
UJIAN TENGAH SEMESTER (UTS) PRAKTIKUM

Mata Kuliah: Pemrograman Mobile - Aplikasi OCR

Semester: Ganjil Tahun Akademik: 2025/2026

Nama	Deanissa Sherly Sabilla
Kelas/NIM	SIB 3C/2341760187
Absen	08

Soal 1: Modifikasi Struktur Navigasi dan Aliran (30 Poin)

Tujuan: Menyederhanakan alur navigasi dan meningkatkan pengalaman pengguna di HomeScreen.

1. Pengubahan Navigasi Home (15 Poin):

- Ubah **ElevatedButton** di HomeScreen (*lib/screens/home_screen.dart*) menjadi Widget, ListTile.
- Atur ListTile: leading: Icon(Icons.camera_alt, color: Colors.blue); title: Text('Mulai Pindai Teks Baru').
- Fungsi onTap harus menggunakan Navigator.push() untuk ke ScanScreen.

```
15  child: Card(  
16    margin: const EdgeInsets.all(20),  
17    //Modifikasi menjadi ListTile  
18    child: ListTile(  
19      leading: const Icon(Icons.camera_alt, color: Colors.blue),  
20      title: const Text('Mulai Pindai Teks Baru'),  
21      onTap: () {  
22        Navigator.push(  
23          context,  
24          MaterialPageRoute(builder: (_) => const ScanScreen()),  
25        );  
26      },  
27    ), // ListTile  
28  ), // Card  
29  ), // Center  
30 ); // Scaffold
```

2. Teks Utuh dan Navigasi Balik (15 Poin):

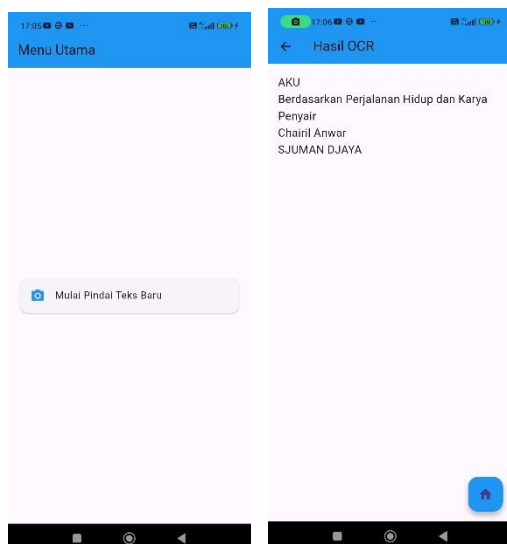
- Di ResultScreen (*lib/screens/result_screen.dart*), hapus fungsi `ocrText.replaceAll` (agar hasil teks ditampilkan dengan baris baru (\n) yang utuh).
- Tambahkan `FloatingActionButton` dengan ikon `Icons.home`.
- Ketika tombol ditekan, navigasi harus kembali langsung ke HomeScreen menggunakan `Navigator.pushAndRemoveUntil()` (atau metode yang setara) untuk *menghapus semua halaman di atasnya dari stack navigasi*.

```

18         child: SingleChildScrollView(
19           child: SelectableText(
20             ocrText.isEmpty
21               ? 'Tidak ada teks ditemukan.'
22               : ocrText, // baris baru (\n) dipertahankan
23             style: const TextStyle(
24               fontSize: 18,
25               height: 1.5,
26             ), // TextStyle
27           ), // SelectableText
28         ), // SingleChildScrollView
29       ), // Padding
30
31       // Tambahkan FloatingActionButton
32       floatingActionButton: FloatingActionButton(
33         backgroundColor: Colors.blue,
34         onPressed: () {
35           Navigator.pushAndRemoveUntil(
36             context,
37             MaterialPageRoute(builder: (_) => const HomeScreen()),
38             (route) => false, // Menghapus semua halaman sebelumnya
39           );
40         },
41         child: const Icon(Icons.home),
42       ), // FloatingActionButton
43     ); // Scaffold

```

Hasil :



Hasil akan muncul ikon kamera dan teks ‘Mulai Pindai Teks Baru’

Github :



Soal 2: Penyesuaian Tampilan dan Penanganan State/Error (40 Poin)

Tujuan: Memperbaiki tampilan loading dan memberikan feedback error yang lebih jelas.

1. Custom Loading Screen di ScanScreen (20 Poin):

- Di ScanScreen (lib/screens/scan_screen.dart), modifikasi tampilan loading yang muncul sebelum kamera siap (if (!controller.value.isInitialized)) :
- Latar Belakang: Scaffold(backgroundColor: Colors.grey[900]).
- Isi: Di dalam Center, tampilkan Column berisi CircularProgressIndicator(color: Colors.yellow).
- Di bawah indikator, tambahkan Text('Memuat Kamera... Harap tunggu.', style: TextStyle(color: Colors.white, fontSize: 18)).

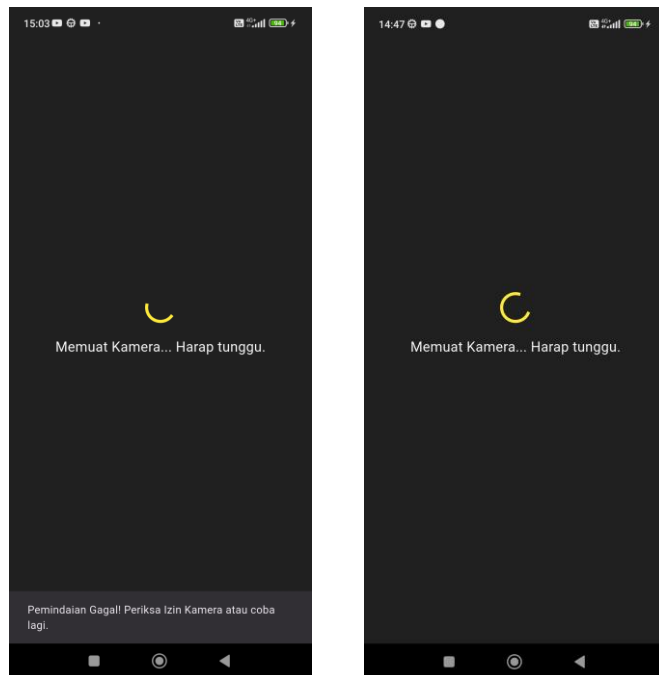
```
92 @override
93 Widget build(BuildContext context) {
94   //Tampilan Loading sebelum kamera siap
95   if (_controller == null || !_controller!.value.isInitialized) {
96     return Scaffold(
97       backgroundColor: Colors.grey[900],
98       body: const Center(
99         child: Column(
100           mainAxisAlignment: MainAxisAlignment.min,
101           children: [
102             CircularProgressIndicator(color: Colors.yellow),
103             SizedBox(height: 20),
104             Text(
105               'Memuat Kamera... Harap tunggu.',
106               style: TextStyle(color: Colors.white, fontSize: 18),
107             ), // Text
108           ],
109         ), // Column
110       ), // Center
111     ); // Scaffold
112   }
```

2. Spesifikasi Pesan Error (20 Poin):

- Di fungsi _takePicture() pada ScanScreen, modifikasi blok catch (e) untuk mengubah pesan error pada Snackbar.
- Pesan Snackbar harus berbunyi: "Pemindaian Gagal! Periksa Izin Kamera atau coba lagi." (Hilangkan variabel error (\$e)).

```
    } catch (e) {
      debugPrint('Error saat mengambil foto: $e');
      //Pesan Snackbar diperbarui sesuai instruksi
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(
          content: Text(
            'Pemindaian Gagal! Periksa Izin Kamera atau coba lagi.',
          ), // Text
        ), // SnackBar
      );
    }
  }
```

Hasil:



Hasil akan muncul loading berwarna kuning, dan menampilkan teks ‘Memuat Kamera.... Harap Tunggu’. Dan jika kamera tidak diizinkan untuk akses, maka dibawah akan muncul alert ‘Pemindaian Gagal! Periksa Izin Kamera atau coba lagi.’

Github:



Soal 3: Implementasi Plugin Text-to-Speech (TTS) (30 Poin)

Tujuan: Mengintegrasikan fitur membaca teks secara lisan menggunakan plugin flutter_tts.

1. Instalasi Plugin (5 Poin):

- Tambahkan plugin flutter_tts ke dalam file pubspec.yaml (gunakan versi terbaru yang kompatibel).
- Jalankan flutter pub get.

```
34 google_mlkit_text_recognition: ^0.11.0
35 camera: ^0.10.5+5
36 path_provider: ^2.1.2
37 path: ^1.8.3
38 flutter_tts: ^4.2.3
39
40
```

PROBLEMS 2 OUTPUT DEBUG CONSOLE **TERMINAL** PORTS QUERY RESULTS

```
PS C:\src\ocr_sederhana> flutter pub get
Resolving dependencies...
Downloading packages...
  camera 0.10.6 (0.11.2 available)
  characters 1.4.0 (1.4.1 available)
  flutter_lints 5.0.0 (6.0.0 available)
! google_mlkit_commons 0.7.1 (overridden) (0.11.0 available)
  google_mlkit_text_recognition 0.11.0 (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)
  test_api 0.7.6 (0.7.7 available)
Got dependencies!
9 packages have newer versions incompatible with dependency constraints.
Try 'flutter pub outdated' for more information.
PS C:\src\ocr_sederhana>
```

2. Konversi Widget dan Inisialisasi (10 Poin):

- Ubah ResultScreen dari StatelessWidget menjadi StatefulWidget.
- Di initState(), inisialisasi FlutterTts dan atur bahasa pembacaan menjadi Bahasa Indonesia.
- Implementasikan dispose() untuk menghentikan mesin TTS saat halaman ditutup.

```
8 const ResultScreen({super.key, required this.ocrText});
9
10 @override
11 State<ResultScreen> createState() => _ResultScreenState();
12 }
13
14 class _ResultScreenState extends State<ResultScreen> {
15   late FlutterTts flutterTts;
16
17   @override
18   void initState() {
19     super.initState();
20
21     //Inisialisasi FlutterTts dan atur bahasa Indonesia
22     flutterTts = FlutterTts();
23     _initTts();
24   }
25
26   Future<void> _initTts() async {
27     await flutterTts.setLanguage("id-ID"); // Bahasa Indonesia
28     await flutterTts.setSpeechRate(0.5); // Kecepatan bicara
29     await flutterTts.setPitch(1.0); // Nada suara normal
30   }
31
32   @override
33   void dispose() {
34     // Hentikan TTS saat halaman ditutup
35     flutterTts.stop();
36     super.dispose();
37   }
38
39   Future<void> _speak() async {
40     if (widget.ocrText.isNotEmpty) {
41       await flutterTts.speak(widget.ocrText);
42     }
43   }
44 }
```

```

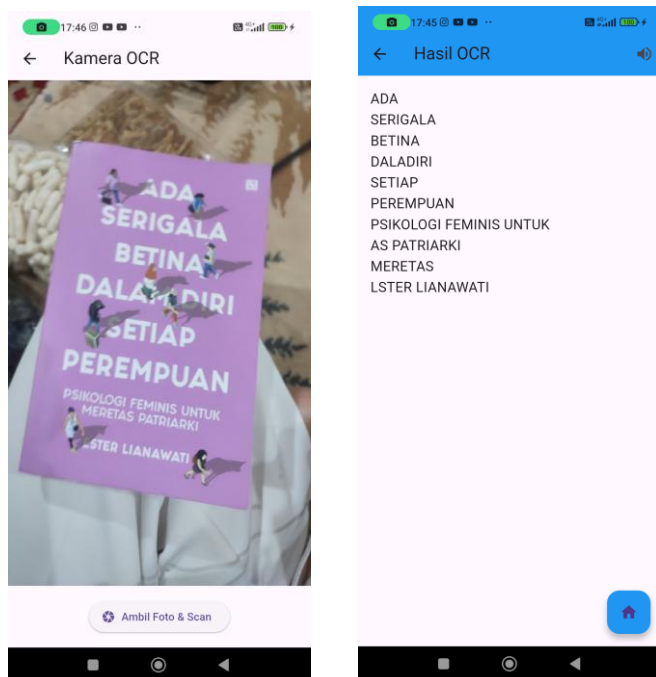
67 //Dua tombol: Home dan Baca Teks
68 FloatingActionButton: Column(
69   mainAxisAlignment: MainAxisAlignment.end,
70   children: [
71     FloatingActionButton(
72       heroTag: "speak",
73       backgroundColor: Colors.green,
74       onPressed: _speak,
75       child: const Icon(Icons.volume_up),
76     ), // FloatingActionButton
77     const SizedBox(height: 10),
78     FloatingActionButton(
79       heroTag: "home",
80       backgroundColor: Colors.blue,
81       onPressed: () {
82         Navigator.pushAndRemoveUntil(
83           context,
84           MaterialPageRoute(builder: (_) => const HomeScreen()),
85           (route) => false,
86         );
87       },
88       child: const Icon(Icons.home),
89     ), // FloatingActionButton
90   ],
91 );

```

3. Fungsionalitas Pembacaan (15 Poin):

- Tambahkan FloatingActionButton kedua di ResultScreen (atau ganti AppBar dengan action button) dengan ikon Icons.volume_up.
- Ketika tombol ditekan, panggil fungsi speak() pada FlutterTts untuk membacakan seluruh isi ocrText.

Hasil:



Hasil yang sudah di Scan, maka akan muncul translate dan terdapat ikon speaker pada kiri atas, jika di tekan, teks akan otomatis keluar suara.

Github:

