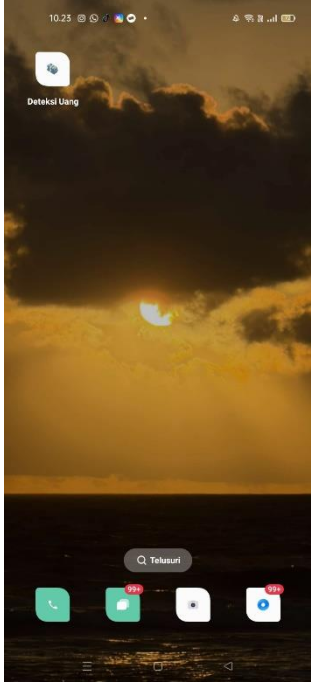
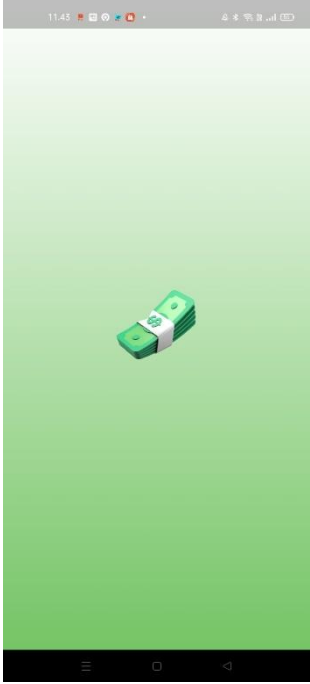


LAMPIRAN

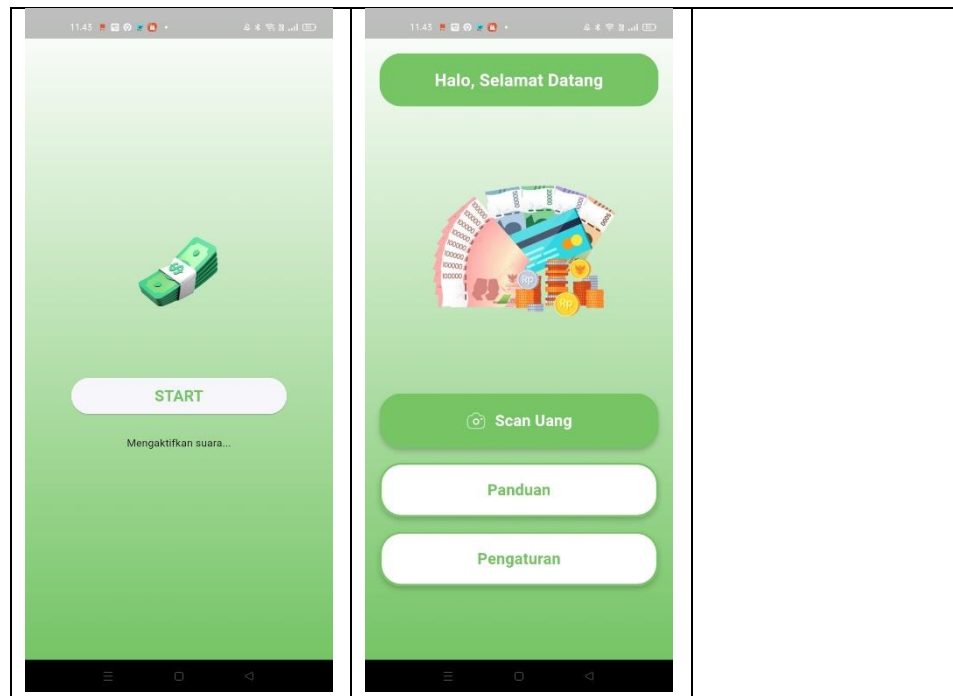
1. Hasil Pengujian Fungsional Aplikasi

a. Pengujian pada menu *splash*

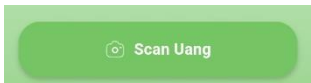
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Tampilan splash screen 	Menampilkan logo aplikasi sebagai pembuka aplikasi mulai berjalan. 	Berhasil

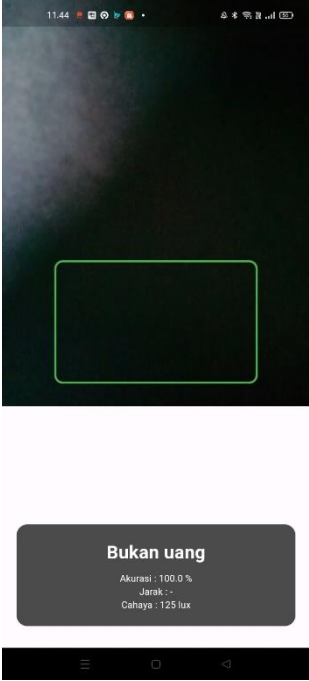
b. Pengujian pada menu masuk

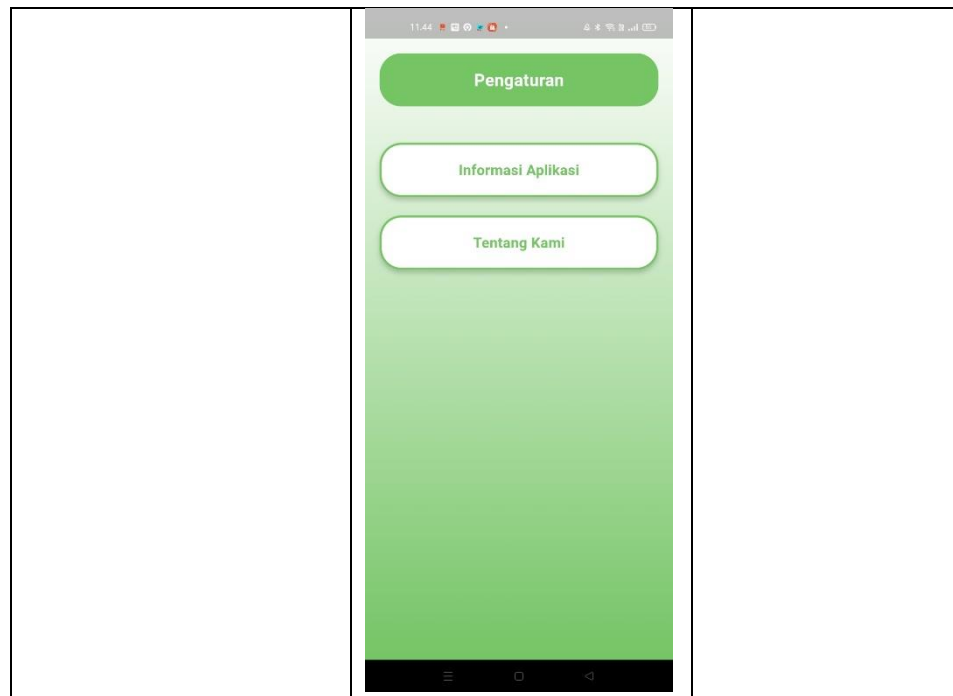
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses button masuk	Masuk berhasil dan mengakses dashboard	Berhasil



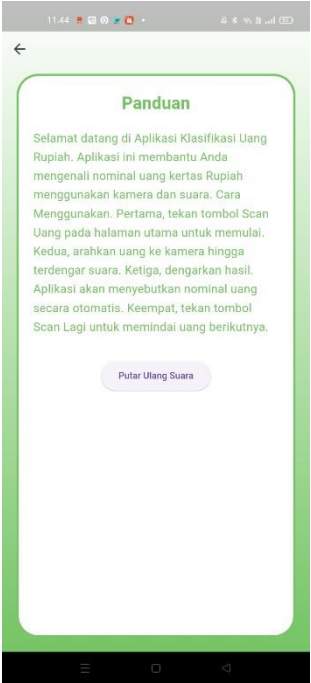
c. Pengujian pada menu dashboard

Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses tampilan dashboard 	Menampilkan menu utama 	Berhasil
Akses scan uang 	Menampilkan kamera, mengeluarkan hasil teks dan suara	Berhasil

		
Akses panduan 	Menampilkan teks dan suara panduan 	Berhasil
Akses pengaturan 	Menampilkan button 'Informasi Aplikasi' dan 'Tentang Kami'	Berhasil



d. Pengujian pada menu panduan

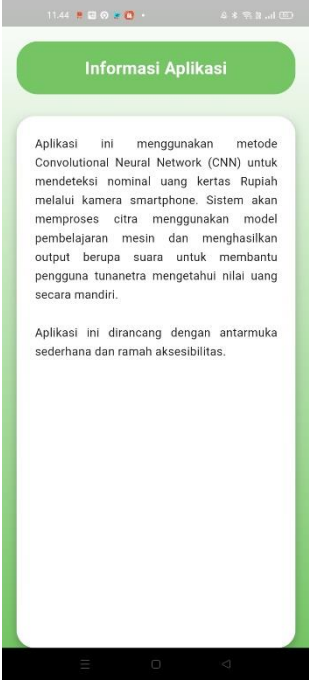
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses panduan 	Menampilkan teks dan suara panduan 	Berhasil

e. Pengujian pada menu pengaturan

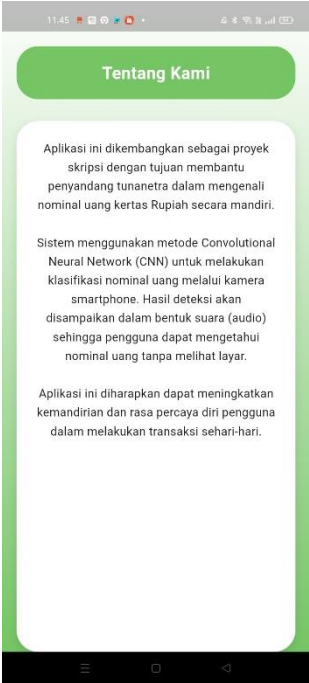
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses pengaturan 	Menampilkan button 'Informasi Aplikasi' dan 'Tentang Kami' 	Berhasil

f. Pengujian pada menu informasi aplikasi

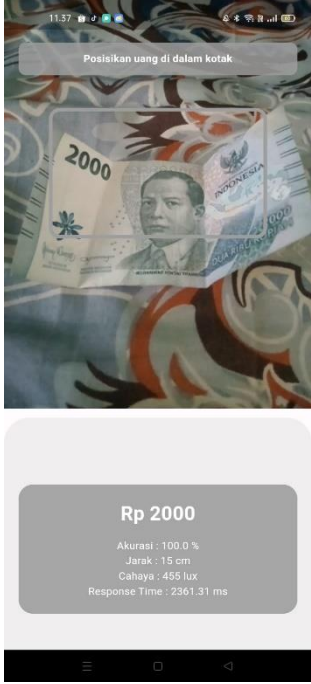
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses informasi aplikasi 	Menampilkan teks dan suara informasi aplikasi.	Berhasil

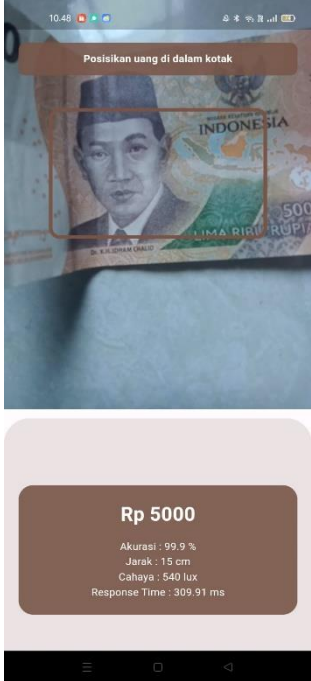
		
--	--	--

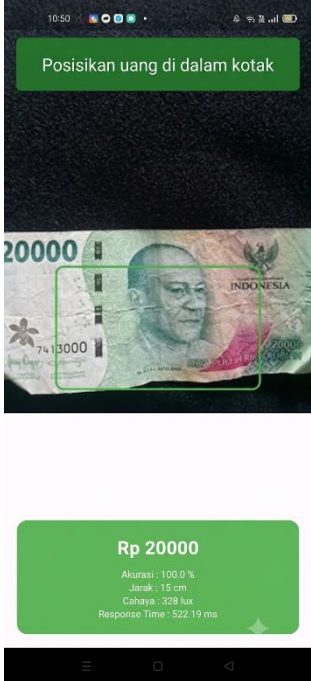
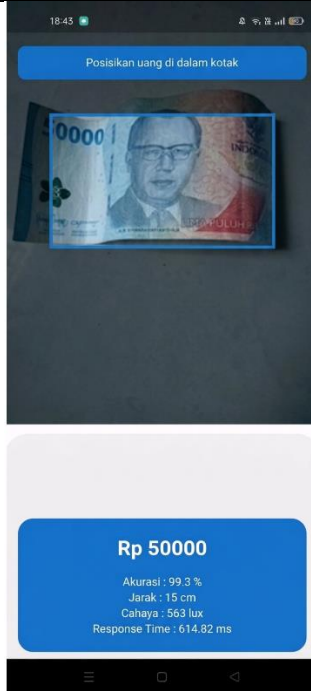
g. Pengujian pada menu tentang kami

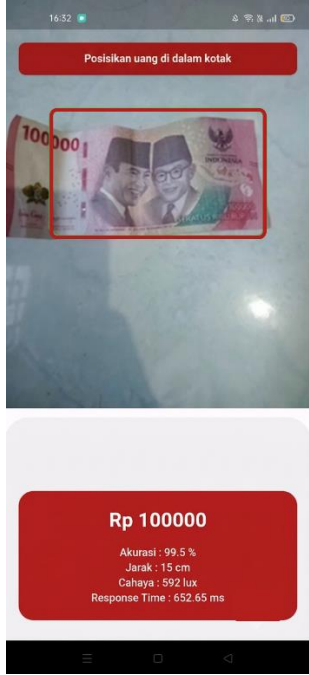
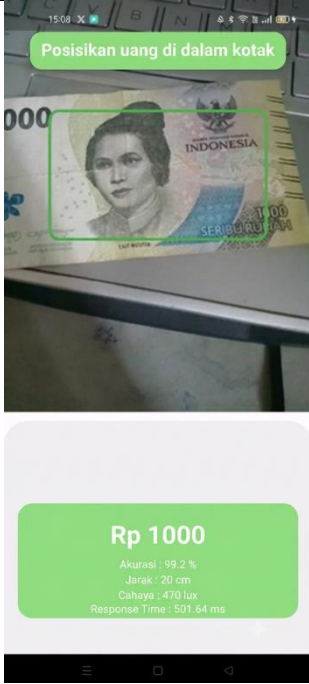
Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian
Akses tentang kami 	Menampilkan teks dan suara tentang kami 	Berhasil

2. Pengujian Sistem Berdasarkan Jarak

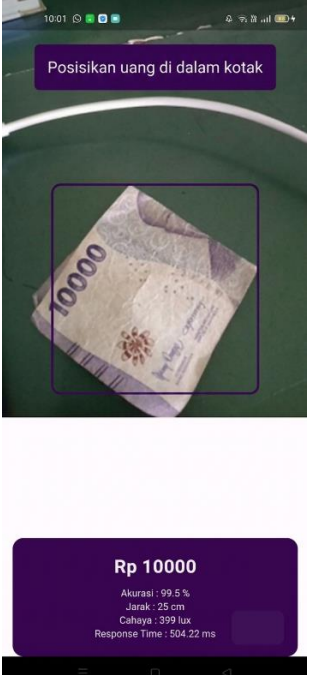
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
± 10 cm	15 cm	Rp 1.000		99.5%	Berhasil
	15 cm	Rp 2.000		100.0%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	15 cm	Rp 5.000		99.9%	Berhasil
	15 cm	Rp 10.000		99.2%	Berhasil

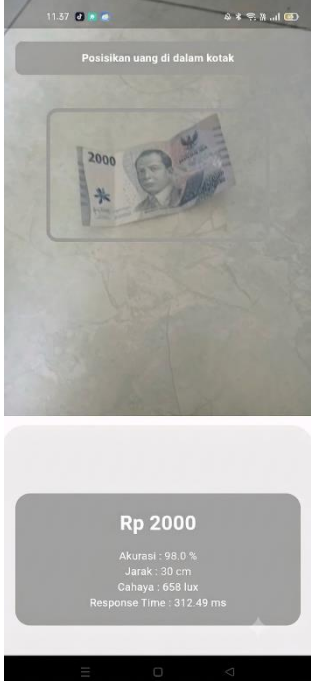
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	15 cm	Rp 20.000		100.0%	Berhasil
	15 cm	Rp 50.000		99.3%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	15 cm	Rp 100.000		99.5%	Berhasil
± 20 cm	20 cm	Rp 1.000		99.2%	Berhasil

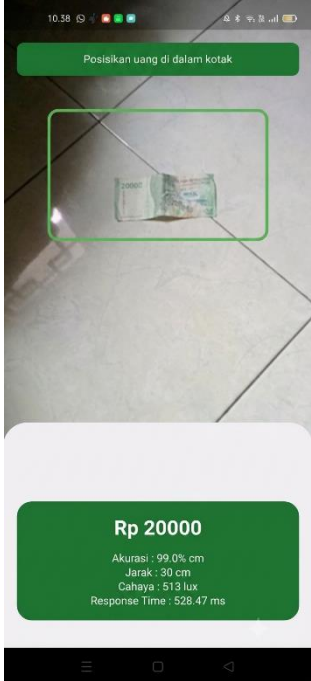
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	20 cm	Rp 2.000		99.5%	Berhasil
	20 cm	Rp 5.000		99.2%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	25 cm	Rp 10.000		99.5%	Berhasil
	23 cm	Rp 20.000		98.8%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	25 cm	Rp 50.000		99.0%	Berhasil
	17 cm	Rp 100.000		99.1%	Berhasil

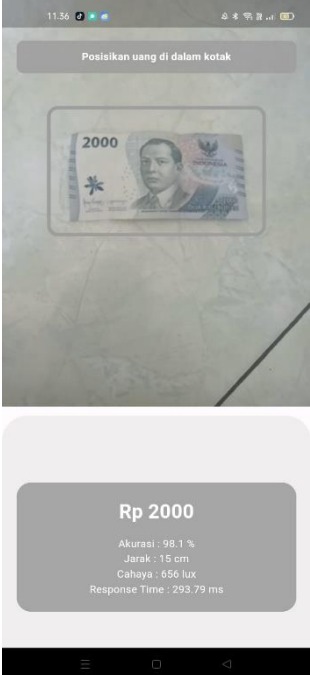
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
± 30 cm	30 cm	Rp 1.000		98.2%	Berhasil
	30 cm	Rp 2.000		98.0%	Berhasil

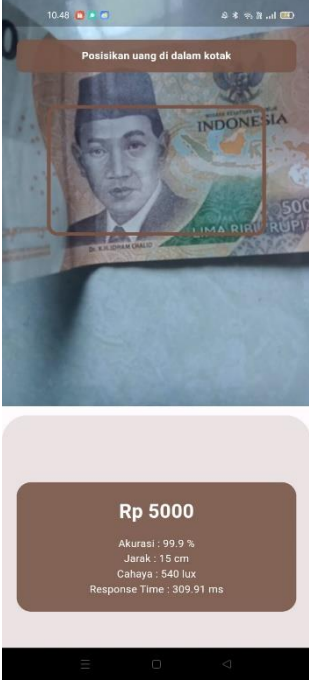
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	30 cm	Rp 5.000		97.5%	Berhasil
	31 cm	Rp 10.000		99.5%	Berhasil

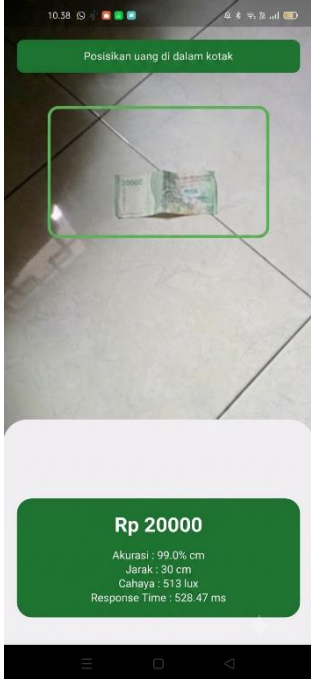
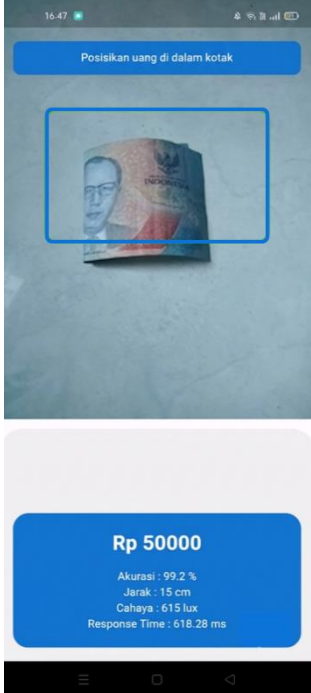
Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	30 cm	Rp 20.000		99.0%	Berhasil
	30 cm	Rp 50.000		99.2%	Berhasil

Jarak Kamera		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
	30 cm	Rp 100.000		99.0%	Berhasil
> 40 cm	Rp 1.000	Tidak Terdeteksi			Gagal
	Rp 2.000				
	Rp 5.000				
	Rp 10.000				
	Rp 20.000				
	Rp 50.000				
	Rp 100.000				

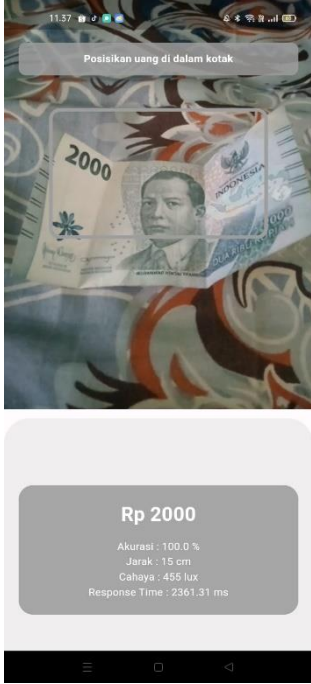
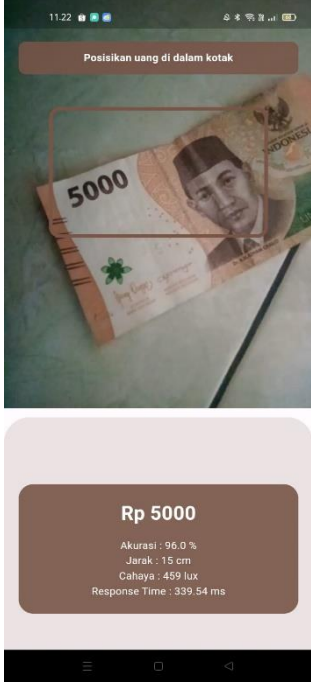
3. Pengujian Sistem Berdasarkan Cahaya

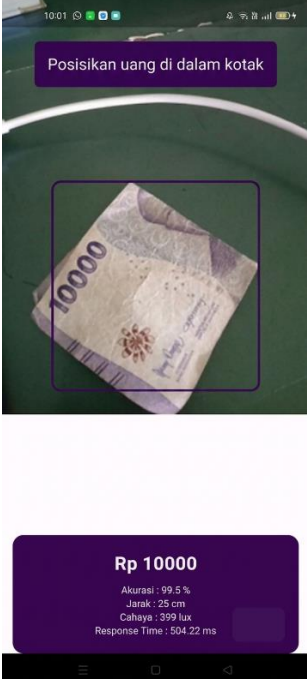
Cahaya		Nominal	Hasil Deteksi Sistem	Akurasi	Ket
> 500 lux	612 lux	Rp 1.000		99.5%	Berhasil
	656 lux	Rp 2.000		98.1%	Berhasil

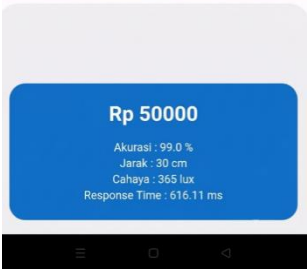
	540 lux	Rp 5.000		99.9%	Berhasil
	590 lux	Rp 10.000		99.5%	Berhasil

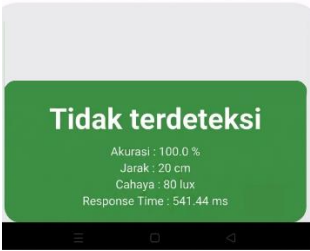
	513 lux	Rp 20.000		99.0%	Berhasil
	615 lux	Rp 50.000		99.2%	Berhasil

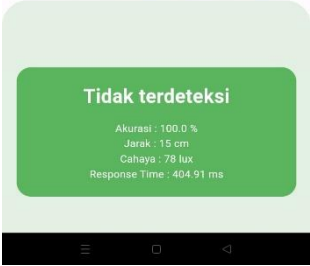
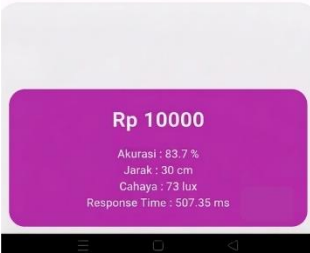
	551 lux	Rp 100.000		99.1%	Berhasil
100 – 500 lux	442 lux	Rp 1.000		100.0%	Berhasil

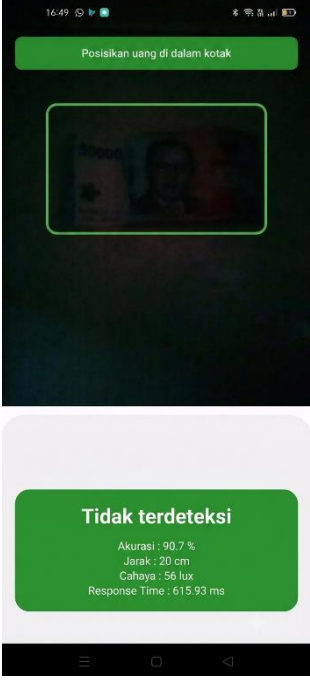
	455 lux	Rp 2.000		100.0%	Berhasil
	459 lux	Rp 5.000		96.0%	Berhasil

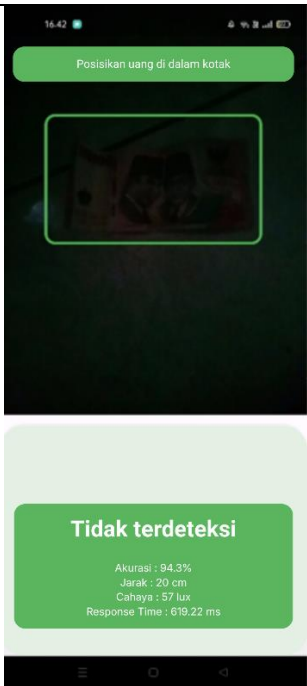
	399 lux	Rp 10.000		99.5%	Berhasil
	405 lux	Rp 20.000		98.8%	Berhasil

	365 lux	Rp 50.000	 	99.0%	Berhasil
	401 lux	Rp 100.000	 	99.7%	Berhasil

10 — 100 lux	80 lux	Rp 1.000	 	100.0%	Gagal
	79 lux	Rp 2.000	 	100.0%	Gagal

	78 lux	Rp 5.000	 	100.0%	Gagal
	73 lux	Rp 10.000	 	83.7%	Berhasil

	75 lux	Rp 20.000		84.0%	Berhasil
	56 lux	Rp 50.000		90.7%	Gagal

	57 lux	Rp 100.000		94.3%	Gagal
< 10 lux	Rp 1.000		Tidak Terdeteksi		Gagal
	Rp 2.000				
	Rp 5.000				
	Rp 10.000				
	Rp 20.000				
	Rp 50.000				
	Rp 100.000				

4. Kode program

a. main.dart

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

Future<void> main() async {
  WidgetsFlutterBinding.ensureInitialized();
  runApp(const MyApp());
}

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

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      debugShowCheckedModeBanner: false,
      home: SplashPage(),
    );
  }
}
```

b. splash.dart

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

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

  @override
  State<SplashPage> createState() => _SplashPageState();
}

class _SplashPageState extends State<SplashPage> {
  @override
  void initState() {
    super.initState();

    // Splash tampil 5 detik
    Timer(const Duration(seconds: 3), () {
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(
          builder: (context) => const MulaiPage(),
        ),
      );
    });
  }

  @override
  Widget build(BuildContext context) {
    final size = MediaQuery.of(context).size;

    return Scaffold(
      body: Container(
        width: size.width,
        height: size.height,
        decoration: const BoxDecoration(
          gradient: LinearGradient(
            begin: Alignment.topCenter,
            end: Alignment.bottomCenter,
            colors: [
              Colors.white,
              Color(0xFF75C565),
            ],
          ),
        ),
      child: Center(
        child: Image.asset(
          'assets/img/money.png',
          width: size.width * 0.35,
          fit: BoxFit.contain,
        ),
      ),
    );
  }
}
```

```

        ),
    ),
);
}
}

```

c. mulai.dart

```

import 'package:flutter/material.dart';
import 'package:speech_to_text/speech_to_text.dart' as stt;
import 'dashboard.dart';

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

  @override
  State<MulaiPage> createState() => _MulaiPageState();
}

class _MulaiPageState extends State<MulaiPage> {
  late stt.SpeechToText _speech;
  bool _isListening = false;

  @override
  void initState() {
    super.initState();
    _speech = stt.SpeechToText();
    _initSpeech();
  }

  Future<void> _initSpeech() async {
    bool available = await _speech.initialize();
    if (available && mounted) {
      _startListening();
    }
  }

  void _startListening() {
    if (!_isListening) {
      setState(() => _isListening = true);

      _speech.listen(
        localeId: "id_ID",
        onResult: (result) async {
          String command =
result.recognizedWords.toLowerCase();

          if (command.contains("mulai") ||
              command.contains("masuk") ||
              command.contains("start")) {

```

```

        await _speech.stop();
        await _speech.cancel();

        Future.delayed(const Duration(milliseconds: 300),
() {
            if (mounted) {
                _goToDashboard();
            }
        });
    },
);
}

void _goToDashboard() {
    Navigator.pushReplacement(
        context,
        MaterialPageRoute(
            builder: (context) => const DashboardPage(),
        ),
    );
}

@override
void dispose() {
    _speech.stop();
    _speech.cancel();
    super.dispose();
}

@override
Widget build(BuildContext context) {
    final size = MediaQuery.of(context).size;

    return Scaffold(
        body: Container(
            width: size.width,
            height: size.height,
            decoration: const BoxDecoration(
                gradient: LinearGradient(
                    begin: Alignment.topCenter,
                    end: Alignment.bottomCenter,
                    colors: [
                        Colors.white,
                        Color(0xFF75C565),
                    ],
                ),
        ),
        child: Column(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [

```

```

        Image.asset(
          'assets/img/money.png',
          width: size.width * 0.35,
        ),

        SizedBox(height: size.height * 0.08),

        SizedBox(
          width: size.width * 0.7,
          height: 50,
          child: ElevatedButton(
            onPressed: () async {
              await _speech.stop();
              await _speech.cancel();
              _goToDashboard();
            },
            style: ElevatedButton.styleFrom(
              backgroundColor: Colors.white,
              shape: RoundedRectangleBorder(
                borderRadius: BorderRadius.circular(35),
              ),
            ),
            child: const Text(
              'START',
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.bold,
                color: Color(0xFF75C565),
              ),
            ),
          ),
        ),
      ),
    ),
  ),
  const SizedBox(height: 30),

  Text(
    _isListening
      ? "Mendengarkan... Ucapkan 'Mulai'"
      : "Mengaktifkan suara...",
    style: const TextStyle(fontSize: 16),
  ),
),
],
),
),
);
}
}

```

d. dashboard.dart

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:flutter_tts/flutter_tts.dart';
import 'package:speech_to_text/speech_to_text.dart' as stt;

import 'hasil.dart';
import 'panduan.dart';
import 'pengaturan.dart';

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

  @override
  State<DashboardPage> createState() => _DashboardPageState();
}

class _DashboardPageState extends State<DashboardPage> {
  final FlutterTts tts = FlutterTts();
  late stt.SpeechToText _speech;

  bool isListening = false;
  bool isSpeaking = false;
  bool isActive = false;

  String statusText = "Menunggu perintah...";
  String lastSpoken = "";

  @override
  void initState() {
    super.initState();
    initAll();
  }

  Future<void> initAll() async {
    _speech = stt.SpeechToText();
    await _speech.initialize();

    await tts.setLanguage("id-ID");
    await tts.setSpeechRate(0.5);
    await tts.awaitSpeakCompletion(true);

    await speak("Aplikasi siap. Ucapkan halo aplikasi untuk
mulai");

    startListening();
  }

  Future<void> speak(String text) async {
    if (text == lastSpoken) return; // 🔥 anti spam
    lastSpoken = text;

    setState(() {
      isSpeaking = true;
      statusText = text;
    });
  }
}
```

```

    });

    await tts.stop();
    await tts.speak(text);

    setState(() => isSpeaking = false);
  }

  void startListening() {
    if (isListening) return;

    setState(() {
      isListening = true;
      statusText = "Mendengarkan...";
    });

    _speech.listen(
      localeId: "id_ID",
      listenMode: stt.ListenMode.dictation,
      listenFor: const Duration(seconds: 30),
      pauseFor: const Duration(seconds: 5),
      cancelOnError: true,
      onResult: (result) async {
        if (!result.finalResult) return;

        String command = result.recognizedWords.toLowerCase();
        await processCommand(command);
      },
    );
  }

  Future<void> processCommand(String command) async {
    await _speech.stop();
    setState(() => isListening = false);

    HapticFeedback.mediumImpact();

    // WAKE WORD
    if (!isActive) {
      if (command.contains("halo")) {
        isActive = true;
        await speak(
          "Silakan pilih menu. Scan uang, panduan, atau pengaturan");
      } else {
        startListening();
      }
      return;
    }

    if (command.contains("scan") || command.contains("kamera"))
    {
      await navigate("Membuka kamera", const HasilPage());
    }
  }

```

```

    } else if (command.contains("panduan")) {
        await navigate("Membuka panduan", const PanduanPage());
    } else if (command.contains("pengaturan")) {
        await navigate("Membuka pengaturan", const
PengaturanPage());
    } else if (command.contains("kembali")) {
        if (Navigator.canPop(context)) {
            Navigator.pop(context);
            await speak("Kembali");
        } else {
            await speak("Sudah di dashboard");
        }
    } else {
        await speak("Perintah tidak dikenali");
    }

    startListening();
}

Future<void> navigate(String message, Widget page) async {
    await speak(message);

    if (!mounted) return;

    await Navigator.push(
        context,
        MaterialPageRoute(builder: (_) => page),
    );

    await speak("Kembali ke dashboard");
    startListening();
}

@override
void dispose() {
    _speech.stop();
    tts.stop();
    super.dispose();
}

// 🔥 BUTTON FULL AKSESIBILITAS
Widget buildButton(String text, VoidCallback onTap) {
    return Padding(
        padding: const EdgeInsets.symmetric(horizontal: 20,
vertical: 10),
        child: GestureDetector(
            onTapDown: (_) async {
                // 🗣️ 1x sentuh → baca
                await tts.stop();
                await tts.speak(text);
            },
            onTap: () async {

```

```

        // 📢 2x → eksekusi
        HapticFeedback.heavyImpact();

        await tts.stop();
        await tts.speak("Membuka $text");

        await Future.delayed(const Duration(milliseconds:
300));

        onTap();
      },
      child: Container(
        height: 70,
        decoration: BoxDecoration(
          color: Colors.white,
          borderRadius: BorderRadius.circular(25),
          border: Border.all(color: const Color(0xFF4CAF50),
width: 2),
          boxShadow: const [
            BoxShadow(
              blurRadius: 5,
              color: Colors.black12,
              offset: Offset(0, 3),
            )
          ],
        ),
      child: Center(
        child: Text(
          text,
          style: const TextStyle(
            fontSize: 20,
            fontWeight: FontWeight.bold,
            color: Color(0xFF4CAF50),
          ),
        ),
      ),
    ),
  ),
),
);
}

@override
Widget build(BuildContext context) {
  final size = MediaQuery.of(context).size;

  return Scaffold(
    body: Container(
      width: size.width,
      height: size.height,
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          colors: [Colors.white, Color(0xFF4CAF50)],

```

```

        begin: Alignment.topCenter,
        end: Alignment.bottomCenter,
    ),
),
child: SafeArea(
    child: Column(
        children: [
            const SizedBox(height: 20),

            Container(
                padding: const EdgeInsets.all(20),
                margin: const EdgeInsets.symmetric(horizontal:
20),

                decoration: BoxDecoration(
                    color: const Color(0xFF4CAF50),
                    borderRadius: BorderRadius.circular(30),
                ),
                child: const Center(
                    child: Text(
                        "Dashboard Aksesibilitas",
                        style: TextStyle(
                            fontSize: 22,
                            color: Colors.white,
                            fontWeight: FontWeight.bold,
                        ),
                    ),
                ),
            ),
        ],
    ),
),

const SizedBox(height: 30),

Image.asset(
    'assets/img/uang.png',
    width: size.width * 0.6,
),

const SizedBox(height: 30),

buildButton("Scan Uang", () {
    navigate("Membuka kamera", const HasilPage());
}),
buildButton("Panduan", () {
    navigate("Membuka panduan", const
PanduanPage());
}),
buildButton("Pengaturan", () {
    navigate("Membuka pengaturan", const
PengaturanPage());
}),

const Spacer(),

```

```

        Text(
          isSpeaking
            ? "🔊 $statusText"
            : isListening
              ? "🔊 Mendengarkan..."
              : statusText,
          style: const TextStyle(fontSize: 16),
        ),
        const SizedBox(height: 20),
      ],
    ),
  ),
);
}
}

```

e. hasil.dart

```

import 'package:flutter/material.dart';
import 'package:camera/camera.dart';
import 'package:tflite_v2/tflite_v2.dart';
import 'package:flutter_tts/flutter_tts.dart';

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

  @override
  State<HasilPage> createState() => _HasilPageState();
}

class _HasilPageState extends State<HasilPage> {
  CameraController? cameraController;
  FlutterTts tts = FlutterTts();

  bool detecting = false;

  String nominal = "Scanning...";
  String akurasi = "-";
  String jarak = "-";
  String cahaya = "-";

  String lastSpoken = "";

  Color beezColor = Colors.green;

  @override
  void initState() {
    super.initState();
    initAll();
  }
}

```

```

    }

    Color getMoneyColor(String nominal) {
        switch (nominal) {
            case "1000":
                return const Color(0xFF4CAF50);

            case "2000":
                return const Color(0xFF9E9E9E);

            case "5000":
                return const Color(0xFF795548);

            case "10000":
                return const Color(0xFF9C27B0);

            case "20000":
                return const Color(0xFF2E7D32);

            case "50000":
                return const Color(0xFF1976D2);

            case "100000":
                return const Color(0xFFD32F2F);

            case "bukan_uang":
                return Colors.grey;

            default:
                return Colors.teal;
        }
    }

    String getNominal2022(String label) {
        if (label.contains("_2022")) {
            return label.replaceAll("_2022", "");
        }
        return "";
    }

    Future initAll() async {
        final cameras = await availableCameras();

        cameraController = CameraController(
            cameras[0],
            ResolutionPreset.medium,
            enableAudio: false,
        );

        await cameraController!.initialize();

        await cameraController!.setFocusMode(

```

```

        FocusMode.auto,
    );

    await Tflite.loadModel(
      model: "assets/model/model_uang.tflite",
      labels: "assets/model/labels.txt",
    );

    await tts.setLanguage("id-ID");
    await tts.setSpeechRate(0.5);

    startStream();

    setState(() {});
  }

  void startStream() {
    cameraController!.startImageStream((image) {
      if (!detecting) {
        detecting = true;
        runModel(image);
      }
    });
  }

  Future runModel(CameraImage image) async {
    try {
      var result = await Tflite.runModelOnFrame(
        bytesList: image.planes.map((e) => e.bytes).toList(),
        imageHeight: image.height,
        imageWidth: image.width,
        imageMean: 127.5,
        imageStd: 127.5,
        rotation: 90,
        numResults: 3,
        threshold: 0.95,
      );

      double brightness = image.planes[0].bytes.reduce((a, b)
=> a + b) /
        image.planes[0].bytes.length;

      double lux = brightness * 4;

      if (result != null && result.isNotEmpty) {
        result.sort(
          (a, b) =>
            (b['confidence'] as
double).compareTo(a['confidence'] as double),
        );

        var res = result[0];

```

```

print(result);

String label = res["label"].toString().trim();

double conf = (res["confidence"] as double) * 100;

print("LABEL   : $label");
print("CONF    : $conf");

if (conf < 95) {
  setState(() {
    nominal = "Tidak terdeteksi";
    akurasi = "${conf.toStringAsFixed(1)} %";
    jarak = "-";
    cahaya = "${lux.toStringAsFixed(0)}";
    beezColor = Colors.green;
  });

  detecting = false;
  return;
}

double jarakCm = 15;

if (conf >= 95) {
  jarakCm = 15;
} else if (conf >= 90) {
  jarakCm = 20;
} else if (conf >= 85) {
  jarakCm = 25;
} else if (conf >= 80) {
  jarakCm = 30;
} else {
  jarakCm = 40;
}

if (label == "bukan_uang") {
  setState(() {
    nominal = "Tidak terdeteksi";
    akurasi = "${conf.toStringAsFixed(1)} %";
    jarak = "${jarakCm.toStringAsFixed(0)} cm";
    cahaya = "${lux.toStringAsFixed(0)} lux";
    beezColor = Colors.green;
  });

  if (lastSpoken != "bukan_uang") {
    lastSpoken = "bukan_uang";

    await tts.stop();

    await tts.speak("Bukan uang");
  }
}

```

```

    } else {
        setState(() {
            nominal = "Rp $label";
            akurasi = "${conf.toStringAsFixed(1)} %";
            jarak = "${jarakCm.toStringAsFixed(0)} cm";
            cahaya = "${lux.toStringAsFixed(0)} lux";
            beezColor = getMoneyColor(label);
        });

        if (lastSpoken != "label") {
            lastSpoken = "label";

            await tts.stop();

            String spokenText = label.replaceAll("_2022", "");

            await tts.speak("Uang ${spokenText} rupiah");

        }
    }
} catch (e) {
    print("ERROR : $e");
}

await Future.delayed(
    const Duration(milliseconds: 30),
);

detecting = false;
}

@override
void dispose() {
    cameraController?.dispose();
    Tflite.close();
    tts.stop();
    super.dispose();
}

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

    return Scaffold(
        body: Stack(

```

```

children: [
  CameraPreview(cameraController!),
  Positioned(
    top: 150,
    left: 0,
    right: 0,
    child: Center(
      child: AnimatedContainer(
        duration: const Duration(milliseconds: 400),
        width: 300,
        height: 180,
        decoration: BoxDecoration(
          border: Border.all(
            color: beezColor,
            width: 5,
          ),
          borderRadius: BorderRadius.circular(15),
        ),
      ),
    ),
  ),
  Positioned(
    top: 60,
    left: 20,
    right: 20,
    child: AnimatedContainer(
      duration: const Duration(milliseconds: 400),
      padding: const EdgeInsets.all(12),
      decoration: BoxDecoration(
        color: beezColor.withOpacity(0.85),
        borderRadius: BorderRadius.circular(10),
      ),
      child: const Text(
        "Posisikan uang di dalam kotak",
        textAlign: TextAlign.center,
        style: TextStyle(
          color: Colors.white,
          fontSize: 16,
          fontWeight: FontWeight.bold,
        ),
      ),
    ),
  ),
  Align(
    alignment: Alignment.bottomCenter,
    child: AnimatedContainer(
      duration: const Duration(milliseconds: 400),
      height: 320,
      width: double.infinity,
      decoration: BoxDecoration(
        color: beezColor.withOpacity(0.15),
        borderRadius: const BorderRadius.only(
          topLeft: Radius.circular(35),

```

```

        topRight: Radius.circular(35),
      ),
    ),
  ),
),
Positioned(
  bottom: 50,
  left: 20,
  right: 20,
  child: AnimatedContainer(
    duration: const Duration(milliseconds: 400),
    padding: const EdgeInsets.all(20),
    decoration: BoxDecoration(
      color: beezColor.withOpacity(0.9),
      borderRadius: BorderRadius.circular(20),
    ),
    child: Column(
      mainAxisAlignment: MainAxisAlignment.min,
      children: [
        Text(
          nominal,
          style: const TextStyle(
            color: Colors.white,
            fontSize: 30,
            fontWeight: FontWeight.bold,
          ),
        ),
        const SizedBox(height: 15),
        Text(
          "Akurasi : $akurasi",
          style: const TextStyle(
            color: Colors.white,
            fontSize: 16,
          ),
        ),
        Text(
          "Jarak : $jarak",
          style: const TextStyle(
            color: Colors.white,
            fontSize: 16,
          ),
        ),
        Text(
          "Cahaya : $cahaya",
          style: const TextStyle(
            color: Colors.white,
            fontSize: 16,
          ),
        ),
      ],
    ),
  ),
),
),

```

```

        ),
      ],
    ),
  );
}
}

```

f. panduan.dart

```

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

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

  @override
  State<PanduanPage> createState() => _PanduanPageState();
}

class _PanduanPageState extends State<PanduanPage> {
  final FlutterTts flutterTts = FlutterTts();

  final String teksPanduan =
    "Selamat datang di Aplikasi Klasifikasi Uang Rupiah. "
    "Aplikasi ini membantu Anda mengenali nominal uang kertas "
    "Rupiah menggunakan kamera dan suara. "
    "Cara Menggunakan. "
    "Pertama, tekan tombol Scan Uang pada halaman utama untuk "
    "memulai. "
    "Kedua, arahkan uang ke kamera hingga terdengar suara. "
    "Ketiga, dengarkan hasil. Aplikasi akan menyebutkan "
    "nominal uang secara otomatis. "
    "Keempat, tekan tombol Scan Lagi untuk memindai uang "
    "berikutnya.";

  @override
  void initState() {
    super.initState();
    speakPanduan();
  }

  Future<void> speakPanduan() async {
    await flutterTts.setLanguage("id-ID");
    await flutterTts.setSpeechRate(0.5);
    await flutterTts.setVolume(1.0);
    await flutterTts.speak(teksPanduan);
  }

  @override
  void dispose() {
    flutterTts.stop();
    super.dispose();
  }
}

```

```

    }

    @override
    Widget build(BuildContext context) {
      return Scaffold(
        body: Container(
          width: double.infinity,
          decoration: const BoxDecoration(
            gradient: LinearGradient(
              begin: Alignment.topCenter,
              end: Alignment.bottomCenter,
              colors: [
                Colors.white,
                Color(0xFF75C565),
              ],
            ),
          ),
        child: SafeArea(
          child: Column(
            children: [
              Align(
                alignment: Alignment.centerLeft,
                child: IconButton(
                  icon: const Icon(Icons.arrow_back),
                  onPressed: () {
                    flutterTts.stop();
                    Navigator.pop(context);
                  },
                ),
              ),
              const SizedBox(height: 10),
              Expanded(
                child: Container(
                  margin: const
EdgeInsets.symmetric(horizontal: 20),
                  padding: const EdgeInsets.all(20),
                  decoration: BoxDecoration(
                    color: Colors.white,
                    borderRadius: BorderRadius.circular(30),
                    border: Border.all(
                      color: const Color(0xFF75C565),
                      width: 3,
                    ),
                  ),
                ),
                child: SingleChildScrollView(
                  child: Column(
                    children: [
                      const Text(
                        "Panduan",
                        style: TextStyle(
                          fontSize: 26,
                          fontWeight: FontWeight.bold,
                          color: Color(0xFF75C565),

```

```

        ),
      ),
      const SizedBox(height: 20),
      Text(
        teksPanduan,
        style: const TextStyle(
          fontSize: 18,
          height: 1.6,
          color: Color(0xFF75C565),
        ),
      ),
    ),
    const SizedBox(height: 30),

    // Tombol ulangi suara
    ElevatedButton(
      onPressed: speakPanduan,
      child: const Text("Putar Ulang
Suara"),
    ),
  ],
),
),
),
),
const SizedBox(height: 20),
],
),
),
),
);
}
}

```

g. pengaturan.dart

```

import 'package:flutter/material.dart';
import 'package:flutter_tts/flutter_tts.dart';
import 'informasiaplikasi.dart';
import 'tentangkami.dart';

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

  @override
  State<PengaturanPage> createState() =>
    _PengaturanPageState();
}

class _PengaturanPageState extends State<PengaturanPage> {
  final FlutterTts flutterTts = FlutterTts();

  @override

```

```

void initState() {
  super.initState();
  speak("Halaman pengaturan. Silakan pilih menu.");
}

Future<void> speak(String text) async {
  await flutterTts.setLanguage("id-ID");
  await flutterTts.setSpeechRate(0.5);
  await flutterTts.speak(text);
}

@override
void dispose() {
  flutterTts.stop();
  super.dispose();
}

@override
Widget build(BuildContext context) {
  final size = MediaQuery.of(context).size;

  return Scaffold(
    body: Container(
      width: size.width,
      height: size.height,
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topCenter,
          end: Alignment.bottomCenter,
          colors: [Colors.white, Color(0xFF75C565)],
        ),
      ),
      child: SafeArea(
        child: Padding(
          padding: const EdgeInsets.symmetric(horizontal:
20),
          child: Column(
            children: [
              const SizedBox(height: 20),

              Container(
                width: double.infinity,
                padding: const EdgeInsets.symmetric(vertical:
20),
                decoration: BoxDecoration(
                  color: const Color(0xFF75C565),
                  borderRadius: BorderRadius.circular(30),
                ),
                child: const Center(
                  child: Text(
                    "Pengaturan",
                    style: TextStyle(
                      fontSize: 26,

```



```

        decoration: BoxDecoration(
          color: Colors.white,
          borderRadius: BorderRadius.circular(30),
          border: Border.all(color: const Color(0xFF75C565),
width: 3),
          boxShadow: [
            BoxShadow(
              color: Colors.black.withOpacity(0.2),
              blurRadius: 5,
              offset: const Offset(0, 4),
            ),
          ],
        ),
        child: Center(
          child: Text(
            text,
            style: const TextStyle(
              fontSize: 22,
              fontWeight: FontWeight.bold,
              color: Color(0xFF75C565),
            ),
          ),
        ),
      ),
    ),
  );
}
}

```

h. informasiaplikasi.dart

```

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

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

  @override
  State<InformasiAplikasiPage> createState() =>
    _InformasiAplikasiPageState();
}

class _InformasiAplikasiPageState
  extends State<InformasiAplikasiPage> {

  final FlutterTts flutterTts = FlutterTts();

  @override
  void initState() {
    super.initState();
    speak();
  }
}

```

```

Future<void> speak() async {
  await flutterTts.setLanguage("id-ID");
  await flutterTts.setSpeechRate(0.5);
  await flutterTts.speak(
    "Halaman informasi aplikasi. "
    "Aplikasi ini menggunakan metode Convolutional Neural
Network "
    "untuk mendeteksi nominal uang kertas Rupiah dan
memberikan output suara."
  );
}

@override
void dispose() {
  flutterTts.stop();
  super.dispose();
}

@override
Widget build(BuildContext context) {
  final size = MediaQuery.of(context).size;

  return Scaffold(
    body: Container(
      width: size.width,
      height: size.height,
      decoration: const BoxDecoration(
        gradient: LinearGradient(
          begin: Alignment.topCenter,
          end: Alignment.bottomCenter,
          colors: [
            Colors.white,
            Color(0xFF75C565),
          ],
        ),
      ),
    child: SafeArea(
      child: Padding(
        padding: const EdgeInsets.symmetric(horizontal:
20),

        child: Column(
          children: [
            const SizedBox(height: 20),

            // HEADER
            Container(
              width: double.infinity,
              padding: const EdgeInsets.symmetric(vertical:
20),

              decoration: BoxDecoration(
                color: const Color(0xFF75C565),
                borderRadius: BorderRadius.circular(30),

```

```

    ),
    child: const Center(
      child: Text(
        "Informasi Aplikasi",
        style: TextStyle(
          fontSize: 26,
          fontWeight: FontWeight.bold,
          color: Colors.white,
        ),
      ),
    ),
  ),
),

const SizedBox(height: 30),

// HALAMAN PUTIH (CARD)
Expanded(
  child: Container(
    width: double.infinity,
    padding: const EdgeInsets.all(25),
    decoration: BoxDecoration(
      color: Colors.white,
      borderRadius: BorderRadius.circular(25),
      boxShadow: [
        BoxShadow(
          color: Colors.black26,
          blurRadius: 10,
          offset: Offset(0, 5),
        ),
      ],
    ),
  ),
  child: const SingleChildScrollView(
    child: Text(
      "Aplikasi ini menggunakan metode  

      Convolutional Neural Network (CNN) "  

      "untuk mendeteksi nominal uang kertas  

      Rupiah melalui kamera smartphone. "  

      "Sistem akan memproses citra  

      menggunakan model pembelajaran mesin "  

      "dan menghasilkan output berupa suara  

      untuk membantu pengguna tunanetra "  

      "mengetahui nilai uang secara  

      mandiri.\n\n"  

      "Aplikasi ini dirancang dengan  

      antarmuka sederhana dan ramah aksesibilitas.",
      style: TextStyle(
        fontSize: 18,
        height: 1.6,
      ),
      textAlign: TextAlign.justify,
    ),
  ),
),
),

```

```

        ),
      ],
    ),
  ),
),
);
}
}

```

i. tentangkami.dart

```

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

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

  @override
  State<TentangKamiPage> createState() =>
    _TentangKamiPageState();
}

class _TentangKamiPageState extends State<TentangKamiPage> {

  final FlutterTts flutterTts = FlutterTts();

  @override
  void initState() {
    super.initState();
    speak();
  }

  Future<void> speak() async {
    await flutterTts.setLanguage("id-ID");
    await flutterTts.setSpeechRate(0.5);
    await flutterTts.speak(
      "Halaman tentang kami. "
      "Aplikasi ini dikembangkan sebagai proyek skripsi "
      "untuk membantu penyandang tunanetra mengenali nominal
uang secara mandiri."
    );
  }

  @override
  void dispose() {
    flutterTts.stop();
    super.dispose();
  }

  @override
  Widget build(BuildContext context) {

```

```

final size = MediaQuery.of(context).size;

return Scaffold(
  body: Container(
    width: size.width,
    height: size.height,
    decoration: const BoxDecoration(
      gradient: LinearGradient(
        begin: Alignment.topCenter,
        end: Alignment.bottomCenter,
        colors: [
          Colors.white,
          Color(0xFF75C565),
        ],
      ),
    ),
  child: SafeArea(
    child: Padding(
      padding: const EdgeInsets.symmetric(horizontal:
20),

      child: Column(
        children: [
          const SizedBox(height: 20),

          // Header
          Container(
            width: double.infinity,
            padding: const EdgeInsets.symmetric(vertical:
20),

            decoration: BoxDecoration(
              color: const Color(0xFF75C565),
              borderRadius: BorderRadius.circular(30),
            ),
            child: const Center(
              child: Text(
                "Tentang Kami",
                style: TextStyle(
                  fontSize: 26,
                  fontWeight: FontWeight.bold,
                  color: Colors.white,
                ),
              ),
            ),
          ),

          const SizedBox(height: 30),

          // Halaman Putih seperti Panduan
          Expanded(
            child: Container(
              width: double.infinity,
              padding: const EdgeInsets.all(25),
              decoration: BoxDecoration(

```

