PRAKTIKUM PEMROGRAMAN PERANGKAT BERGERAK

GUIDED & UNGUIDED

MODUL IX

API PERANGKAT KERAS

Disusun Oleh :

Maria Nathasya Desfera Pangestu / 2211104008

SE0601

Asisten Praktikum :

Muhammad Faza Zulian Gesit Al Barru

Aisyah Hasna Aulia

Dosen Pengampu :

Yudha Islami Sulistya, S.Kom., M.Cs.

PROGRAM STUDI S1 SOFTWARE ENGINEERING

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

GUIDED

1. Source code Image Picker

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

class ImageFromGalleryEx extends StatefulWidget {
  final ImageSourceType type;
  ImageFromGalleryEx(this.type);

  @override
  ImageFromGalleryExState createState() => ImageFromGalleryExState(this.type);
}

class ImageFromGalleryExState extends State<ImageFromGalleryEx> {
  File? _image;
  late ImagePicker imagePicker;
  final ImageSourceType type;

  ImageFromGalleryExState(this.type);

  @override
  void initState() {
    super.initState();
    imagePicker = ImagePicker();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(type == ImageSourceType.camera
            ? "Image from Camera"
            : "Image from Gallery"),
      ),
      body: Column(
        children: <Widget>[
          SizedBox(height: 52),
          Center(
            //mengambil gambar dari camera atau gallery
            child: GestureDetector(
              onTap: () async {
                //operasi ternary untuk memilih sumber gambar
                var source = type == ImageSourceType.camera
                    ? ImageSource.camera
                    : ImageSource.gallery;

                //menyimpan gambar pada variabel image
                XFile? image = await imagePicker.pickImage(
                    source: source,
                    imageQuality: 50,
                    preferredCameraDevice: CameraDevice.front);

                if (image != null) {
                  setState(() {
                    _image = File(image.path);
                  });
                }
              },
              child: Container(
                width: 200,
                height: 200,
                decoration: BoxDecoration(
                  color: Colors.red[200],
                ),

                // menampilkan gambar dari camera atau gallery
                child: _image != null
                    ? Image.file(
                        _image!,
                        width: 200.0,
                        height: 200.0,
                        fit: BoxFit.fitHeight,
                      )

                    // jika tidak ada gambar yang dipilih
                    : Container(
                        decoration: BoxDecoration(
                          color: Colors.red[200],
                        ),
                        width: 200,
                        height: 200,
                        child: Icon(
                          Icons.camera_alt,
                          color: Colors.grey[800],
                        ),
                      ),
              ),
            ),
          ),
        ],
      ),
    );
  }
}

enum ImageSourceType { camera, gallery }
```

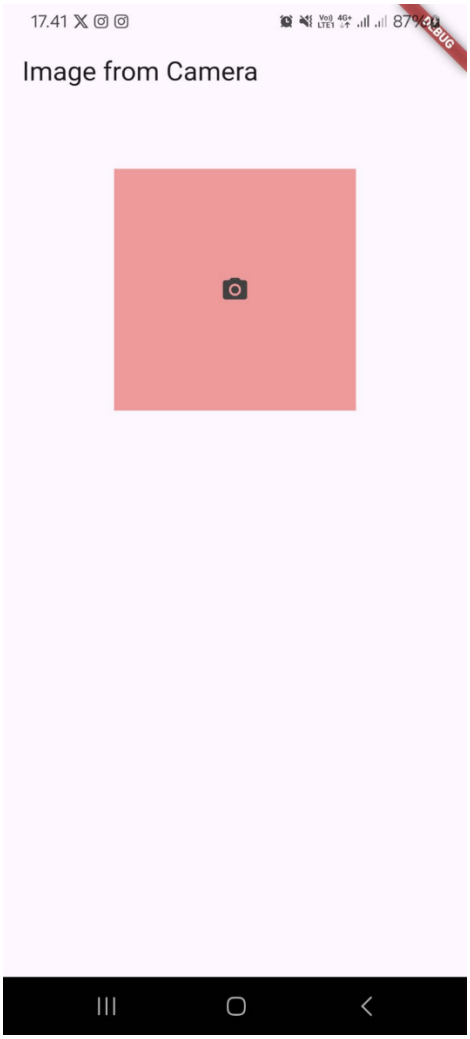
2. Source code Main Dart

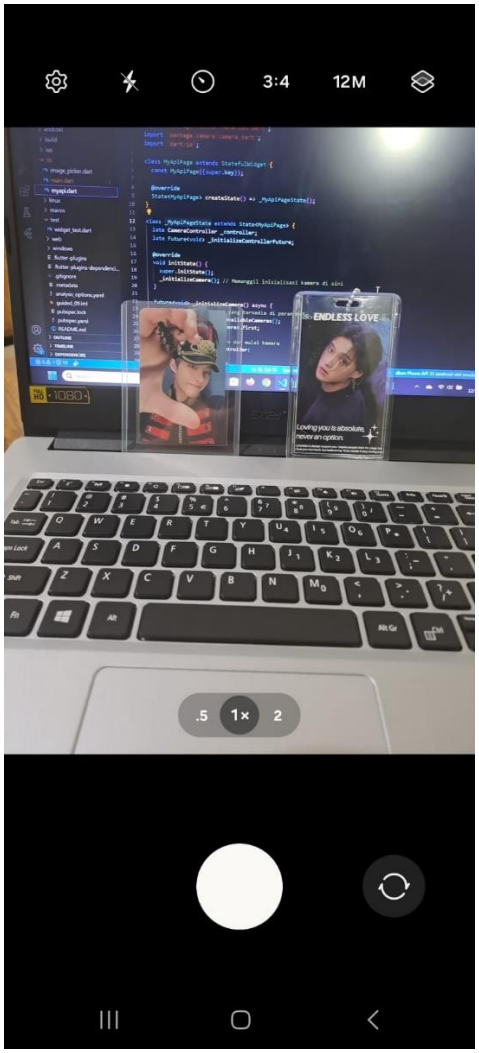
```dart
import 'package:flutter/material.dart';
import 'package:guided_09/image_picker.dart';
import 'package:guided_09/myapi.dart';

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

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

  // This widget is the root of your application.
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Week 9',
      theme: ThemeData(
        colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
        useMaterial3: true,
      ),
      //home: const MyApiPage(),
      home: ImageFromGalleryEx(ImageSourceType.camera),
    );
  }
}
```

3. Source code My API

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

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

  @override
  State<MyApiPage> createState() => _MyApiPageState();
}

class _MyApiPageState extends State<MyApiPage> {
  late CameraController _controller;
  late Future<void> _initializeControllerFuture;

  @override
  void initState() {
    super.initState();
    _initializeCamera(); // Memanggil inisialisasi kamera di sini
  }

  Future<void> _initializeCamera() async {
    // Ambil daftar kamera yang tersedia di perangkat
    final cameras = await availableCameras();
    final firstCamera = cameras.first;

    // Buat kontroler kamera dan mulai kamera
    _controller = CameraController(
      firstCamera,
      ResolutionPreset.high,
    );

    // Inisialisasi kontroler kamera
    _initializeControllerFuture = _controller.initialize();
    setState(() {}); // Pastikan state diperbarui setelah kamera diinisialisasi
  }

  @override
  void dispose() {
    // Bersihkan kontroler ketika widget dihapus
    _controller.dispose();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("API Perangkat Keras"),
        centerTitle: true,
        backgroundColor: Colors.amberAccent,
      ),
      body: FutureBuilder<void>(
        future: _initializeControllerFuture, // Menunggu inisialisasi kamera
        builder: (context, snapshot) {
          if (snapshot.connectionState == ConnectionState.done) {
            // Jika kamera sudah siap, tampilkan pratinjau
            return CameraPreview(_controller);
          } else {
            // Jika kamera belum siap, tampilkan loading
            return Center(child: CircularProgressIndicator());
          }
        },
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: () async {
          try {
            // Pastikan kamera sudah diinisialisasi
            await _initializeControllerFuture;

            // Ambil gambar
            final image = await _controller.takePicture();

            // Tampilkan / gunakan gambar
            Navigator.push(
              context,
              MaterialPageRoute(
                builder: (context) =>
                    DisplayPictureScreen(imagePath: image.path),
              ),
            );
          } catch (e) {
            print(e);
          }
        },
        child: Icon(Icons.camera_alt),
      ),
    );
  }
}

class DisplayPictureScreen extends StatelessWidget {
  final String imagePath;

  const DisplayPictureScreen({Key? key, required this.imagePath})
      : super(key: key);

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: Text('Display Picture')),
      body: Image.file(File(imagePath)),
    );
  }
}
```

4. Hasil output dari guided from camera

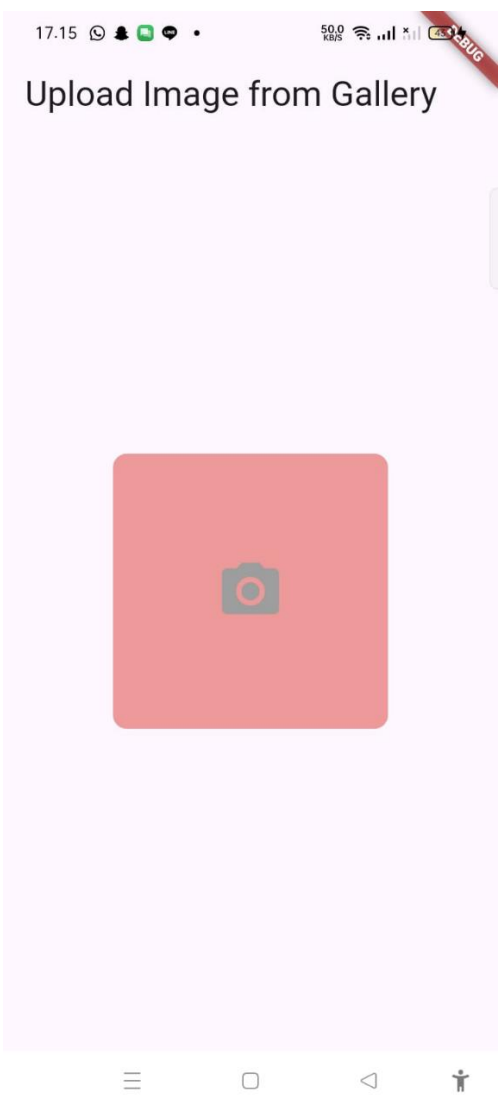





ENDLESS LOVE
Coba lagi
OK


17.43
87%
DEBUG
Image from Camera

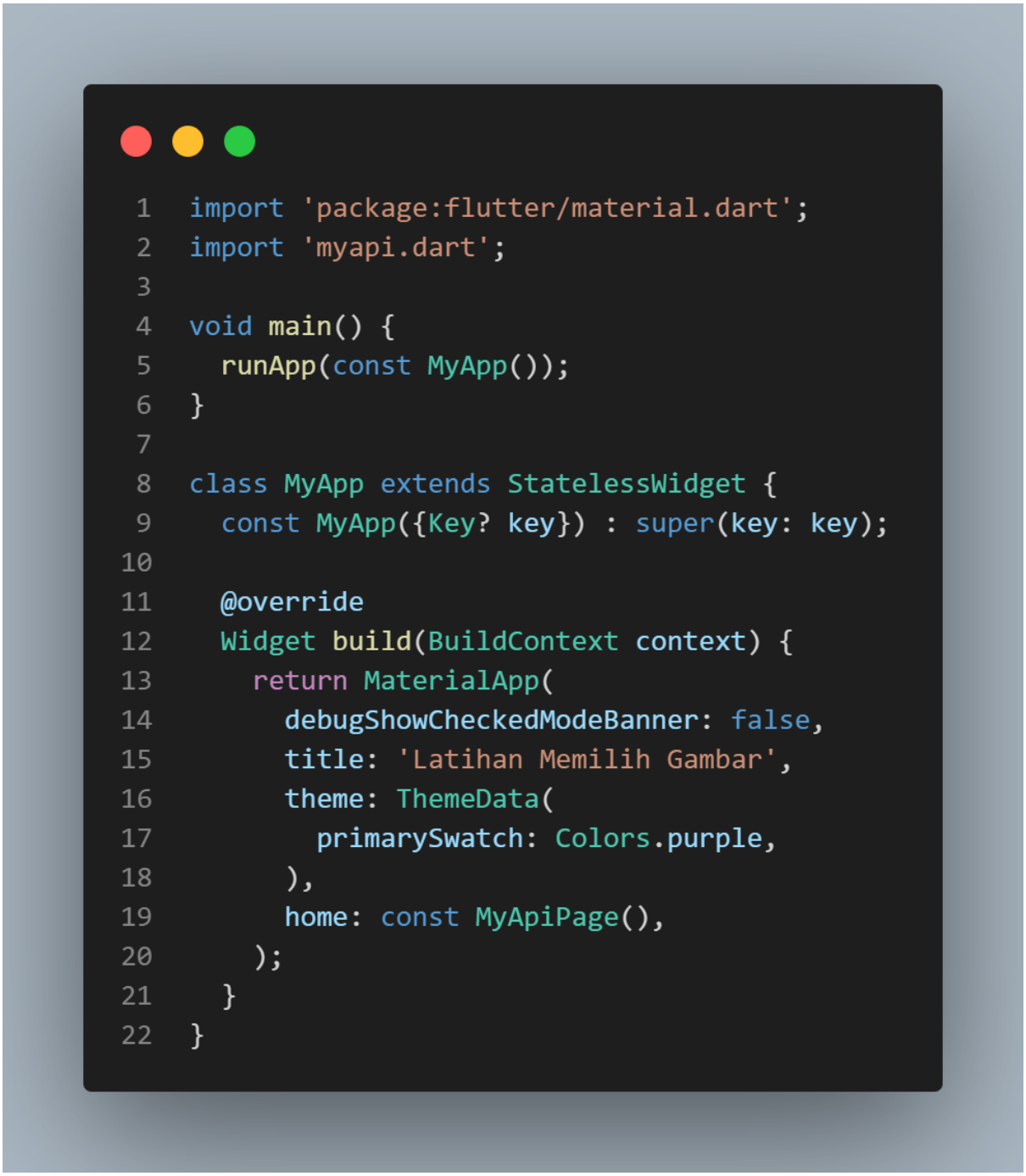
5. Hasil output guided dari gallery



UNGUIDED

1. Main Dart

   Sreenshot code main dart



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

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      title: 'Latihan Memilih Gambar',
      theme: ThemeData(
        primarySwatch: Colors.purple,
      ),
      home: const MyApiPage(),
    );
  }
}
```

2. Image Picker Dart

   Sreenshot code image picker dart

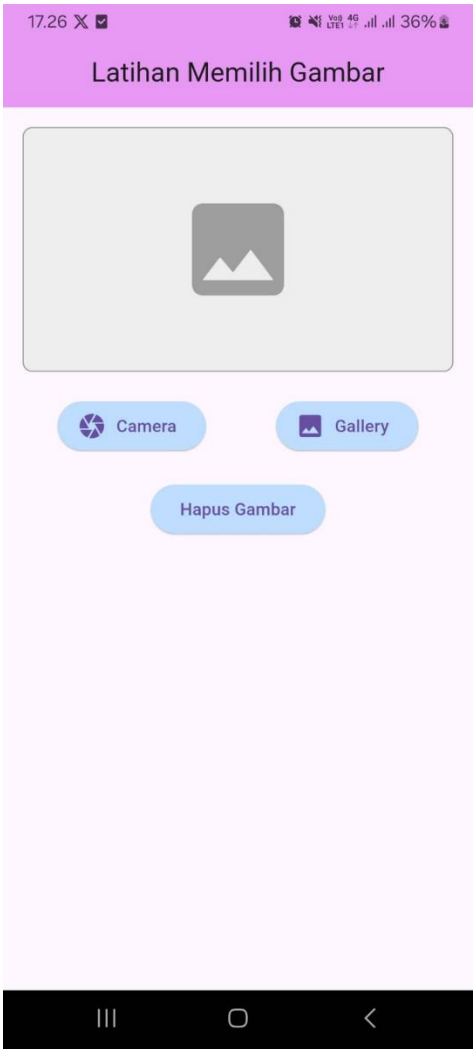
```dart
import 'dart:io';
import 'package:image_picker/image_picker.dart';

class ImagePickerPage {
  static final ImagePicker _picker = ImagePicker();

  // Fungsi untuk mengambil gambar dari galeri
  static Future<String?> pickFromGallery() async {
    try {
      final pickedFile = await _picker.pickImage(source: ImageSource.gallery);
      return pickedFile?.path;
    } catch (e) {
      print("Error picking image from gallery: $e");
      return null;
    }
  }

  // Fungsi untuk mengambil gambar dari kamera
  static Future<String?> pickFromCamera() async {
    try {
      final pickedFile = await _picker.pickImage(source: ImageSource.camera);
      return pickedFile?.path;
    } catch (e) {
      print("Error picking image from camera: $e");
      return null;
    }
  }
}

```

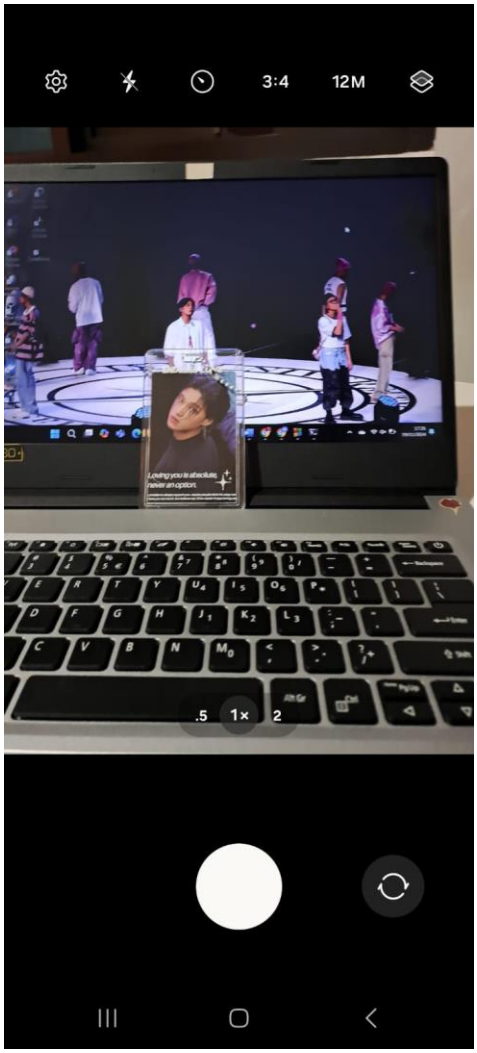
3. Myapi Dart

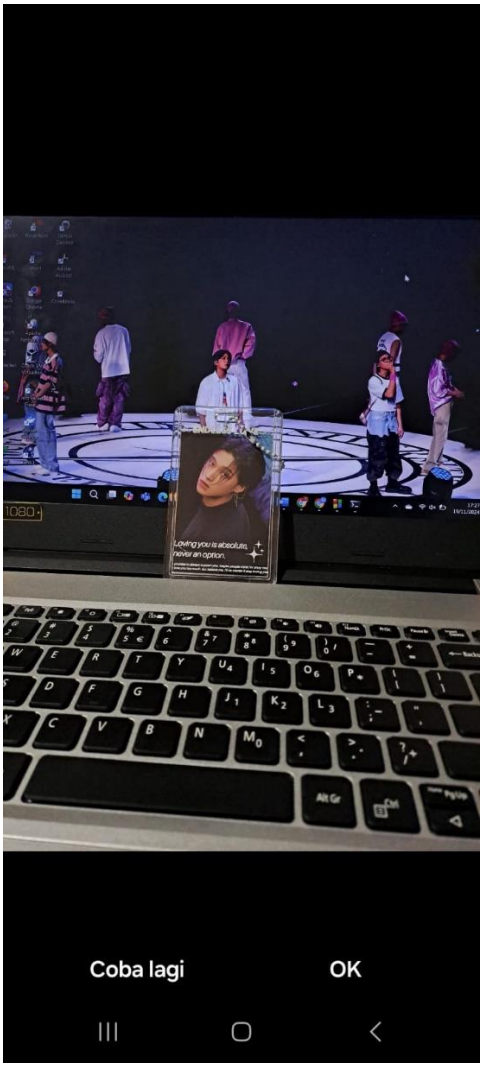
   Sreenshot code myapi dart

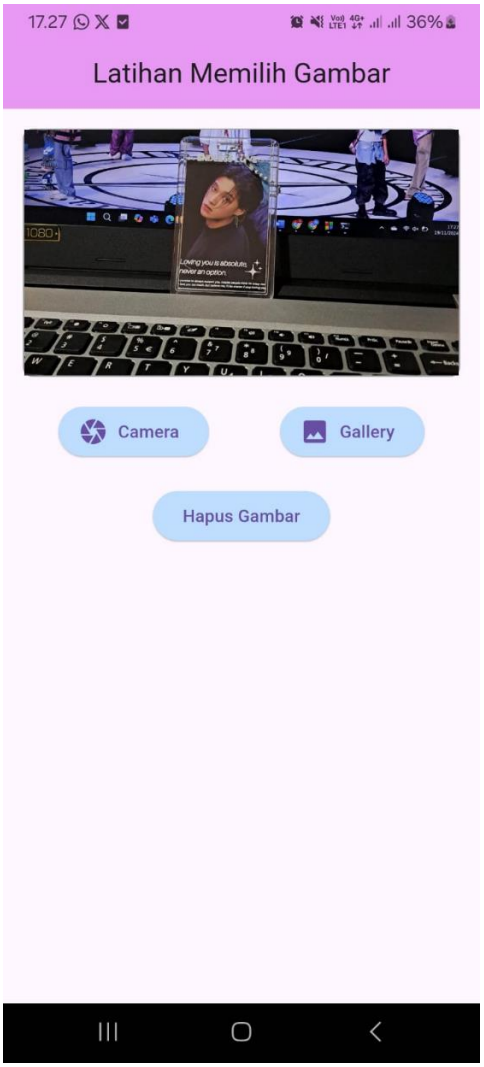
```dart
import 'dart:io';

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

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

  @override
  _MyApiPageState createState() => _MyApiPageState();
}

class _MyApiPageState extends State<MyApiPage> {
  String? _selectedImage; // Untuk menyimpan path gambar yang dipilih

  void _updateImage(String? imagePath) {
    setState(() {
      _selectedImage = imagePath;
    });
  }

  void _clearImage() {
    setState(() {
      _selectedImage = null;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Latihan Memilih Gambar'),
        centerTitle: true,
        backgroundColor: const Color.fromARGB(255, 232, 153, 244),
      ),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          children: [
            Container(
              height: 200,
              width: double.infinity,
              decoration: BoxDecoration(
                color: Colors.grey[200],
                border: Border.all(color: Colors.grey),
                borderRadius: BorderRadius.circular(8.0),
              ),
              child: _selectedImage != null
                  ? Image.file(
                      File(_selectedImage!),
                      fit: BoxFit.cover,
                    )
                  : const Icon(
                      Icons.image,
                      size: 100,
                      color: Colors.grey,
                    ),
            ),
            const SizedBox(height: 20),
            Row(
              mainAxisAlignment: MainAxisAlignment.spaceAround,
              children: [
                ElevatedButton.icon(
                  onPressed: () async {
                    final imagePath = await ImagePickerPage.pickFromCamera();
                    _updateImage(imagePath);
                  },
                  icon: const Icon(Icons.camera),
                  label: const Text('Camera'),
                  style: ElevatedButton.styleFrom(
                    backgroundColor: const Color.fromARGB(255, 191, 221, 255),
                  ),
                ),
                ElevatedButton.icon(
                  onPressed: () async {
                    final imagePath = await ImagePickerPage.pickFromGallery();
                    _updateImage(imagePath);
                  },
                  icon: const Icon(Icons.photo),
                  label: const Text('Gallery'),
                  style: ElevatedButton.styleFrom(
                    backgroundColor: Color.fromARGB(255, 191, 221, 255),
                  ),
                ),
              ],
            ),
            const SizedBox(height: 20),
            ElevatedButton(
              onPressed: _clearImage,
              child: const Text('Hapus Gambar'),
              style: ElevatedButton.styleFrom(
                backgroundColor: Color.fromARGB(255, 191, 221, 255),
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

4. Hasil Output dari Camera

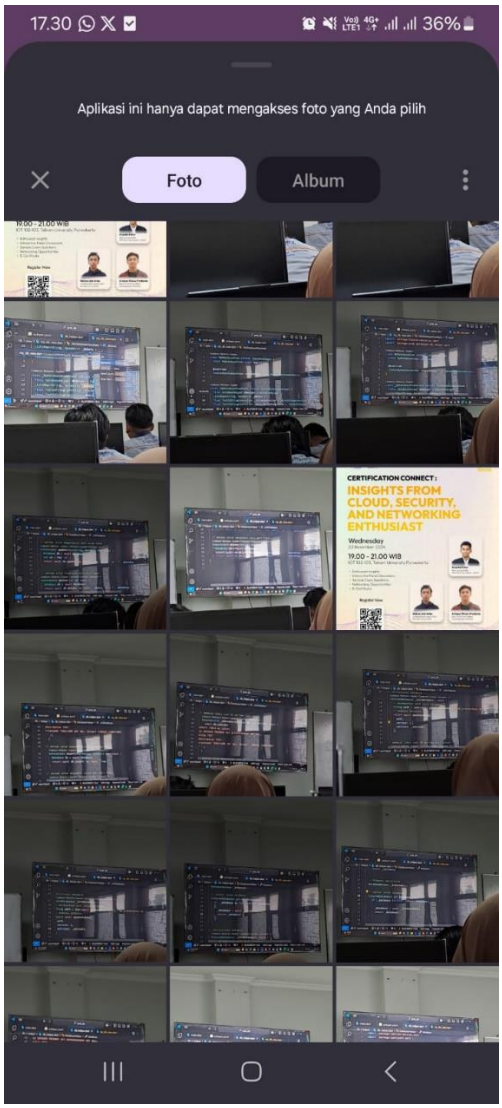


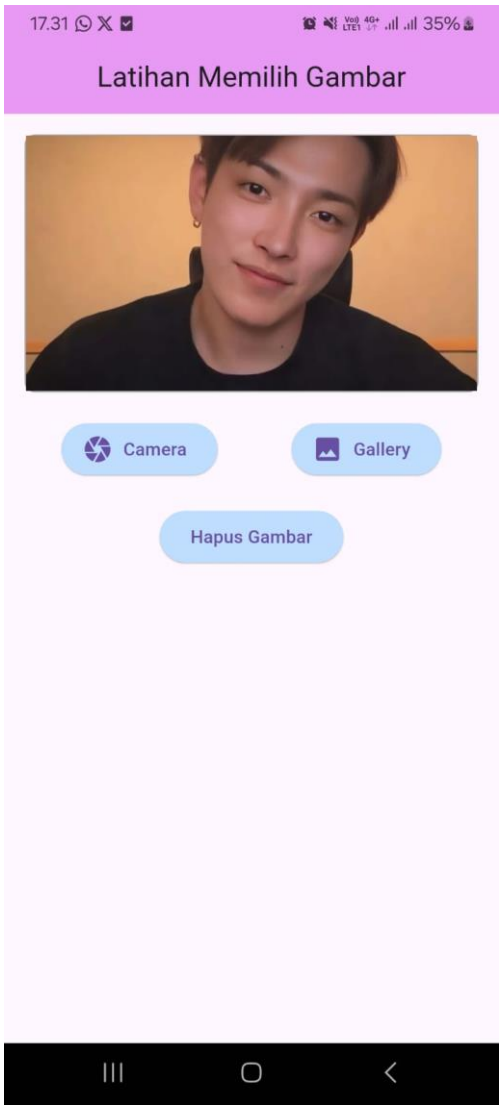



Coba lagi
OK

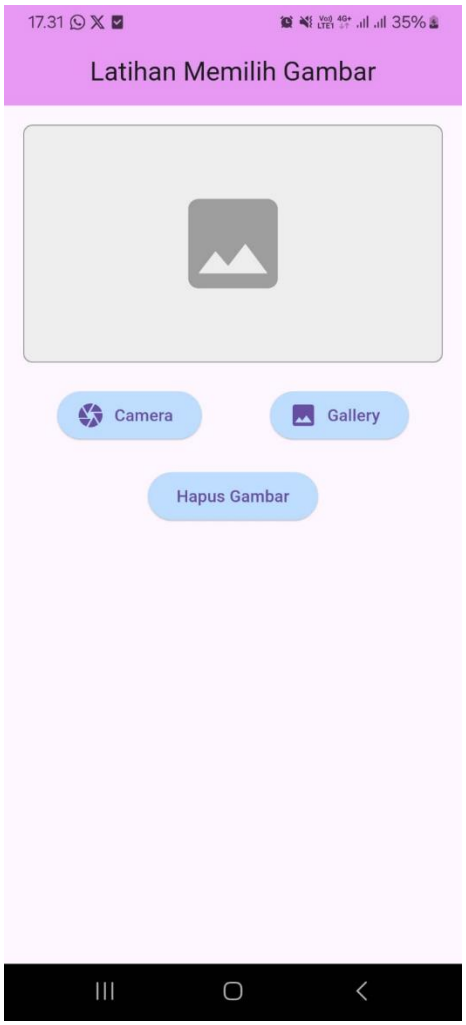

17.27
36%
Latihan Memilih Gambar
Camera
Gallery
Hapus Gambar

5. Hasil Output dari Gallery





6. Hasil Output dari Tampilan Awal dan saat Gambar dihapus



7. Penjelasan

   Kode tersebut gunanya untuk menampilkan gambar pada Flutter. Bagian atasnya untuk mengatur tema dan tampilan utama aplikasi. Aplikasi ini menggunakan warna ungu sebagai warna utama dan menggunakan halaman MyApiPage sebagai halaman awal. Di bagian bawah kode terdapat fungsi-fungsi yang memungkinkan pengguna memilih gambar dari galeri ataupun kamera dari handphone mereka; fungsi ini menggunakan library image_picker sehingga memungkinkan aplikasi berinteraksi dengan kamera atau galeri handphone. Intinya kode ini untuk aplikasi Flutter yang bisa membuat pengguna memilih gambar dari handphone mereka.