JOBSHEET – APLIKASI OCR SEDERHANA DENGAN FLUTTER

1. IDENTITAS PRAKTIKAN

Komponen	Isi
Nama	**Dhevina Agustina**
Kelas / NIM	**SIB 3F/2341760065**
Tanggal	**14 Oktober 2025**
Guru / Dosen	**Pak Ade Ismail, S.Kom., M.TI.**

2. TUJUAN PRAKTIKUM

Setelah menyelesaikan jobsheet ini, siswa/mahasiswa mampu:

1. Membuat aplikasi Flutter multi-halaman.
2. Menggunakan plugin kamera untuk mengambil gambar.
3. Mengintegrasikan **OCR (Optical Character Recognition)** menggunakan library `google_mlkit_text_recognition`.
4. Menampilkan hasil OCR di halaman terpisah.
5. Menerapkan navigasi dasar antar layar menggunakan Navigator.

3. ALAT DAN BAHAN

- Laptop/komputer dengan Flutter SDK terinstal
- VS Code atau Android Studio
- Emulator Android atau perangkat Android fisik
- Koneksi internet (untuk instalasi dependensi)

Link Github: https://github.com/dhevinaagustina/ocr_sederhana.git

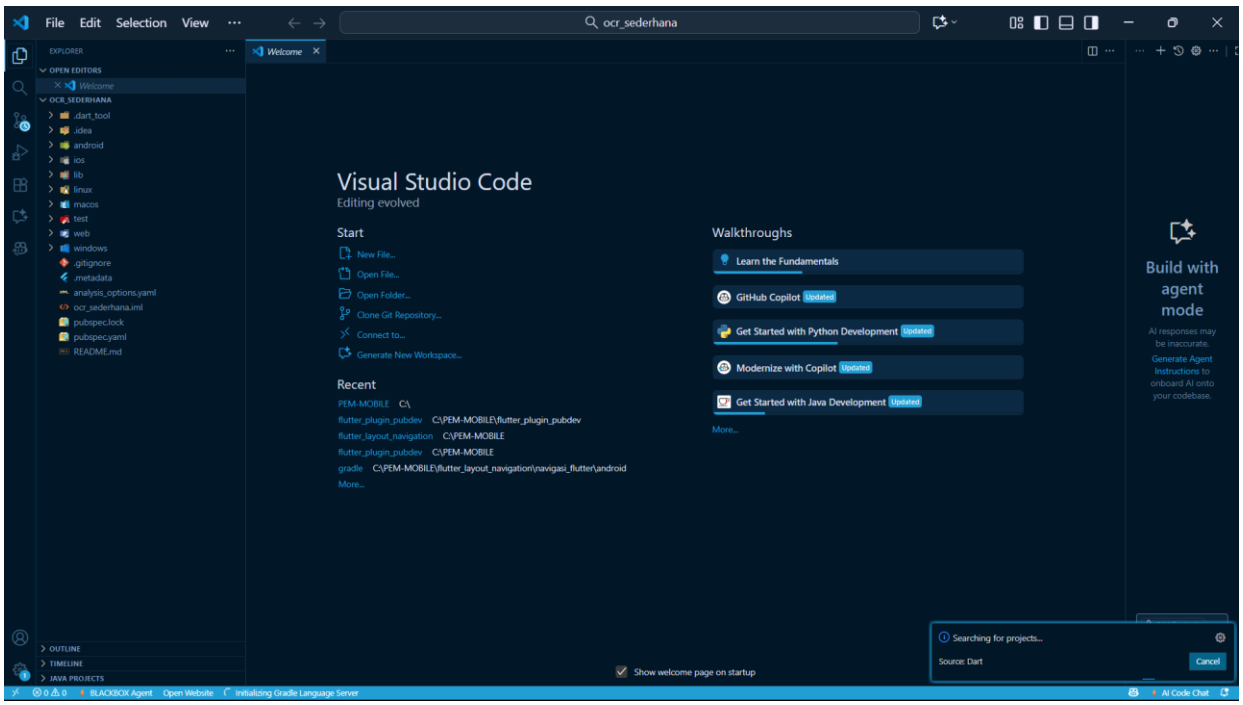
4. LANGKAH KERJA

4.1. Langkah 1: Buat Proyek Baru

Buka terminal, lalu jalankan:

```
flutter create ocr_sederhana
cd ocr_sederhana
```

Listing 1: Membuat proyek Flutter



4.2. Langkah 2: Tambahkan Plugin

Buka file pubspec.yaml, lalu tambahkan dependensi berikut di bawah bagian dependencies:

```
dependencies:
  flutter:
    sdk: flutter
  google_mlkit_text_recognition: ^0.10.0
  camera: ^0.10.5+5
  path_provider: ^2.1.2
  path: ^1.8.3
```

Listing 2: pubspec.yaml - dependencies

Simpan file, lalu jalankan:

```
flutter pub get
```

```
29   # versions available, run `flutter pub outdated`.
30 dependencies:
31   flutter:
32     sdk: flutter
33     google_mlkit_text_recognition: ^0.13.0
34   camera: ^0.10.5+5
35   path_provider: ^2.1.2
36   path: ^1.8.3
37
+ camera_platform_interface 2.11.0
+ camera_web 0.3.5
  characters 1.4.0 (1.4.1 available)
+ cross_file 0.3.4+2
  flutter_lints 5.0.0 (6.0.0 available)
+ flutter_plugin_android_lifecycle 2.0.32
+ flutter_web_plugins 0.0.0 from sdk flutter
+ google_mlkit_commons 0.8.1 (0.11.0 available)
+ google_mlkit_text_recognition 0.13.1 (0.15.0 available)
  lints 5.1.1 (6.0.0 available)
  material_color_utilities 0.11.1 (0.13.0 available)
  meta 1.16.0 (1.17.0 available)
+ plugin_platform_interface 2.1.8
+ stream_transform 2.1.1
  test_api 0.7.6 (0.7.7 available)
+ web 1.1.1
Changed 13 dependencies!
9 packages have newer versions incompatible with dependency constraints.
Try `flutter pub outdated` for more information.
PS C:\PEM-MOBILE\ocr_sederhana>
```

4.3. Langkah 3: Tambahkan Izin Kamera (Android)

Buka file: android/app/src/main/AndroidManifest.xml

Tambahkan baris berikut di dalam tag <manifest>, sebelum <application>:

```
<uses-permission android:name="android.permission.CAMERA" />
```

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
    <uses-permission android:name="android.permission.CAMERA"/>
```

4.4. Langkah 4: Buat Struktur Folder

Di dalam folder lib/, buat struktur berikut:

```
lib/
        main.dart
        screens/
            splash_screen.dart
            home_screen.dart
            scan_screen.dart
            result_screen.dart
```

5. KODE PROGRAM

5.1. File: lib/main.dart

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

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

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

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'OCR Sederhana',
      theme: ThemeData(primarySwatch: Colors.blue),
      home: const SplashScreen(),
      debugShowCheckedModeBanner: false,
    );
  }
}
```

Listing 3: main.dart

5.2. File: lib/screens/splash_screen.dart

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

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

  @override
  State<SplashScreen> createState() => _SplashScreenState();
}

class _SplashScreenState extends State<SplashScreen> {
  @override
```

```
  void initState() {
    super.initState();
    Timer(const Duration(seconds: 2), () {
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(builder: (_) => const HomeScreen()),
      );
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.blue,
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: const [
            CircularProgressIndicator(color: Colors.white),
            SizedBox(height: 20),
            Text('OCR Scanner',
                style: TextStyle(color: Colors.white, fontSize:
 24)),
          ],
        ),
      ),
    );
  }
}
```

Listing 4: splash_screen.dart

5.3. File: lib/screens/home_screen.dart

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

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

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

```
    return Scaffold(
      appBar: AppBar(title: const Text('Menu Utama')),
      body: Center(
        child: ElevatedButton(
          onPressed: () {
            Navigator.push(
              context,
              MaterialPageRoute(builder: (_) => const ScanScreen
   ()),
            );
          },
          child: const Text('Mulai Scan Teks'),
        ),
      ),
    );
  }
}
```

Listing 5: home_screen.dart

5.4. File: lib/screens/scan_screen.dart

```
import 'dart:io';
import 'package:flutter/material.dart';
import 'package:camera/camera.dart';
import 'package:google_mlkit_text_recognition/google_mlkit_text_
   recognition.dart';
import 'package:path/path.dart' as path;
import 'package:path_provider/path_provider.dart';
import 'result_screen.dart';

late List<CameraDescription> cameras;

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

  @override
  State<ScanScreen> createState() => _ScanScreenState();
}

class _ScanScreenState extends State<ScanScreen> {
  late CameraController _controller;
```

```
20  late Future<void> _initializeControllerFuture;
21
22  @override
23  void initState() {
24    super.initState();
25    _initCamera();
26  }
27
28  void _initCamera() async {
29    cameras = await availableCameras();
30    _controller = CameraController(cameras[0], ResolutionPreset.
     medium);
31    _initializeControllerFuture = _controller.initialize();
32    if (mounted) {
33      setState(() {});
34    }
35  }
36
37  @override
38  void dispose() {
39    _controller.dispose();
40    super.dispose();
41  }
42
43  Future<String> _ocrFromFile(File imageFile) async {
44    final inputImage = InputImage.fromFile(imageFile);
45    final textRecognizer = TextRecognizer(script:
     TextRecognitionScript.latin);
46    final RecognizedText recognizedText = await textRecognizer.
     processImage(inputImage);
47    textRecognizer.close();
48    return recognizedText.text;
49  }
50
51  Future<void> _takePicture() async {
52    try {
53      await _initializeControllerFuture;
54
55      if (!mounted) return;
56      ScaffoldMessenger.of(context).showSnackBar(
```

```
57          const SnackBar(content: Text('Memproses OCR, mohon
    tunggu...'), duration: Duration(seconds: 2)));
58
59      final XFile image = await _controller.takePicture();
60
61      final ocrText = await _ocrFromFile(File(image.path));
62
63      if (!mounted) return;
64      Navigator.push(
65        context,
66        MaterialPageRoute(builder: (_) => ResultScreen(ocrText:
    ocrText)),
67      );
68    } catch (e) {
69      if (!mounted) return;
70      ScaffoldMessenger.of(context).showSnackBar(SnackBar(content
    : Text('Error saat mengambil/memproses foto: $e')));
71    }
72  }
73
74  @override
75  Widget build(BuildContext context) {
76    if (!_controller.value.isInitialized) {
77      return const Scaffold(body: Center(child:
    CircularProgressIndicator()));
78    }
79
80    return Scaffold(
81      appBar: AppBar(title: const Text('Kamera OCR')),
82      body: Column(
83        children: [
84          Expanded(
85            child: AspectRatio(
86              aspectRatio: _controller.value.aspectRatio,
87              child: CameraPreview(_controller),
88            ),
89          ),
90          Padding(
91            padding: const EdgeInsets.all(16.0),
92            child: ElevatedButton.icon(
93              onPressed: _takePicture,
```

```
                icon: const Icon(Icons.camera),
                label: const Text('Ambil Foto & Scan'),
              ),
            ),
          ],
        ),
      );
  }
}
```

Listing 6: scan_screen.dart

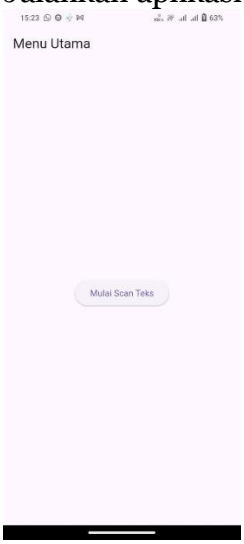
5.5. File: lib/screens/result_screen.dart

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

class ResultScreen extends StatelessWidget {
  final String ocrText;

  const ResultScreen({super.key, required this.ocrText});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(title: const Text('Hasil OCR')),
      body: Padding(
        padding: const EdgeInsets.all(16.0),
        child: SingleChildScrollView(
          child: SelectableText(
            ocrText.isEmpty
                ? 'Tidak ada teks ditemukan.'
                : ocrText.replaceAll('\n', ' '),
            style: const TextStyle(fontSize: 18),
          ),
        ),
      ),
    );
  }
}
```

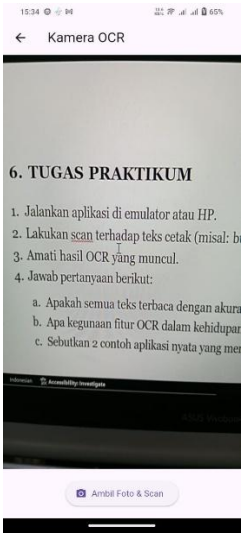
Listing 7: result_screen.dart

6. TUGAS PRAKTIKUM

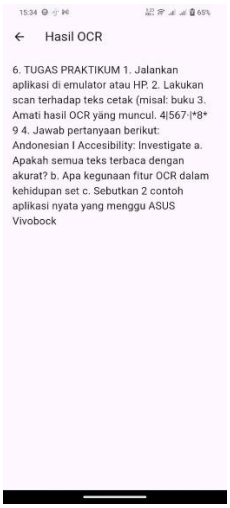
1. Jalankan aplikasi di emulator atau HP.



2. Lakukan scan terhadap teks cetak (misal: buku, koran, atau layar HP).



3. Amati hasil OCR yang muncul.



4. Jawab pertanyaan berikut:
    a. Apakah semua teks terbaca dengan akurat? Mengapa?

Jawaban: Tidak semua teks terbaca dengan akurat. Hal ini terlihat dari adanya karakter yang salah terbaca, misalnya “4|567:|*8*9 4.” yang tidak ada di teks asli, serta kata “Andonesian I Accesibility” yang seharusnya tidak muncul. Kesalahan ini bisa terjadi karena kualitas foto kurang jelas, pencahayaan tidak merata, atau font teks yang sulit dikenali oleh sistem OCR.

b. Apa kegunaan fitur OCR dalam kehidupan sehari-hari?

   Jawaban: Fitur OCR (Optical Character Recognition) berguna untuk mengubah teks dari gambar atau dokumen cetak menjadi teks digital sehingga bisa diedit, disalin, atau dianalisis.

   Contoh: Mempermudah digitalisasi dokumen (misalnya KTP, faktur, atau kwitansi), Membantu pencarian data otomatis dari arsip kertas, Mempercepat input data tanpa harus mengetik ulang.

c. Sebutkan 2 contoh aplikasi nyata yang menggunakan OCR!

   Jawaban:

   1) Google Lens – digunakan untuk membaca teks dari gambar dan menerjemahkannya secara langsung.
   2) Microsoft Office Lens – digunakan untuk memindai dokumen atau papan tulis dan mengubahnya menjadi teks yang bisa disimpan ke Word atau PDF.

7. CATATAN PENTING

- Pastikan kamera perangkat dalam kondisi baik dan pencahayaan cukup.
- Plugin google_mlkit_text_recognition bekerja **offline** dan mendukung bahasa Latin (termasuk Indonesia).
- Jika muncul error saat pertama kali buka kamera, pastikan izin kamera sudah diizinkan di pengaturan HP.

8. PENILAIAN

Aspek	**Skor (1–5)**
Kelengkapan kode	
Aplikasi berjalan lancar	
Jawaban tugas	
Ketepatan waktu	
Total	

Nilai Akhir = Total Skor × 5

Selamat mengerjakan!