

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

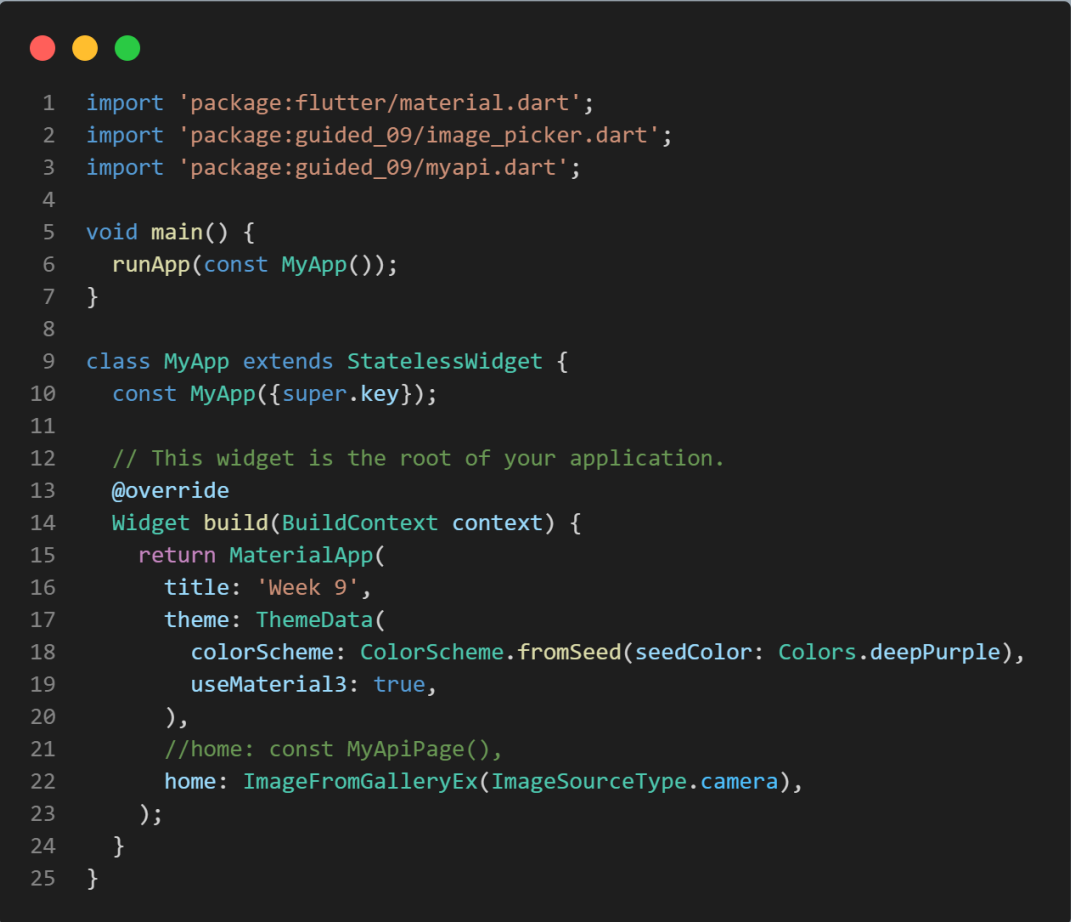
2024

GUIDED

1. Source code Image Picker

```
1 import 'dart:io';
2 import 'package:flutter/material.dart';
3 import 'package:image_picker/image_picker.dart';
4
5 class ImageFromGalleryEx extends StatefulWidget {
6   final ImageSourceType type;
7   ImageFromGalleryEx(this.type);
8
9   @override
10  ImageFromGalleryExState createState() => ImageFromGalleryExState(this.type);
11 }
12
13 class ImageFromGalleryExState extends State<ImageFromGalleryEx> {
14   File? _image;
15   late ImagePicker imagePicker;
16   final ImageSourceType type;
17
18   ImageFromGalleryExState(this.type);
19
20   @override
21   void initState() {
22     super.initState();
23     imagePicker = ImagePicker();
24   }
25
26   @override
27   Widget build(BuildContext context) {
28     return Scaffold(
29       appBar: AppBar(
30         title: Text(type == ImageSourceType.camera
31           ? "Image from Camera"
32           : "Image from Gallery"),
33       ),
34       body: Column(
35         children: <Widget>[
36           SizedBox(height: 52),
37           Center(
38             //mengambil gambar dari camera atau gallery
39             child: GestureDetector(
40               onTap: () async {
41                 //operasi ternary untuk memilih sumber gambar
42                 var source = type == ImageSourceType.camera
43                   ? ImageSource.camera
44                   : ImageSource.gallery;
45
46                 //menyimpan gambar pada variabel image
47                 XFile? image = await imagePicker.pickImage(
48                   source: source,
49                   imageQuality: 50,
50                   preferredCameraDevice: CameraDevice.front);
51
52                 if (image != null) {
53                   setState(() {
54                     _image = File(image.path);
55                   });
56                 }
57               },
58             child: Container(
59               width: 200,
60               height: 200,
61               decoration: BoxDecoration(
62                 color: Colors.red[200],
63               ),
64             ),
65             // menampilkan gambar dari camera atau gallery
66             child: _image != null
67               ? Image.file(
68                 _image!,
69                 width: 200.0,
70                 height: 200.0,
71                 fit: BoxFit.fitHeight,
72               ),
73             // jika tidak ada gambar yang dipilih
74             : Container(
75               decoration: BoxDecoration(
76                 color: Colors.red[200],
77               ),
78             ),
79             width: 200,
80             height: 200,
81             child: Icon(
82               Icons.camera_alt,
83               color: Colors.grey[800],
84             ),
85           ),
86         ],
87       ),
88     );
89   },
90 );
91
92 }
93
94
95 enum ImageSourceType { camera, gallery }
```

2. Source code Main Dart



```
1  import 'package:flutter/material.dart';
2  import 'package:guided_09/image_picker.dart';
3  import 'package:guided_09/myapi.dart';
4
5  void main() {
6    runApp(const MyApp());
7  }
8
9  class MyApp extends StatelessWidget {
10   const MyApp({super.key});
11
12   // This widget is the root of your application.
13   @override
14   Widget build(BuildContext context) {
15     return MaterialApp(
16       title: 'Week 9',
17       theme: ThemeData(
18         colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
19         useMaterial3: true,
20       ),
21       //home: const MyApiPage(),
22       home: ImageFromGalleryEx(ImageSourceType.camera),
23     );
24   }
25 }
```

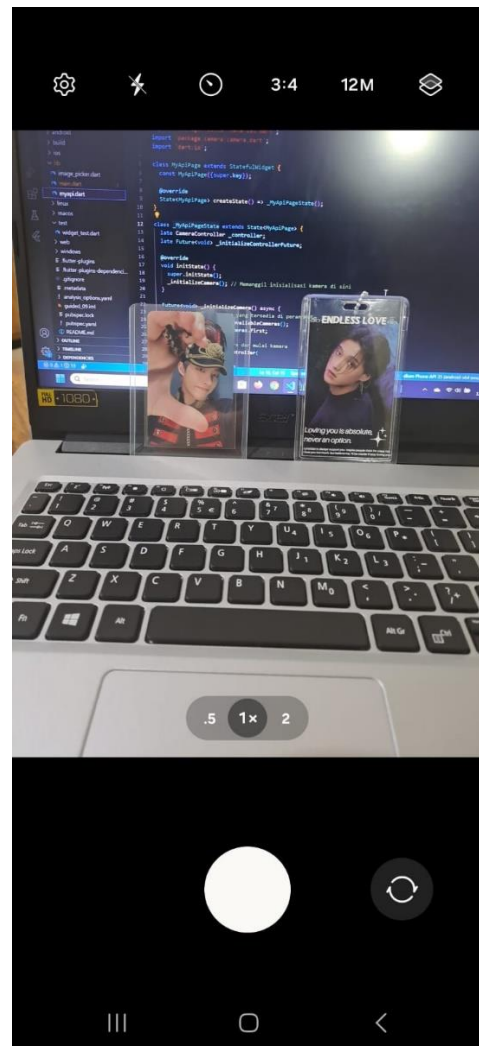
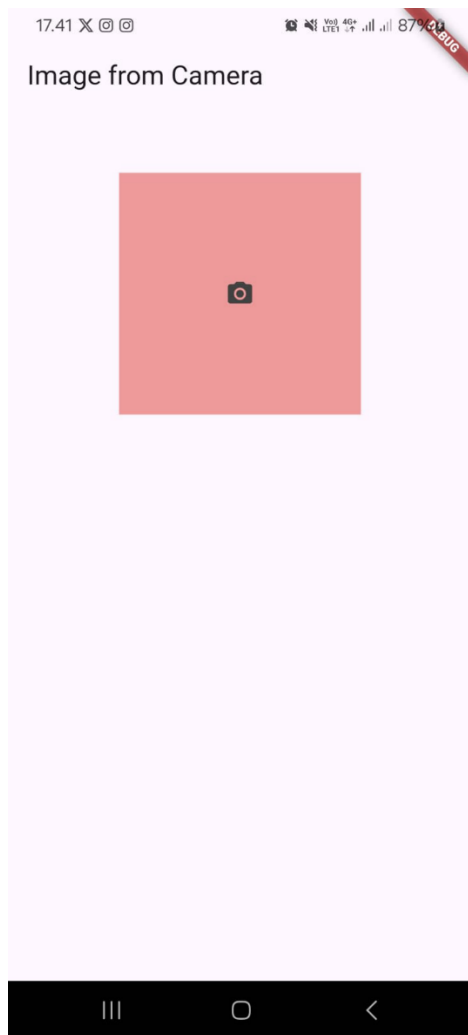
3. Source code My API

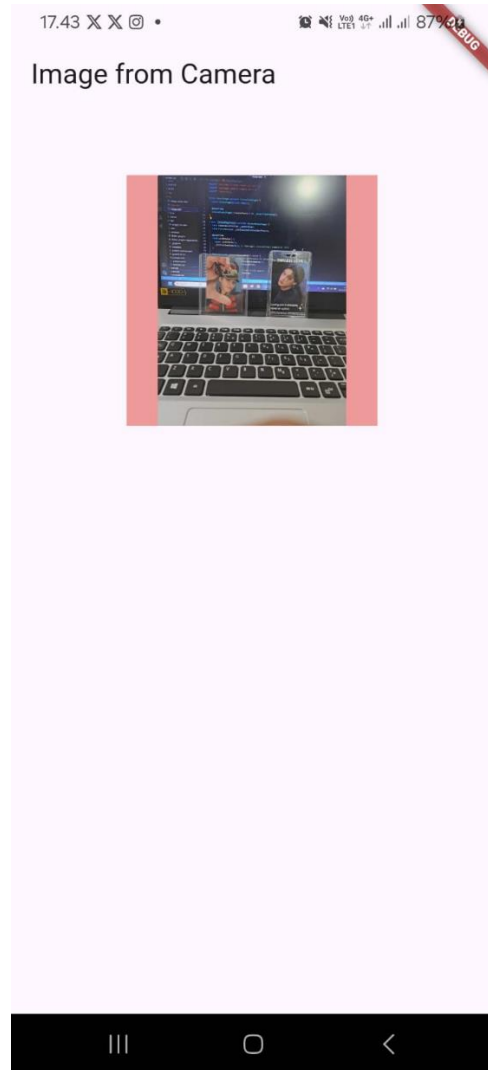
```

1  import 'package:flutter/material.dart';
2  import 'package:camera/camera.dart';
3  import 'dart:io';
4
5  class MyApiPage extends StatefulWidget {
6    const MyApiPage({super.key});
7
8    @override
9    State<MyApiPage> createState() => _MyApiPageState();
10 }
11
12 class _MyApiPageState extends State<MyApiPage> {
13   late CameraController _controller;
14   late Future<void> _initializeControllerFuture;
15
16   @override
17   void initState() {
18     super.initState();
19     _initializeCamera(); // Memanggil inisialisasi kamera di sini
20   }
21
22   Future<void> _initializeCamera() async {
23     // Ambil daftar kamera yang tersedia di perangkat
24     final cameras = await availableCameras();
25     final firstCamera = cameras.first;
26
27     // Buat kontroler kamera dan mulai kamera
28     _controller = CameraController(
29       firstCamera,
30       ResolutionPreset.high,
31     );
32
33     // Inisialisasi kontroler kamera
34     _initializeControllerFuture = _controller.initialize();
35     setState(() {}); // Pastikan state diperbarui setelah kamera diinisialisasi
36   }
37
38   @override
39   void dispose() {
40     // Bersihkan kontroler ketika widget dihapus
41     _controller.dispose();
42     super.dispose();
43   }
44
45   @override
46   Widget build(BuildContext context) {
47     return Scaffold(
48       appBar: AppBar(
49         title: Text("API Perangkat Keras"),
50         centerTitle: true,
51         backgroundColor: Colors.amberAccent,
52       ),
53       body: FutureBuilder<void>({
54         future: _initializeControllerFuture, // Menunggu inisialisasi kamera
55         builder: (context, snapshot) {
56           if (snapshot.connectionState == ConnectionState.done) {
57             // Jika kamera sudah siap, tampilkan pratinjau
58             return CameraPreview(_controller);
59           } else {
60             // Jika kamera belum siap, tampilkan loading
61             return Center(child: CircularProgressIndicator());
62           }
63         },
64       ),
65       floatingActionButton: FloatingActionButton(
66         onPressed: () async {
67           try {
68             // Pastikan kamera sudah diinisialisasi
69             await _initializeControllerFuture;
70
71             // Ambil gambar
72             final image = await _controller.takePicture();
73
74             // Tampilkan / gunakan gambar
75             Navigator.push(
76               context,
77               MaterialPageRoute(
78                 builder: (context) =>
79                   DisplayPictureScreen(imagePath: image.path),
80               ),
81             );
82           } catch (e) {
83             print(e);
84           }
85         },
86         child: Icon(Icons.camera_alt),
87       ),
88     );
89   }
90 }
91
92 class DisplayPictureScreen extends StatelessWidget {
93   final String imagePath;
94
95   const DisplayPictureScreen({Key? key, required this.imagePath})
96     : super(key: key);
97
98   @override
99   Widget build(BuildContext context) {
100     return Scaffold(
101       appBar: AppBar(title: Text('Display Picture')),
102       body: Image.file(File(imagePath)),
103     );
104   }
105 }

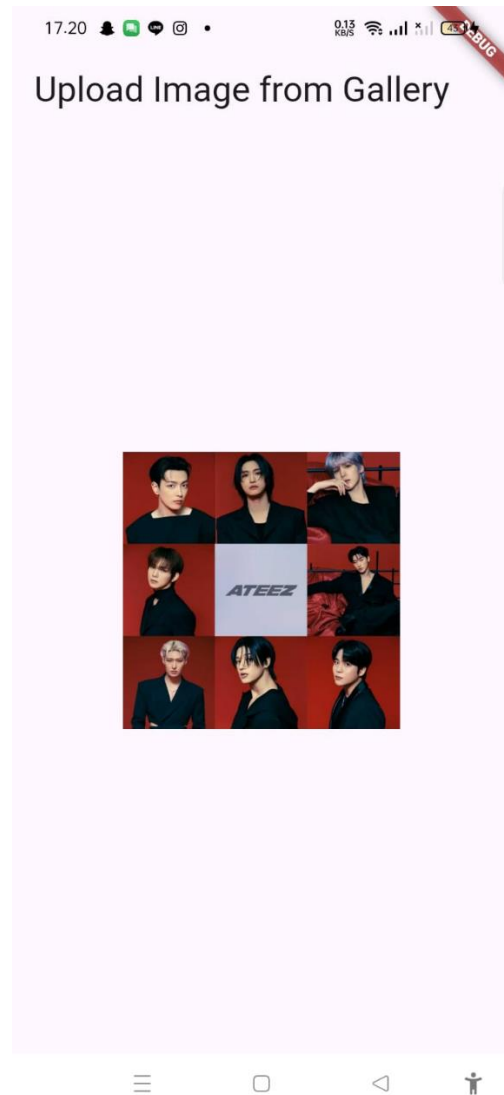
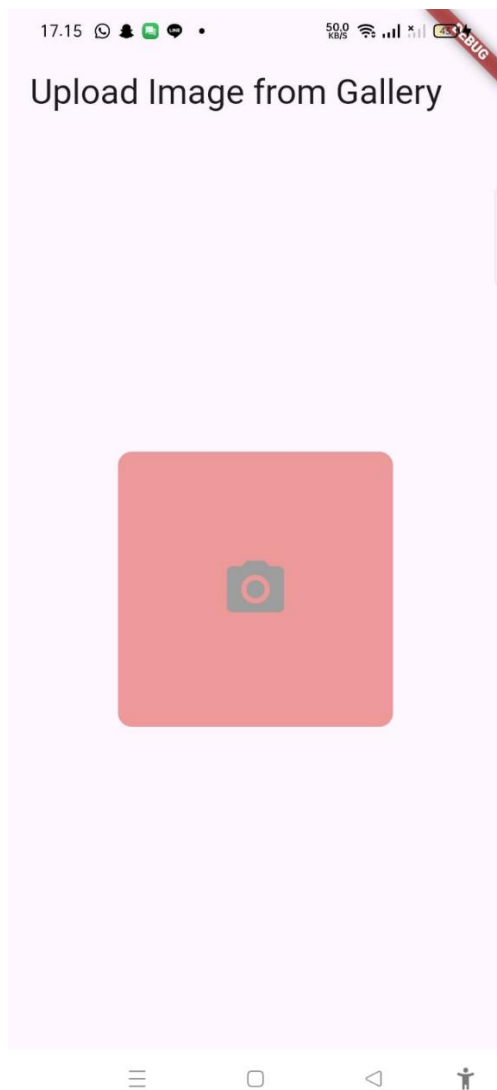
```

4. Hasil output dari guided from camera





5. Hasil output guided dari gallery



UNGUIDED

1. Main Dart

Screenshot code main dart



```
1  import 'package:flutter/material.dart';
2  import 'myapi.dart';
3
4  void main() {
5    runApp(const MyApp());
6  }
7
8  class MyApp extends StatelessWidget {
9    const MyApp({Key? key}) : super(key: key);
10
11    @override
12    Widget build(BuildContext context) {
13      return MaterialApp(
14        debugShowCheckedModeBanner: false,
15        title: 'Latihan Memilih Gambar',
16        theme: ThemeData(
17          primarySwatch: Colors.purple,
18        ),
19        home: const MyApiPage(),
20      );
21    }
22  }
```

2. Image Picker Dart

Screenshot code image picker dart

A screenshot of a code editor window with a dark background. At the top left, there are three colored circles (red, yellow, green) representing window control buttons. The code is written in Dart and is numbered from 1 to 29 on the left side. It defines a class `ImagePickerPage` with two static methods: `pickFromGallery()` and `pickFromCamera()`. Both methods use `ImagePicker` to pick an image from either the gallery or the camera. The code includes error handling with `try/catch` blocks and prints error messages if an exception occurs.

```
1 import 'dart:io';
2 import 'package:image_picker/image_picker.dart';
3
4 class ImagePickerPage {
5   static final ImagePicker _picker = ImagePicker();
6
7   // Fungsi untuk mengambil gambar dari galeri
8   static Future<String?> pickFromGallery() async {
9     try {
10       final pickedFile = await _picker.pickImage(source: ImageSource.gallery);
11       return pickedFile?.path;
12     } catch (e) {
13       print("Error picking image from gallery: $e");
14       return null;
15     }
16   }
17
18   // Fungsi untuk mengambil gambar dari kamera
19   static Future<String?> pickFromCamera() async {
20     try {
21       final pickedFile = await _picker.pickImage(source: ImageSource.camera);
22       return pickedFile?.path;
23     } catch (e) {
24       print("Error picking image from camera: $e");
25       return null;
26     }
27   }
28 }
29
```

3. Myapi Dart

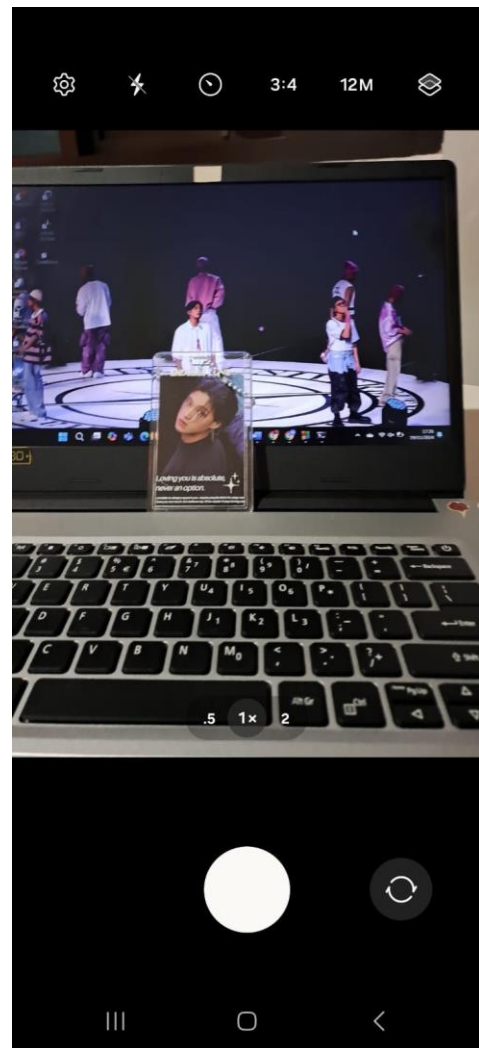
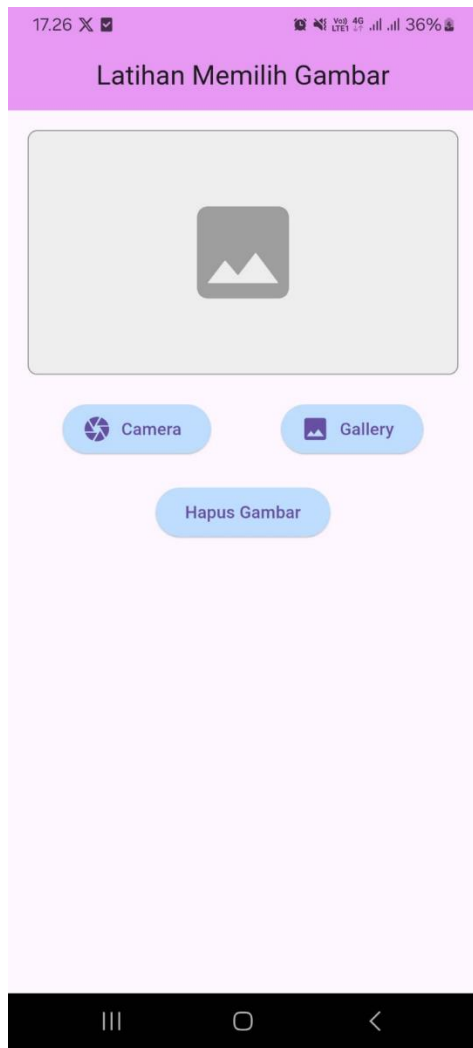
Sreenshot code myapi dart

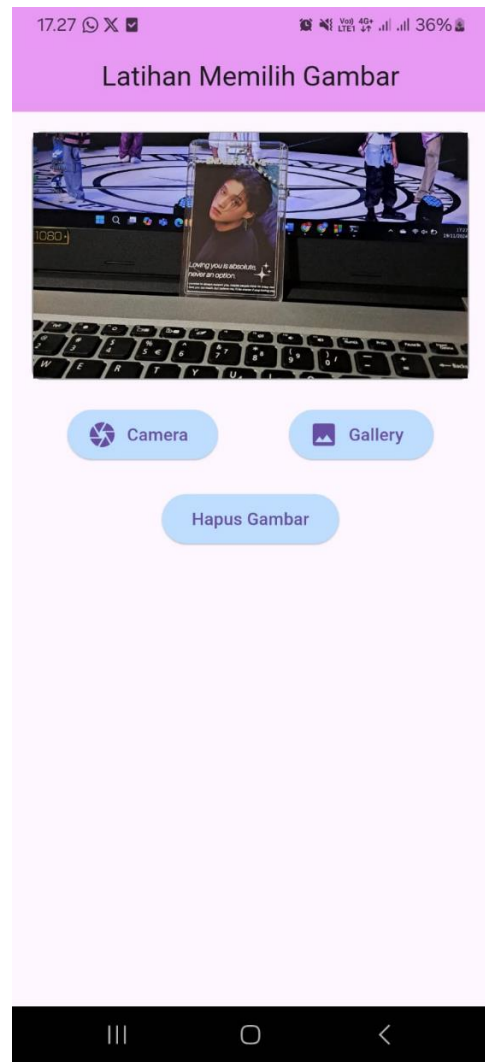
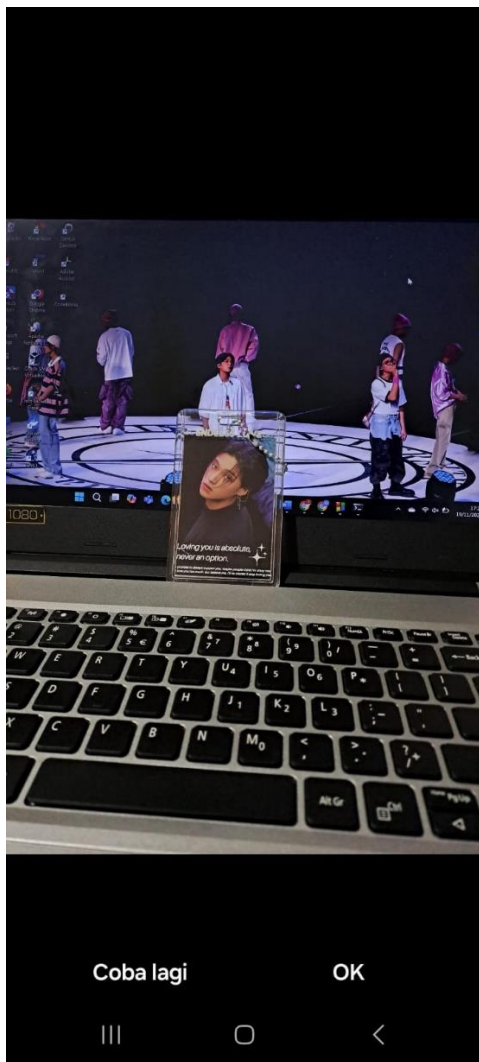
```

1  import 'dart:io';
2
3  import 'package:flutter/material.dart';
4  import 'image_picker.dart';
5
6  class MyApiPage extends StatefulWidget {
7    const MyApiPage({Key? key}) : super(key: key);
8
9    @override
10   _MyApiPageState createState() => _MyApiPageState();
11 }
12
13 class _MyApiPageState extends State<MyApiPage> {
14   String? _selectedImage; // Untuk menyimpan path gambar yang dipilih
15
16   void _updateImage(String? imagePath) {
17     setState(() {
18       _selectedImage = imagePath;
19     });
20   }
21
22   void _clearImage() {
23     setState(() {
24       _selectedImage = null;
25     });
26   }
27
28   @override
29   Widget build(BuildContext context) {
30     return Scaffold(
31       appBar: AppBar(
32         title: const Text('Latihan Memilih Gambar'),
33         centerTitle: true,
34         backgroundColor: const Color.fromARGB(255, 232, 153, 244),
35       ),
36       body: Padding(
37         padding: const EdgeInsets.all(16.0),
38         child: Column(
39           children: [
40             Container(
41               height: 200,
42               width: double.infinity,
43               decoration: BoxDecoration(
44                 color: Colors.grey[200],
45                 border: Border.all(color: Colors.grey),
46                 borderRadius: BorderRadius.circular(8.0),
47               ),
48               child: _selectedImage != null
49                 ? Image.file(
50                     File(_selectedImage!),
51                     fit: BoxFit.cover,
52                   )
53                 : const Icon(
54                     Icons.image,
55                     size: 100,
56                     color: Colors.grey,
57                   ),
58             ),
59             const SizedBox(height: 20),
60             Row(
61               mainAxisAlignment: MainAxisAlignment.spaceAround,
62               children: [
63                 ElevatedButton.icon(
64                   onPressed: () async {
65                     final imagePath = await ImagePickerPage.pickFromCamera();
66                     _updateImage(imagePath);
67                   },
68                   icon: const Icon(Icons.camera),
69                   label: const Text('Camera'),
70                   style: ElevatedButton.styleFrom(
71                     backgroundColor: const Color.fromARGB(255, 191, 221, 255),
72                   ),
73                 ),
74                 ElevatedButton.icon(
75                   onPressed: () async {
76                     final imagePath = await ImagePickerPage.pickFromGallery();
77                     _updateImage(imagePath);
78                   },
79                   icon: const Icon(Icons.photo),
80                   label: const Text('Gallery'),
81                   style: ElevatedButton.styleFrom(
82                     backgroundColor: Color.fromARGB(255, 191, 221, 255),
83                   ),
84                 ),
85               ],
86             ),
87             const SizedBox(height: 20),
88             ElevatedButton(
89               onPressed: _clearImage,
90               child: const Text('Hapus Gambar'),
91               style: ElevatedButton.styleFrom(
92                 backgroundColor: Color.fromARGB(255, 191, 221, 255),
93               ),
94             ),
95           ],
96         ),
97       ),
98     );
99   }
100 }

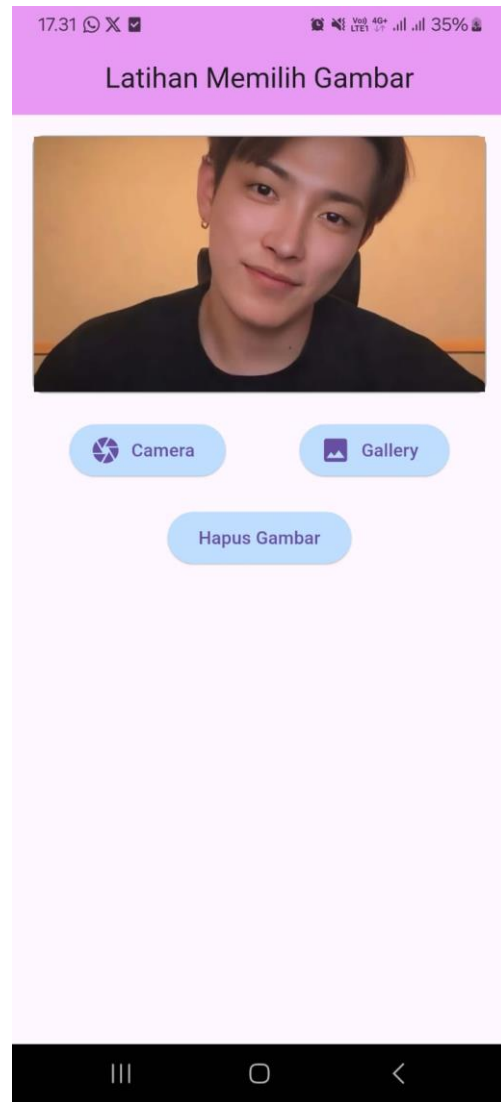
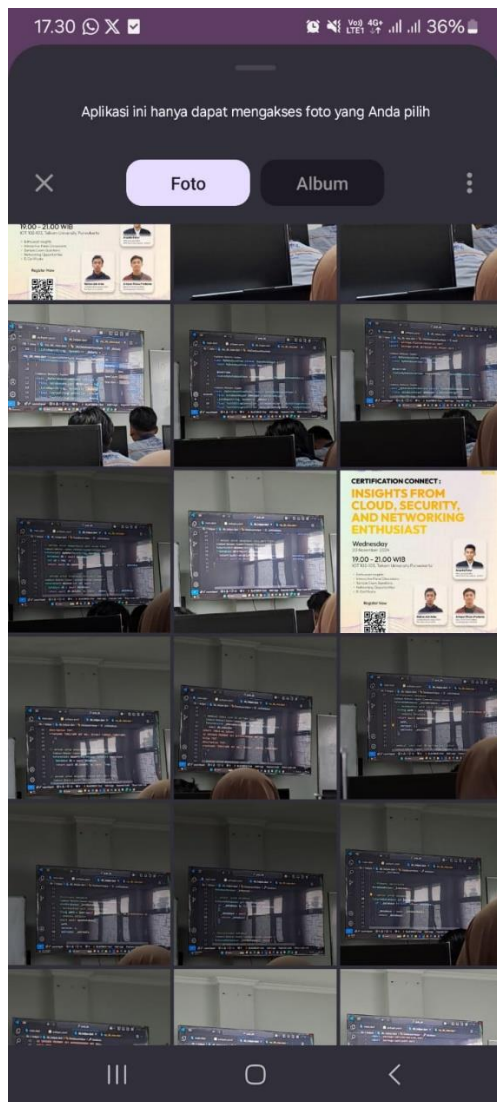
```

4. Hasil Output dari Camera

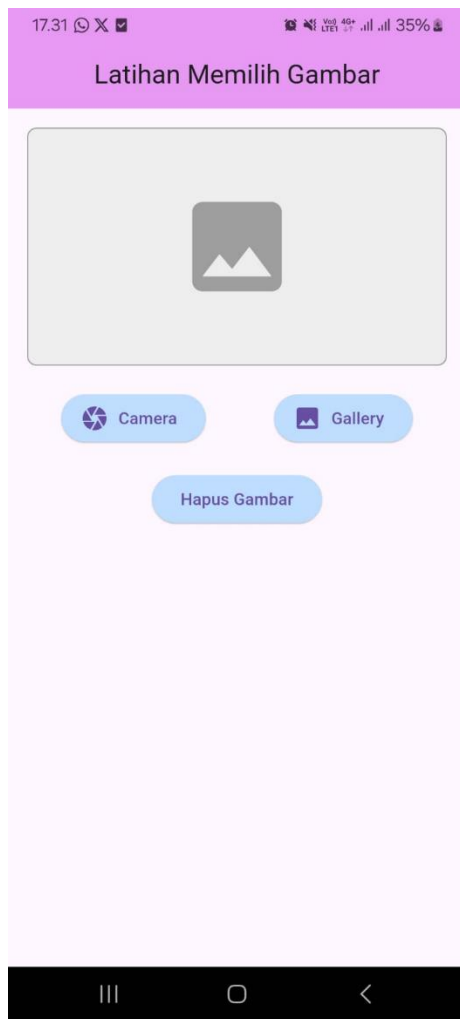




5. Hasil Output dari Gallery



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.