



REAL TIME SYSTEM AND INTERNET OF THINGS FINAL PROJECT REPORT
DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITAS INDONESIA

TRACK GUARD

GROUP A-10

M Farrel Athaillah Nugroho	2106733875
Shabrina Kamiliya Wiyana	2106733894
Prima Shalih	2106636962
Muhammad Fajri Alqomaril	2106651635

PREFACE

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa atas karunia dan rahmat-Nya yang melimpah, sehingga kami, kelompok A10, dapat menyelesaikan proyek akhir praktikum Sistem Waktu Nyata dan IOT dengan judul “TRACK GUARD” sesuai dengan waktu yang telah ditentukan. Dedikasi kami dalam menghasilkan laporan ini bertujuan untuk memberikan kontribusi positif di bidang keamanan dan pertahanan.

Tujuan utama penulisan laporan ini adalah memenuhi tugas akhir praktikum Sistem Waktu Nyata dan IOT. Kami ingin mengucapkan terima kasih setinggi-tingginya kepada Bapak F. Astha Ekadiyanto, S.T., M.Sc., yang telah memberikan bimbingan dalam mata kuliah Sistem Waktu Nyata dan IOT. Penghargaan juga kami sampaikan kepada para asisten laboratorium digital yang turut membimbing kami dari awal praktikum hingga penyelesaian proyek akhir ini.

Kami tidak lupa mengucapkan terima kasih kepada Kak Windiarta, mentor kami, yang dengan penuh dedikasi telah memberikan bantuan dan arahan dalam perjalanan kami menjalankan proyek akhir ini. Kami sadar bahwa laporan yang kami susun masih memiliki kekurangan, dan dengan rendah hati kami mengundang kritik serta saran untuk membantu menyempurnakan dokumen ini.

Semoga laporan ini tidak hanya memenuhi tuntutan akademis, tetapi juga dapat memberikan nilai tambah di dunia nyata. Terima kasih kepada semua pihak yang telah membantu dan mendukung kami dalam perjalanan ini.

Depok, December 8, 2023

Group A-10

TABLE OF CONTENTS

CHAPTER 1.....	4
INTRODUCTION.....	4
1.1 PROBLEM STATEMENT.....	4
1.3 ACCEPTANCE CRITERIA.....	5
1.4 ROLES AND RESPONSIBILITIES.....	5
1.5 TIMELINE AND MILESTONES.....	5
CHAPTER 2.....	7
IMPLEMENTATION.....	7
2.1 HARDWARE DESIGN AND SCHEMATIC.....	7
2.2 SOFTWARE DEVELOPMENT.....	7
2.3 HARDWARE AND SOFTWARE INTEGRATION.....	8
CHAPTER 3.....	9
TESTING AND EVALUATION.....	9
3.1 TESTING.....	9
3.2 RESULT.....	9
3.3 EVALUATION.....	10
CHAPTER 4.....	11
CONCLUSION.....	11

CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

Dalam era teknologi perangkat pemantauan lokasi yang semakin maju, meskipun banyak perangkat pemantau lokasi yang telah tersedia dalam berbagai bentuk, kenyataannya sebagian besar dari perangkat tersebut masih menghadapi keterbatasan dalam mengatasi potensi tantangan keamanan yang mungkin timbul. Sebagai contoh, ketika terjadi pencurian pada suatu benda yang telah dilengkapi dengan sistem pemantau lokasi, apabila perangkat tersebut terputus dari akses internet, hal ini menyebabkan sistem pemantauan tidak mampu beroperasi sebagaimana mestinya, yaitu memberikan lokasi secara *real-time*. Situasi ini dapat menimbulkan kendala, seperti kesulitan dalam menemukan benda yang hilang, hal ini karena perangkat pemantau lokasi harus selalu terhubung ke jaringan internet, seperti WiFi, untuk memberikan informasi lokasi keberadaan perangkat secara *real-time*.

Selain itu, permasalahan lainnya muncul terkait dengan kurangnya mekanisme respons darurat pada sistem pemantau lokasi. Kurangnya efektivitas dalam merespons keadaan darurat dapat menjadi sumber masalah keamanan tambahan. Dengan kata lain, sistem tersebut mungkin tidak cukup responsif dalam situasi yang memerlukan tindakan cepat, seperti adanya ancaman atau kejadian kejahatan. Oleh karena itu, perlu adanya inovasi pada sistem pemantau lokasi untuk meningkatkan responsivitas terhadap potensi keamanan, tidak hanya sebatas memberikan lokasi secara *real-time*, tetapi juga memberikan mekanisme respons darurat yang efektif.

1.2 PROPOSED SOLUTION

Sebagai solusi atas tantangan tersebut, proyek "TrackGuard" mempersembahkan serangkaian inovasi yang terintegrasi. Pertama, modul SIM800L digunakan untuk menghubungkan ESP32 dengan jaringan seluler melalui SIM card hal ini berguna untuk mengatasi keterbatasan akses internet seperti WiFi yang tidak selalu ada pada semua tempat. Kedua, penggunaan ESP32 sebagai platform utama memastikan optimasi daya dan portabilitas, memungkinkan sistem ini dapat beroperasi secara efisien dan mudah dibawa ke

berbagai lokasi. Ketiga, integrasi modul GPS NEO-6M memastikan pendapatan lokasi yang akurat dan pemantauan *real-time* melalui platform ThingsBoard sehingga dengan hal ini dapat memberikan visualisasi yang dapat mudah dimengerti. Keempat, respons darurat yang ditingkatkan dengan integrasi buzzer dan LED, yang memberikan sinyal visual dan auditori ketika ESP32 sudah terhubung ke internet. Dengan solusi ini, diharapkan proyek ini dapat menjadi solusi yang holistik dan adaptif terhadap dinamika kebutuhan pemantauan dan pengamanan.

1.3 ACCEPTANCE CRITERIA

Kriteria penerimaan proyek ini adalah sebagai berikut:

1. Sistem mampu memanfaatkan SIM card untuk mengakses internet melalui modul SIM800L.
2. Sistem mampu mengirimkan data lokasi dari modul GPS NEO-6M ke platform ThingsBoard.
3. Sistem dapat menampilkan lokasi keberadaan perangkat secara real-time pada platform ThingsBoard.
4. Sistem mampu mengaktifkan LED dan Buzzer ketika terhubung ke internet.

1.4 ROLES AND RESPONSIBILITIES

The roles and responsibilities assigned to the group members are as follows:

Roles	Responsibilities	Person
Rangkaian dan Code	Membantu membeli alat dan perlengkapan, membantu membuat code, Thingsboard, dan membantu membuat rangkaian	M Farrel Athaillah Nugroho
Rangkaian dan Dokumen	Membantu membeli sebagian besar alat dan perlengkapan, membuat	Shabrina Kamiliya Wiyana

	skematik rangkaian, membantu merangkai perangkat, membuat laporan, membuat PPT, dan readmeMD	
Rangkaian dan Code	Membantu membeli alat dan perlengkapan, membantu membuat code, Thingsboard, dan membantu membuat rangkaian	Prima Shalih
Dokumen dan Rangkaian	Membantu membeli alat dan perlengkapan, membantu membuat rangkaian, dan membuat user guide.	Muhammad Fajri Alqomaril

Table 1. Roles and Responsibilities

1.5 TIMELINE AND MILESTONES

Month	November										December									
	21	22	23	24	25	26	27	28	29	30	1	2	3	4	5	6	7	8	9	10
Progress																				
Hardware Design completion																				
Software Development																				
Integration and Testing of Hardware and Software																				
Final Product Assembly and Testing																				

Table 2. Ganttchart

- a) Hardware Design completion: 21 November 2023 - 5 Desember 2023

Pada tahap ini kami sudah mengembangkan ide terkait produk yang akan dibuat, mempersiapkan alat, membuat skematik desain rangkaian, dan merangkai alat

- b) Software Development: 29 November 2023 - 7 Desember 2023

Pada tahap ini kami sudah menyelesaikan code yang selanjutnya akan dimasukan kedalam alat, serta membuat *user-interface* pada *thingsboard*

- c) Integration and Testing of Hardware and Software: 5 - 8 Desember 2023

Pada tahap ini kami sudah memasukan code yang sebelumnya sudah dibuat, kedalam alat yang sudah dirangkai.

- d) Final Product Assembly and Testing: 9 - 10 Desember 2023

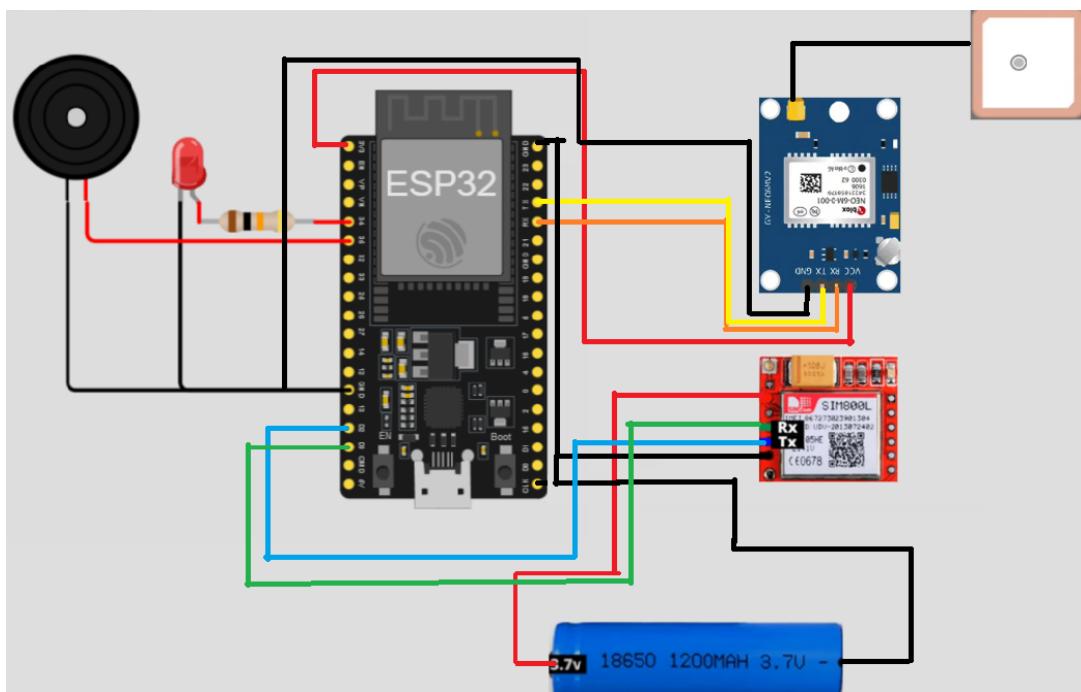
Pada tahap ini kami melakukan testing dan memvalidasi input yang dihasilkan pada alat.

CHAPTER 2

IMPLEMENTATION

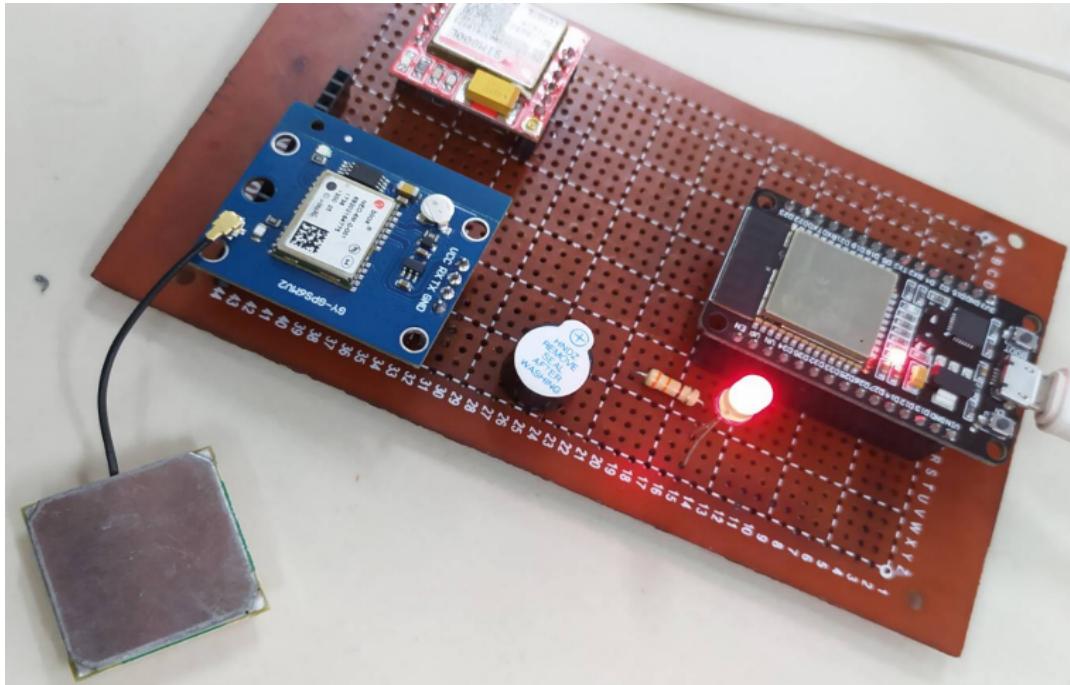
2.1 HARDWARE DESIGN AND SCHEMATIC

Proses perencanaan alat dimulai dengan merancang skematik menyesuaikan pin-pin yang digunakan dan komponen yang akan digunakan, dan selanjutnya, setelah prototipe selesai dibuat, selanjutnya dilakukan perancangan langsung. Tujuan dari langkah ini adalah untuk mencegah kerusakan pada rangkaian asli.



Gambar 1 (Skematik Rangkaian)

Proses selanjutnya dalam pengembangan alat melibatkan penyusunan menggunakan PCB dan melakukan penyolderan sesuai dengan skematik yang telah dirancang sebelumnya.



Gambar 2 (Rangkaian)

Dalam proyek "TrackGuard", kami menghadirkan serangkaian inovasi terintegrasi untuk mengatasi tantangan keamanan pada perangkat pemantau lokasi. Berikut adalah deskripsi untuk setiap komponen yang digunakan:

1. ESP32, Sebagai otak utama sistem, ESP32 digunakan untuk mengontrol dan mengkoordinasikan fungsi-fungsi perangkat, dalam proyek ini ESP32 memiliki peran kunci dalam mengumpulkan data dari modul GPS dan mengirimkannya ke platform ThingsBoard untuk pemantauan lokasi
2. LED, Memberikan sinyal visual bahwa ESP32 sudah terhubung ke internet
3. Buzzer, Memberikan sinyal auditori untuk indikasi bahwa ESP32 sudah terhubung ke internet
4. Resistor, Resistor dalam rangkaian proyek "TrackGuard" dihubungkan ke buzzer dan LED dengan tujuan untuk membatasi arus listrik dan mencegah kerusakan pada kedua komponen tersebut.
5. SIM800L, modul SIM800L dihubungkan ke ESP32 untuk memungkinkan ESP32 terhubung dengan jaringan seluler melalui kartu SIM. Sehingga dapat menyediakan akses jaringan seluler bagi ESP32, memungkinkan perangkat untuk berkomunikasi di lokasi tanpa ketersediaan jaringan WiFi.

6. GPS NEO 6-M, Modul GPS NEO-6M diintegrasikan dengan ESP32 melalui koneksi serial, memungkinkan ESP32 untuk terus memantau dan membaca data lokasi yang diterima dari GPS.
7. Battery, Menyediakan daya untuk menjalankan seluruh sistem secara mobile.
8. PCB, Membantu penyusunan komponen dan mengoptimalkan koneksi antara perangkat

2.2 SOFTWARE DEVELOPMENT

Program ini dirancang untuk mengakuisisi data GPS, mengelola koneksi WiFi dan ThingsBoard, serta memberikan tampilan status melalui LED dan buzzer. Program berfokus pada pengumpulan dan pengiriman data lokasi ke ThingsBoard untuk pemantauan secara real-time, dimana program ini akan dimasukan kedalam ESP32 yang berperan sebagai otak rangkaian.

Berikut merupakan yang digunakan:

```
#include <ArduinoJson.h>
#include <TinyGPS++.h>
#include <HardwareSerial.h>
#include <WiFi.h>
#include <PubSubClient.h>

#define RXPin 16
#define TXPin 17
#define GPSBaud 9600
#define LED_PIN 22
#define BUZZER_PIN 23 // Define buzzer pin

TinyGPSPlus gps;
HardwareSerial mygps(1);

WiFiClient espClient;
PubSubClient client(espClient);

const char* server = "thingsboard.cloud";
const char* token = "mqkb1ts57g5kwbstv04m";

float latitude;
float longitude;

void setup() {
  Serial.begin(115200);
  mygps.begin(GPSBaud, SERIAL_8N1, RXPin, TXPin);

  if (!mygps) {
```

```

        Serial.println("Failed to initialize GPS");
    } else {
        Serial.println("Success to initialize GPS");
    }

    // Print GPS data for 30 seconds
    for (unsigned long startMillis = millis(); millis() - startMillis < 30000; ) {
        while (mygps.available()) {
            Serial.write(mygps.read());
        }
    }

    client.setServer(server, 1883);

    WiFi.begin("ketum", "imesukses");
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.println("Connecting to WiFi...");
    }
    Serial.println("Connected to WiFi");

    pinMode(LED_PIN, OUTPUT);
    pinMode(BUZZER_PIN, OUTPUT); // Set buzzer pin as output
}

void loop() {
    while (mygps.available() > 0) {
        if (gps.encode(mygps.read())) {
            displayInfo();
        }
    }

    if (!client.connected()) {
        reconnect();
    }
    client.loop();

    if (WiFi.status() == WL_CONNECTED) {
        digitalWrite(LED_PIN, HIGH);
        digitalWrite(BUZZER_PIN, HIGH); // Turn buzzer ON
    } else {
        digitalWrite(LED_PIN, LOW);
        digitalWrite(BUZZER_PIN, LOW); // Turn buzzer OFF
    }
}

void displayInfo() {
    if (gps.location.isValid()) {
        latitude = (gps.location.lat());
        longitude = (gps.location.lng());

        Serial.print("Latitude: ");
        Serial.println(latitude, 6);
        Serial.print("Longitude: ");
        Serial.println(longitude, 6);
    }
}

```

```

String latStr = String(latitude, 6);
String lonStr = String(longitude, 6);

char latChar[latStr.length() + 1];
char lonChar[lonStr.length() + 1];

latStr.toCharArray(latChar, latStr.length() + 1);
lonStr.toCharArray(lonChar, lonStr.length() + 1);

client.publish("v1/devices/me/telemetry/latitude", latChar);
client.publish("v1/devices/me/telemetry/longitude",
lonChar);
} else {
Serial.println("GPS data is not valid");
}
}

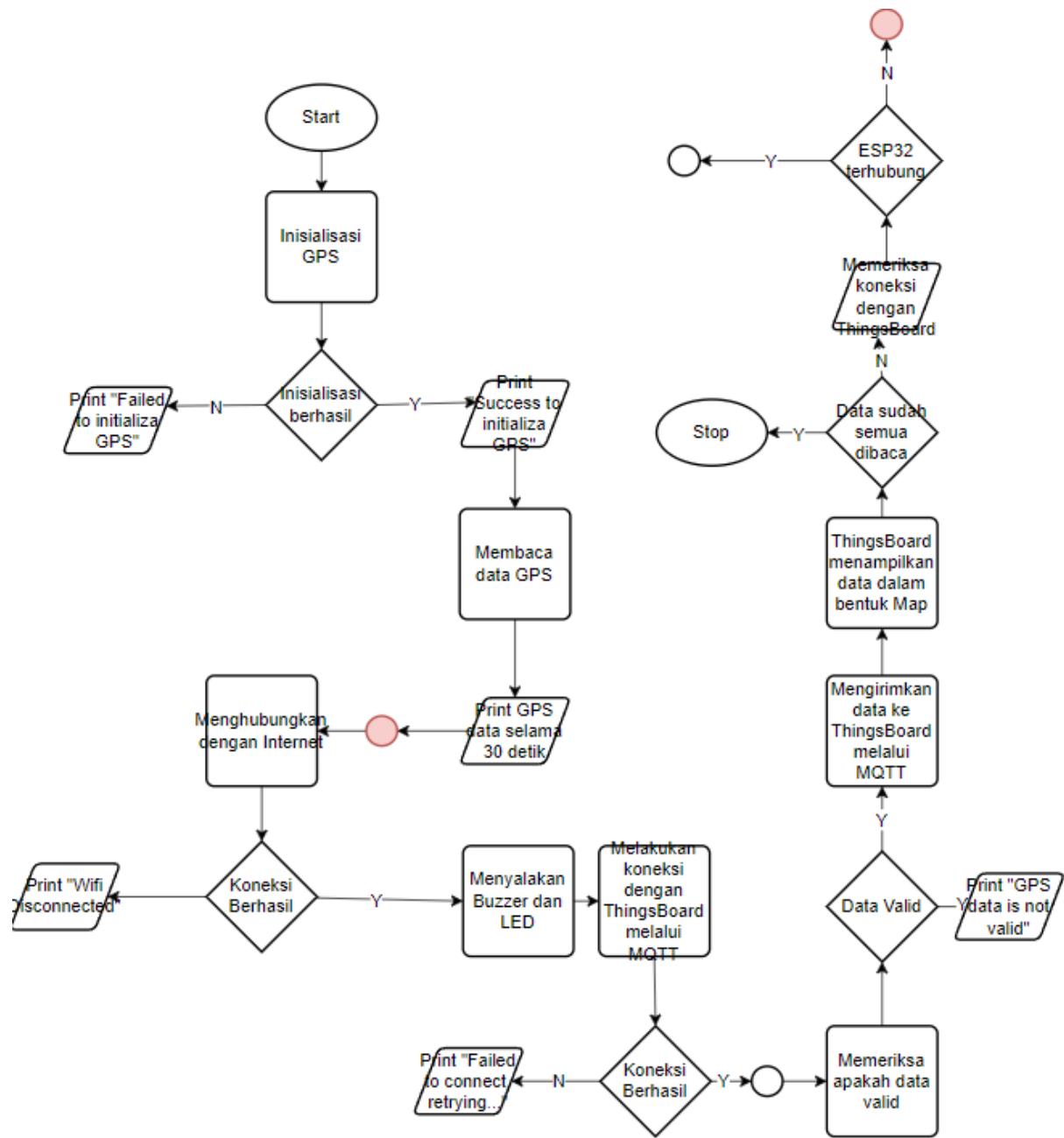
void reconnect() {
while (!client.connected()) {
if (client.connect("ESP32Client", token, NULL)) {
Serial.println("Connected to Thingsboard");
} else {
Serial.println("Failed to connect, retrying...");
delay(5000);
}
}
}
}
}

```

Pada kode tersebut diawali dengan melakukan *include* pada beberapa library, seperti ArduinoJson.h yang digunakan untuk komunikasi dengan platform IOT, TinyGPS++.h yang digunakan untuk melakukan *parsing* NMEA dari GPS dan mengekstraksikan informasi lokasi, HardwareSerial.h yang menyediakan akses ke serial komunikasi hardware pada ESP32, WiFi.h yang digunakan untuk menghubungkan ESP32 kedalam jaringan WiFi, dan PubSubClient.h yang digunakan untuk komunikasi, dan digunakan untuk mengirimkan data ke platform Thingsboard. Pada bagian fungsi setup, dilakukan inisialisasi komunikasi serial, menghubungkan GPS, dan memeriksa apakah inisialisasi berhasil, selanjutnya GPS akan mengirimkan data ke serial monitor selama 30 detik untuk memverifikasi komunikasi, dan MQTT client dikonfigurasikan ke server Thingsboard, dan LED serta Buzzer diatur sebagai output yang akan memberikan indikator. Kode ini akan melakukan looping pada pembacaan data GPS dari module NEO-6M dan men-decode menggunakan library TinyGPS++, dan jika data GPS valid, maka fungsi displayInfo akan dipanggil, dimana fungsi ini akan melakukan pemrosesan dan menampilkan data GPS, dimana *latitude* dan *longitude* akan di ekstraksikan dan ditampilkan pada serial monitor. Data dari GPS berupa *latitude* dan *longitude* ini dikonversikan menjadi string dan dipublish ke platform ThingsBoard menggunakan MQTT client. Terakhir, terdapat fungsi reconnect yang akan digunakan untuk melakukan percobaan

untuk melakukan koneksi ulang ke ThingsBoard, jika MQTT client tidak terhubung, dan koneksi ini akan dilakukan dengan delay 5 detik pada tiap percobaan koneksi.

Untuk memberikan gambaran visual yang jelas tentang alur dan interaksi antar bagian-bagian utama dalam program, kami membuat *flowchart* sebagai berikut:



Gambar 3 (Flowchart Program)

2.3 HARDWARE AND SOFTWARE INTEGRATION

Proyek ini mengintegrasikan sejumlah komponen *hardware* yang meliputi GPS NEO-6M, SIM800L, ESP32, buzzer, LED, dan baterai, dengan *software* yang diimplementasikan dalam kode bersamaan dengan penggunaan ThingsBoard sebagai platform pemantauan. GPS NEO-8M berfungsi sebagai sensor utama yang memberikan data lokasi yang akurat, sementara SIM800L memungkinkan koneksi seluler melalui kartu SIM, memperluas jangkauan proyek ke lokasi tanpa Wi-Fi. ESP32 berperan sebagai otak sistem yang mengelola pembacaan GPS, manajemen koneksi seluler, serta mentransmisikan data ke ThingsBoard yang dikirimkan melalui topik MQTT yang sesuai dengan topik-topiknya yaitu v1/devices/me/telemetry/latitude dan v1/devices/me/telemetry/longitude. Selain itu, buzzer dan LED berfungsi sebagai indikator visual dan suara untuk memberikan respons terhadap status koneksi dan keadaan proyek. Baterai memberikan daya agar proyek dapat beroperasi secara mobile di lokasi tanpa sumber daya listrik. Selain itu integrasi *hardware* dengan *software* yang memanfaatkan *library* ArduinoJson, TinyGPS++, dan PubSubClient untuk manajemen data JSON, *parsing* data GPS, dan komunikasi dengan server MQTT ThingsBoard. Selain itu pemanfaatan pin yang terdefinisi dengan jelas di dalam kode memberikan struktur yang kokoh untuk integrasi hardware. Selain proyek ini juga memanfaatkan Serial Monitor untuk memantau status koneksi, data GPS, dan pesan *debugging* secara langsung.

Dengan integrasi yang baik antara hardware dan software, proyek ini menciptakan sistem yang andal dan responsif untuk pemantauan lokasi real-time. Kombinasi penggunaan GPS, SIM800L, dan ThingsBoard memberikan solusi yang holistik untuk memenuhi dinamika kebutuhan pemantauan dan pengiriman data IoT di berbagai lingkungan.

CHAPTER 3

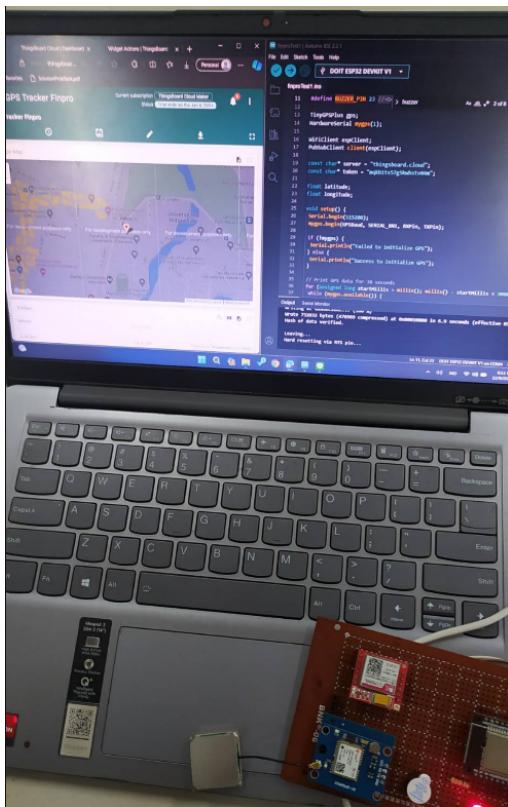
TESTING AND EVALUATION

3.1 TESTING

Setelah berhasilnya integrasi antara perangkat keras dan perangkat lunak, langkah selanjutnya akan melibatkan proses pengujian secara menyeluruh. Pengujian ini dirancang untuk mengevaluasi sejauh mana fungsionalitas dan kinerja sistem telah terimplementasi dengan baik. Akan dilakukan serangkaian tes, termasuk uji stabilitas koneksi, akurasi pemantauan lokasi, responsivitas terhadap perubahan kondisi, dan kemampuan pemulihan setelah gangguan sinyal.

Selama langkah pengujian, kami menguji fungsionalitas perangkat dengan meletakkan rangkaian pada dua tempat yang berbeda, yaitu

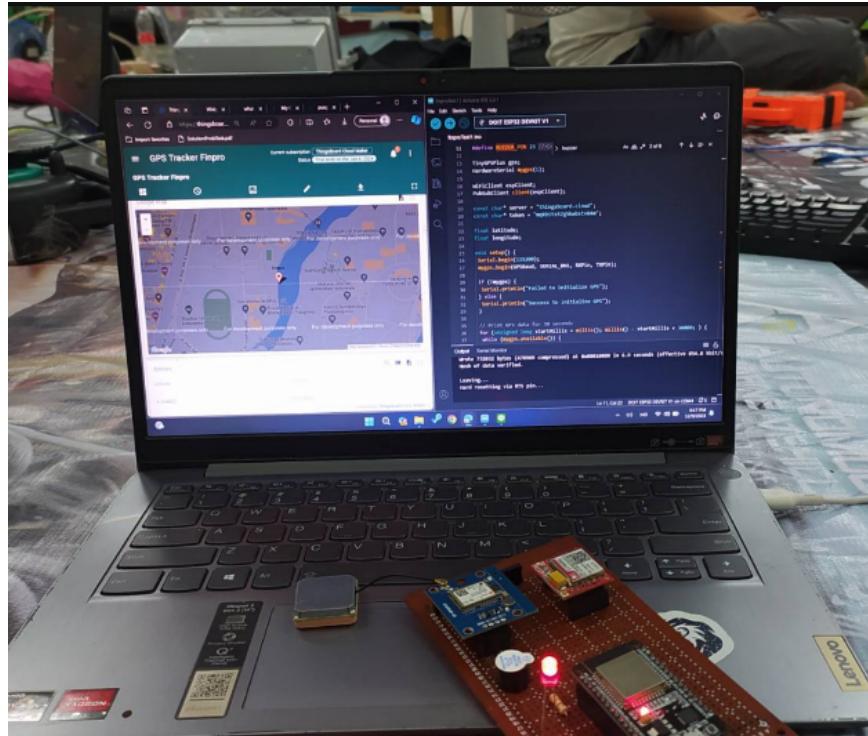
- Lokasi Departemen Teknik Elektro.



Gambar 4 (Pengujian TrackGuard di DTE)

Pada pengujian ini, setelah perangkat terhubung kedalam jaringan internet, maka LED menyala dan buzzer menyala sebagai indikator. Lalu terlihat ketika membuka platform thingsboard, maka akan tampil gambaran maps, yang ditandai dengan simbol lokasi yang berada di sekitaran FT. Hal ini menyatakan bahwa proses rangkaian telah tersusun dan terintegrasi dengan benar.

- Lokasi Pusgiwa

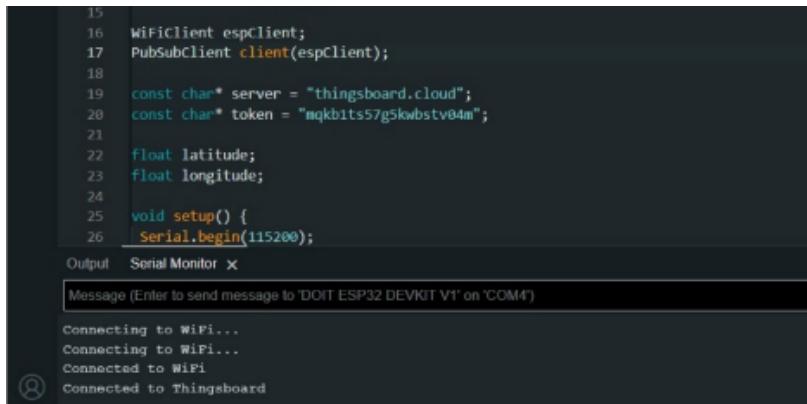


Gambar 5 (Pengujian TrackGuard di Pusgiwa)

Pada pengujian ini, setelah perangkat terhubung kedalam jaringan internet, maka LED menyala dan buzzer menyala sebagai indikator. Lalu terlihat ketika membuka platform thingsboard, maka akan tampil gambaran maps, yang ditandai dengan simbol lokasi yang berada di sekitaran pusgiwa dekat FT. Hal ini menyatakan bahwa proses rangkaian telah tersusun dan terintegrasi dengan benar.

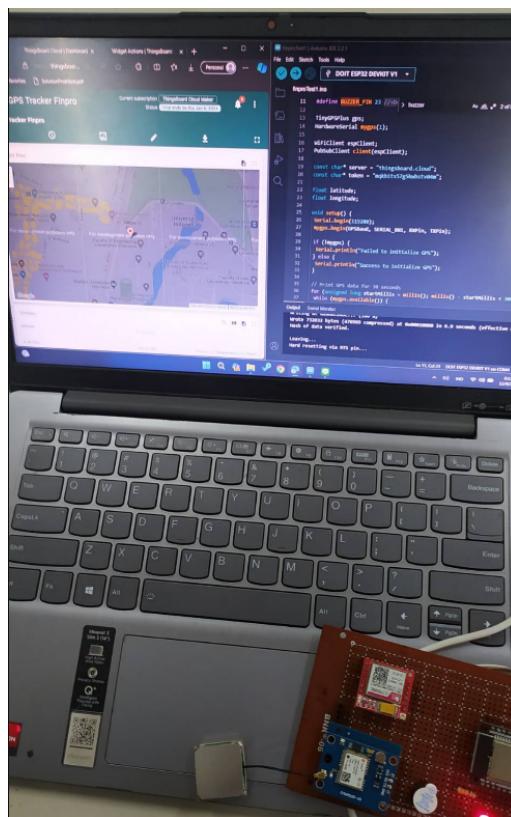
3.2 RESULT

- Lokasi : Departemen Teknik Elektro

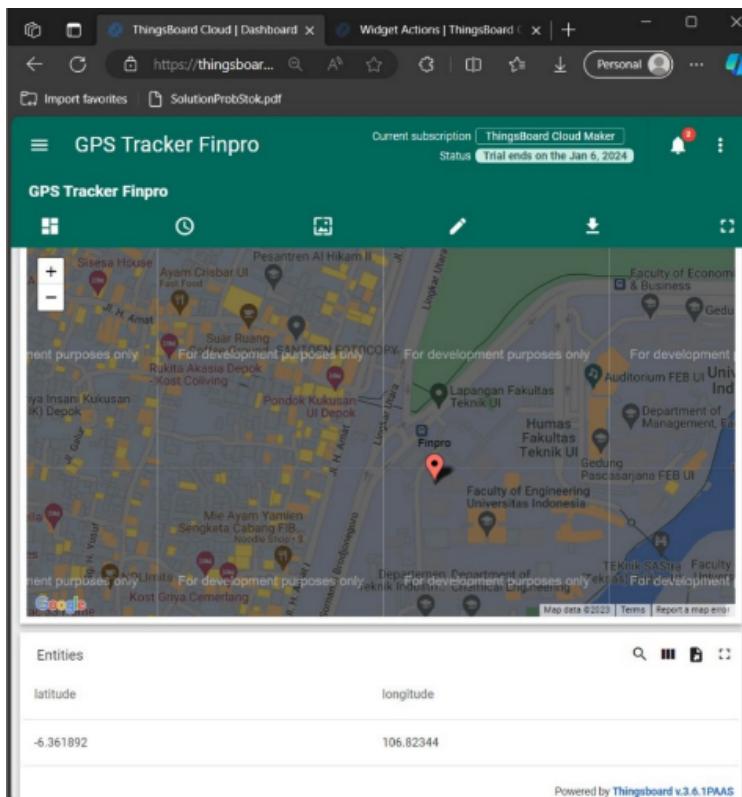


```
15
16 WiFiClient espClient;
17 PubSubClient client(espClient);
18
19 const char* server = "thingsboard.cloud";
20 const char* token = "mqkb1ts57g5kwbstv04m";
21
22 float latitude;
23 float longitude;
24
25 void setup() {
26   Serial.begin(115200);
Output  Serial Monitor x
Message (Enter to send message to 'DOIT ESP32 DEVKIT V1' on 'COM4')
Connecting to WiFi...
Connecting to WiFi...
Connected to WiFi
Connected to Thingsboard
```

Gambar 6 (Tampilan Serial Monitor)



Gambar 4 (Pengujian TrackGuard di DTE)



Gambar 7 (Tampilan ThingsBoard)

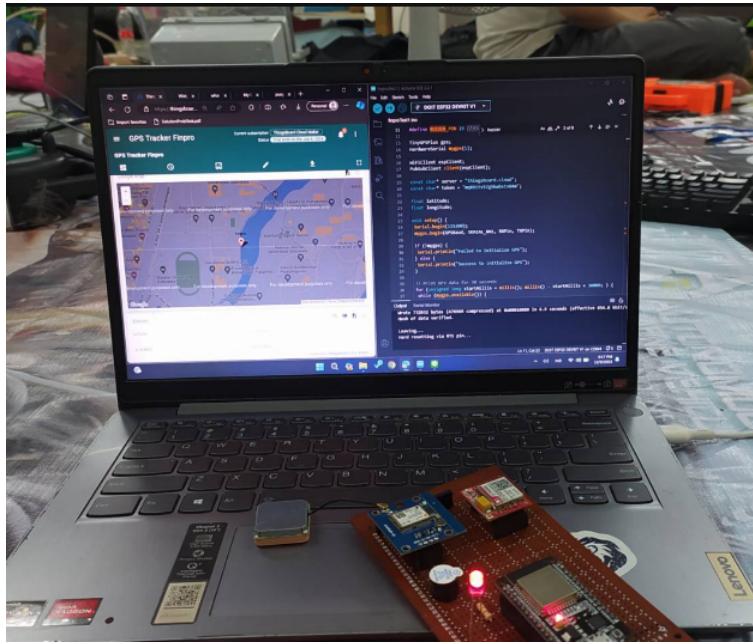
- Lokasi : Pusgiwa (Rubot)

```

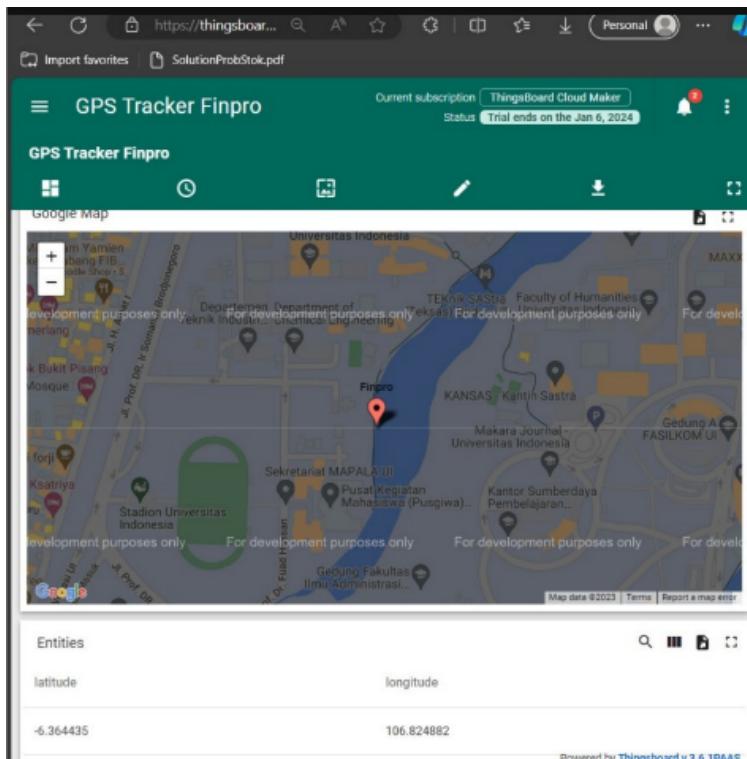
15
16 WiFiclient espClient;
17 PubSubClient client(espClient);
18
19 const char* server = "thingsboard.cloud";
20 const char* token = "mqkb1ts57g5kwbstv04m";
21
22 float latitude;
23 float longitude;
24
25 void setup() {
26   Serial.begin(115200);
Output  Serial Monitor ×
Message (Enter to send message to 'DOIT ESP32 DEVKIT V1' on 'COM4')
Connecting to WiFi...
Connecting to WiFi...
Connected to WiFi
Connected to Thingsboard

```

Gambar 6 (Tampilan Serial Monitor)



Gambar 5 (Pengujian TrackGuard di Pusgiwa)



Gambar 8 (Tampilan ThingsBoard)

Berdasarkan hasil pengujian, setelah menyalakan perangkat, terlihat bahwa LED dan Buzzer menyala, menunjukkan bahwa perangkat telah berhasil terhubung dengan jaringan internet. Selanjutnya, melalui Serial Monitor, dapat diamati bahwa proses koneksi dengan ThingsBoard melalui MQTT client berjalan dengan sukses. Selanjutnya pengujian divalidasi

dengan membuka platform ThingsBoard, terlihat bahwa ThingsBoard telah mendapatkan koordinat yang didapatkan oleh komponen GPS NEO-6M dan yang telah diproses sedemikian rupa oleh ESP32 kedalam koordinat yang mudah dibaca, lalu dapat terlihat pula ketika perangkat berpindah, maka koordinat berubah juga secara *real-time*, dan lokasi yang ditampilkan pada peta turut berpindah mengikuti keberadaan perangkat “TrackGuard”.

Pengamatan ini menegaskan bahwa TrackGuard mampu memberikan pemantauan lokasi yang akurat dan responsif dalam situasi perpindahan. Dengan demikian, dapat diambil kesimpulan bahwa berdasarkan hasil pengujian, “TrackGuard” telah berhasil memenuhi fungsi dan tujuan utamanya secara efektif.

3.3 EVALUATION

Meskipun ukuran fisik “TrackGuard” tergolong cukup besar untuk standar perangkat GPS, hal ini diimbangi dengan kemampuan dan fungsionalitas yang luar biasa dalam memberikan pemantauan lokasi yang akurat dan responsif. Mesin pemrosesan ESP32 yang digunakan memastikan kinerja yang optimal, sementara modul GPS NEO-6M memberikan ketepatan koordinat yang diperlukan. Walaupun perangkat ini mungkin tidak ideal untuk kebutuhan yang memerlukan perangkat yang sangat kecil, namun dalam konteks pemantauan dan pelacakan, kehandalan dan kemampuan TrackGuard menjadi nilai tambah yang signifikan. Hal ini dapat dilihat berdasarkan hasil pengujian yang dilakukan pada lokasi Departemen Teknik Elektro dan Pusgiwa, “TrackGuard” telah menunjukkan performa yang sangat baik. Setelah perangkat berhasil terhubung ke jaringan internet, indikator LED dan Buzzer menyala sebagai tanda bahwa integrasi perangkat keras dan lunak telah berlangsung dengan sukses. Selanjutnya, melalui pemantauan Serial Monitor, dapat diamati bahwa koneksi dengan ThingsBoard melalui MQTT client berjalan lancar.

CHAPTER 4

CONCLUSION

Dalam proyek "TrackGuard," kami berhasil menghadirkan solusi inovatif untuk mengatasi beberapa tantangan dalam sistem pemantauan lokasi. Mengintegrasikan ESP32, SIM800L, dan GPS NEO-6M, proyek ini memberikan solusi terkini yang memastikan pemantauan lokasi yang akurat dan responsif. Melalui uji coba di Departemen Teknik Elektro dan Pusgiwa, TrackGuard terbukti mampu memberikan performa yang sangat baik.

Meskipun ukuran fisik TrackGuard tergolong cukup besar untuk perangkat GPS standar, kehandalan dan kemampuannya dalam memberikan pemantauan lokasi menjadi nilai tambah yang signifikan. Mesin pemrosesan ESP32 dan modul GPS NEO-6M menjadikan proyek ini efisien dan handal dalam menghasilkan data koordinat yang akurat. Dengan demikian, walaupun mungkin kurang ideal untuk kebutuhan perangkat yang sangat kecil, TrackGuard tetap memberikan solusi efektif dalam konteks pemantauan dan pelacakan.

Pengujian di Departemen Teknik Elektro dan Pusgiwa berhasil memvalidasi koneksi yang stabil, akurasi pemantauan lokasi, responsivitas terhadap perubahan kondisi, dan kemampuan pemulihan setelah gangguan sinyal. Hasil evaluasi menegaskan bahwa TrackGuard berhasil memenuhi fungsi dan tujuan utamanya secara efektif. Integrasi perangkat keras dan lunak berlangsung dengan sukses, dan antarmuka ThingsBoard memberikan pemantauan lokasi real-time yang dapat diandalkan.

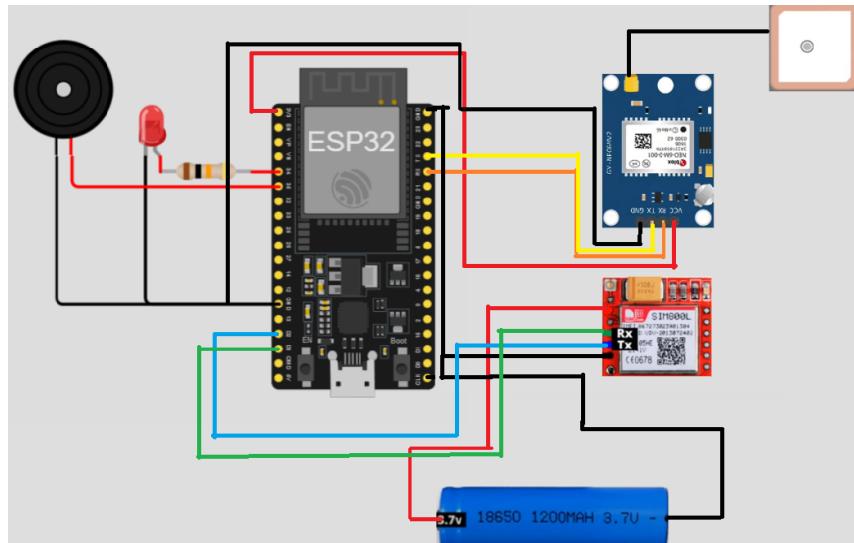
Dengan demikian, proyek TrackGuard diharapkan dapat memberikan solusi yang holistik dan adaptif terhadap dinamika kebutuhan pemantauan dan pengamanan dalam sistem pemantauan lokasi.

REFERENCES

- [1] “ESP32 SIM800L GPS Tracker | Vehicle Tracking System using ESP32 | ,” *Ahmad Logs.* Sept, 13 2020. <https://www.youtube.com/watch?v=Lrbu9oA3Do&t=100s> [Accessed: Dec 10, 2023]
- [2] “GSM/GPRS based GPS Tracker using Blynk with Calling & SMS features” *techiesms.* Sept, 13 2020. <https://www.youtube.com/watch?v=Lrbu9oA3Do&t=100s> [Accessed: Dec 10, 2023]
- [3] “GPS asset tracking with ThingsBoard (PART 2),” *mikroTik.* Sept, 13 2020. <https://www.youtube.com/watch?v=48gC4T2lTjs> [Accessed: Dec 10, 2023]
- [4] Das, Debasish, “Interfacing SIM800L GSM/GPRS Module with ESP32,” *circuitdigest,* <https://circuitdigest.com/microcontroller-projects/interfacing-sim800l-module-with-es-p32> [Accessed: Dec 10, 2023]

APPENDICES

Appendix A: Project Schematic



Appendix B: Documentation

