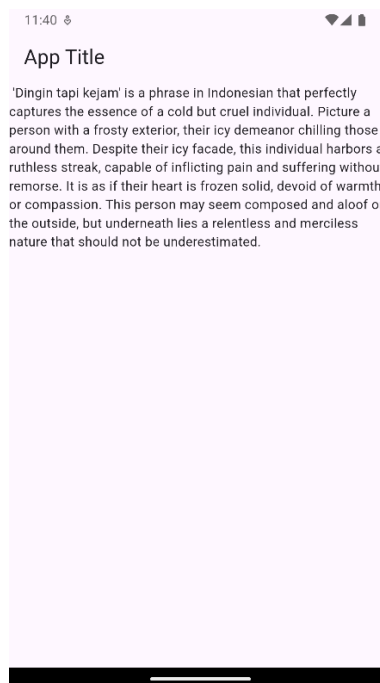
```
import 'package:flutter/material.dart';

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: Text('App Title'),
        ),
        body: Text("'Dingin tapi kejam' is a phrase in Indonesian that perfectly captures the essence of a cold but cruel individual. Picture a person with a frosty exterior, their icy demeanor chilling those around them. Despite their icy facade, this individual harbors a ruthless streak, capable of inflicting pain and suffering without remorse. It is as if their heart is frozen solid, devoid of warmth or compassion. This person may seem composed and aloof on the outside, but underneath lies a relentless and merciless nature that should not be underestimated. "),
      ),
    );
  }
}
```

Hasil :



CENTER

Sebuah widget yang memusatkan anaknya di dalam dirinya sendiri. Widget ini akan menjadi sebesar mungkin jika dimensinya dibatasi dan widthFactor serta heightFactor bernilai nol. Jika suatu dimensi tidak dibatasi dan faktor ukuran terkait adalah nol, maka widget akan cocok dengan ukuran turunannya dalam dimensi tersebut. Jika faktor ukuran bukan nol, maka dimensi yang sesuai pada widget ini akan menjadi hasil kali dimensi anak dan faktor ukuran. Misalnya jika widthFactor adalah 2.0 maka lebar widget ini akan selalu dua kali lebar anaknya.

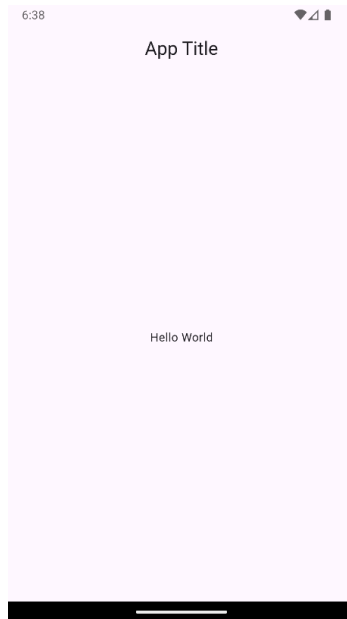
```
import 'package:flutter/material.dart';

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: Text('App Title'),
          centerTitle: true,
        ),
        body: Center(
          child: Text("Hello World"),
        ),
      ),
    );
  }
}
```


Hasil :



FLOATING ACTION BUTTON

Tombol aksi mengambang Desain Material. Tombol tindakan mengambang adalah tombol ikon melingkar yang melayang di atas konten untuk mendorong tindakan utama dalam aplikasi. Tombol tindakan mengambang paling umum digunakan di bidang Scaffold.floatingActionButton. Gunakan paling banyak satu tombol tindakan mengambang per layar. Tombol tindakan mengambang sebaiknya digunakan untuk tindakan positif seperti "buat", "bagikan", atau "navigasi". (Jika lebih dari satu tombol tindakan mengambang digunakan dalam suatu Rute, pastikan bahwa setiap tombol memiliki heroTag unik, jika tidak, pengecualian akan diberikan.) Jika panggilan balik onPressed adalah null, maka tombol tersebut akan dinonaktifkan dan tidak akan bereaksi terhadap sentuhan. Sangat tidak disarankan untuk menonaktifkan tombol tindakan mengambang karena tidak ada indikasi kepada pengguna bahwa tombol tersebut dinonaktifkan. Pertimbangkan untuk mengubah backgroundColor jika menonaktifkan tombol tindakan mengambang.

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

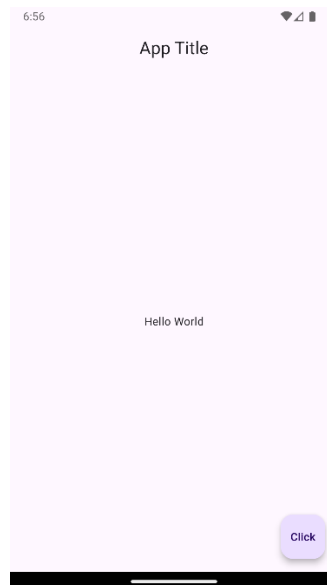
void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
```

```
debugShowCheckedModeBanner: false,  
home: Scaffold(  
  appBar: AppBar(  
    title: Text('App Title'),  
    centerTitle: true,  
  ),  
  body: Center(  
    child: Text("Hello World"),  
  ),  
  floatingActionButton: FloatingActionButton(  
    onPressed: () {},  
    child: Text("Click")),  
),  
);  
}
```

Hasil



COLOR

Konstanta Warna dan ColorSwatch yang mewakili palet warna desain Material. Daripada menggunakan warna absolut dari palet ini, pertimbangkan untuk menggunakan Theme.of untuk mendapatkan ThemeData.colorScheme lokal, yang menentukan warna yang digunakan sebagian besar komponen Material secara default. Kebanyakan contoh memiliki warna dari 100 hingga 900 dengan kelipatan seratus, ditambah warna 50. Semakin kecil angkanya, semakin pucat warnanya. Semakin besar angkanya, semakin gelap warnanya. Contoh aksent (misalnya redAccent) hanya memiliki nilai 100, 200, 400, dan 700. Selain itu, serangkaian warna hitam dan putih dengan kekeruhan umum juga tersedia. Misalnya, black54 adalah hitam murni dengan opacity 54%.

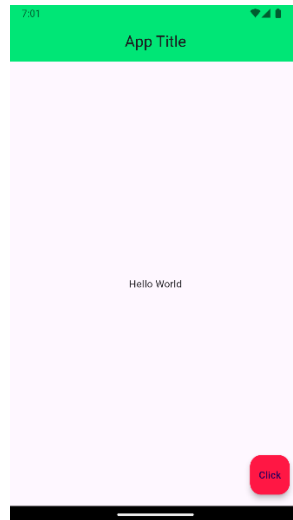
```
import 'package:flutter/material.dart';

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: Text('App Title'),
          centerTitle: true,
          backgroundColor: Colors.greenAccent[400],
        ),
        body: Center(
          child: Text("Hello World"),
        ),
        floatingActionButton: FloatingActionButton(
          onPressed: () {},
          child: Text("Click"),
          backgroundColor: Colors.redAccent[400],
        ),
      ),
    );
  }
}
```

Hasil



TEXT STYLES

Gaya abadi yang menjelaskan cara memformat dan melukis teks. Style yang disediakan adalah background, backgroundColor, color, debugLabel, decoration, decorationColor, decorationStyle, decorationThickness, fontFamily, fontFamilyFallback, fontFeatures, fontSize, fontStyle, fontVariations, fontWeight, foreground, hashCode, height, inherit, leadingDistribution, letterSpacing, locale, overflow, runtimeType, shadows, textBaseline, wordSpacing

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

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: Text('App Title'),
          centerTitle: true,
          backgroundColor: Colors.greenAccent[400],
        ),
        body: Center(
          child: Text("Hello World",
```

```
        style: TextStyle(
          fontSize: 22.5,
          fontWeight: FontWeight.bold,
          letterSpacing: 2.0,
          color: Colors.purple[400],
        )),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () {},
        child: Text("Click"),
        backgroundColor: Colors.redAccent[400],
      ),
    ),
  );
}
```

Hasil

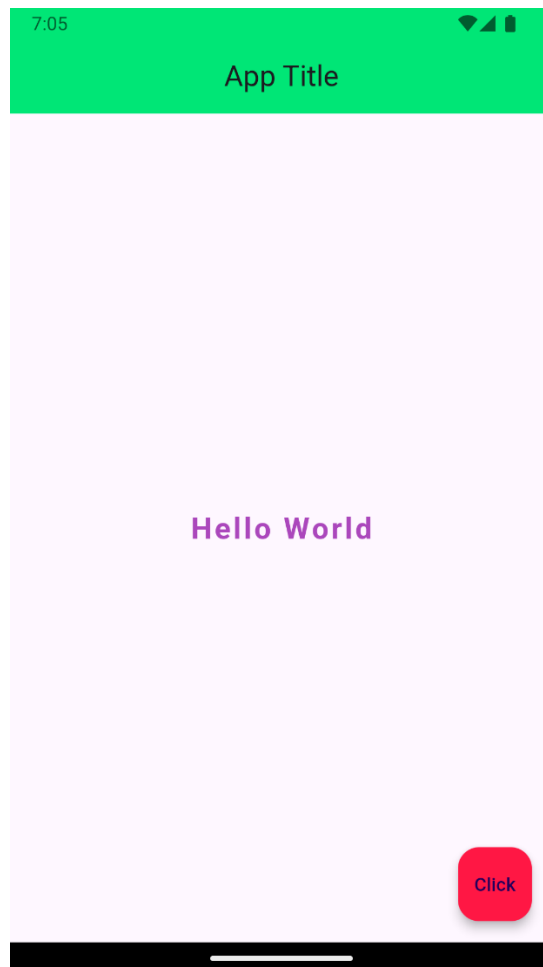


IMAGE (Network Image)

Menampilkan gambar merupakan hal mendasar bagi sebagian besar aplikasi seluler. Flutter menyediakan widget Gambar untuk menampilkan berbagai jenis gambar.

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

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: Text('App Title'),
          centerTitle: true,
          backgroundColor: Colors.greenAccent[400],
        ),
        body: Center(
          child: Image(
            image: NetworkImage(
              "https://images.unsplash.com/photo-1486010586814-abd061e90cf9?q=80&w=3087&auto=format&fit=crop&ixlib=rb-4.0.3&ixid=M3wxMjA3fDB8MHxwaG90by1wYWdlfHx8fGVuZDB8fHx8fA%3D%3D")),
            floatingActionButton: FloatingActionButton(
              onPressed: () {},
              child: Text("Click"),
              backgroundColor: Colors.redAccent[400],
            ),
          ),
        ),
      );
  }
}
```

Hasil :

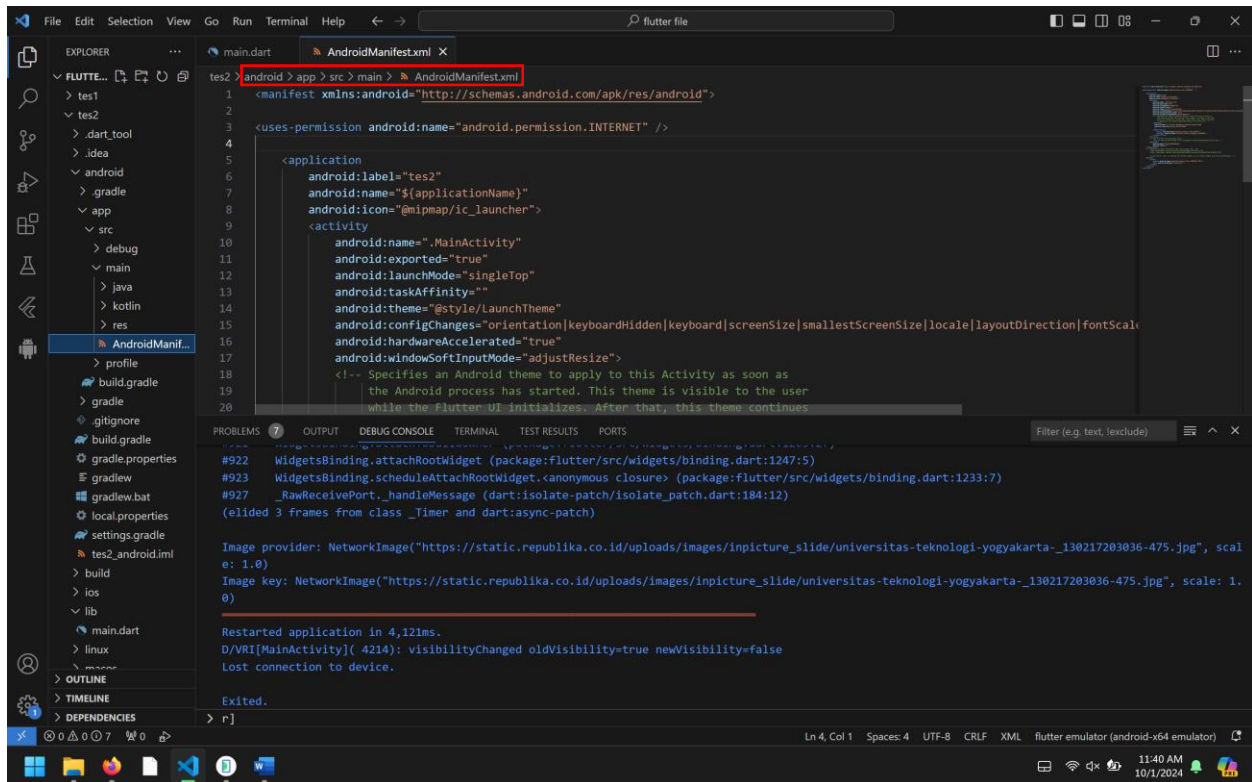


Jika error seperti di bawah,

SocketException (SocketException: Failed host lookup: 'images.unsplash.com' (OS Error: No address associated with hostname, errno = 7))

Android emulator harus diberi internet terlebih dahulu dengan cara berikut,

1. Pergi ke file flutter → android → src → main → AndroidManifest.xml
2. Tambahkan perintah `<uses-permission android:name="android.permission.INTERNET" />`

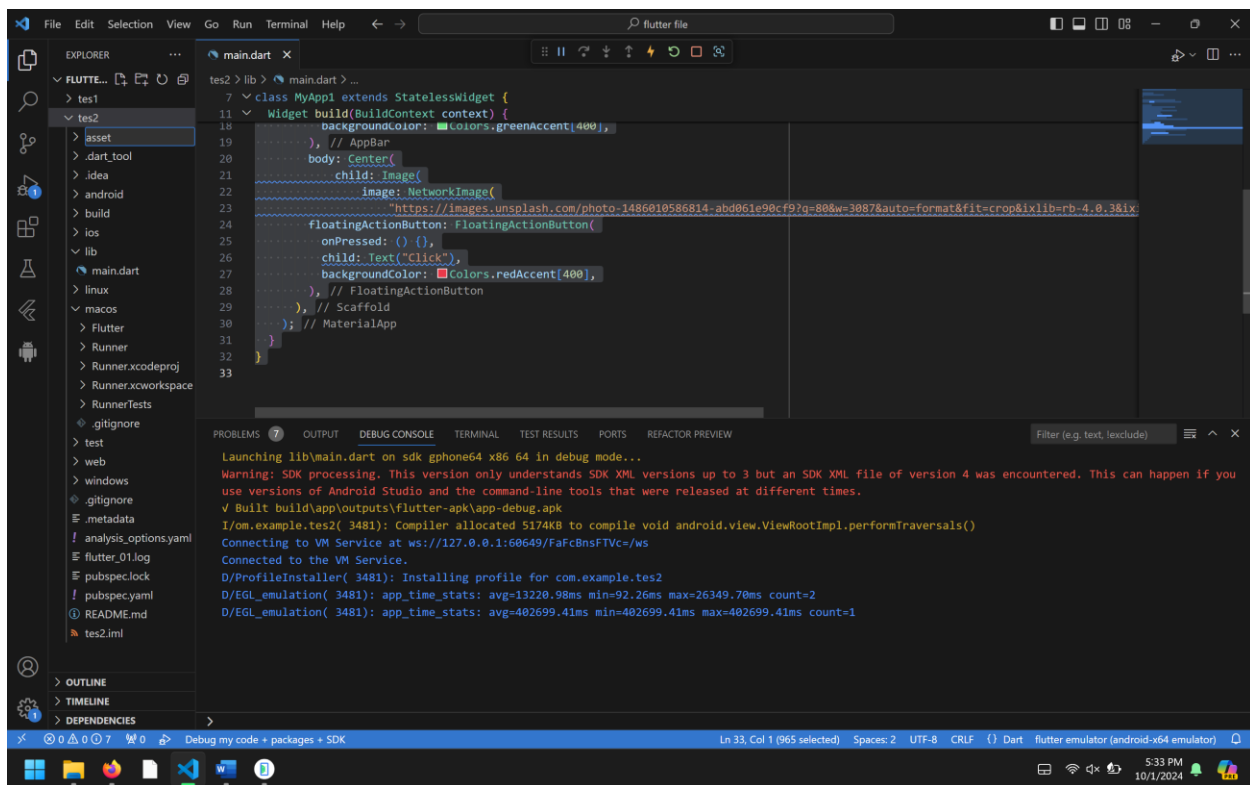


3. Booting ulang android emulator

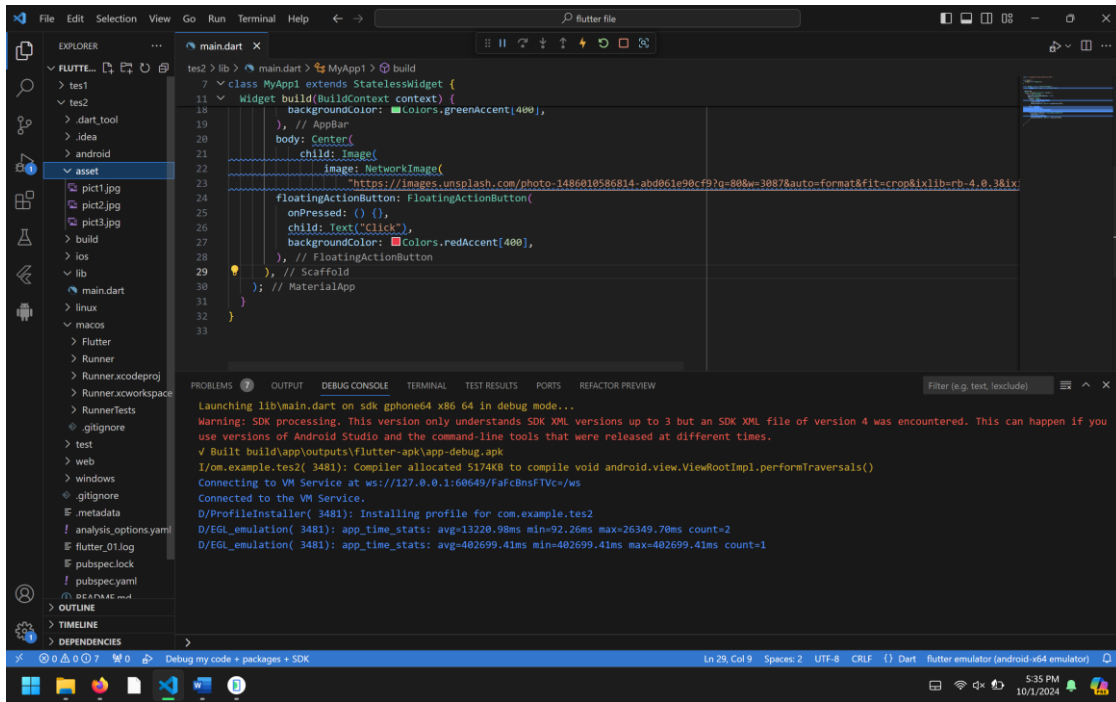
IMAGE (Image Asset)

Aplikasi Flutter dapat menyertakan kode dan aset (terkadang disebut sumber daya). Aset adalah file yang dipaketkan dan disebarkan dengan aplikasi Anda, dan dapat diakses saat runtime. Jenis aset yang umum mencakup data statis (misalnya, file JSON), file konfigurasi, ikon, dan gambar (JPEG, WebP, GIF, WebP/GIF animasi, PNG, BMP, dan WBMP). Flutter menggunakan file `pubspec.yaml`, yang terletak di root proyek Anda, untuk mengidentifikasi aset yang diperlukan oleh suatu aplikasi.

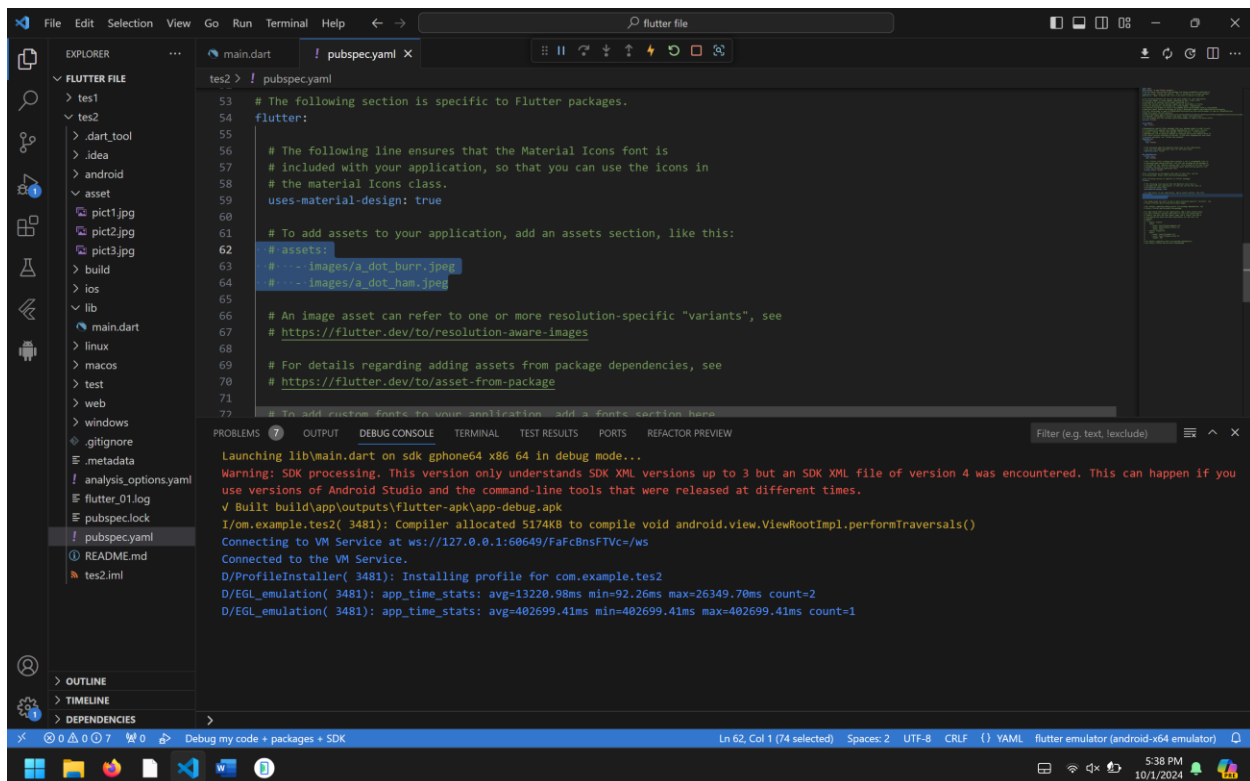
1. Buat directory di root bebas Namanya (contoh : asset)



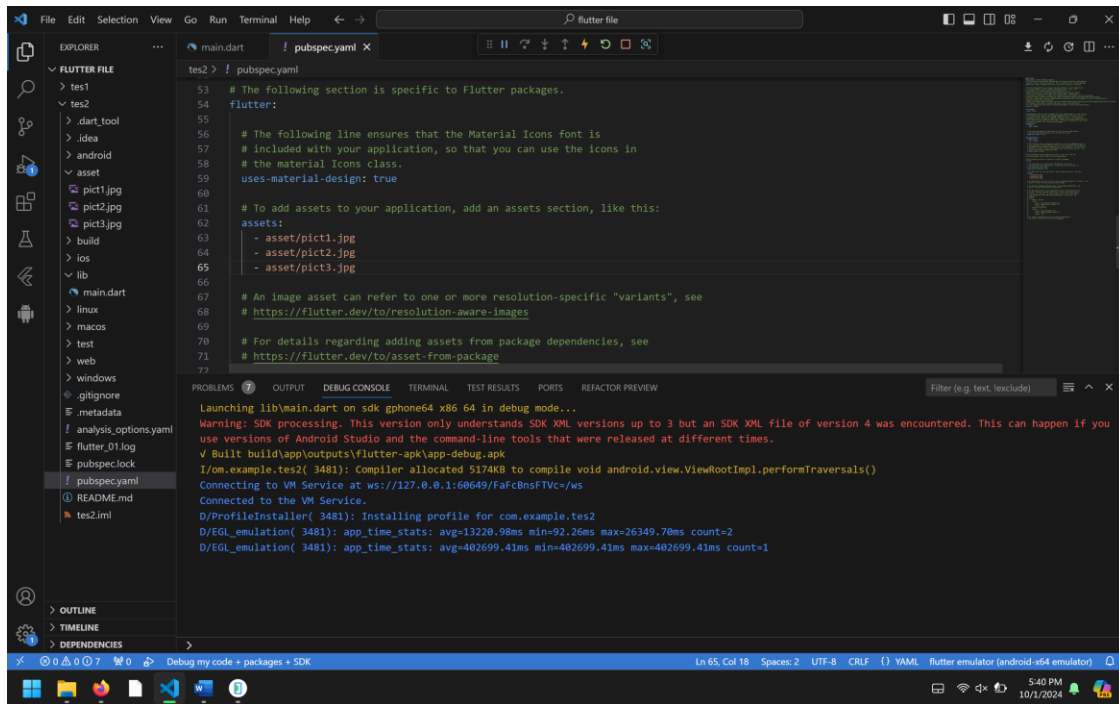
2. Pindahkan file gambar ke directory tersebut



3. Lalu pergi ke pubspec.yaml dan cari kata asset,



4. Ubah ke dalam bentuk perintah dan tambahkan path file gambar di bawah assets seperti berikut,



Kode :

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

void main() {
  runApp(const MyApp1());
}

class MyApp1 extends StatelessWidget {
  const MyApp1({Key? key}) : super(key: key);

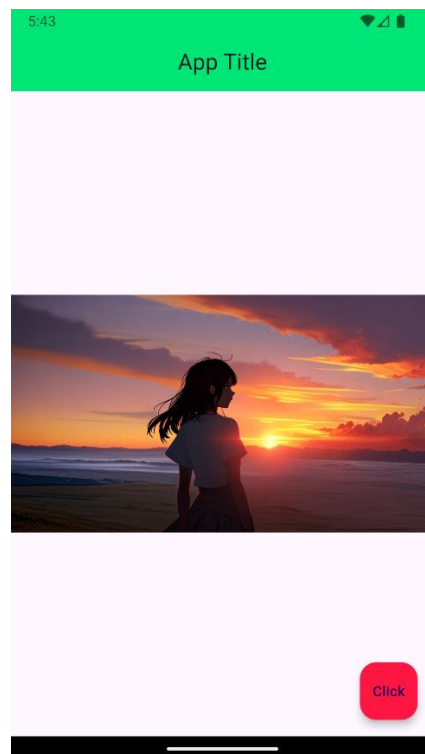
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        appBar: AppBar(
          title: Text('App Title'),
          centerTitle: true,
          backgroundColor: Colors.greenAccent[400],
        ),
        body: Center(
          child: Image(
```

```

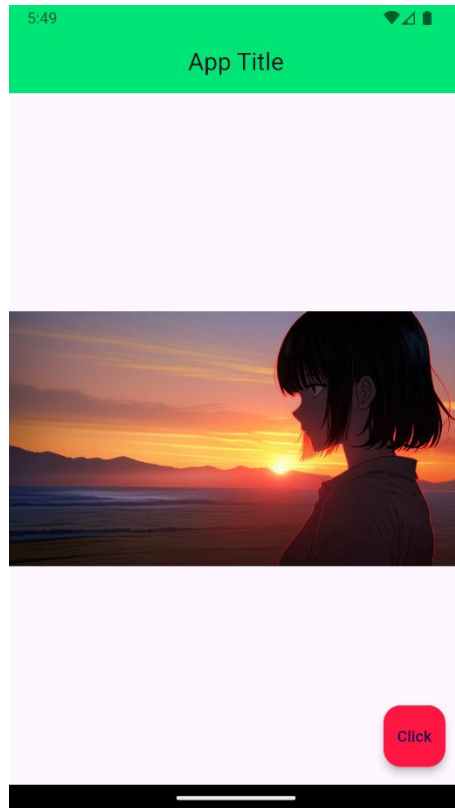
        image: AssetImage('asset/pict1.jpg'))),
floatingActionButton: FloatingActionButton(
  onPressed: () {},
  child: Text("Click"),
  backgroundColor: Colors.redAccent[400],
),
),
);
}
}

```

Hasil :



Misal pict2



Misal pict3

5:50



App Title



Click