PROJECT-BASED LEARNING REPORT

TEKNOLOGI REKAYASA MULTIMEDIA
POLITEKNIK NEGERI BATAM
2024

GROUP MEMBER

TRM 4A (REGULER NIGHT) :

1. 4312211004 - IRWAN SIHOMBING
2. 4312211003 - MEYSTAVIA BERTA WAHYU RENGGANIS
3. 4312211008 - MAULANA RAVI AKHBAR
4. 4312211026 - MUHAMMAD NAUFAL RASIKH YANDA
5. 4312211030 - MOHAMED FARHAN AHMED SHAHEEN

PROJECT IDENTITY

Project Title	: Mobile Application “Baby Guard”
Project Owner	:
Project Manager	: Agung Riyadi. S.Si., M.Kom
Project Co-Manager	:
Client	: Agung Riyadi. S.Si., M.Kom
Outputs	:

√	Final Report
√	Product: *Mobile Application*/Hardware/video*
√	Demo video /trailer*
√	Scientific Poster
√	Intellectual Property Rights Document
√	Handover Document
	Contest Proposal (optional)

Approved by,
Batam, June 26th 2024

Agung Riyadi. S.Si., M.Kom

NIK. 119221

LIST OF CONTENT

PROJECT IDENTITY ..... *ii*
LIST OF CONTENT ..... *iii*
LIST OF PICTURES ..... *iv*
LIST OF TABLES ..... *vi*
1. PROJECT-BASED LEARNING PRODUCT ..... *vii*
1.1 Product Description vii
1.2 Product Design vii
3. PRODUCT IMPLEMENTATION ..... *xiii*
3.1 Product Implementation xiii
4. CONCLUSION *1*
4.1 Obstacle 1
4.2 Learning Process 1
APPENDIX I – LOGBOOK *2*
APPENDIX II – TEAM SCHEDULE *5*
APPENDIX III – PROJECT BOARD *6*
APPENDIX IV – PRESENTATION SLIDES *7*
APPENDIX V - LIST OF FILE LINKS *22*

LIST OF PICTURES

FIGURE 1 DIAGRAM USE CASE ..... *ix*
FIGURE 2 LOADING PAGE ..... *x*
FIGURE 3 MAIN VIEW ..... *xi*
FIGURE 4 SYSTEM DESIGN ..... *xiv*
FIGURE 5 LOGBOOK *2*
FIGURE 6 SCHEDULE *5*
FIGURE 7 BOARD *6*
FIGURE 8 PRESENTATION SLIDE 1 ..... *7*
FIGURE 9 PRESENTATION SLIDE 2 ..... *7*
FIGURE 10 PRESENTATION SLIDE 3 ..... *8*
FIGURE 11 PRESENTATION SLIDE 4 ..... *8*
FIGURE 12 PRESENTATION SLIDE 5 ..... *9*
FIGURE 13 PRESENTATION SLIDE 6 ..... *9*
FIGURE 14 PRESENTATION SLIDE 7 ..... *10*
FIGURE 15 PRESENTATION SLIDE 8 ..... *10*
FIGURE 16 PRESENTATION SLIDE 9 ..... *11*
FIGURE 17 PRESENTATION SLIDE 10 ..... *11*
FIGURE 18 PRESENTATION SLIDE 11 ..... *12*
FIGURE 19 PRESENTATION SLIDE 12 ..... *12*
FIGURE 20 PRESENTATION SLIDE 13 ..... *13*
FIGURE 21 PRESENTATION SLIDE 14 ..... *13*
FIGURE 22 PRESENTATION SLIDE 15 ..... *14*
FIGURE 23 PRESENTATION SLIDE 16 ..... *14*
FIGURE 24 PRESENTATION SLIDE 17 ..... *15*
FIGURE 25 PRESENTATION SLIDE 18 ..... *15*
FIGURE 26 PRESENTATION SLIDE 19 ..... *16*
FIGURE 27 PRESENTATION SLIDE 20 ..... *16*
FIGURE 28 PRESENTATION SLIDE 21 ..... *17*
FIGURE 29 PRESENTATION SLIDE 22 ..... *17*
FIGURE 30 PRESENTATION SLIDE 23 ..... *18*
FIGURE 31 PRESENTATION SLIDE 24 ..... *18*

FIGURE 32 PRESENTATION SLIDE 25 ..... *19*
FIGURE 33 PRESENTATION SLIDE 26 ..... *19*
FIGURE 34 PRESENTATION SLIDE 27 ..... *20*
FIGURE 35 PRESENTATION SLIDE 28 ..... *20*
FIGURE 36 PRESENTATION SLIDE 29 ..... *21*

LIST OF TABLES

TABLE 1 FUNCTIONAL SYSTEM REQUIREMENTS.....*viii*
TABLE 2 TESTING SOUND SENSOR ..... *xxxi*
TABLE 3 TESTING MOTION SENSOR ..... *xxxi*
TABLE 4 TESTING TEMPERATURE SENSOR..... *xxxii*
TABLE 5 LOGBOOK.....*2*

1. PROJECT-BASED LEARNING PRODUCT

1.1 Product Description

Baby Guard is the result of a mobile application design project aimed at providing an innovative solution for baby monitoring. Designed with advanced IoT technology, Baby Guard integrates sound, motion, and temperature sensors with the ESP8266 module, and a mobile application that can be easily accessed using Android Studio.

With a focus on baby safety and comfort, Baby Guard allows parents to connect in real-time with their baby's environment. The sensitive sound sensor detects crying or unusual sounds, while the motion sensor provides information about the baby's activities. The temperature and humidity sensors ensure optimal comfort for the baby's sleep.

The Baby Guard mobile application, developed with Flutter and Dart, provides easy access to baby monitoring information. Parents can receive real-time notifications about significant changes detected by the sensors, ensuring a quick response to their baby's needs.

Through Wi-Fi connectivity, sensor data is sent to a secure cloud server, allowing easy access via the mobile application. Thus, Baby Guard provides parents with peace of mind, ensuring that their baby is always under proper and safe supervision.

1.2 Product Design

Baby Guard is a mobile application designed to provide advanced and secure baby monitoring for parents. Using sensor technology and internet connectivity, this app allows parents to monitor sound, movement, temperature, and humidity around their baby in real-time. Its aim is to give parents peace of mind and ensure the comfort and safety of their baby while they sleep.

a. Components and Tools

1. Hardwares:

- NodeMCU ESP8266: A microcontroller that acts as the central controller, collecting data from sensors and sending it to the mobile application.
- Sound Sensor (KY-037): Detects baby sounds or crying.
- Motion Sensor (PIR Sensor): Detects the baby's movements.
- Temperature Sensor (DHT11 or DHT22): Measures the temperature of the environment where the baby sleeps.
- Breadboard and Jumper Cables: For connecting the sensors to the ESP8266.

- USB Cable: For connecting the NodeMCU to a laptop for programming and power.

2. Softwares:
   - Arduino IDE: Used for writing and uploading code to the NodeMCU ESP8266.
   - Android Studio: Used for developing the mobile application.
   - Flutter and Dart: Flutter is the UI toolkit used to build natively compiled applications for mobile from a single codebase, and Dart is the programming language used to write the application code. This combination ensures a smooth and efficient development process for creating a responsive and attractive mobile application.

b. Functional System Requirements

Table 1 Functional System Requirements

Source: personal document

No.	Requirement Description	Achieved (√) / Not Achieved (x)
1.	The application must be able to receive data from sound, motion, temperature, and humidity sensors.	√
2.	The application must provide real-time notifications to parents about significant changes in the baby's environment, such as crying or unusual movements.	√
3.	The application must have an intuitive and user-friendly interface, allowing parents to easily monitor their baby's status.	√
4.	The application must connect to a cloud server for secure data storage, accessible from anywhere.	√

c. Use Case

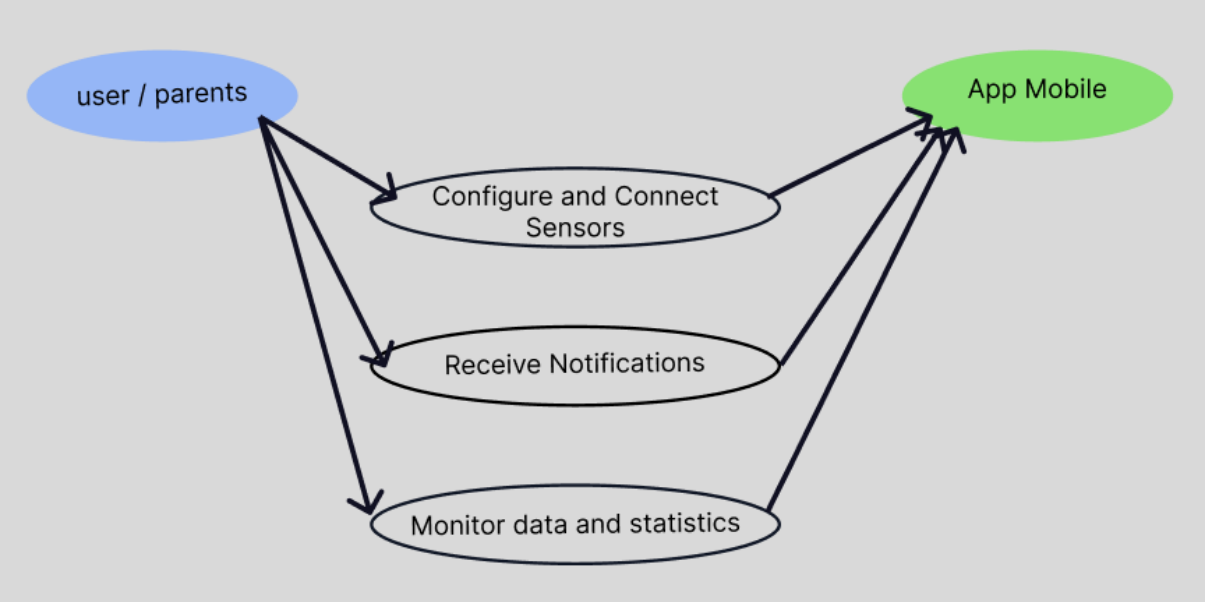

Figure 1 Diagram Use Case
Source: personal document

Here's a use case diagram for the "Baby Guard" mobile application, illustrating the interaction between the "Orang Tua" (Parents) and the mobile app:

1. User/Parents:
    - Arrange and connect sound, motion, temperature, and humidity sensors with the application.
    - Receive direct notifications to their devices when sensors detect significant changes in the baby's environment.
    - Monitor data and statistics about the baby's sound, movement, temperature, and humidity through the app interface.
2. Mobile App:
    - Processes and displays the data received from sensors.
    - Sends notifications to the parents' devices.

d. Product interface/architecture design.

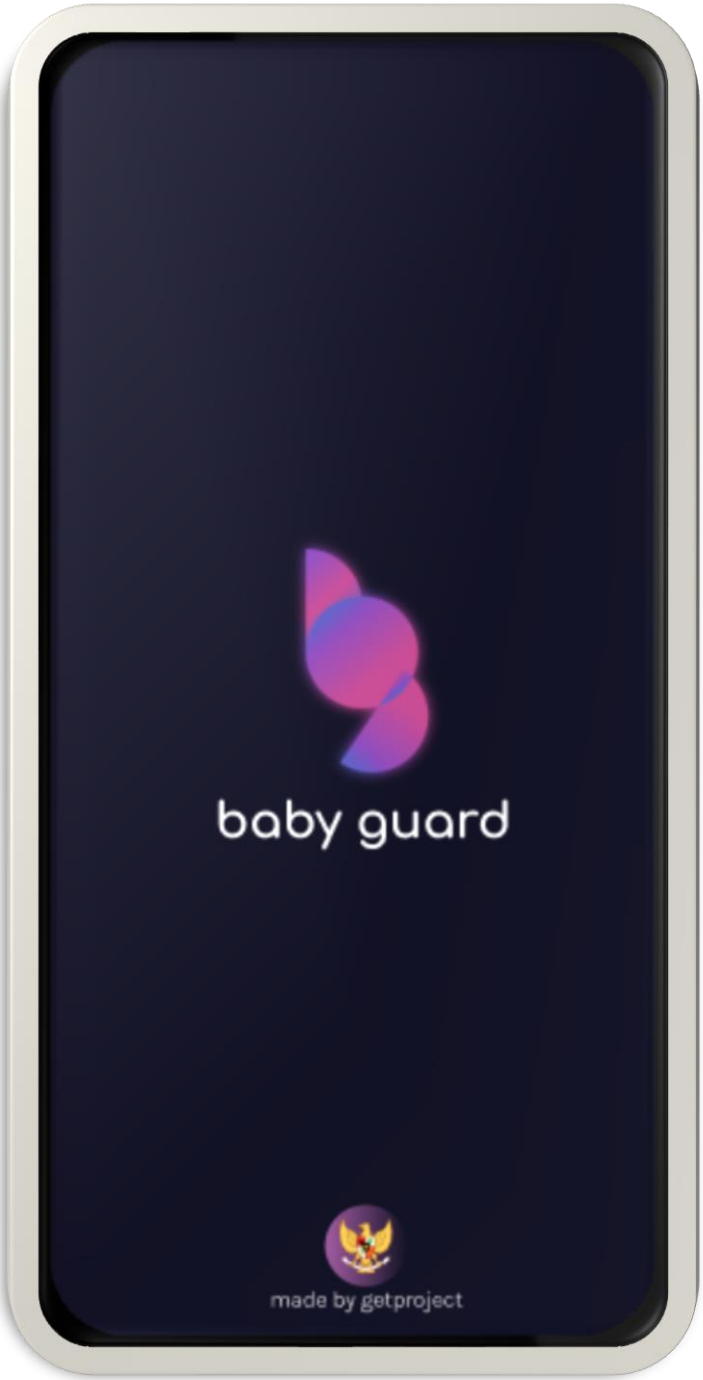

Figure 2 Loading page

Source: personal document

Figure 3 main view

Source: personal document

e. Programming language.

The programming languages used in the Baby Guard project are Flutter and Dart. Flutter is the UI toolkit used for building natively compiled applications for mobile from a single codebase, while Dart is the programming language used to write the application code. Flutter was chosen for its ability to create visually attractive and highly responsive interfaces with a single codebase that runs on both Android and iOS, reducing development time and effort. Dart

complements Flutter by providing a robust and efficient programming language with a smooth learning curve. By leveraging Flutter and Dart, the Baby Guard project can produce a responsive, stable, and easily developed mobile application.

3. PRODUCT IMPLEMENTATION

3.1 Product Implementation

a. Problem Analysis and Requirements

1. Identification of Key Needs for Baby Monitoring:
    - Sound Detection: Identifying cries or other sounds produced by the baby to determine if the baby is awake or needs attention.
    - Motion Detection: Monitoring the baby's movements to ensure the baby remains in a safe and comfortable sleeping position, as well as detecting unusual activity.
    - Temperature Monitoring: Checking the environmental temperature where the baby sleeps to ensure conditions remain comfortable and safe for the baby, avoiding temperatures that are too hot or too cold.
2. Gathering Information about the Most Suitable Sensors for This Purpose:
    - Sound Sensor (KY-037): The sound sensor (KY-037) functions to detect sound intensity around the baby. This sensor is easy to use with microcontrollers like NodeMCU ESP8266, has adjustable sensitivity, and is affordable. However, calibration is required to avoid detection of irrelevant sounds.
    - Motion Sensor (PIR - Passive Infrared Sensor): The motion sensor (PIR) functions to detect infrared motion generated by the baby's body. This sensor has advantages such as low power consumption, fast response to motion changes, and wide area coverage. However, sensitivity to motion may require adjustment for the baby's sleeping environment.
    - Temperature Sensor (DHT11): The DHT11 temperature and humidity sensor function to measure temperature and humidity around the baby and are easy to integrate with NodeMCU ESP8266. Sensor placement should ensure it is not directly exposed to hot or cold air to maintain accurate measurements.

b. System Design

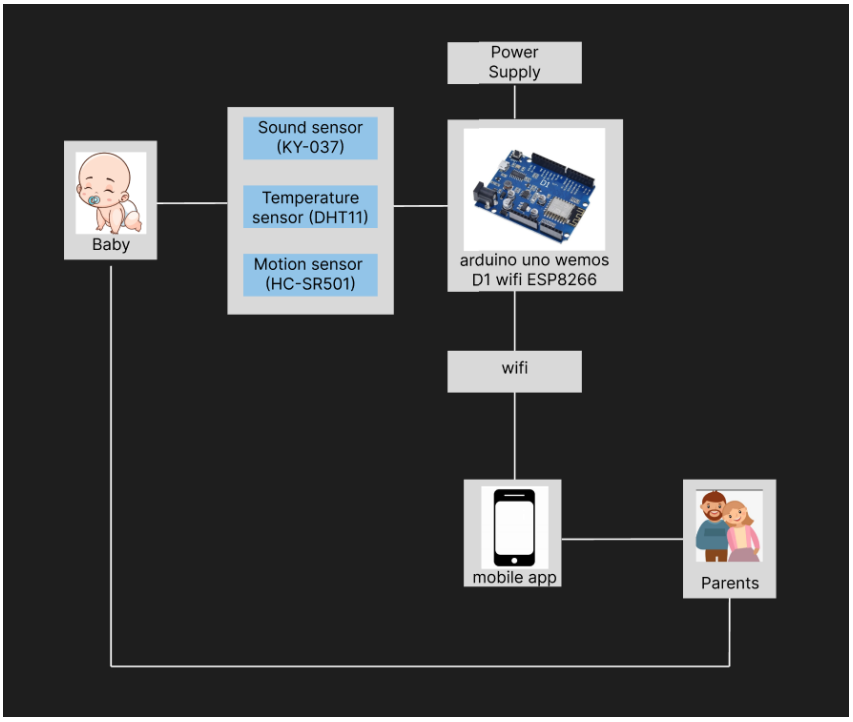

Figure 4 System design

Source: personal document

The Sound Sensor (KY-037), Motion Sensor (PIR), and Temperature Sensor (DHT11/DHT22) are connected to a NodeMCU ESP8266, which collects data from these sensors. Wi-Fi allows the NodeMCU to send data to a cloud server, which receives and stores sensor data, then provides an interface for the mobile application. The 'Baby Guard' Mobile Application in Android Studio, using Java and Kotlin programming languages, accesses data from the cloud server and displays baby monitoring information to the user.

c. Programming code

1. Code 3 combined sensors DHT 11, PIR, and KY-037

```
#include <ESP8266WiFi.h>
#include <WiFiManager.h>
#include <DHT.h>
#include <WebSocketsServer.h>

#define DHTPIN D4
#define DHTTYPE DHT11
#define PIR_PIN D6
```

```
#define LED_PIN D13
#define SOUND_SENSOR_PIN D2
#define SOUND_LED_PIN D11

DHT dht(DHTPIN, DHTTYPE);
WebSocketsServer webSocket = WebSocketsServer(81);

bool motionDetected = false;
bool pirLedOn = false;
bool soundLedOn = false;
unsigned long led_on_time = 0;
unsigned long led_off_time = 0;
const unsigned long on_duration = 10000; // Duration LED on (10 seconds)
const unsigned long off_duration = 5000; // Duration LED off (5 seconds)

void handleWebSocketMessage(uint8_t num, WStype_t type, uint8_t * payload, size_t
length) {
  // Handle messages from the WebSocket
}

void setup() {
  Serial.begin(9600);
  pinMode(PIR_PIN, INPUT);
  pinMode(LED_PIN, OUTPUT);
  pinMode(SOUND_SENSOR_PIN, INPUT);
  pinMode(SOUND_LED_PIN, OUTPUT);

  dht.begin();

  WiFiManager wifiManager;
  wifiManager.resetSettings();
  wifiManager.autoConnect("AutoConnectAP");
```

```
  webSocket.begin();
  webSocket.onEvent(handleWebSocketMessage);
  Serial.println("WebSocket server started");
}

void loop() {
  webSocket.loop();

  // Read PIR sensor state
  int pirState = digitalRead(PIR_PIN);

  if (pirState == HIGH) {
    Serial.println("Motion detected!");
    motionDetected = true;
    pirLedOn = true;
  } else {
    Serial.println("No motion detected");
    motionDetected = false;
    pirLedOn = false;
  }

  digitalWrite(LED_PIN, pirLedOn ? HIGH : LOW);

  // Read sound sensor state
  int soundData = digitalRead(SOUND_SENSOR_PIN);

  if (soundData == 1 && !soundLedOn) {
    digitalWrite(SOUND_LED_PIN, HIGH);
    soundLedOn = true;
    led_on_time = millis();
  }

  if (soundLedOn && millis() - led_on_time < on_duration) {
```

```
    Serial.println("Sound detected!"); // Output sound detected when sound is detected
  }

  if (soundLedOn && millis() - led_on_time >= on_duration) {
    digitalWrite(SOUND_LED_PIN, LOW);
    soundLedOn = false;
    led_off_time = millis();
  }

  if (!soundLedOn && millis() - led_off_time >= off_duration) {
    Serial.println("No sound detected"); // Output no sound detected when no sound is
detected
  }

  // Reading temperature and humidity
  float humidity = dht.readHumidity();
  float temperature = dht.readTemperature();

  // Check if any reads failed and exit early
  if (!isnan(humidity) && !isnan(temperature)) {
    Serial.print(F("Humidity: "));
    Serial.print(humidity);
    Serial.print(F("%\tTemperature: "));
    Serial.println(temperature);

    String soundStatus = (soundData == 1) ? "Sound Detected" : "No Sound";
    String data = "{\"temperature\": " + String(temperature) + ", \"humidity\": " +
String(humidity) + ", \"motion\": " + (motionDetected ? "true" : "false") + ", \"sound\": \""
+ soundStatus + "\"}";
    webSocket.broadcastTXT(data);
  } else {
    Serial.println(F("Failed to read from DHT sensor!"));
  }
```

```
  delay(1000);
}
```

2. The main.dart code (the application's interface)

```
import 'package:flutter/material.dart';
import 'package:web_socket_channel/io.dart';
import 'dart:convert';

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'IoT Sensor Data',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: LoadingScreen(),
    );
  }
}

class LoadingScreen extends StatefulWidget {
  @override
  _LoadingScreenState createState() => _LoadingScreenState();
}

class _LoadingScreenState extends State<LoadingScreen> {
  @override
```

```
void initState() {
  super.initState();
  Future.delayed(Duration(seconds: 5), () {
    Navigator.pushReplacement(
      context,
      MaterialPageRoute(builder: (context) => MyHomePage()),
    );
  });
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    body: Stack(
      children: [
        Container(
          width: double.infinity,
          height: double.infinity,
          decoration: BoxDecoration(
            image: DecorationImage(
              image: AssetImage('assets/loadscreen.png'),
              fit: BoxFit.cover,
            ),
          ),
        ),
        Align(
          alignment: Alignment.bottomCenter,
          child: Padding(
            padding: const EdgeInsets.only(bottom: 215.0),
            child: CircularProgressIndicator(
              valueColor: AlwaysStoppedAnimation<Color>(Colors.white),
            ),
          ),
```

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

class MyHomePage extends StatefulWidget {
  @override
  _MyHomePageState createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  late IOWebSocketChannel channel;
  String temperature = '';
  String humidity = '';
  String motionStatus = 'No Motion Detected';
  String soundStatus = 'No Sound Detected';
  bool soundDetected = false;
  late DateTime soundDetectedTime;

  @override
  void initState() {
    super.initState();
    // Replace IP and port with your ESP8266 WebSocket server details
    channel = IOWebSocketChannel.connect('ws://172.20.10.5:81');

    channel.stream.listen((data) {
      final sensorData = jsonDecode(data);
      setState(() {
        temperature = sensorData['temperature'].toString();
        humidity = sensorData['humidity'].toString();
```

```
        bool motionValue = sensorData['motion'];
        motionStatus = motionValue? 'Motion Detected' : 'No Motion Detected';

        if (sensorData['sound'] == 'Sound Detected') {
          soundDetected = true;
          soundDetectedTime = DateTime.now();
        }

        if (soundDetected &&
DateTime.now().difference(soundDetectedTime).inSeconds <= 10) {
          soundStatus = 'Sound Detected';
        } else {
          soundDetected = false;
          soundStatus = 'No Sound Detected';
        }
      });
    }, onError: (error) {
      print('Error: $error');
    });
  }

  @override
  void dispose() {
    channel.sink.close();
    super.dispose();
  }

  Widget buildSensorCard(String title, String value, IconData icon, Color color) {
    return Card(
      color: color.withOpacity(0.7),
      shape: CircleBorder(),
      child: Padding(
        padding: const EdgeInsets.all(20.0),
```

```
          child: Column(
            mainAxisSize: MainAxisSize.min,
            children: <Widget>[
              Stack(
                alignment: Alignment.center,
                children: [
                  CircularProgressIndicator(
                    value: value == 'No Motion Detected' || value == 'No Sound Detected'? 0.0 :
1.0,
                    color: color,
                    backgroundColor: Colors.grey[200],
                    strokeWidth: 10,
                  ),
                  Container(
                    width: 80,
                    height: 80,
                    decoration: BoxDecoration(
                      shape: BoxShape.circle,
                      color: Colors.white,
                      boxShadow: [
                        BoxShadow(
                          color: color.withOpacity(0.5),
                          blurRadius: 10,
                          spreadRadius: 2,
                        ),
                      ],
                    ),
                    child: Icon(
                      icon,
                      color: color,
                      size: 40,
                    ),
                  ),
```

```
            ],
          ),
          SizedBox(height: 10),
          Text(
            value,
            style: TextStyle(color: Colors.white, fontSize: 14),
          ),
          SizedBox(height: 5),
          Text(
            title,
            style: TextStyle(color: Colors.white, fontSize: 12),
          ),
        ],
      ),
    ),
  );
}

 Widget buildCircularIndicator(String title, String value, IconData icon, double
percentage, Color color) {
  return Card(
    color: color.withOpacity(0.7),
    shape: CircleBorder(),
    child: Padding(
      padding: const EdgeInsets.all(20.0),
      child: Column(
        mainAxisSize: MainAxisSize.min,
        children:<Widget>[
          Stack(
            alignment: Alignment.center,
            children: [
              CircularProgressIndicator(
                value: percentage / 100,
```

```
          color: color,
          backgroundColor: Colors.grey[200],
          strokeWidth: 10,
        ),
        Container(
          width: 80,
          height: 80,
          decoration: BoxDecoration(
            shape: BoxShape.circle,
            color: Colors.white,
            boxShadow: [
              BoxShadow(
                color: color.withOpacity(0.5),
                blurRadius: 10,
                spreadRadius: 2,
              ),
            ],
          ),
          child: Icon(
            icon,
            color: color,
            size: 40,
          ),
        ),
      ],
    ),
    SizedBox(height: 10),
    Text(
      value,
      style: TextStyle(color: Colors.white, fontSize: 18),
    ),
    SizedBox(height: 5),
    Text(
```

```
            title,
            style: TextStyle(color: Colors.white, fontSize: 14),
          ),
        ],
      ),
    ),
  );
}

@override
Widget build(BuildContext context) {
  double tempValue = double.tryParse(temperature) ?? 0.0;
  double humValue = double.tryParse(humidity) ?? 0.0;

  return Scaffold(
    extendBodyBehindAppBar: true,
    backgroundColor: Colors.transparent,
    appBar: AppBar(
      elevation: 0,
      backgroundColor: Colors.transparent,
    ),
    body: Container(
      width: double.infinity,
      height: double.infinity,
      decoration: BoxDecoration(
        image: DecorationImage(
          image: AssetImage('assets/mainbg.png'),
          fit: BoxFit.fill,
        ),
      ),
      child: Center(
        child: SingleChildScrollView(
          child: Column(
```

```
            mainAxisAlignment: MainAxisAlignment.center,
            children: <Widget>[
              SizedBox(height: 20),
              Row(
                mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                children: <Widget>[
                  Expanded(
                    child: buildCircularIndicator('Temperature', '$temperature°C',
Icons.thermostat, tempValue, Colors.orange),
                  ),
                  SizedBox(width: 20),
                  Expanded(
                    child: buildCircularIndicator('Humidity', '$humidity%', Icons.water_drop,
humValue, Colors.blue),
                  ),
                ],
              ),
              SizedBox(height: 20),
              Row(
                mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                children: <Widget>[
                  Expanded(
                    child: buildSensorCard('Sound', soundStatus, Icons.volume_up,
Colors.red),
                  ),
                  SizedBox(width: 20),
                  Expanded(
                    child: buildSensorCard('Motion', motionStatus, Icons.directions_run,
Colors.green),
                  ),
                ],
              ),
              SizedBox(height: 20),
```

```
                Padding(
                  padding: const EdgeInsets.symmetric(horizontal: 20.0),
                  child: Row(
                    mainAxisAlignment: MainAxisAlignment.spaceBetween,
                    children: [
                      Column(
                        children: [
                          Stack(
                            alignment: Alignment.center,
                            children: [
                              Image.asset(
                                tempValue > 25? 'assets/thermostat_hot.png' :
'assets/thermostat_cold.png',
                                width: 60,
                                height: 60,
                                fit: BoxFit.contain,
                              ),
                            ],
                          ),
                          SizedBox(height: 5),
                          Text(
                            tempValue > 25? 'Hot' : 'Normal',
                            style: TextStyle(color: Colors.white, fontSize: 14),
                          ),
                        ],
                      ),
                      Column(
                        children: [
                          Stack(
                            alignment: Alignment.center,
                            children: [
                              Image.asset(
                                humValue > 70? 'assets/water_high.png' : 'assets/water_low.png',
```

```
                    width: 60,
                    height: 60,
                    fit: BoxFit.contain,
                  ),
                ],
              ),
              SizedBox(height: 5),
              Text(
                'Kelembapan: $humidity%',
                style: TextStyle(color: Colors.white, fontSize: 14),
              ),
            ],
          ),
        ],
      ),
    ),
    SizedBox(height: 20),
    Padding(
      padding: const EdgeInsets.symmetric(horizontal: 20.0),
      child: Row(
        mainAxisAlignment: MainAxisAlignment.spaceBetween,
        children: [
          Column(
            children: [
              Stack(
                alignment: Alignment.center,
                children: [
                  Image.asset(
                    soundDetected? 'assets/audio_on.png' : 'assets/audio_off.png',
                    width: 60,
                    height: 60,
                    fit: BoxFit.contain,
                  ),
```

```
                ],
              ),
              SizedBox(height: 5),
              Text(
                soundDetected? 'Sound Detected' : 'No Sound Detected',
                style: TextStyle(color: Colors.white, fontSize: 14),
              ),
            ],
          ),
          Column(
            children: [
              Stack(
                alignment: Alignment.center,
                children: [
                  Image.asset(
                    'assets/motion.png',
                    width: 60,
                    height: 60,
                    fit: BoxFit.contain,
                  ),
                ],
              ),
              SizedBox(height: 5),
              Text(
                motionStatus,
                style: TextStyle(color: Colors.white, fontSize: 14),
              ),
            ],
          ),
        ],
      ),
    ),
  ],
```

```
            ),
          ),
        ),
      ),
    );
  }
}
```

3. The code Pubsspec.yaml

```
name: BabyGuarddd
description: A sample command-line application with basic argument parsing.
version: 0.0.1
# repository: https://github.com/my_org/my_repo

environment:
  sdk: '^3.4.2'

# Add regular dependencies here.
dependencies:
  flutter:
    sdk: flutter
  web_socket_channel: ^2.1.0
  cupertino_icons: ^1.0.2
  http: ^1.2.1
  args: ^2.3.0

dev_dependencies:
  flutter_test:
    sdk: flutter
  lints: ^3.0.0
  test: ^1.24.0

flutter_lints: ^2.0.0
```

```
flutter:
  uses-material-design: true
  assets:
    - assets/loadscreen.png
    - assets/mainbg.png
    - assets/thermostat_hot.png
    - assets/thermostat_cold.png
    - assets/water_high.png
    - assets/water_low.png
    - assets/audio_on.png
    - assets/audio_off.png
    - assets/motion.png
```

d. System Testing

1. Sound sensor

Table 2 Testing sound sensor

Source: personal document

Date test	Sound source	Distance with Sensor	View
25/05/2024	Trial 1	10 cm	Detected
25/05/2024	Trial 2	20 cm	Detected
25/05/2024	Trial 3	30 cm	Detected
25/05/2024	Trial 4	40 cm	Detected
25/05/2024	Trial 5	50 cm	Not detected
25/05/2024	Trial 6	60 cm	Not detected

2. Motion Sensor

Table 3 Testing Motion sensor

Source: personal document

Date test	Motion source	Motion detected	Distance with Sensor	View

26/05/2024	Trial 1	Yes	10 cm	Detected
26/05/2024	Trial 2	Yes	20 cm	Detected
26/05/2024	Trial 3	Yes	30 cm	Detected
26/05/2024	Trial 4	Yes	40 cm	Detected
26/05/2024	Trial 5	No	50 cm	Not Detected
26/05/2024	Trial 6	No	60 cm	Not Detected

3. Temperature Sensor

Table 4 Testing Temperature sensor

Source: personal document

Date test	Temperature source	Time	Humadity	Temperature
26/05/2024	Trial 1	08:00 AM	55 %	22 °C
26/05/2024	Trial 2	09:00 AM	54 %	23 °C
26/05/2024	Trial 3	10:00 AM	53 %	24 °C
26/05/2024	Trial 4	11:00 AM	52 %	24 °C
26/05/2024	Trial 5	12:00 PM	50 %	25 °C
26/05/2024	Trial 6	13:00 PM	51 %	25 °C

4. CONCLUSION

4.1 Obstacle

One obstacle encountered in the design of this project was ensuring seamless communication between the mobile application and the IoT devices. Integrating various sensors, such as sound, motion, and temperature sensors, with the NodeMCU ESP8266 microcontroller posed a challenge in terms of establishing reliable data transmission and synchronization. Additionally, configuring the server-side infrastructure to handle real-time data streaming and storage presented another obstacle, especially in ensuring scalability and data security. Overcoming these obstacles required thorough testing, debugging, and optimization of both hardware and software components to ensure the smooth operation of the Baby Guard system.

4.2 Learning Process

Working on the PBL project provided several new insights and learning experiences. One of the most significant aspects was gaining practical experience in developing an IoT-based mobile application from start to finish. This included understanding the hardware components involved, such as sensors and microcontrollers, as well as the software aspect, including Flutter framework and Dart language for Android development.

Furthermore, the project offered insights into the complexities of integrating various technologies to create a cohesive system. Understanding how to establish communication between different devices and ensuring data transmission and processing were essential aspects that I learned during the project.

Relating this to the courses taken in this semester, we found that concepts from courses such as Software Engineering, IoT Systems, and Mobile App Development were directly applicable to the project. Concepts learned in Software Engineering, such as requirement analysis, system design, and testing methodologies, were crucial in planning and executing the project effectively. Similarly, knowledge gained from the IoT Systems course helped in understanding the underlying principles of IoT architecture and sensor integration. Finally, the Mobile App Development course equipped me with the necessary skills to develop the user interface and functionality of the mobile application using Java and Kotlin.

Overall, working on the PBL project provided a holistic learning experience that reinforced and applied the theoretical knowledge gained from the courses, enabling me to tackle real-world challenges in developing IoT-based mobile applications.

APPENDIX I – LOGBOOK

ID	Stages	Work Details	Ouput	Begin	Finish	Progress	#
1	Planning	Brainstorming. The first meeting with the manpro to discuss the tools and materials needed during the work	Note	2024-03-08	2024-03-08	5%	Happy
12	Implementation	Finishing and retesting. Ensures all components work properly. Ensure all needs before AAS is 100% completed	zip, pdf	2024-06-13	2024-06-20	10%	Happy
8	Design	designing an intuitive user interface/UI and ensuring a good user experience/UX and ensuring ease of use and accessibility	note	2024-05-02	2024-05-15	15%	Happy
6	Implementation	Preparing the needs for progress presentations before UTS, namely reports in the form of PPT and then taking presentation videos according to existing rules.	PPT, mp4	2024-03-26	2024-03-27	10%	Happy
7	Implementation	Frontend and Backend development and integrating camera features for video monitoring and other sensors such as sound, temperature and humidity.	code	2024-04-01	2024-05-01	20%	Happy
9	Implementation	Testing. Testing each component of the application separately to ensure they function properly as well as testing the interactions between the various components to ensure they work well together	Note, Finished network	2024-05-16	2024-05-23	7%	Happy
10	Analisys	Data collection, starting from planning to test results for the purpose of the final report, and preparing power points for presentations ahead of UAS.	ppt, word	2024-05-24	2024-05-30	5%	Happy
2	Planning	The second meeting with manpro to discuss the RPP and discuss stores for the purchase of tools as well as compare between offline and online stores	Document/Word	2024-03-14	2024-03-14	5%	Happy
4	Planning	The fourth meeting with manpro to introduce tools and materials directly, then the group began to buy the necessary tools and materials directly (offline)	Note	2024-03-28	2024-03-29	5%	Happy
5	Planning	The fifth meeting with manpro to demonstrate the process of assembling tools and materials to become one product	Note	2024-03-07	2024-03-07	5%	Happy
3	Planning	The third meeting with the group to discuss UI/UX display design using Figma	Figma	2024-03-21	2024-03-13	5%	Happy
11	Analisys	There are a few obstacles in the program, this week the team conducted a visitation to the lecturers of the Mobile Device Course to find solutions related to the obstacles that occurred	Note	2024-06-01	2024-06-07	8%	Happy

Showing 1 to 12 of 12 entries

Previous 1 Next

Figure 5 Logbook

Source: Siap-PBL

Table 5 Logbook

Source: Siap-PBL

No.	Step	Detail tasks	Output	Start	Finish	Progress
1	Planning	First Meeting with Project Manager: Discuss the tools and materials needed during the project execution.	Note	2024-03-08	2024-03-08	5%
2	Planning	Second Meeting with Project Manager: Discuss the lesson plan (RPP) and discuss stores for purchasing tools, comparing offline and online stores.	Doc	2024-03-14	2024-03-14	5%

3	Planning	Third Meeting with the Group: Discuss the design of the UI/UX using Figma.	Figma	2024-03-21	2024-03-13	5%
4	Planning	Fourth Meeting with Project Manager: Introduce the tools and materials directly, and then the group starts purchasing the necessary tools and materials directly (offline).	Note	2024-03-28	2024-03-29	5%
5	Planning	Fifth Meeting with Project Manager: Demonstrate the process of assembling the tools and materials into a single product.	Note	2024-03-07	2024-03-07	5%
6	Implementation	Preparing for the Pre-Midterm Progress Presentation: Create a report in the form of a PowerPoint presentation and record a video presentation according to the given guidelines.	PPT, mp4	2024-03-26	2024-03-27	10%
7	Implementation	Frontend and Backend Development: Integrate camera features for video monitoring and other sensors such as sound, temperature, and humidity.	code	2024-04-01	2024-05-01	20%
8	Design	Designing an Intuitive User Interface (UI):	note	2024-05-02	2024-05-15	15%

		Ensure a good User Experience (UX) and make sure the interface is user-friendly and accessible.				
9	Implementation	Testing: Test each component of the application individually to ensure they function properly, and test the interactions between various components to ensure they work well together.	Note, Rangkaian jadi	2024-05-16	2024-05-23	7%
10	Analisys	Data Collection: From planning to testing results, gather data for the final report, and prepare a PowerPoint presentation for the final assessment.	ppt, word	2024-05-24	2024-05-30	5%
11	Analisys	Encountering a Minor Issue: This week, the team will visit the Mobile Devices course lecturer to find solutions for the encountered issues.	Note	2024-06-01	2024-06-07	8%
12	Implementation	Finishing and Retesting: Ensure all components work well and that everything needed before the final assessment is 100% complete.	zip, pdf	2024-06-13	2024-06-20	10%

APPENDIX II – TEAM SCHEDULE

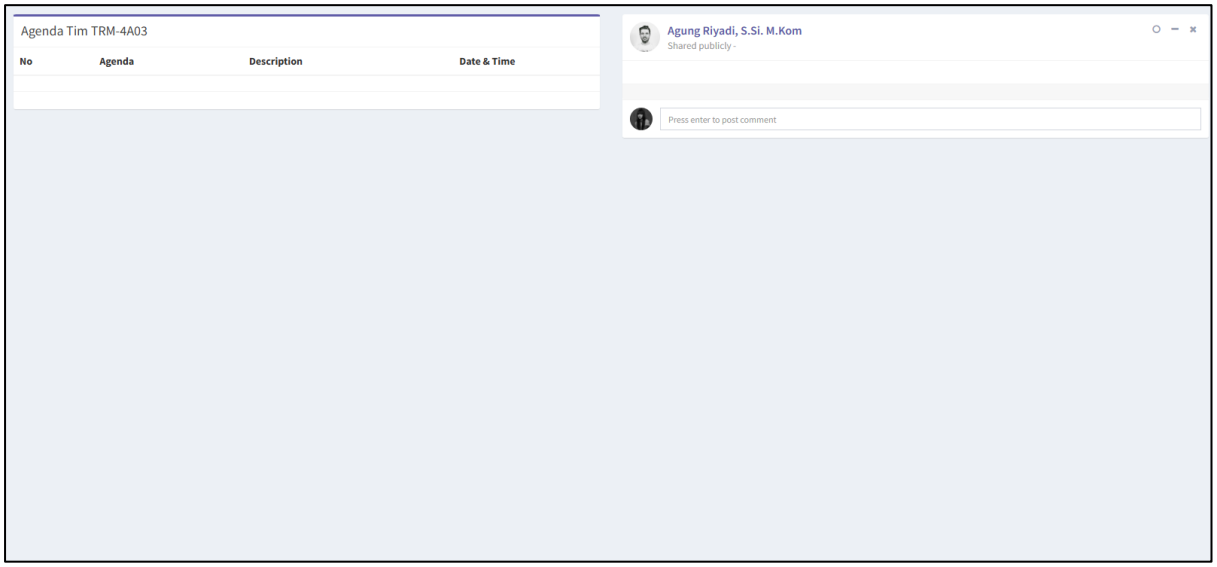

Figure 6 Schedule

Source: Siap-PBL

APPENDIX III – PROJECT BOARD

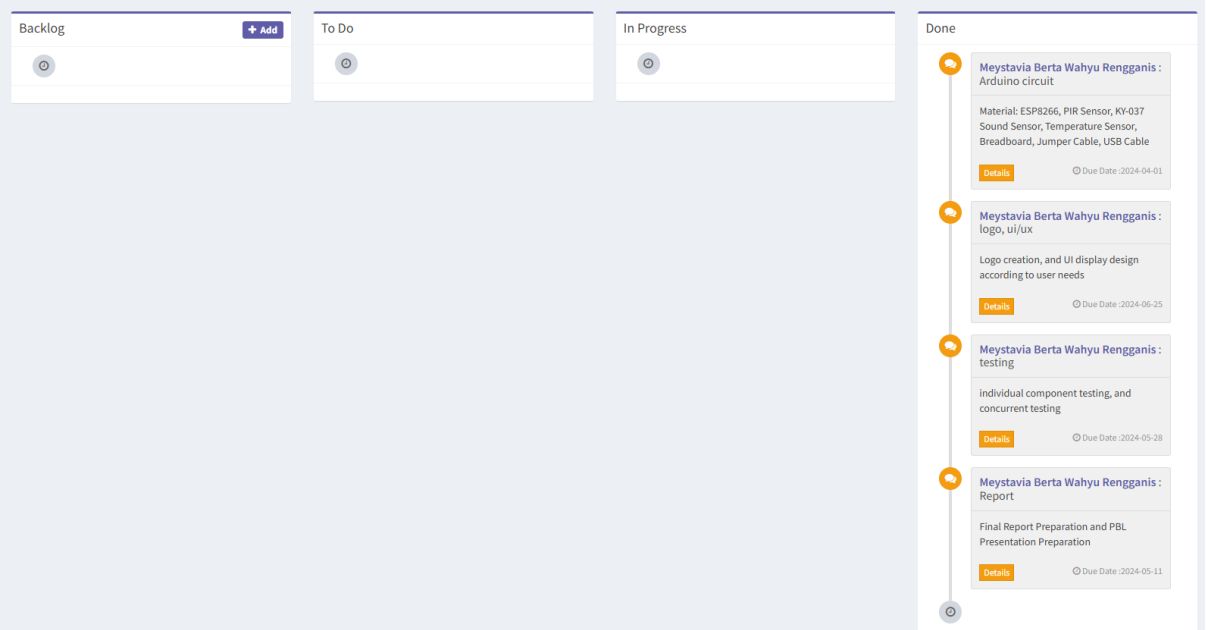

Figure 7 Board

Source: Siap-PBL

APPENDIX IV – PRESENTATION SLIDES

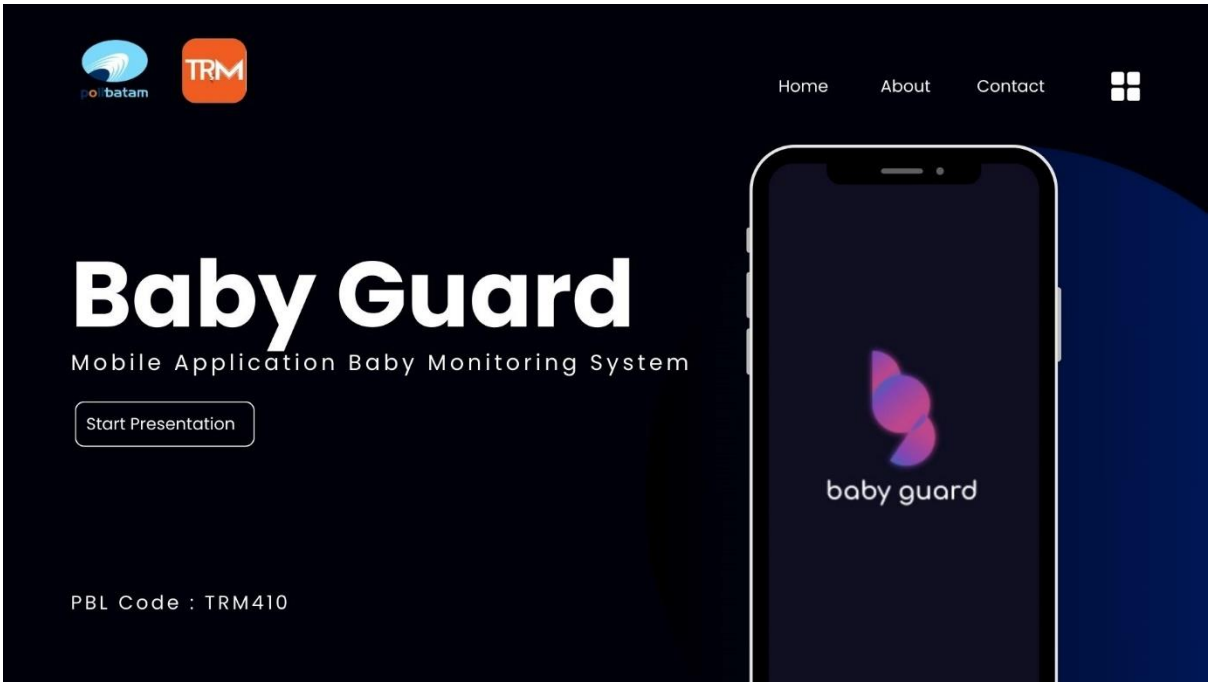

Figure 8 Presentation slide 1

Source: personal document

Figure 9 Presentation slide 2

Source: personal document

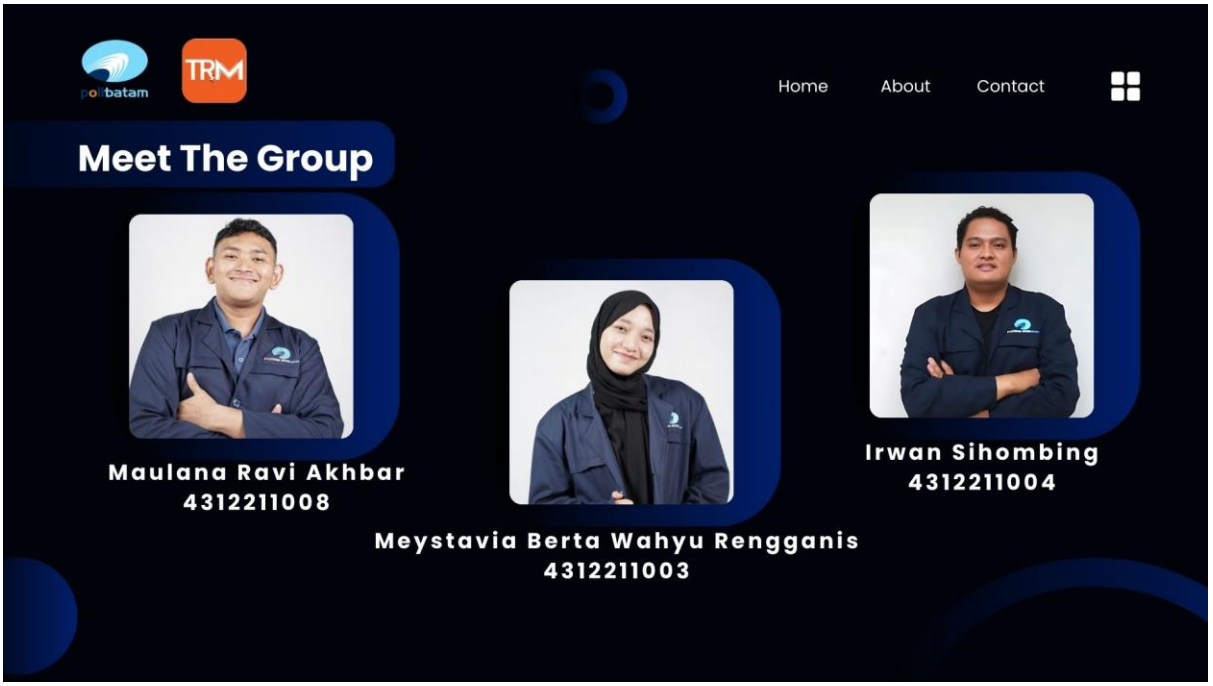

Figure 10 Presentation slide 3

Source: personal document

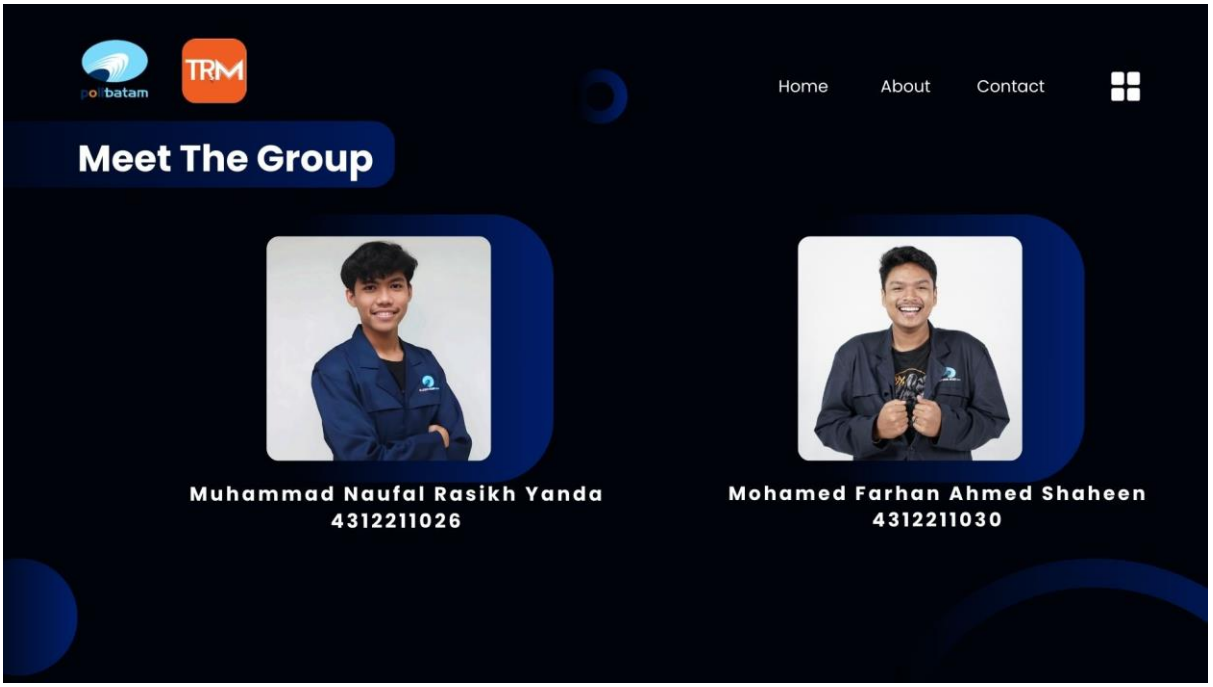

Figure 11 Presentation slide 4

Source: personal document

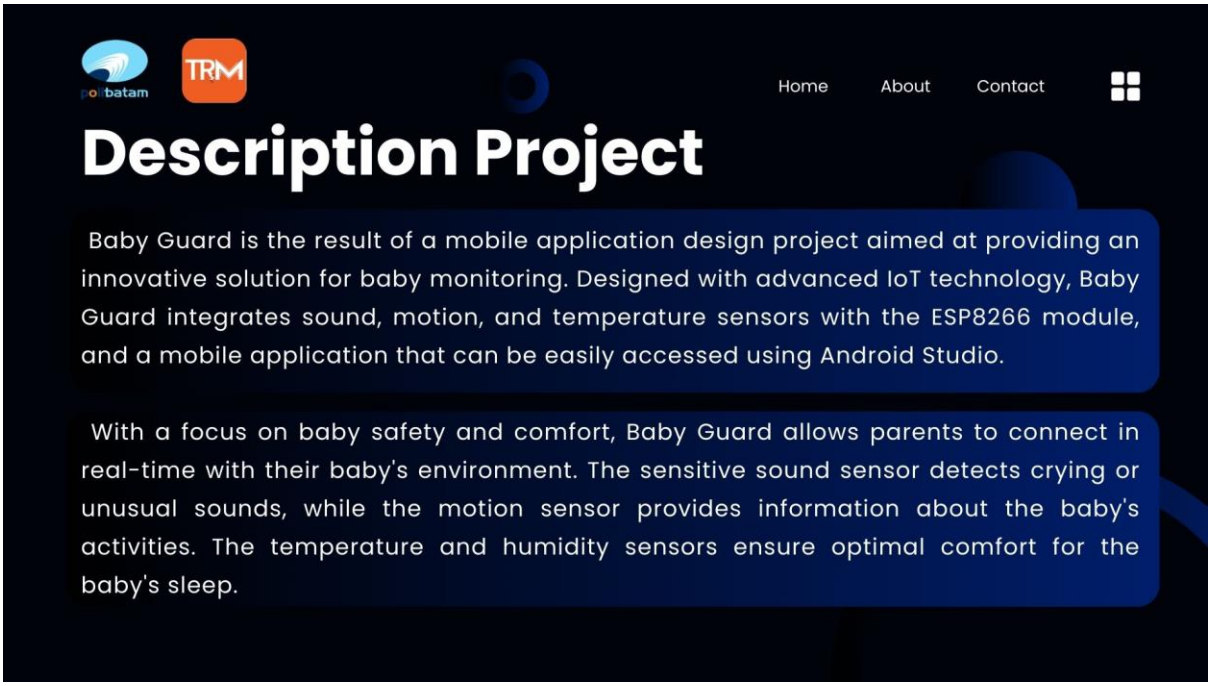

Figure 12 Presentation slide 5

Source: personal document

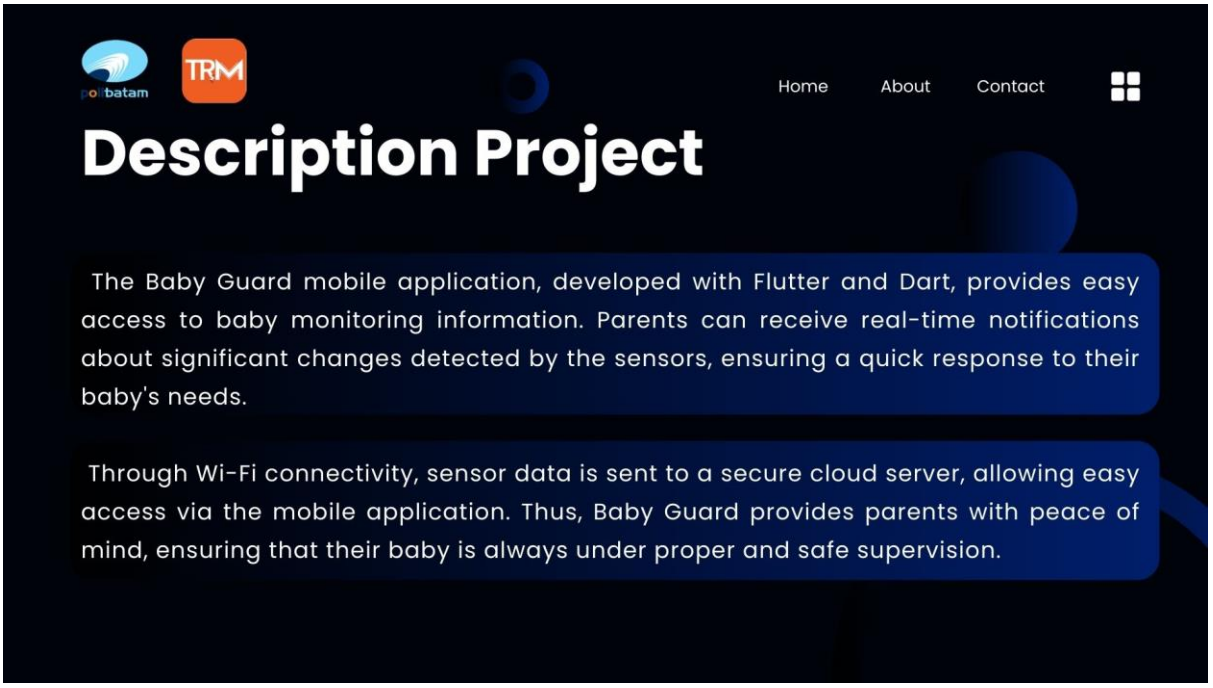

Figure 13 Presentation slide 6

Source: personal document

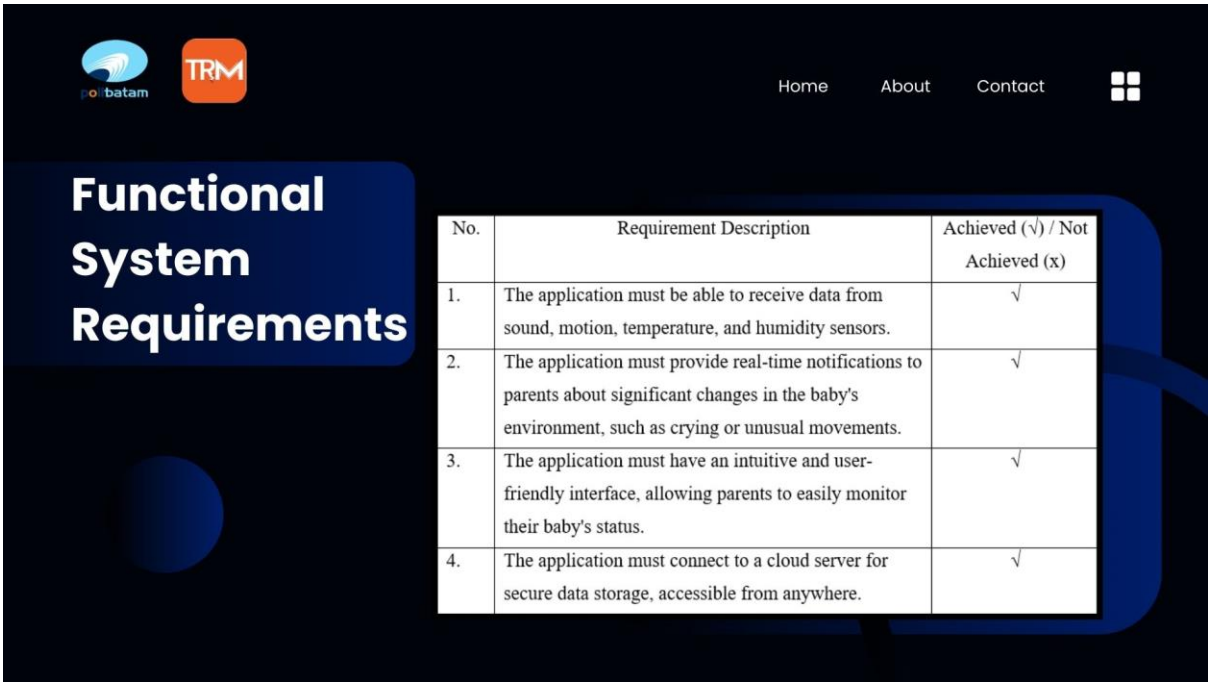

Figure 14 Presentation slide 7

Source: personal document

Figure 15 Presentation slide 8

Source: personal document

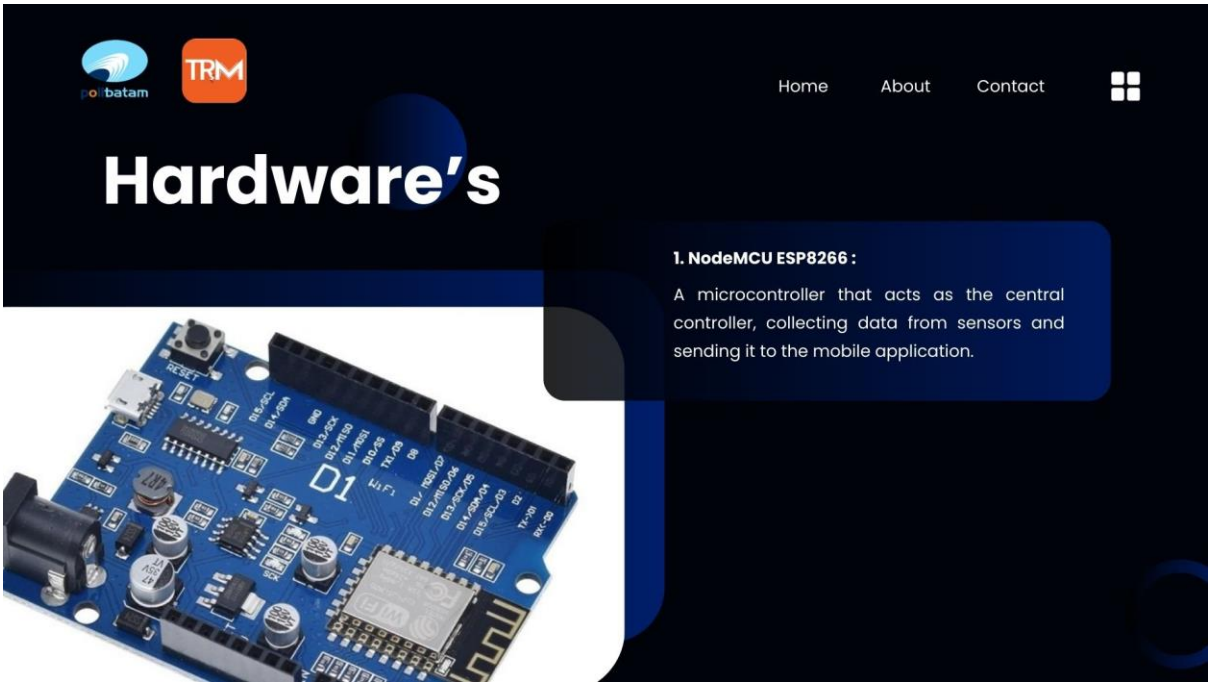

Figure 16 Presentation slide 9

Source: personal document

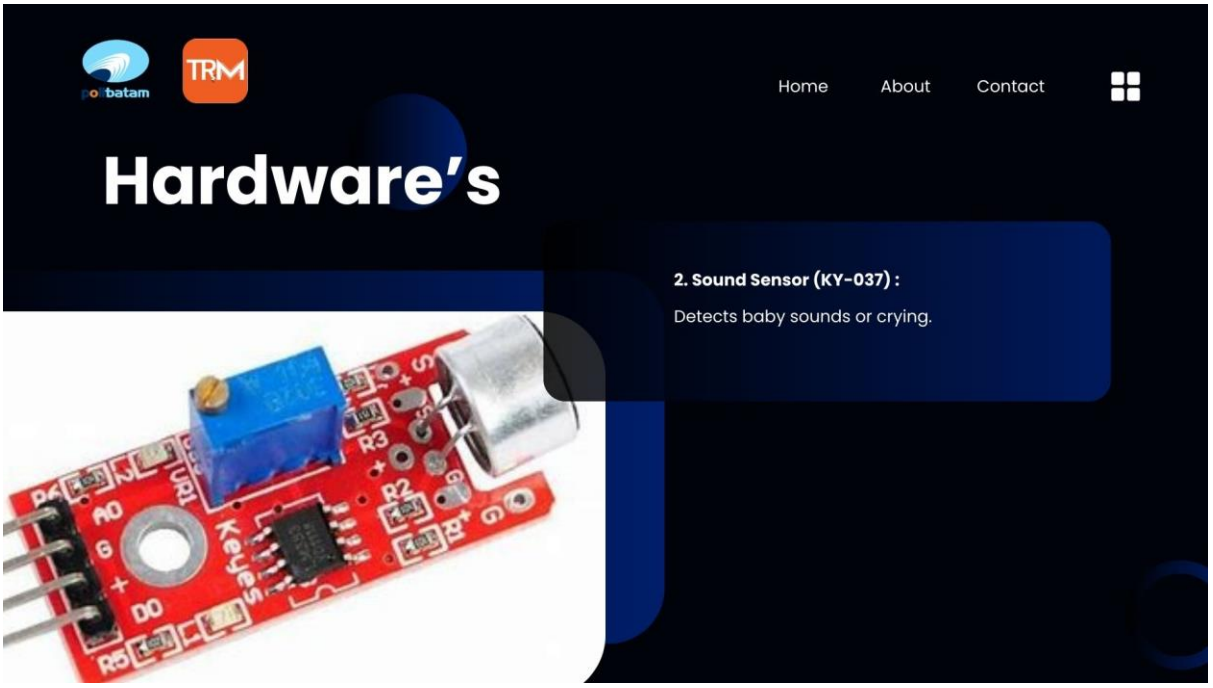

Figure 17 Presentation slide 10

Source: personal document

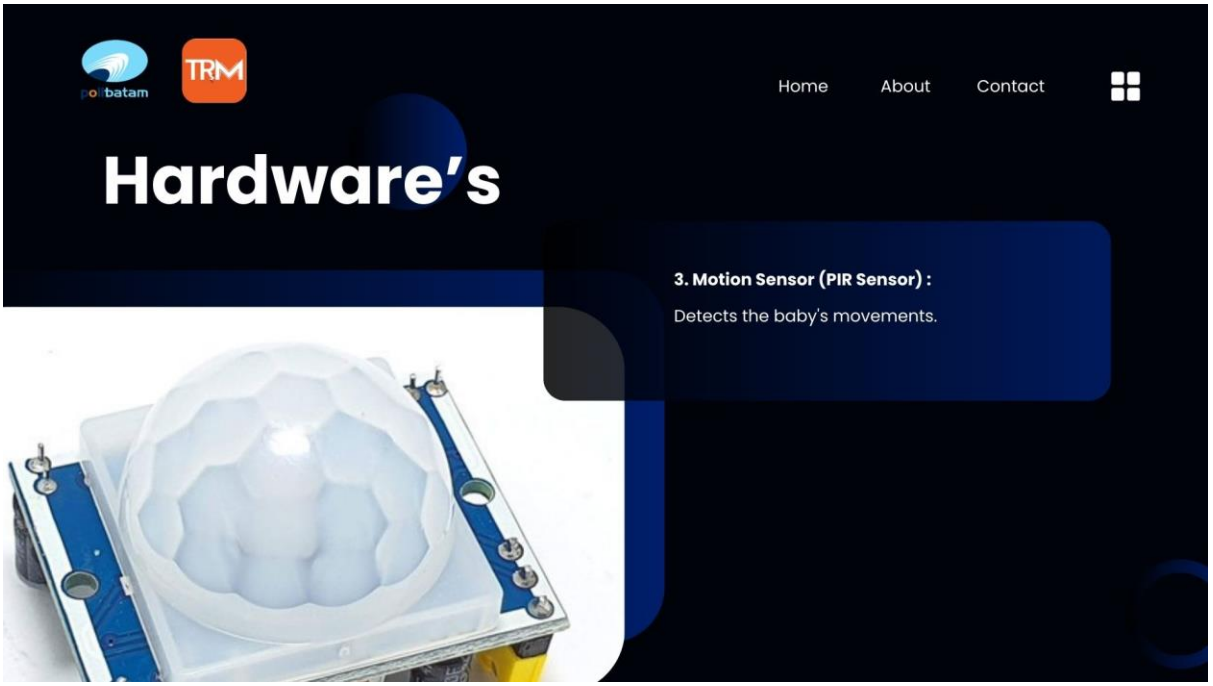

Figure 18 Presentation slide 11

Source: personal document

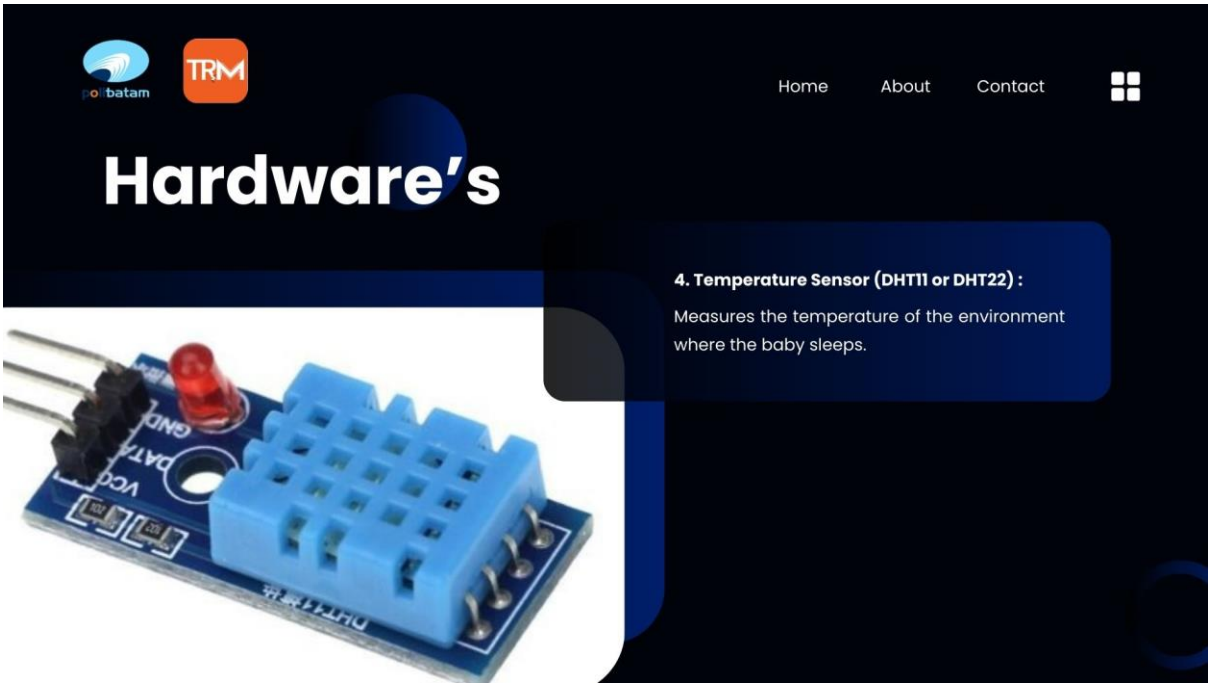

Figure 19 Presentation slide 12

Source: personal document

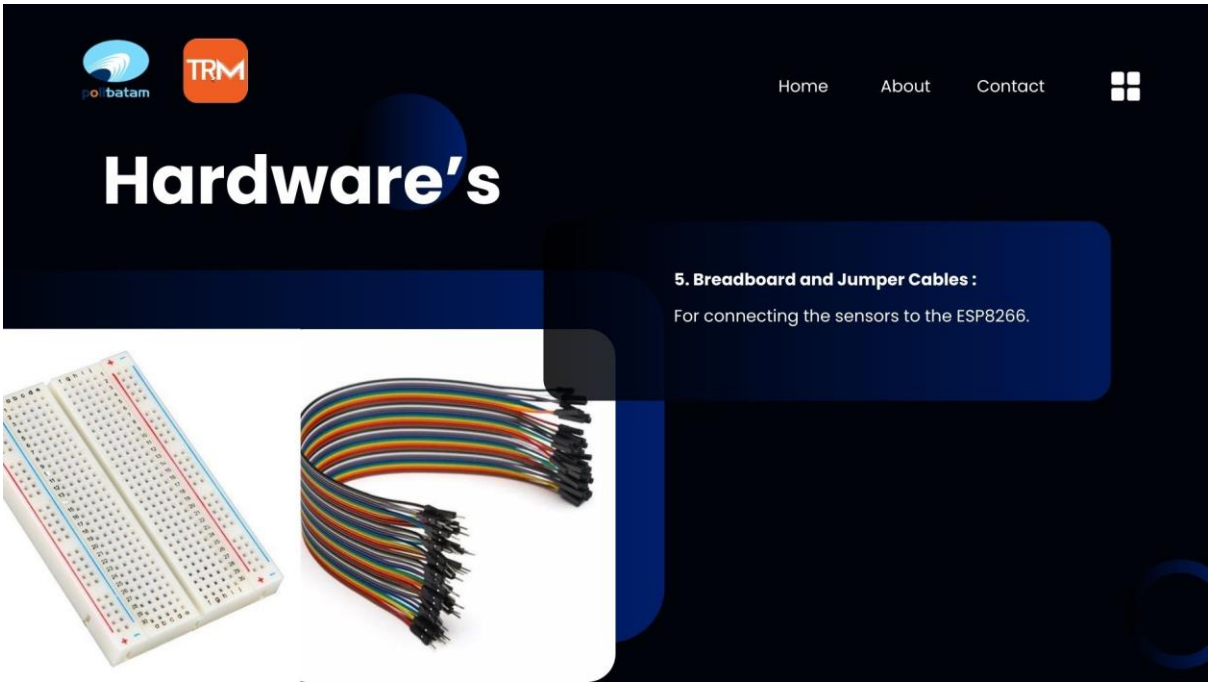

Figure 20 Presentation slide 13

Source: personal document

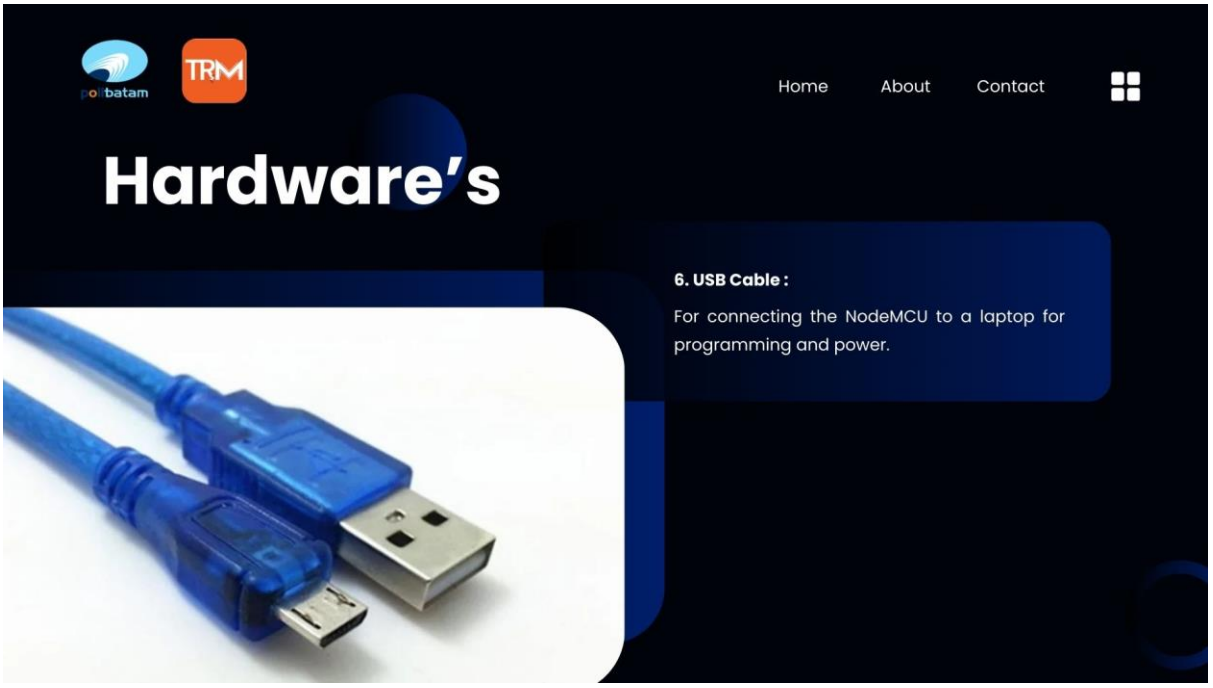

Figure 21 Presentation slide 14

Source: personal document

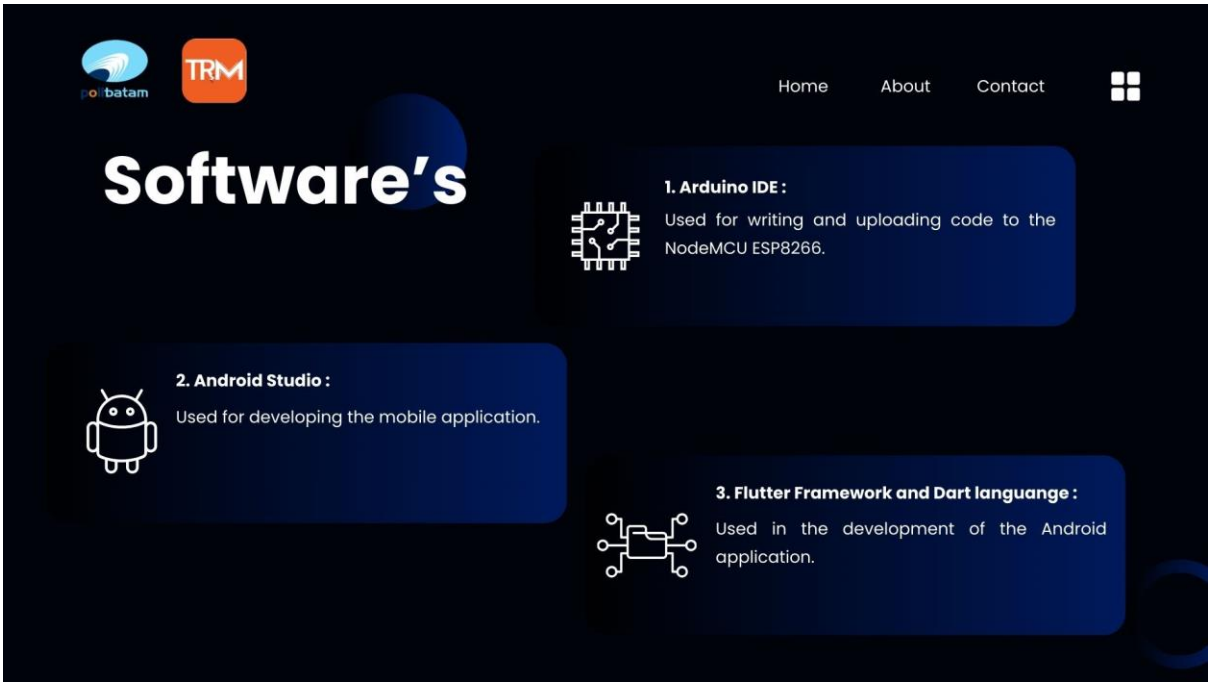

Figure 22 Presentation slide 15

Source: personal document

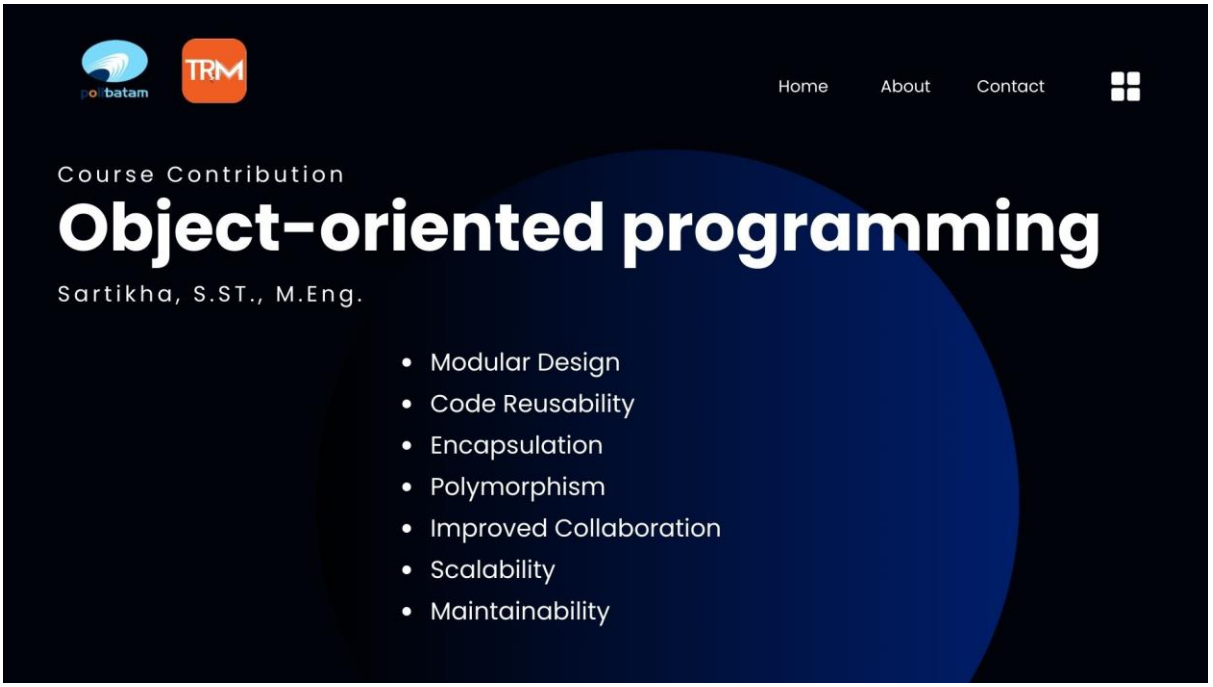

Figure 23 Presentation slide 16

Source: personal document

Figure 24 Presentation slide 17

Source: personal document

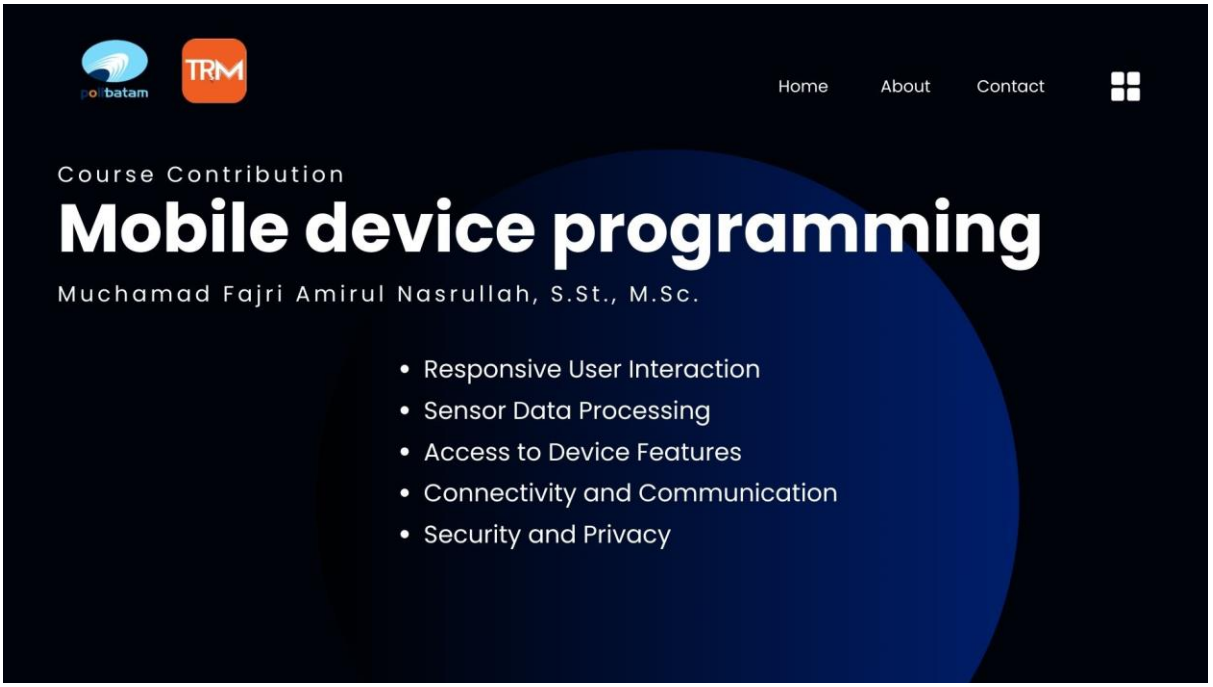

Figure 25 Presentation slide 18

Source: personal document

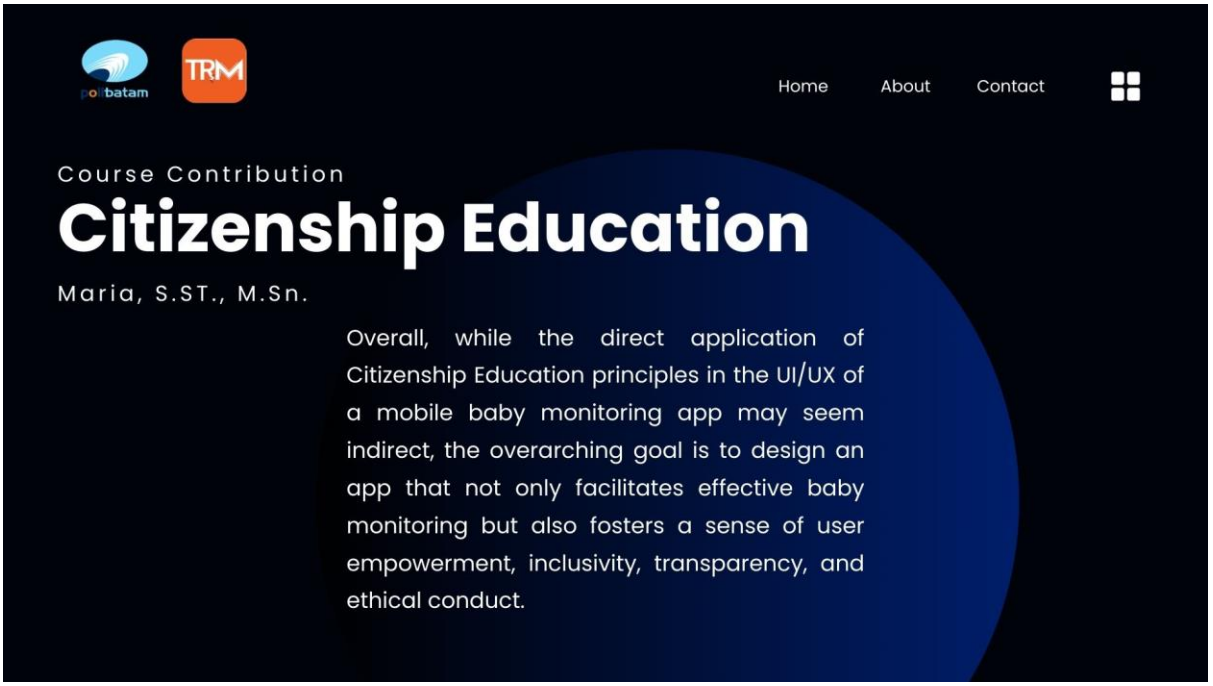

Figure 26 Presentation slide 19

Source: personal document

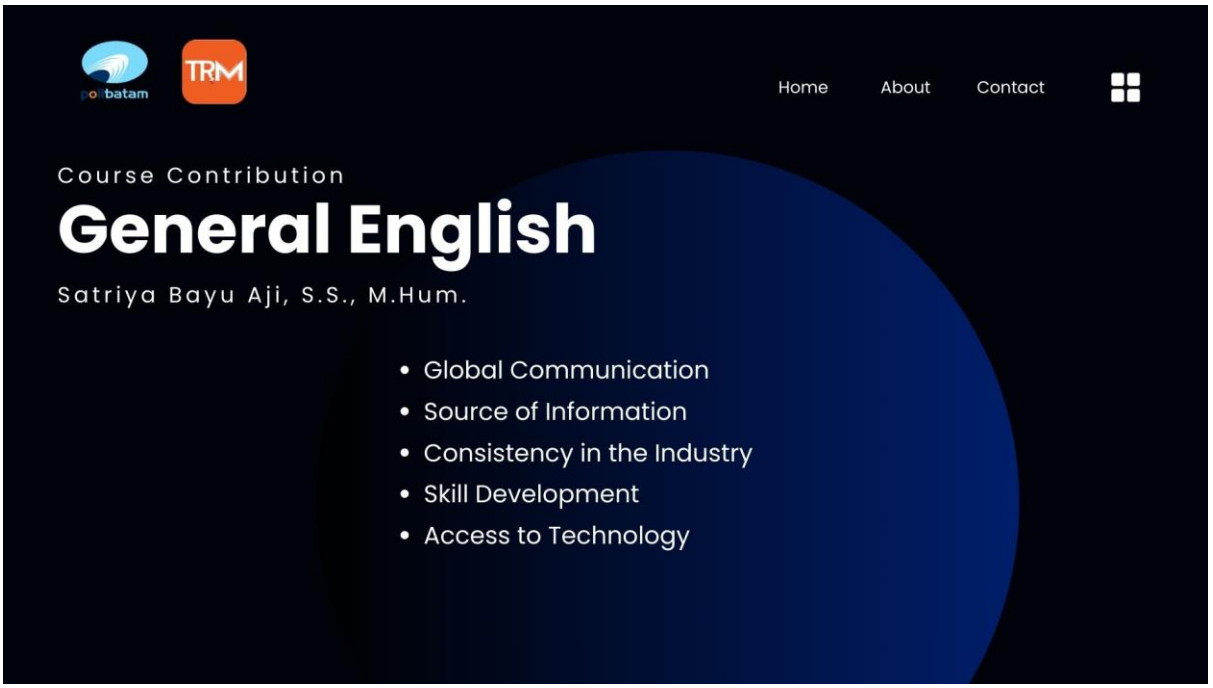

Figure 27 Presentation slide 20

Source: personal document

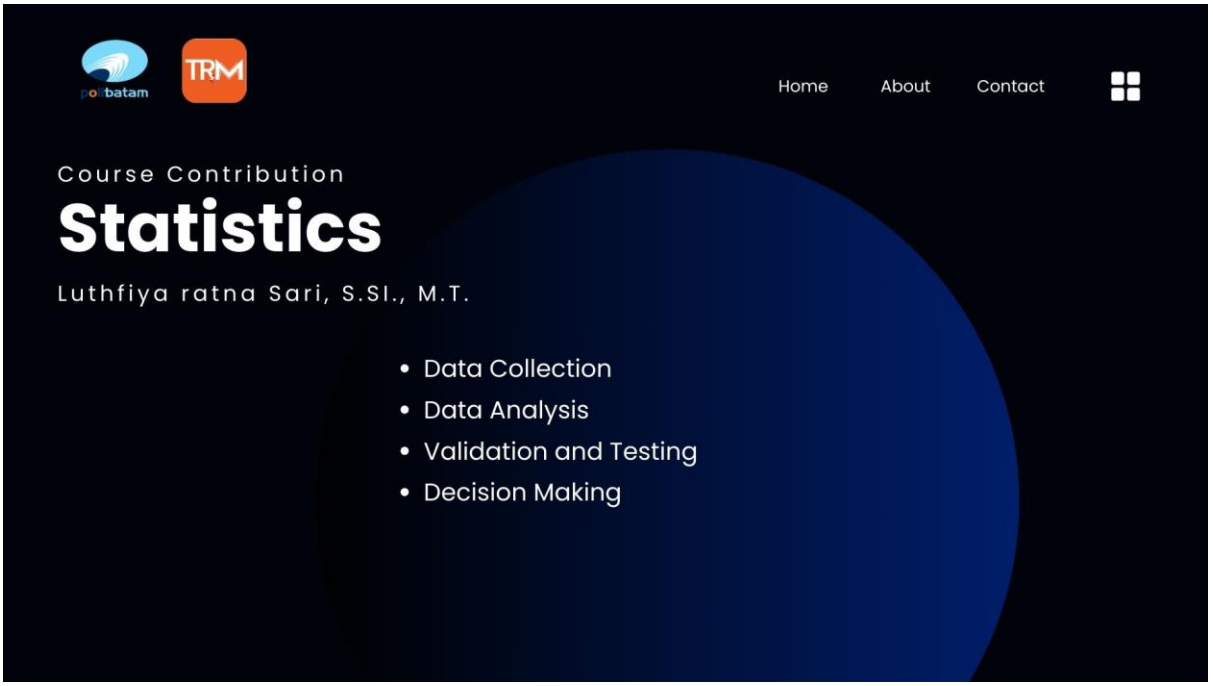

Figure 28 Presentation slide 21

Source: personal document

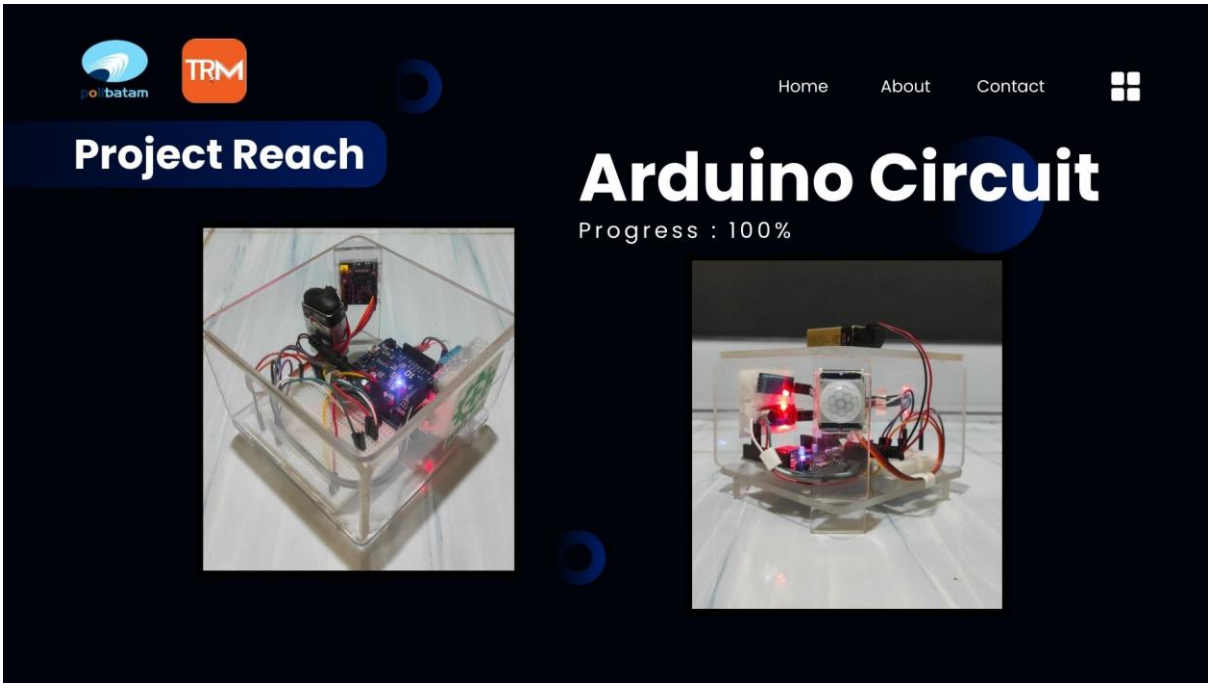

Figure 29 Presentation slide 22

Source: personal document

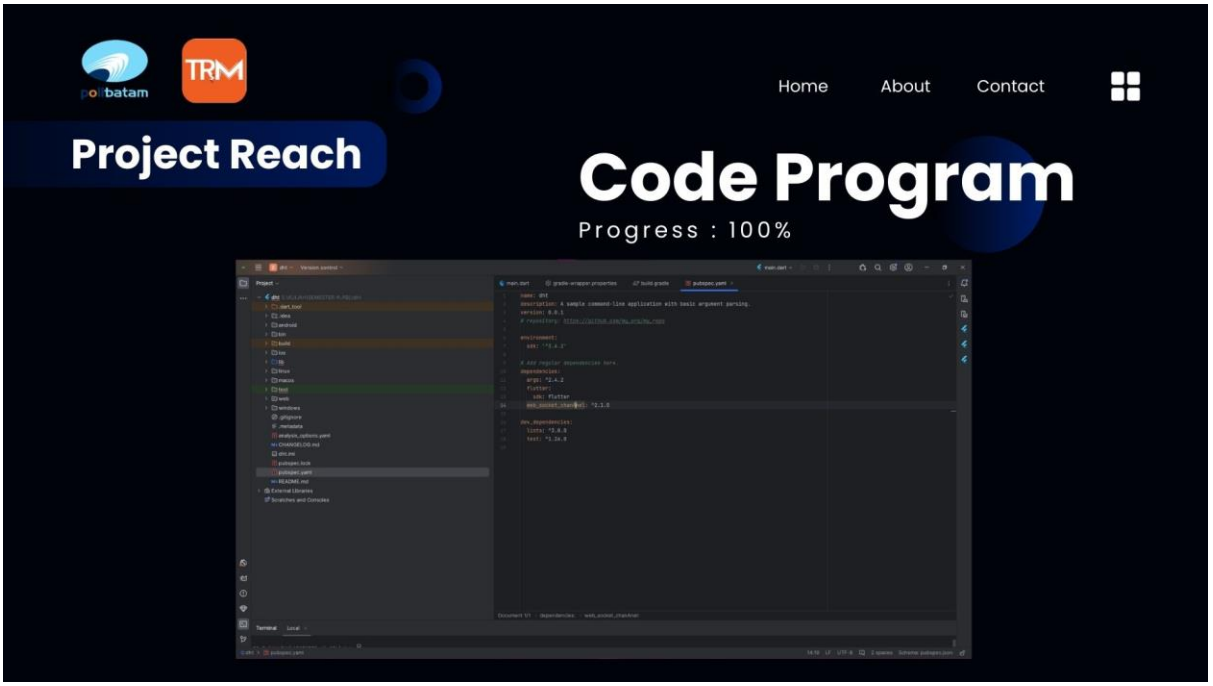

Figure 30 Presentation slide 23

Source: personal document

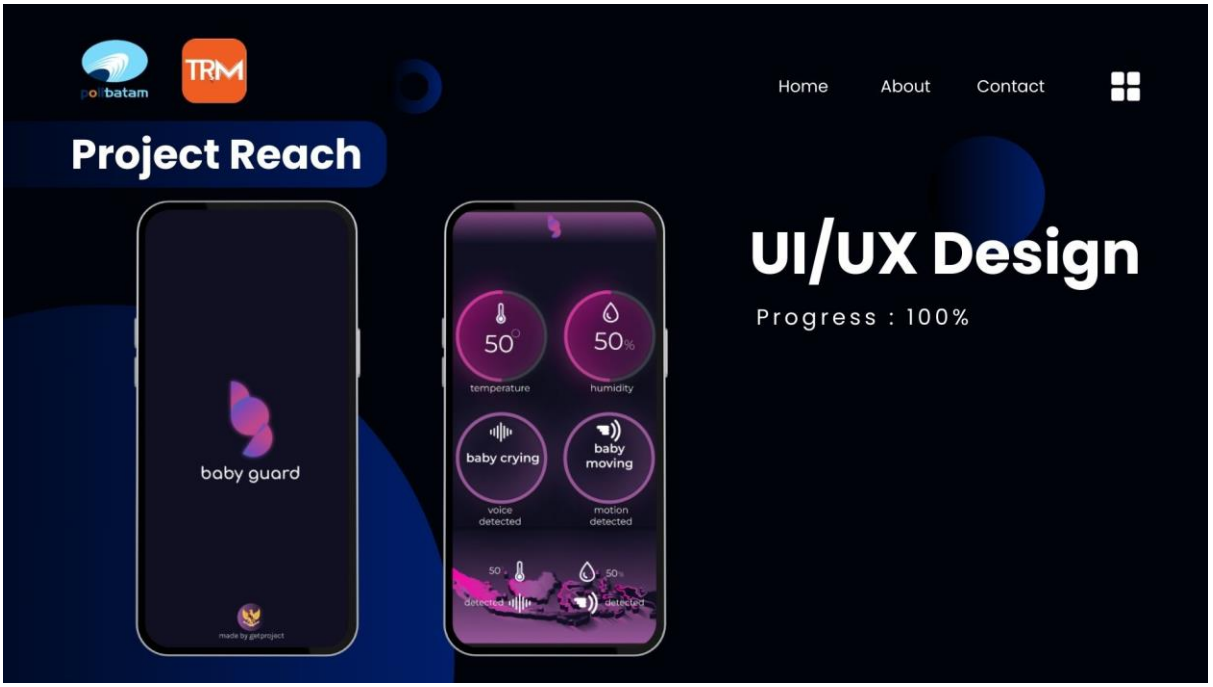

Figure 31 Presentation slide 24

Source: personal document

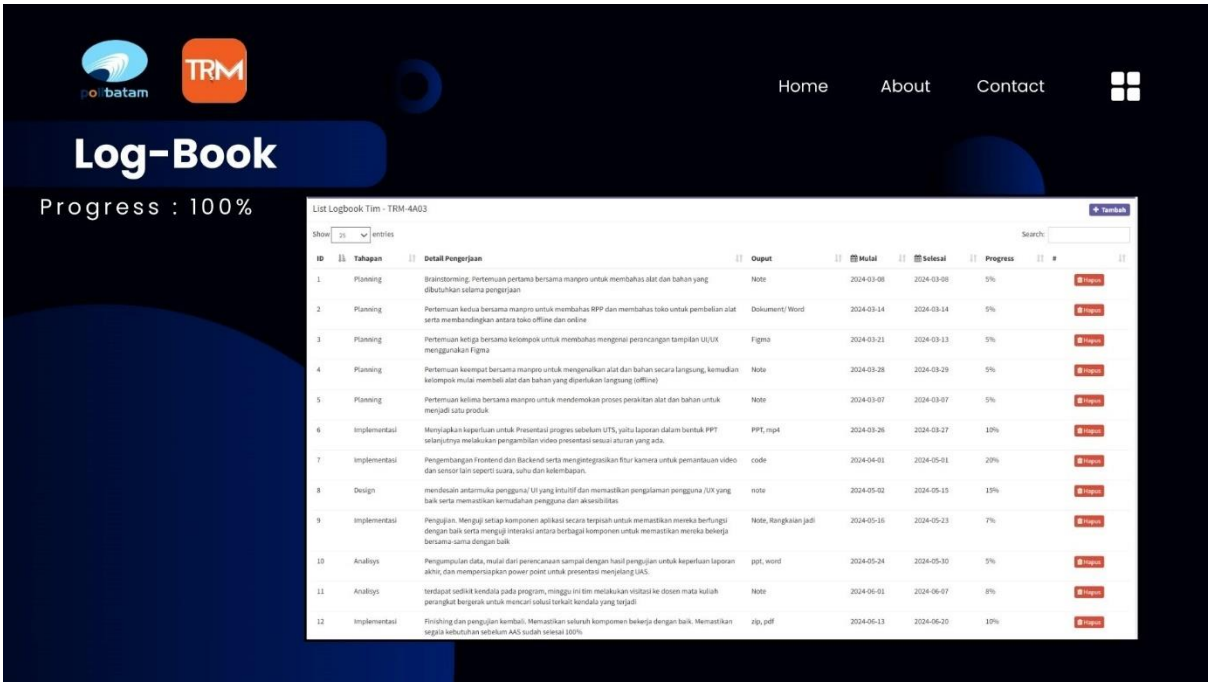

Figure 32 Presentation slide 25

Source: personal document

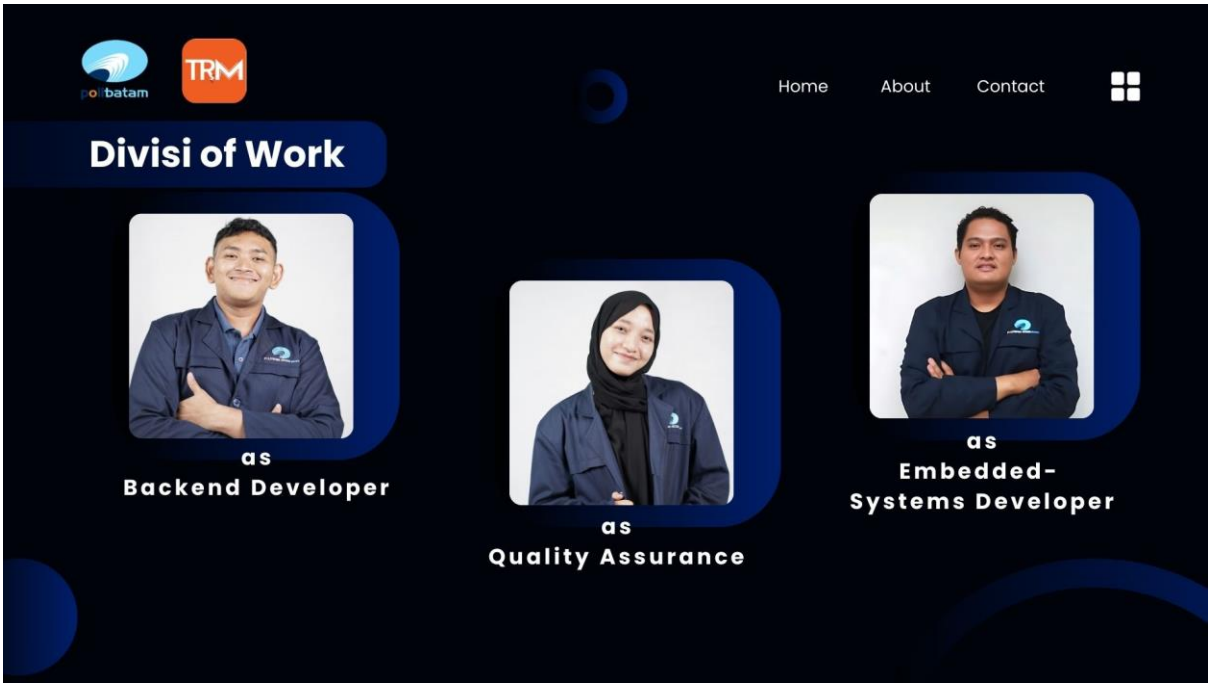

Figure 33 Presentation slide 26

Source: personal document

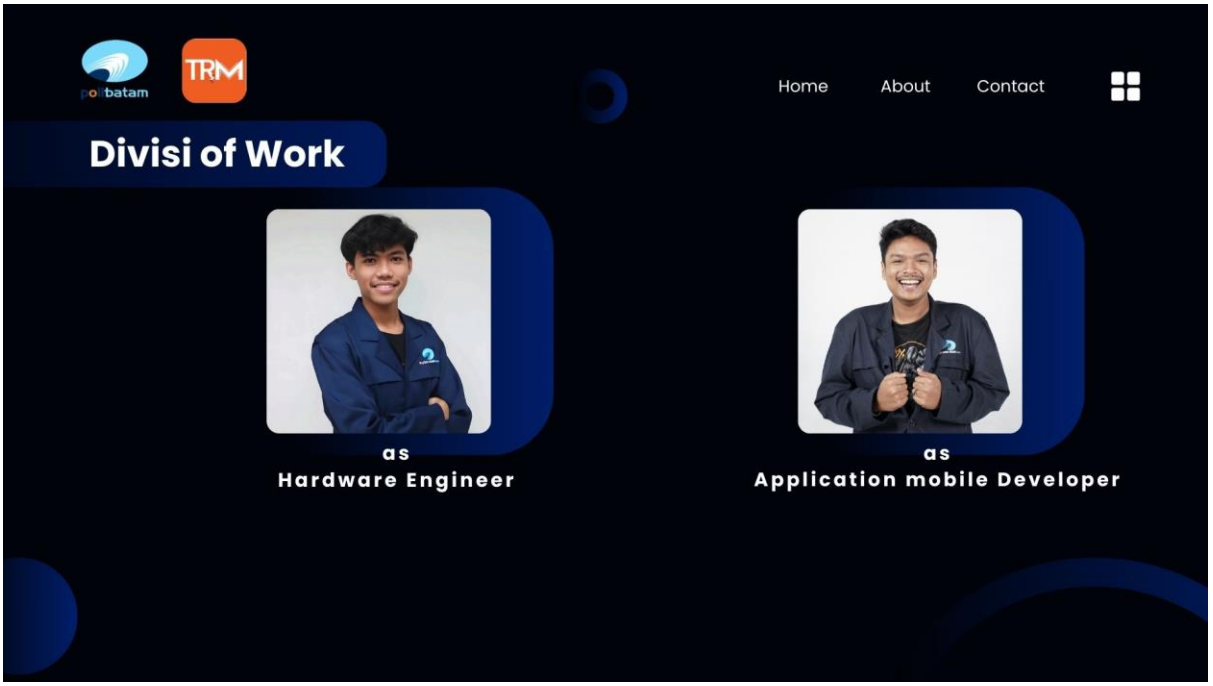

Figure 34 Presentation slide 27

Source: personal document

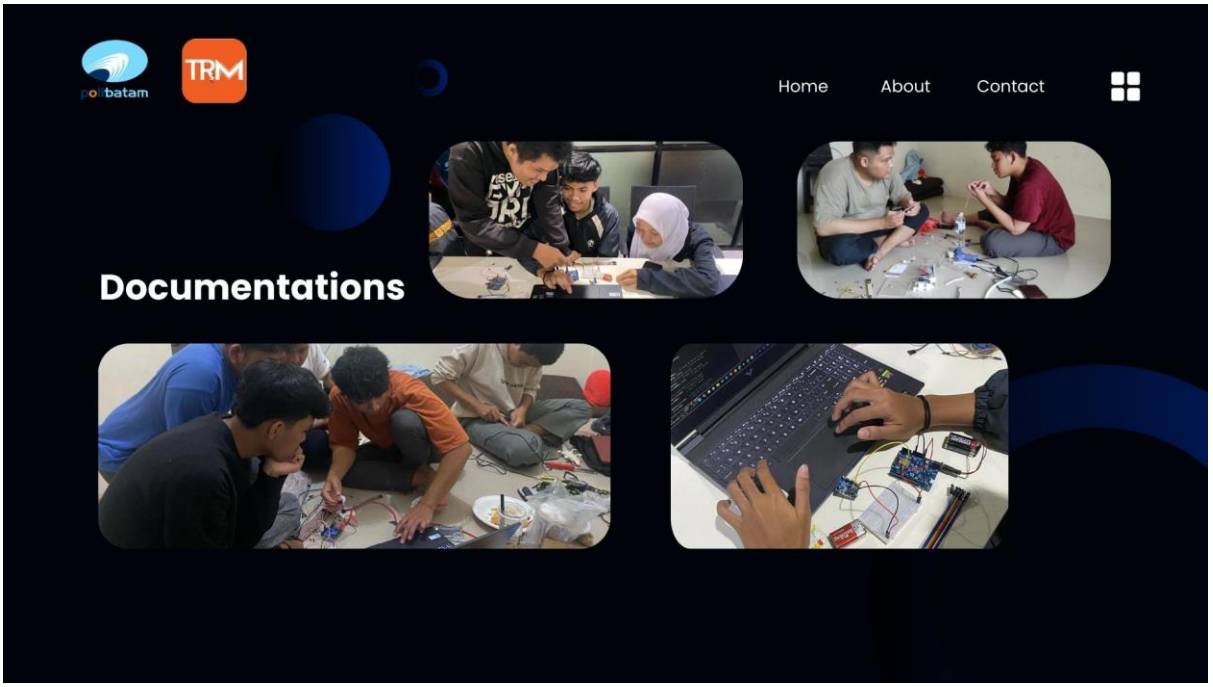

Figure 35 Presentation slide 28

Source: personal document

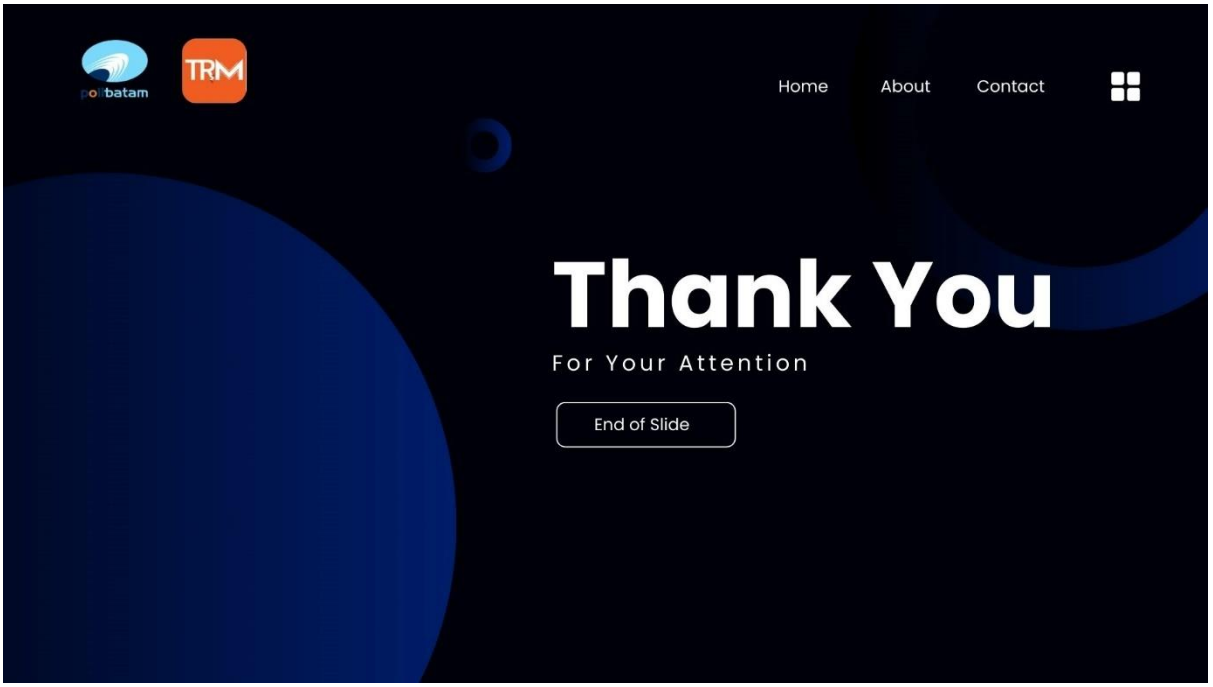

Figure 36 Presentation slide 29

Source: personal document

APPENDIX V - LIST OF FILE LINKS

1. Link of product :
   https://drive.google.com/drive/folders/1RdGCq8-WKTr5BBNPi4rn9k9P59wKmvBl?usp=sharing

2. Link of presentation :
   https://drive.google.com/drive/folders/121ZRTohizdHvaul4ph6QlMOVEA1k_Fna?usp=sharing

3. Link of demo video /teaser :
   https://youtu.be/UCfFrrjk-oM

4. Link of scientific poster:
   https://drive.google.com/drive/folders/1jMwJHaksWgy37a1X94J201K9hF3DAUNl?usp=sharing

5. Link of Intellectual Property Rights Document :
   https://drive.google.com/drive/folders/14HcVL4qUw8W3Xg51z6AheM2ieTvmfneX?usp=sharing

6. Link of handover document scan :
   https://drive.google.com/drive/folders/11fqDpjp_0O5x1B1J2-pe6zBitFzW-NnP?usp=sharing

7. Link of manual book :
   https://drive.google.com/drive/folders/1WX7XaOOvcbdNvqdqDGBYU0_fcQgUhT8O?usp=sharing

trm.polibatam

https://if.polibatam.ac.id/teknologi-rekayasa-multimedia/index.html

kps-trm@polibatam.ac.id