# JOBSHEET – APLIKASI OCR SEDERHANA DENGAN FLUTTER

## 1. IDENTITAS PRAKTIKAN

| Komponen | Isi |
|---|---|
| Nama | Fatima Sitta Maulidia |
| Kelas / NIM | SIB 3F / 2341760167 |
| Tanggal | 15 Oktober 2025 |
| Guru / Dosen | 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)

## 4. LANGKAH KERJA

### 4.1. Langkah 1: Buat Proyek Baru

Buka terminal, lalu jalankan:

```
1  flutter create ocr_sederhana
2  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:

```
1  dependencies:
2    flutter:
3      sdk    :              flutter
4    google_mlkit_text_recognition: ^0.10.0
5    camera: ^0.10.5+5
6    path_provider: ^2.1.2
7    path: ^1.8.3
```

Listing 2: pubspec.yaml - dependencies

Simpan file, lalu jalankan:

```
1  flutter pub get
```



2

### 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>:

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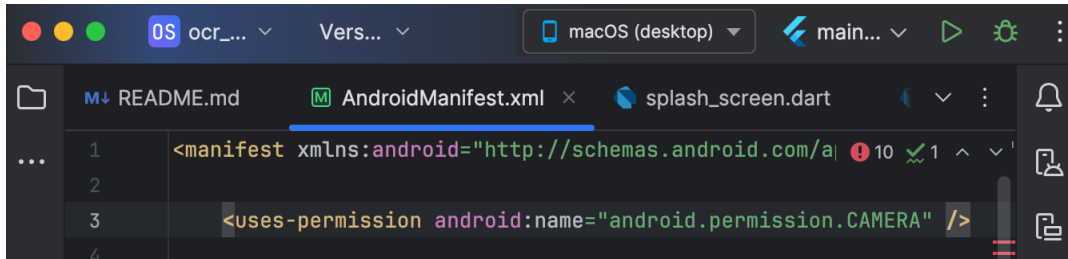


### 4.4. Langkah 4: Buat Struktur Folder

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,
    );
  }
}
```
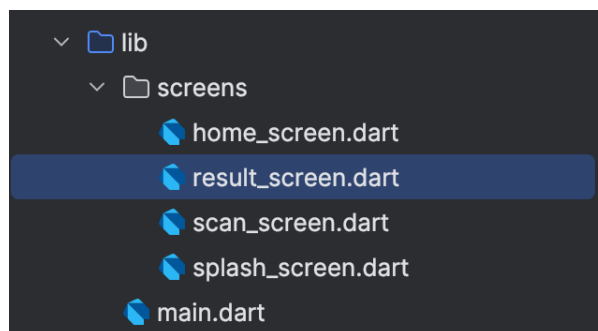
Listing 3: main.dart
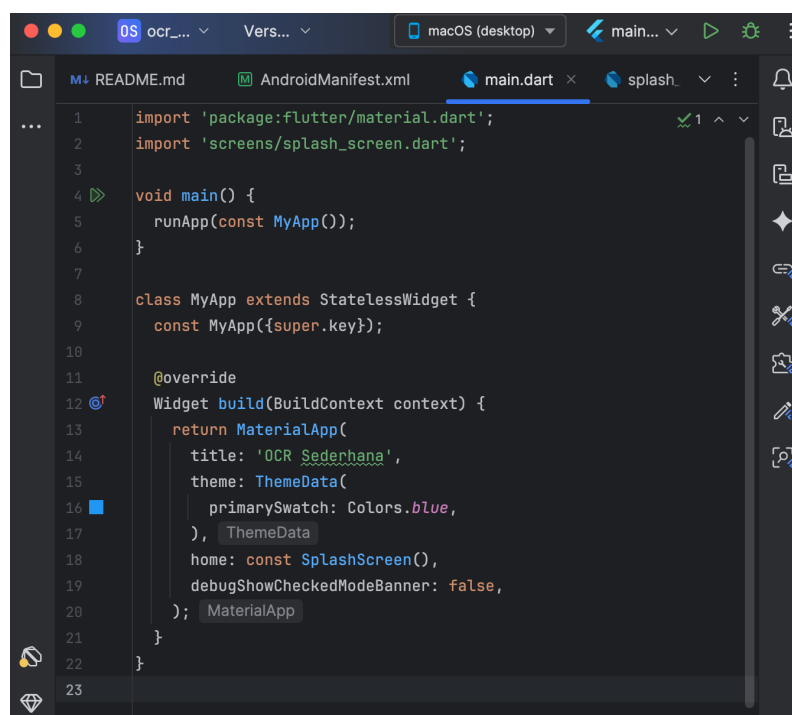
## 5.2. File: lib/screens/splash_screen.dart

```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
```

```dart
14    void initState() {
15      super.initState();
16      Timer(const Duration(seconds: 2), () {
17        Navigator.pushReplacement(
18          context,
19          MaterialPageRoute(builder: (_) => const HomeScreen()),
20        );
21      });
22    }
23
24    @override
25    Widget build(BuildContext context) {
26      return Scaffold(
27        backgroundColor: Colors.blue,
28        body: Center(
29          child: Column(
30            mainAxisAlignment: MainAxisAlignment.center,
31            children: const [
32              CircularProgressIndicator(color: Colors.white),
33              SizedBox(height: 20),
34              Text('OCR Scanner',
35                  style: TextStyle(color: Colors.white, fontSize:
    24)),
36            ],
37          ),
38        ),
39      );
40    }
41  }
```
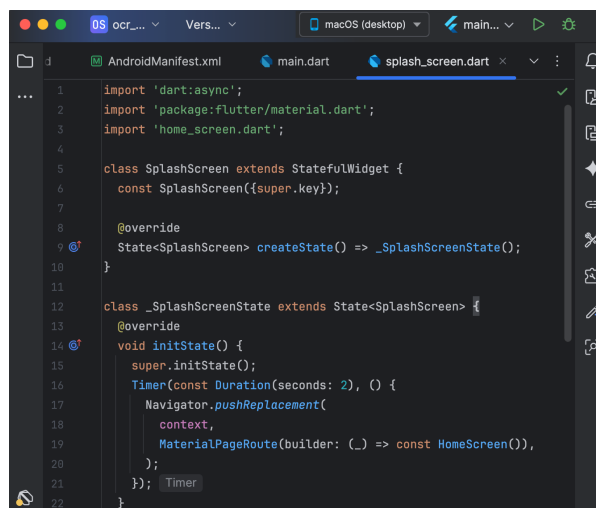
Listing 4: splash_screen.dart



6

### 5.3. File: lib/screens/home_screen.dart

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

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

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

```
 9        return Scaffold(
10          appBar: AppBar(title: const Text('Menu Utama')),
11          body: Center(
12            child: ElevatedButton(
13              onPressed: () {
14                Navigator.push(
15                  context,
16                  MaterialPageRoute(builder: (_) => const ScanScreen
    ()),
17                );
18              },
19              child: const Text('Mulai Scan Teks'),
20            ),
21          ),
22        );
23      }
24 }
```
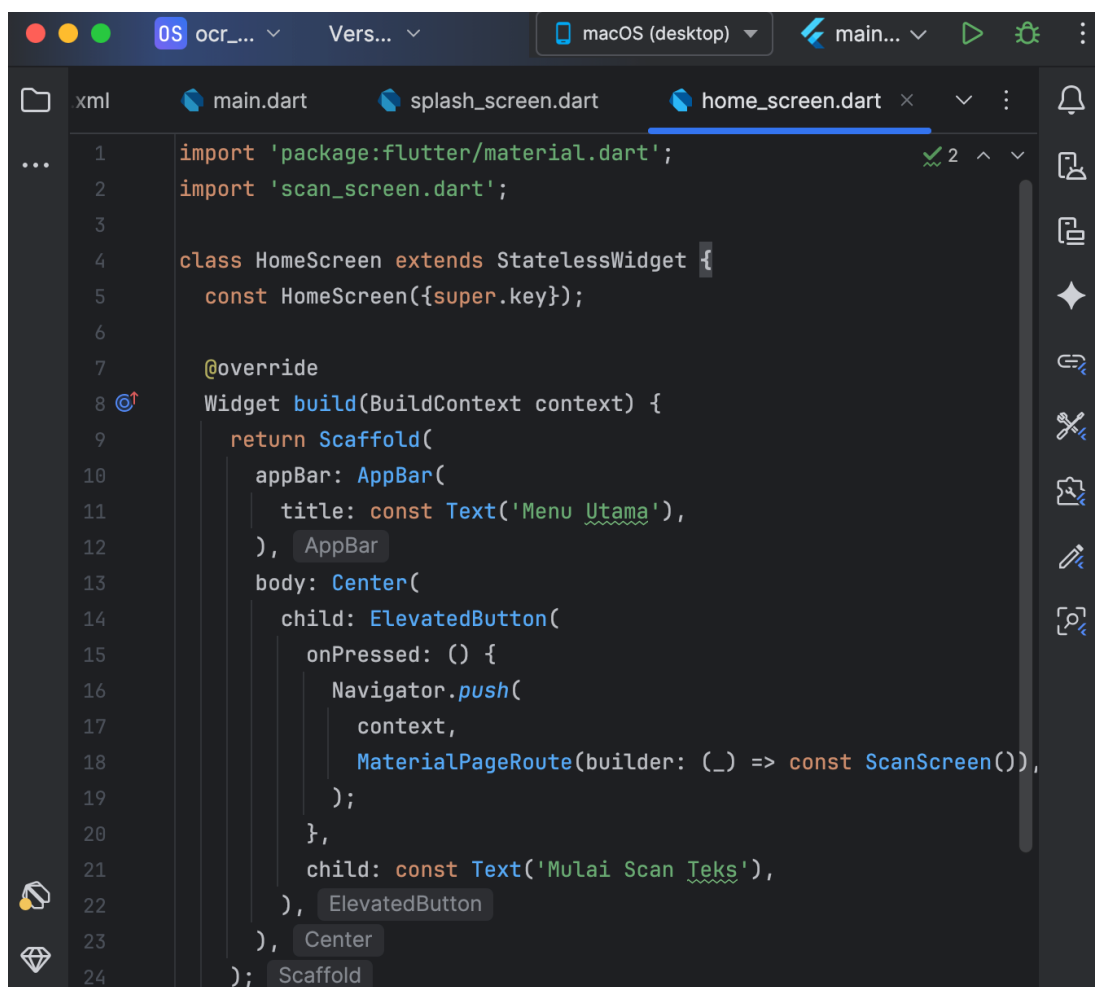
Listing 5: home_screen.dart

```
1   import 'package:flutter/material.dart';
2   import 'scan_screen.dart';
3
4   class HomeScreen extends StatelessWidget {
5     const HomeScreen({super.key});
6
7     @override
8     Widget build(BuildContext context) {
9       return Scaffold(
10        appBar: AppBar(
11          title: const Text('Menu Utama'),
12        ), AppBar
13        body: Center(
14          child: ElevatedButton(
15            onPressed: () {
16              Navigator.push(
17                context,
18                MaterialPageRoute(builder: (_) => const ScanScreen()),
19              );
20            },
21            child: const Text('Mulai Scan Teks'),
22          ), ElevatedButton
23        ), Center
24      ); Scaffold
```

## 5.4. File: lib/screens/scan_screen.dart

```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;
```

```dart
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(
```

```dart
                      const SnackBar(content: Text('Memproses OCR, mohon
    tunggu...'), duration: Duration(seconds: 2)));

         final XFile image = await _controller.takePicture();

         final ocrText = await _ocrFromFile(File(image.path));

         if (!mounted) return;
         Navigator.push(
           context,
           MaterialPageRoute(builder: (_) => ResultScreen(ocrText:
    ocrText)),
         );
       } catch (e) {
         if (!mounted) return;
         ScaffoldMessenger.of(context).showSnackBar(SnackBar(content
    : Text('Error saat mengambil/memproses foto: $e')));
       }
    }

    @override
    Widget build(BuildContext context) {
      if (!_controller.value.isInitialized) {
         return const Scaffold(body: Center(child:
    CircularProgressIndicator()));
      }

      return Scaffold(
         appBar: AppBar(title: const Text('Kamera OCR')),
         body: Column(
           children: [
             Expanded(
               child: AspectRatio(
                 aspectRatio: _controller.value.aspectRatio,
                 child: CameraPreview(_controller),
               ),
             ),
             Padding(
               padding: const EdgeInsets.all(16.0),
               child: ElevatedButton.icon(
                 onPressed: _takePicture,
```
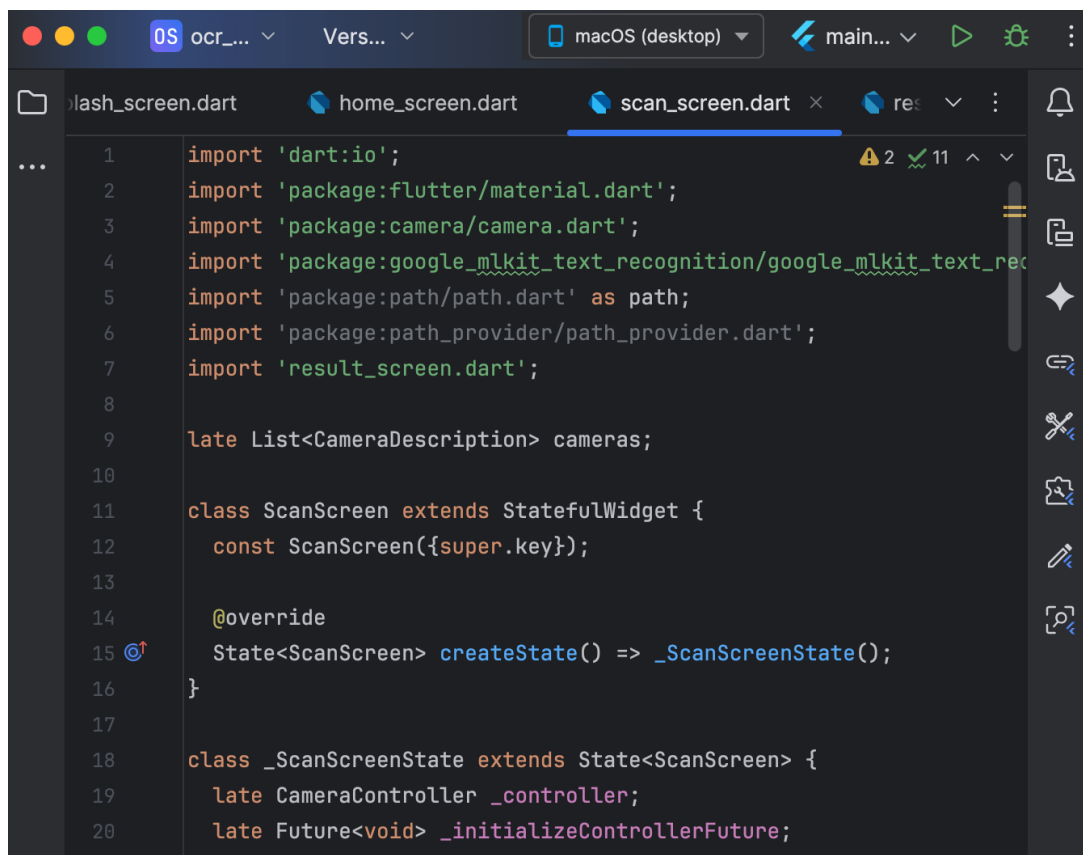
```
94            icon: const Icon(Icons.camera),
95            label: const Text('Ambil Foto & Scan')),
96          ),
97        ),
98      ],
99    ),
100   );
101  }
102 }
```

Listing 6: scan_screen.dart

```dart
import 'dart:io';
import 'package:flutter/material.dart';
import 'package:camera/camera.dart';
import 'package:google_mlkit_text_recognition/google_mlkit_text_rec
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;
  late Future<void> _initializeControllerFuture;
```

## 5.5. File: lib/screens/result screen.dart

```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),
          ),
        ),
      ),
    );
  }
}
```
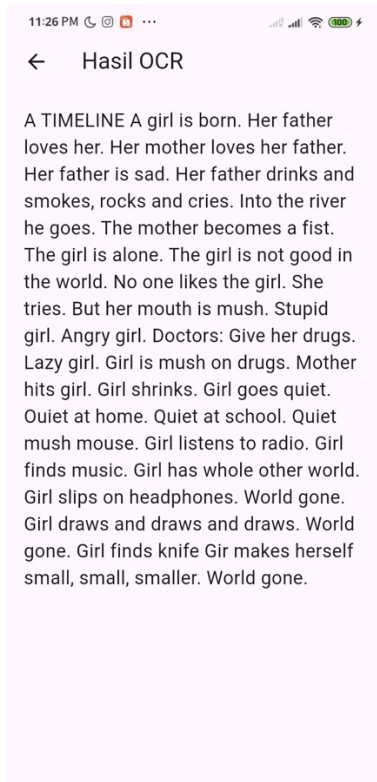
Listing 7: result_screen.dart
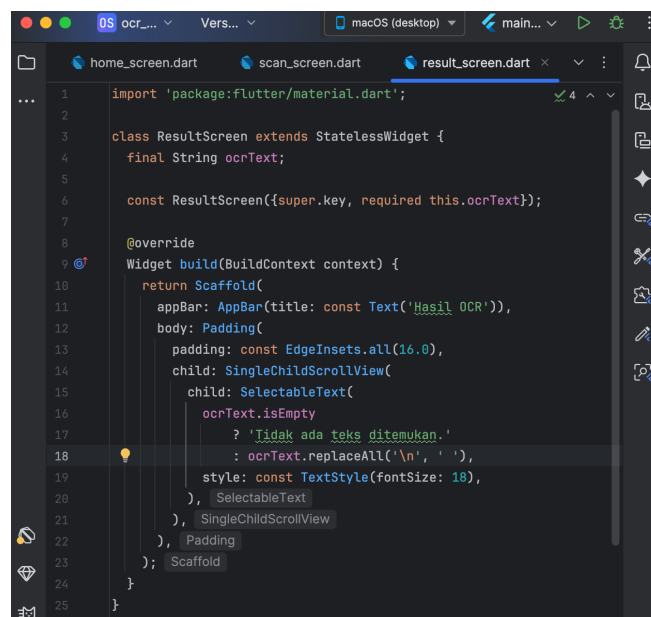
# 6. TUGAS PRAKTIKUM

1. Jalankan aplikasi di emulator atau HP.
2. Lakukan scan terhadap teks cetak (misal: buku, koran, atau layar HP).
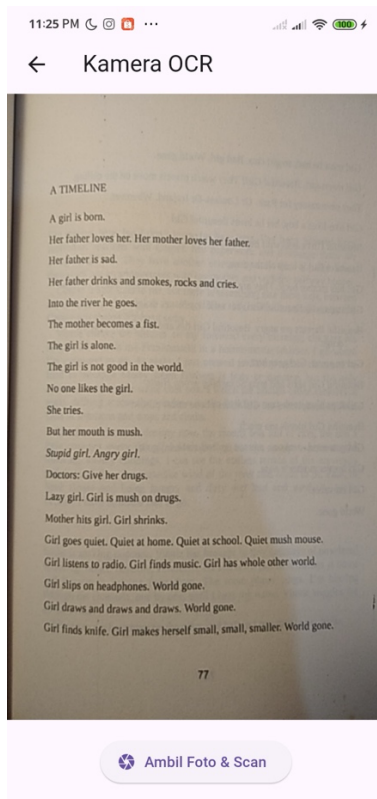


3. Amati hasil OCR yang muncul.

4. Jawab pertanyaan berikut:

   a. Apakah semua teks terbaca dengan akurat? Mengapa?

   # Tidak, tidak semua teks terbaca 100% akurat, meskipun hasilnya sangat baik.

   Jika membandingkan teks asli ("Kamera OCR") dengan teks hasil pemindaian ("Hasil OCR"), ditemukan adanya sedikit ketidakakuratan atau *error*:

   1. Kesalahan Minor (Typo) yaitu terdapat di baris terakhir, kata "Girl" terbaca sebagai "Gir" (Kalimat asli: *Girl finds knife. Girl makes herself...* menjadi Hasil OCR: *Girl finds knife Gir makes herself...*).

   2. Kesalahan Penggabungan Baris yaitu terdapat penggabungan baris pada paragraf yang mencantumkan sifat, di mana seharusnya baris "Stupid girl." dan "Angry girl." berada di baris terpisah, namun di hasil OCR terlihat penggabungan yang tidak sempurna dan sedikit pengulangan baris: *Stupid girl. Angry girl.*

   Alasannya : Ketidaksempurnaan kecil ini sering terjadi karena keterbatasan algoritma OCR dalam membedakan karakter yang sangat mirip atau menginterpretasikan tata letak, meskipun teks asli memiliki kualitas cetakan yang baik.

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

   # Fitur OCR berfungsi untuk mengubah informasi yang terperangkap dalam format visual (seperti gambar, dokumen fisik, atau scan) menjadi data digital yang dapat dicari, disunting, dan diproses.

   c. Sebutkan 2 contoh aplikasi nyata yang menggunakan OCR!

   # Aplikasi Pemindai Dokumen dan Catatan :

   Contoh : Aplikasi seperti Google Lens, Microsoft Lens, atau yang sedang Anda gunakan, yang memungkinkan pengguna memfoto halaman, papan tulis, atau dokumen, lalu mengekstrak teks di dalamnya.

   # Sistem *Know Your Customer* (KYC) atau Pendaftaran *Online* :

   Contoh : Aplikasi perbankan atau dompet digital yang menggunakan kamera ponsel untuk memindai KTP atau kartu identitas, secara otomatis membaca nama, NIK, dan alamat, lalu mengisi formulir pendaftaran tanpa perlu pengetikan manual.

# 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.

https://github.com/sittafatim4/ocr_sederhana

# 8. PENILAIAN

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

**Nilai Akhir** = Total Skor × 5

**Selamat mengerjakan!**