

Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi
 Jl. Ahmad Yani, K.M. 33,5 - Kampus STMIK Banjarbaru
 Loktabat – Banjarbaru (Tlp. 0511 4782881), e-mail: puslit.stmikbjb@gmail.com
 e-ISSN: [2685-0893](#)
 p-ISSN: 2089-3787

Perancangan *Prototype* Pendeteksi Lokasi Pencurian Baterai Tower Berbasis *Internet of Things*

Kevin Kusuma^{1*}, Evasaria Magdalena Sipayung²

Program Studi Informatika, Universitas Bunda Mulia, Tangerang, Indonesia

*e-mail *Corresponding Author*: evasaria.sipayung@gmail.com

Abstract

Tower battery theft is a problem that can cause large losses in Indonesia. Therefore, this research aims to overcome the problem of tower battery theft based on the Internet of Things (IoT). The type of method used in this research is designing a prototype to detect tower battery theft locations. The designed prototype consists of NodeMCU Esp8266, PIR sensor and GPS Neo 6Mv2 with the Telegram application as a communication module. The PIR sensor will detect movement that occurs around the area where this sensor is placed and provide notifications in the form of a Google Maps link via the Telegram application to the user's cellphone if theft or movement occurs. In this research trial, it was carried out to ensure the performance and accuracy of the system in detecting the location of tower battery theft. From the test results, the average distance between the default coordinates from Google Maps and the prototype coordinates is 0.9963 meters from 30 different locations. The average response time of this prototype is 7.73 seconds from 30 different locations and 12.67 seconds based on prototype simulations.

Keyword: *Internet of things; Sensor; Prototype; Sensor PIR; Telegram*

Abstrak

Pencurian baterai tower merupakan masalah yang dapat menyebabkan kerugian yang besar di Indonesia. Oleh karena itu, penelitian ini bertujuan untuk mengatasi masalah pencurian baterai tower berbasis *Internet of Things* (IoT). Jenis metode yang dilakukan pada penelitian ini berupa perancangan prototype pendeteksi lokasi pencurian baterai tower. *Prototype* yang dirancang terdiri dari NodeMCU Esp8266, sensor PIR, dan GPS Neo 6Mv2 dengan aplikasi Telegram sebagai modul komunikasi. Sensor PIR akan mendeteksi gerakan yang terjadi pada sekitar area sensor ini ditempatkan dan memberikan notifikasi berupa link *Google Maps* melalui aplikasi Telegram ke ponsel pengguna jika terjadi pencurian dan pergerakan. Dalam uji coba penelitian ini, dilakukan untuk memastikan kinerja dan akurasi sistem dalam mendeteksi lokasi pencurian baterai tower. Dari hasil pengujian, jarak rata-rata antara koordinat default dari *Google Maps* dan koordinat prototype sejauh 0,9963 meter dari 30 lokasi yang berbeda. Waktu respon rata-rata dari *prototype* ini selama 7,73 detik dari 30 lokasi yang berbeda dan 12,67 detik berdasarkan simulasi *prototype*.

Kata kunci: *Internet of things; Sensor; Prototype; Sensor PIR; Telegram*

1. Pendahuluan

Pencurian baterai pada *tower* BTS (*Base Transceiver Station*) menjadi salah satu masalah yang terjadi di Indonesia. Hal ini disebabkan oleh meningkatnya permintaan dan akses internet yang saat ini membuat perusahaan telekomunikasi harus memperluas jaringannya dengan memasang tower BTS di berbagai tempat. Karena lokasi tower BTS umumnya berada di area yang terpencil dan minim pengawasan, maka hal ini menyebabkan terjadinya pencurian baterai tower. Pencurian ini tidak hanya menyebabkan kerugian finansial, namun juga mempengaruhi kualitas layanan telekomunikasi yang diberikan.

Maraknya pencurian baterai *tower* ini menyulitkan pihak keamanan untuk menemukan lokasi baterai yang dicuri. Selain itu, memerlukan waktu yang cukup lama untuk mengecek satu persatu lokasi baterai. Untuk mengatasi masalah ini, perlu dirancang sebuah *prototype* pendeteksi lokasi baterai *tower* berbasis IoT (*Internet of Things*) dengan Telegram. IoT merupakan suatu infrastruktur berupa koneksi jaringan global, yang mengkoneksikan benda secara fisik dan virtual melalui eksploitasi data capture dan teknologi komunikasi sehingga

membuatnya lebih mudah untuk diakses dan dikendalikan dari jarak jauh. [1]. Telegram merupakan aplikasi *cloud end-to-end, self destruction Messages*, infrastruktur *multi-data center* [2].

Prototype ini dirancang akan menggunakan sensor GPS untuk mendeteksi lokasi baterai tower ini berada. Ketika baterai ini dicuri, maka sensor GPS akan mengirimkan data lokasi ke aplikasi yang Telegram. Sensor GPS ini akan bergerak secara *real-time* yang dimana ketika baterai bergerak, lokasi agar tetap terdeteksi. Haversine Formula juga digunakan untuk menghitung akurasi lokasi *prototype*. Dengan adanya *prototype* ini diharapkan pula petugas keamanan seperti kepolisian dapat meringankan pencarian baterai tower yang dicuri dan juga dapat menangkap pelaku pencurian baterai *tower* ini.

2. Tinjauan Pustaka

Penelitian terdahulu dengan judul “Perancangan dan Implementasi Security dan Sistem Kendali Otomatis Smart Home Menggunakan *NodeMCU*,” mikrokontroler yang digunakan terdiri dari ESP32-CAM dan *NodeMCU* mampu mengendalikan sensor PIR, *relay*, *solenoid doorlock*, laser sensor, Idr sensor dan magnetic sensor dengan baik dan berjalan sesuai dengan fungsi yang ditujukan yaitu membangun sistem keamanan. Implementasi dalam memasang sensor sebagai salah satu cara untuk melakukan pencegahan terjadinya pencurian rumah bisa menjadi pertimbangan dalam melindungi rumah [3].

Penelitian terdahulu dengan judul “Sistem Deteksi Kebakaran Berbasis *Internet of Things* (IoT) dengan Perangkat Arduino”, sistem ini menggunakan mikrokontroler Arduino Uno dan *NodeMCU* sehingga data dari ketiga sensor tersebut dapat dikirimkan melalui jaringan internet dan tampil pada sebuah website serta mampu melakukan notifikasi melalui panggilan telepon. Dengan berbasis internet of things data yang dikirimkan akan lebih cepat sehingga informasi kebakaran dapat diketahui dengan cepat dan musibah kebakaran dapat segera diatasi [4].

Penelitian terdahulu dengan judul “*Internet of Things* (IoT) Kontrol Lampu Rumah Menggunakan *Nodemcu* dan ESP-12E berbasis Telegram Chatbot”, alat ini dirancang dengan menggunakan *NodeMCU* untuk mengatur kerja sistem, ESP-12E sebagai alat penghubung internet ke smartphone, Relay sebagai alat penghantar arus listrik yang dihubungkan ke lampu, dan aplikasi Telegram chatbot yang sudah di-instal pada smartphone [5].

Penelitian terdahulu dengan judul “Alat Pendeteksi Lokasi Kebakaran Berbasis *Internet of Things* (IoT),” alat pendeteksi lokasi kebakaran berhasil mengirimkan sinyal input untuk mengirimkan titik lokasi kebakaran secara realtime dan tepat pada website DAMKAR.GO yang dikelola oleh petugas operator pemadam kebakaran [6].

Penelitian terdahulu dengan judul “Rancang Bangun Sistem Monitoring dan Pendeteksi Lokasi Kebocoran Pipa Berdasarkan Analisis Debit Air Berbasis IoT”, *Water Flow Sensor* digunakan untuk mengetahui debit air yang mengalir berdasarkan area pipa yang berada di antara 2 buah sensor, dengan analisis debit air memungkinkan untuk mengetahui area kebocoran, tingkat kebocoran, serta jumlah area kebocoran jika terjadi penurunan debit air yang diolah dengan mikrokontroler *NodeMCU* berbasis *Internet of things* (IoT). Akurasi dari 7 buah *Water Flow Sensor* setelah dilakukan kalibrasi sebesar 97,53% [7].

Penelitian terdahulu dengan judul “*Classification of heart disease trigger factors using Naive Bayes method to predict the risk of heart disease using IoT-based heart rate sensor* prototipe berbasis IoT yang dirancang untuk mengambil nilai input denyut jantung dan kadar oksigen darah (Spo2) pengguna dengan menggunakan metode Naive Bayes dapat digunakan untuk prediksi risiko penyakit jantung seseorang memiliki akurasi sebesar 96% dari 100 data uji [8].

Penelitian terdahulu dengan judul “Perancangan Aplikasi Cek Radius Outlet PT. IJS Berbasis Web Menggunakan Metode Haversine Formula “, Metode Haversine Formula dapat membuat aplikasi melakukan pengecekan radius calon lokasi mitra PT. IJS dengan outlet mitra yang sudah ada dengan nilai rata-rata persentase akurasi sebesar 94% [9].

Penelitian ini menggunakan perangkat IoT terdiri dari *NodeMCU Esp8266*, sensor PIR, dan GPS Neo 6Mv2 dengan aplikasi Telegram sebagai modul komunikasi. Sensor PIR akan mendeteksi gerakan yang terjadi pada sekitar area sensor ini ditempatkan dan memberikan notifikasi berupa link *Google Maps* melalui aplikasi Telegram ke ponsel pengguna jika terjadi pergerakan sebagai indikasi pencurian.

3. Metodologi

3.1 Analisis Kebutuhan

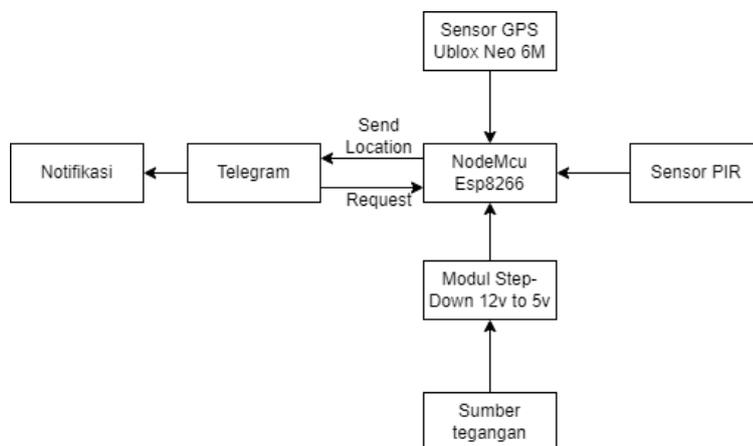
Analisis kebutuhan fungsionalitas prototype pendeteksi lokasi pencurian baterai tower berbasis Internet of Things (IoT) yang dikembangkan terdiri dari:

- 1) Prototype mendapatkan feedback saat chat request “/loc”
- 2) Prototype mengirim informasi saat chat request “/start”.
- 3) Sensor PIR mengirim lokasi saat ada pergerakan
- 4) Prototype dapat mengirimkan lokasi berbentuk link Google Maps ke Telegram.
- 5) Prototype mendapatkan koordinat Longitude.
- 6) Prototype mendapatkan koordinat Latitude
- 7) Prototype dapat terhubung degan API Telegram
- 8) P prototype terhubung dengan jaringan internet.
- 9) Prototype dapat menerima sinyal satelit untuk mendeksi lokasi
- 10) Prototype mendapat tegangan untuk supply komponen Arduino.

3.2 Rancangan Sistem

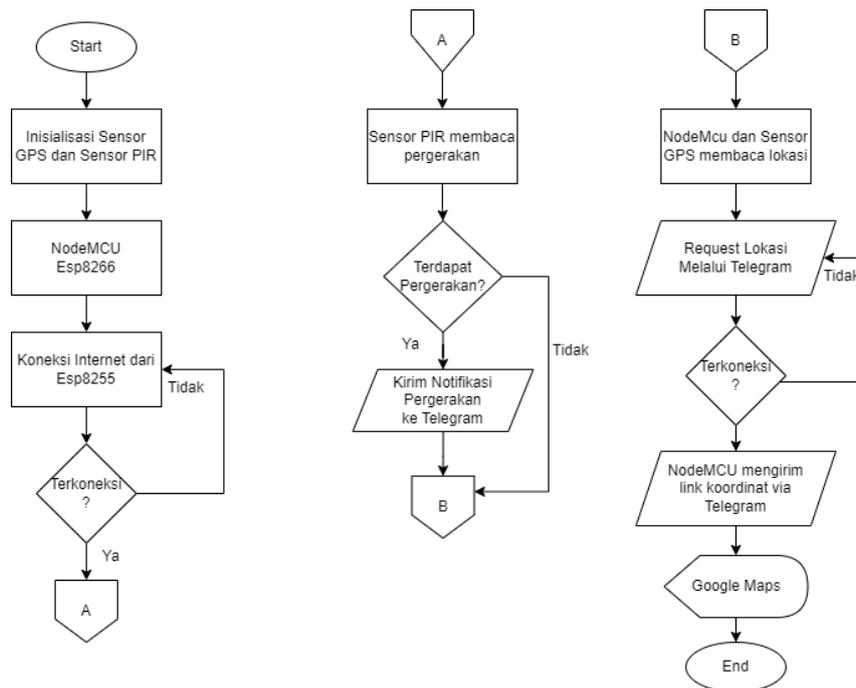
Rancangan desain *interface* dalam bentuk Blok Diagram ditunjukkan pada Gambar 1 dan alur kerja program digambarkan dengan *flowchart* ditunjukkan pada Gambar 2.

Sensor GPS Ublox Neo 6M akan mendeteksi lokasi yang ada pada prototype, NodeMcu Esp8266 merupakan microcontroller yang berfungsi memproses input dan output system. Modul Step-Down 12v to 5v merupakan modul penurun tegangan dari sumber tegangan yang awalnya 12v menjadi 5v. Sensor PIR yang digunakan dalam prototype ini menjadi sensor pergerakan. Ketika pergerakan terdeteksi, maka otomatis Sensor PIR akan mengirim tanda pergerakan ke Telegram tanpa di Request. Telegram yang digunakan dalam prototype ini menjadi alat IoT yang berguna untuk mengirim request lokasi dan menerima link koordinat lokasi.



Gambar 1. Blok Diagram Prototype

Pada alur kerja program diawali dengan Inisialisasi sensor GPS Ublox Neo 6M dan Input inisialisasi ke NodeMcu Esp8266. Sensor PIR membaca pergerakan. Jika pergerakan terbaca, maka prototype akan mengirimkan notifikasi secara otomatis ke Telegram. Jika tidak terdapat pergerakan, maka NodeMcu akan memproses Sensor GPS. Saat internet sudah terhubung, maka NodeMcu akan membaca koordinat lokasi yang ada pada Sensor GPS Ublox Neo 6M. Request lokasi melalui pesan Telegram. Jika Telegram tidak terkoneksi dengan *prototype*, maka Request dapat dilakukan lagi hingga betul. Jika Telegram sudah terkoneksi, maka *prototype* akan mengirimkan link koordinat via Telegram yang sebelumnya sudah di Request. Link dari koordinat ini akan masuk ke dalam *Google Maps*.



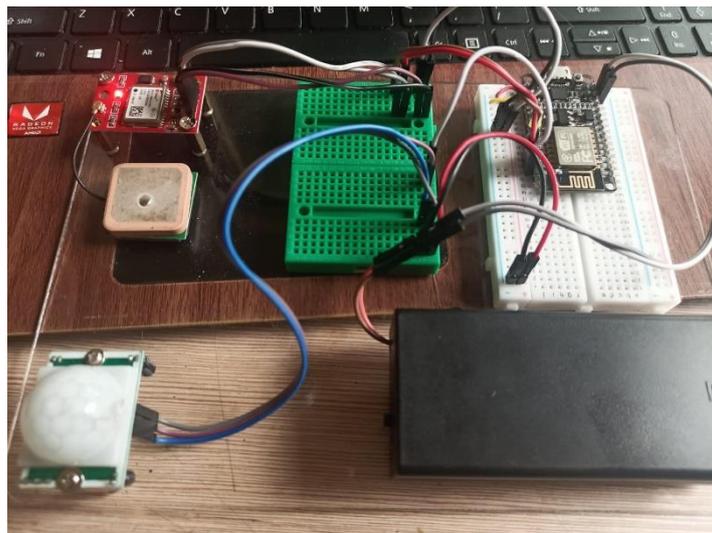
Gambar 2. Flowchart Kerja Sistem

4. Hasil dan Pembahasan

4.1 Ujicoba Sistem

Hasil perancangan perangkat keras *prototype* pendeteksi lokasi pencurian baterai tower berbasis Arduino Uno dengan menggunakan aplikasi Telegram [10][11]

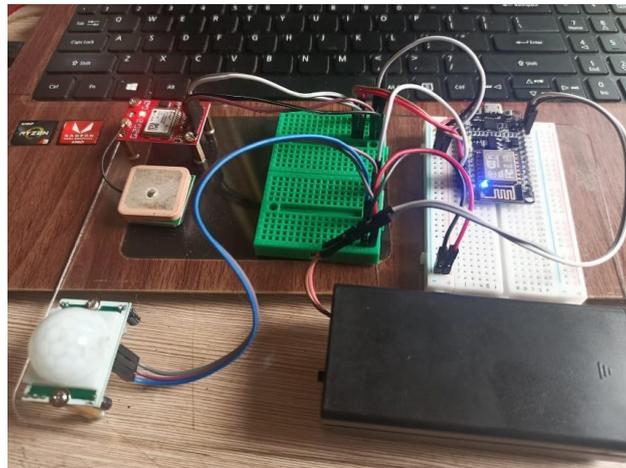
Prototype dirancang di atas kaca akrilik plastic ditunjukkan pada Gambar 3 di mana terdapat mikrokontroler NodeMCU Esp8266 sebagai input/output penghubung semua komponen. Modul GPS beserta Antena EEPROM yang berfungsi menerima sinyal satelit dan mengetahui posisi sebuah baterai *tower*. Sensor PIR untuk mendeteksi pergerakan di sekitar sensor PIR ditempatkan.



Gambar 3. Prototype Dalam Keadaan ON

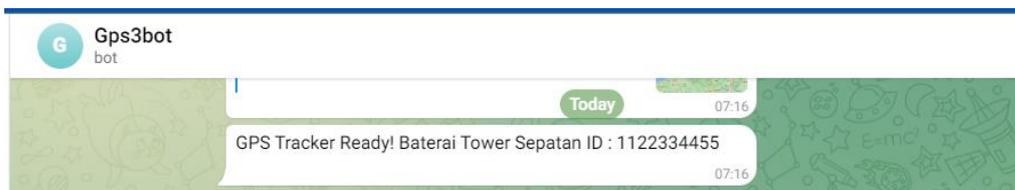
Terdapat sebuah LED yang menyala ditunjukkan pada Gambar 4 berfungsi sebagai indikator sensor PIR, baterai sebagai tegangan masuk dan sumber daya cadangan, serta *breadboard* sebagai penghubung antar komponen yang menggunakan kabel *jumper*. Semua

komponen dirakit di atas kaca akrilik plastik dan diberi dudukan baut agar dapat didesain dengan rapi.



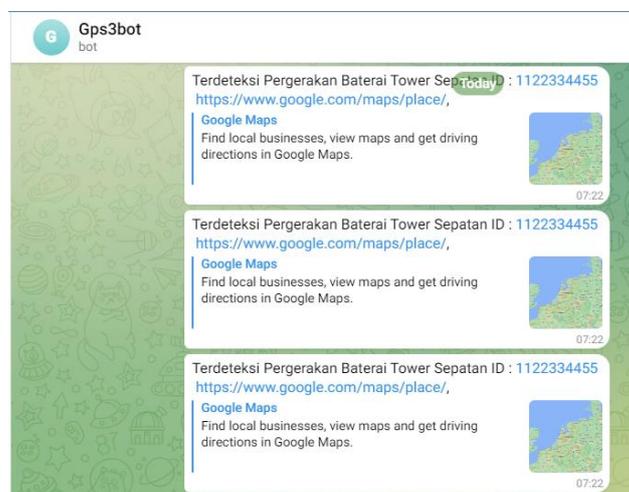
Gambar 4. Prototype Dalam Keadaan ON

Setelah *prototype* diaktifkan maka akan mengirimkan pesan “GPS Tracker Ready! Baterai Tower Sepatan ID : 1122334455” menunjukkan bahwa *prototype* berhasil diaktifkan secara menyeluruh. ID pada pesan tersebut merupakan suatu identitas pengenal baterai. Pesan tersebut tidak akan muncul jika *prototype* mengalami kesalahan dalam program. Hasil tampilan dari aplikasi telegram dan respon dari *prototype* saat pertama kali diaktifkan ditunjukkan pada Gambar 5.



Gambar 5. Tampilan Telegram Saat Prototype Pertama Kali Diaktifkan

Secara kerja sistem, ketika sensor PIR mendeteksi pergerakan di sekitar sensor, maka sistem akan mengirimkan lokasi ke aplikasi Telegram [12]. Pengujian sensor PIR atau sensor gerak mendeteksi terdapat 3x pergerakan dalam 1 waktu. Hasil respon dari sensor PIR yang terdiri dari 3 pergerakan ditunjukkan pada Gambar 6.



Gambar 6 Tampilan Respon Sensor PIR Sebanyak 3x Pergerakan

Sensor PIR mendeteksi 3x pergerakan dalam 1 waktu dan mengirimkan ke aplikasi Telegram. Respon dari Serial Monitor ditunjukkan pada Gambar 7. Serial Monitor memunculkan hasil respon sensor PIR dengan memberi nilai 1 ('true' pada sistem). Sensor PIR ketika tidak mendeteksi pergerakan, maka hasil respon sensor PIR dengan nilai 0 ('false' pada sistem).

```

PIR Sensor = 1
GPS = Invalid Location!

PIR Sensor = 0
GPS = Invalid Location!

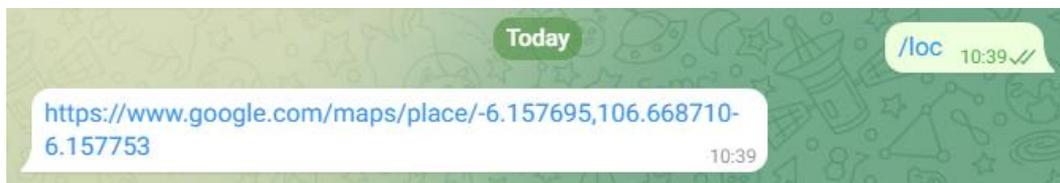
PIR Sensor = 1
GPS = Invalid Location!

PIR Sensor = 1
GPS = Invalid Location!

```

Gambar 7 Tampilan Serial Monitor Sensor PIR

Pengujian *chat request* dari pengguna aplikasi Telegram dengan cara pengguna mengetik "/loc" pada aplikasi Telegram [13]. Hasil dari *chat request* pengguna aplikasi Telegram dan sistem mengirimkan lokasi berupa link akses ke *Google Maps* di mana *Google Maps* menampilkan hasil lokasi *prototype* berada ditunjukkan pada Gambar 8.



Gambar 8 Tampilan Chat Request /loc pada Telegram

Hasil respon dari Serial Monitor ketika pengguna Telegram mengirim chat request "/loc", maka Serial Monitor akan menampilkan respon masuk ke system ditunjukkan pada Gambar 9. Sistem akan mendeteksi chat request dari Telegram dengan memberikan nilai 1 ('true' pada sistem) pada 'Handle New Messages' yang mengartikan sistem menerima chat request dari pengguna Telegram. GPS menampilkan status "Invalid Location!" jika lokasi *prototype* tidak terdeteksi. Hal ini mengartikan bahwa modul GPS tidak menerima sinyal GPS. Jika GPS mendapatkan sinyal, maka GPS akan menampilkan *Longitude* dan *Latitude* di Serial Monitor.

```

Output Serial Monitor x
Message (Enter to send message to 'NodeMCU 1.0 (ESP-12E Module)' on 'COM3')
Handle New Messages: 1
/loc
-6.157753#106.668750
-6.157745#106.668740
-6.157764#106.668750
PIR Sensor = 0
GPS = https://www.google.com/maps/place/-6.157695,106.668710-6.157753

```

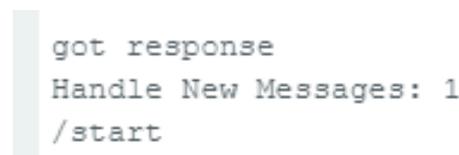
Gambar 9 Tampilan Serial Monitor /loc Pada Serial Monitor

Hasil chat request '/start' pada aplikasi Telegram dengan mengirimkan keterangan yang ada pada sistem *prototype* ditunjukkan pada Gambar 10.



Gambar 10 Tampilan Chat Request /start pada Telegram

Respon pada Serial Monitor ketika pengguna Telegram melakukan *chat request* /start, maka “Handle New Messages” akan bernilai 1 (‘true’ pada sistem). Hal ini yang mengartikan bahwa sistem menerima *chat request* dari pengguna Telegram ditunjukkan pada Gambar 11.



Gambar 11 Tampilan Serial Monitor /start

Pengujian akurasi pada *prototype* ini dilakukan pada 30 tempat yang berbeda. Tabel 1 merupakan lokasi yang ditentukan berdasarkan *Longitude* dan *Latitude* dari *Google Maps* dan *prototype*, serta jarak antara lokasi *default* dengan *prototype* yang didapatkan berdasarkan perhitungan rumus *Haversine*.

Tabel 1. Jarak Lokasi

No	Lokasi	Google Maps		Prototype		Jarak (meter)
		Longitude	Latitude	Longitude	Latitude	
1	Arcadia Sport Club	106.6682444	-6.1569949	106.6682468	-6.1569987	0,50
2	Lotus Tomang	106.5804138	-6.1571802	106.5804127	-6.1571811	0,16
3	Rumah Pribadi	106.5832299	-6.1362438	106.5832285	-6.1362421	0,24
4	Kampus ITI	106.6731423	-6.3437639	106.6731443	-6.3437664	0,35
5	Tarsisius Vireta	106.5724954	-6.1701914	106.5724973	-6.1701998	0,96
6	Danau Regency	106,5839904	-6,1623682	106,5839919	-6,1623598	0,95
7	Bundaran Tomang	106,5740431	-6,1572025	106,5740416	-6,1572102	0,87
8	Amsterdam	106,5812982	-6,1589915	106,5812920	-6,1589901	0,70
9	Sunrise Sunset	106,5841708	-6,1631485	106,5841636	-6,1631458	0,84
10	Saung Lembur	106,5897632	-6,1613724	106,5897698	-6,1613796	1,08
11	Indomart Sakura	106,5570061	-6,1665905	106,5570187	-6,1665856	1,50
12	Wisma Garden	106,5508468	-6,1659571	106,5508516	-6,1659525	0,73
13	Saung Gading	106,5586488	-6,1602313	106,5586476	-6,1602401	0,98
14	Servis Printer	106,5535798	-6,1583838	106,5535891	-6,1583858	1,05
15	Dimsum In's	106,5533778	-6,1572908	106,5533697	-6,1572913	0,90
16	Kebun Latte Ciater	106,6958679	-6,3339808	106,6958709	-6,3339663	1,64
17	Upnormal BSD	106,6772585	-6,3175176	106,6772515	-6,3175186	0,78
18	Stasiun Rawabuntu	106,6745333	-6,3154808	106,6745236	-6,3154869	1,27
19	Mixue Puspitek	106,6888433	-6,3471758	106,6888392	-6,3471912	1,77
20	Indomart Victor	106,6871912	-6,3460155	106,6871941	-6,3460122	0,47
21	Pasar Lama	106,6296821	-6,1787559	106,6296723	-6,1787525	1,14
22	Stasiun Tangerang	106,6302017	-6,1767922	106,6302125	-6,1767936	1,20
23	Roti Bakar 88	106,6259937	-6,1814643	106,6259789	-6,1814623	1,65
24	Pinggir Cisadane	106,6300013	-6,1812292	106,6300101	-6,1812282	0,97
25	Museum Benteng	106,6296823	-6,1787559	106,6296758	-6,1787563	0,71
26	TM Yamaha	106,4953396	-6,1064151	106,4953473	-6,1064136	0,86

No	Lokasi	Google Maps		Prototype		Jarak (meter)
		Longitude	Latitude	Longitude	Latitude	
27	Pasar Sepatan	106,5571427	-6,1066351	106,5571356	-6,1066369	0,81
28	Pasar Mauk	106,5226387	-6,0800926	106,5226486	-6,0800936	1,10
29	Sepatan Residen	106,5593743	-6,0839332	106,5593825	-6,0839299	0,97
30	Taman Kota	106,5571427	-6,1066351	106,5571346	-6,1066355	0,89

Rata-rata jarak koordinat dihitung menggunakan *Haversine* sebesar 0,9363 meter. Lokasi mendapatkan jarak diatas 1 meter disebabkan lokasi tempat tersebut terdapat vegetasi / pohon yang cukup lebat dan tinggi, serta banyaknya jumlah bangunan lokasi tersebut [14] [15]

Rata-rata jarak yang didapatkan dari 30 lokasi yang berbeda didapatkan 0,9363 meter. Contoh perhitungan dari selisih *Radian Longitude* dan *Latitude* yang didapat.

Selisih Radian Latitude =

$$\frac{Rad\ Lat\ 2 - Rad\ Lat\ 1}{2}$$

$$\frac{-0,107459809 - (-0,107456322)}{2}$$

$$-0,000001743$$

Selisih Radian Longitude =

$$\frac{Rad\ Long\ 2 - Rad\ Long\ 1}{2}$$

$$\frac{1,861710541 - 1,8617105}{2}$$

$$0,00000002094$$

$$a = \left(\sqrt{\sin^2\left(\frac{\Delta Lat}{2}\right) + \cos(Rad\ Lat\ 1) * \cos(Rad\ Lat\ 2) * \sin^2\left(\frac{\Delta Long}{2}\right)} \right)$$

$$a = \left(\sqrt{\sin^2(-0,000001743) + \cos(-0,107456322) * \cos(-0,107459809) * \sin^2(0,00000002094)} \right)$$

$$a = 0,000000039157$$

$$d = 2 * r * \sin^{-1}(\sqrt{a})$$

$$d = 2 * 6371000 * \sin^{-1}(0,000000039157)$$

$$d = 0,4989\ m$$

$$Rad\ Lat\ 1 = -6,1569949 * \frac{3,14159}{180} = -0,107456322$$

$$Rad\ Lat\ 2 = -6,1569987 * \frac{3,14159}{180} = -0,107459809$$

$$Rad\ Long\ 1 = 106,6682444 * \frac{3,14159}{180} = 1,8617105$$

$$Rad\ Long\ 2 = 106,6682468 * \frac{3,14159}{180} = 1,861710541$$

Perhitungan pada di atas merupakan salah satu selisih radian *Longitude dan Latitude*. Perhitungan di atas diambil pada salah satu lokasi yaitu Arcadia Sport Club. Selisih ini dimasukan ke dalam rumus utama *Haversine*.

$$d = 2r \sin^{-1} \left(\sqrt{\sin^2 \left(\frac{\Phi_2 - \Phi_1}{2} \right) + \cos(\Phi_1) \cos(\Phi_2) \sin^2 \left(\frac{\lambda_2 - \lambda_1}{2} \right)} \right)$$

Keterangan :

d = jarak (dalam meter)

r = 6371 km

Φ_1 = Latitude 1 (Dalam radian)

Φ_2 = Latitude 2 (dalam Radian)

λ_1 = Longitude 1 (dalam Radian)

λ_2 = Longitude 2 (dalam Radian)

Pengujian waktu respon berupa waktu lama respon *prototype* saat pengguna Telegram mulai mengirimkan *chat* respon. Kecepatan respon tidak hanya dipengaruhi oleh perbedaan lokasi, tetapi juga jaringan internet yang digunakan. Dalam pengujian ini, menggunakan jaringan dari provider XL yaitu Live.On dengan koneksi *hotspot* [16]. Tabel 2 merupakan hasil dari waktu respon yang didapat dari 30 lokasi berbeda yang tercantum pada Tabel 1.

Tabel 2. Waktu Respon Lokasi

No	Lokasi	Prototype		Waktu (detik)
		Longitude	Latitude	
1	Arcadia Sport Club	106.6682468	-6.1569987	5,73
2	Lotus Tomang	106.5804127	-6.1571811	4,34
3	Rumah Pribadi	106.5832285	-6.1362421	7,45
4	Kampus ITI	106.6731443	-6.3437664	5,49
5	Tarsisius Vireta	106.5724973	-6.1701998	9,57
6	Danau Regency	106,5839919	-6,1623598	8,46
7	Pasar Lotus	106,5740416	-6,1572102	7,42
8	Amsterdam	106,5812920	-6,1589901	10,69
9	Sunrise Sunset	106,5841636	-6,1631458	6,33
10	Saung Lembur	106,5897698	-6,1613796	9,12
11	Indomart Sakura	106,5570187	-6,1665856	5,64
12	Wisma Garden	106,5508516	-6,1659525	6,13
13	Saung Gading	106,5586476	-6,1602401	3,88
14	Servis Printer	106,5535891	-6,1583858	7,59
15	Dimsum In's	106,5533697	-6,1572913	7,37
16	Kebun Latte Ciater	106,6958709	-6,3339663	12,74
17	Upnormal BSD	106,6772515	-6,3175186	6,58
18	Stasiun Rawabuntu	106,6745236	-6,3154869	9,79
19	Mixue Puspitek	106,6888392	-6,3471912	9,12
20	Indomart Victor	106,6871941	-6,3460122	7,25
21	Pasar Lama	106,6296723	-6,1787525	9,85
22	Stasiun Tangerang	106,6302125	-6,1767936	11,19
23	Roti Bakar 88	106,6259789	-6,1814623	8,68
24	Pinggir Cisadane	106,6300101	-6,1812282	10,71
25	Museum Benteng	106,6296758	-6,1787563	8,37
26	TM Yamaha	106,4953473	-6,1064136	4,15
27	Pasar Sepatan	106,5571356	-6,1066369	6,84
28	Pasar Mauk	106,5226486	-6,0800936	9,73
29	Sepatan Residen	106,5593825	-6,0839299	6,52
30	Taman Kota	106,5571346	-6,1066355	5,19

Rata-rata waktu respon yang didapatkan dari 30 lokasi yang berbeda didapatkan 7,73 detik. Simulasi *prototype* ini bertujuan untuk mengetahui kemampuan dari *prototype* dengan

cara membawa *prototype* secara *real-time* (diam di suatu tempat) dan melakukan monitoring dari rumah. Rata-rata waktu respon adalah 12.67 detik ditunjukkan pada Tabel 3.

Tabel 1. Rata-rata Waktu Respon

No	Lokasi	Prototype		Waktu (detik)
		Longitude	Latitude	
1	Unknown Location	106.5517739	-6.2245624	13,37
2	Unknown Location	106.5824408	-6.1577742	16,87
3	Unknown Location	106.6602031	-6.2238357	10,75
4	Unknown Location	106.5817978	-6.1282771	12,24
5	Unknown Location	106.574698	-6.1200648	10,14
Rata – Rata Waktu				12,67

4.2 Pengujian Sistem

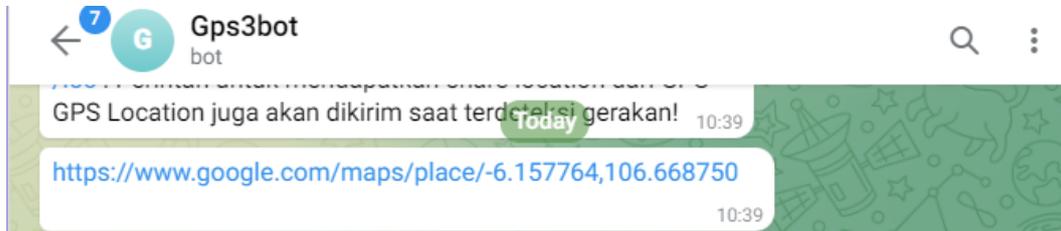
Hasil pengujian fungsionalitas sistem yang dikembangkan dengan *blackbox testing* ditunjukkan pada Tabel 4.

Tabel 1. Hasil Pengujian Fungsionalitas Sistem

No	Tahap Menjalankan Prototype	Hal Yang Diharapkan	Hasil	Keterangan
1	Pengujian <i>prototype</i> mendapatkan <i>feedback</i> saat <i>chat request</i> "/loc"	<i>Prototype</i> memberikan <i>feedback</i> saat <i>chat request</i> "/loc"	Sesuai	<i>Prototype</i> mengirimkan <i>feedback</i> saat <i>chat request</i> "/loc"
2	Pengujian <i>prototype</i> mengirim informasi <i>prototype</i> saat <i>chat request</i> "/start".	<i>Prototype</i> dapat mengirimkan informasi <i>prototype</i> saat pengguna Telegram melakukan <i>chat request</i> .	Sesuai	<i>Prototype</i> dapat mengirimkan informasi <i>prototype</i> saat <i>chat request</i> "/start"
3	Pengujian sensor PIR.	Sensor PIR mengirim lokasi saat ada pergerakan.	Sesuai	Sensor PIR dapat mengirimkan lokasi saat ada pergerakan.
4	Pengujian pengiriman lokasi berbentuk link <i>Google Maps</i> ke Telegram.	<i>Prototype</i> dapat mengirimkan lokasi berbentuk link <i>Google Maps</i> .	Sesuai	<i>Prototype</i> dapat mengirimkan <i>link Google Maps</i> .
5	Pengujian akurasi koordinat <i>Longitude</i> .	Mendapatkan koordinat <i>Longitude</i>	Sesuai	<i>Prototype</i> mendapatkan koordinat <i>Longitude</i> .
6	Pengujian akurasi koordinat <i>Latitude</i>	Mendapatkan koordinat <i>Latitude</i>	Sesuai	<i>Prototype</i> mendapatkan koordinat <i>Latitude</i> .
7	Pengujian API Telegram dengan <i>prototype</i> .	API Telegram terhubung dengan <i>prototype</i> .	Sesuai	Sistem <i>prototype</i> berhasil terhubung dengan API Telegram.
8	Pengujian <i>prototype</i> dengan koneksi jaringan.	Koneksi jaringan internet terhubung dengan sumber internet.	Sesuai	Sistem <i>prototype</i> berhasil terhubung ke sumber internet.
9	Pengujian sinyal satelit yang diterima oleh <i>prototype</i> .	<i>Prototype</i> mendapatkan sinyal satelit untuk mendeteksi lokasi.	Sesuai	<i>Prototype</i> mendapatkan sinyal satelit.
10	Pengujian tegangan untuk <i>supply</i> komponen Arduino.	Tegangan sesuai dengan spesifikasi dari komponen.	Sesuai	Tegangan output dari NodeMCU sesuai dengan komponen penghubung.

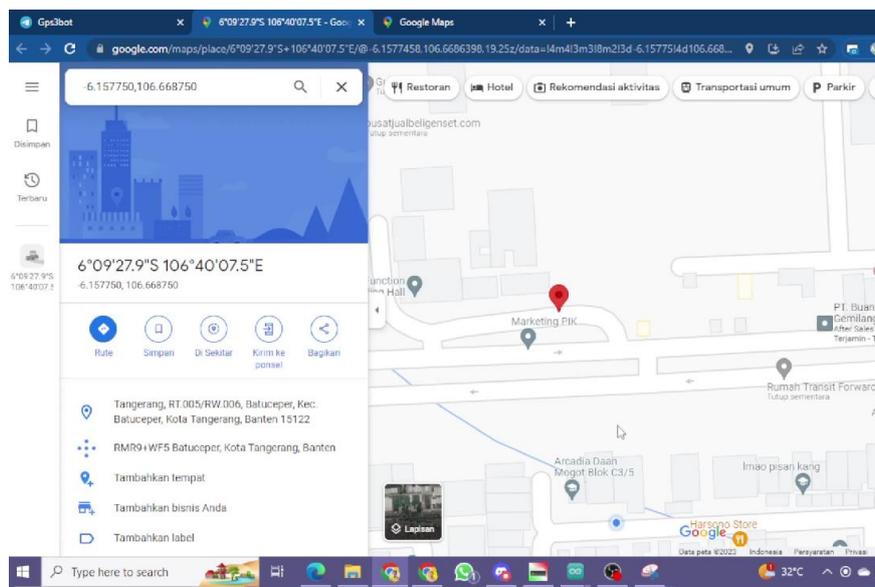
4.3 Pembahasan

Pengguna harus mengirim *chat request* melalui Telegram pada *prototype* untuk mendapatkan lokasi. Dan pada saat *prototype* mendapatkan dan menerima *chat request* melalui aplikasi Telegram, maka secara otomatis *prototype* akan mengirim *link Google Maps* sebagai akses *link* untuk masuk ke menu *Google Maps* [17]. Salah satu hasil pengujian *prototype* di rumah pribadi ditunjukkan pada Gambar 12.



Gambar 12 Tampilan Telegram Saat Respon Sensor PIR

Untuk mendapatkan lokasi, pengguna harus mengirim *chat request* melalui Telegram pada *prototype*. Dan pada saat *prototype* mendapatkan dan menerima *chat request* melalui Telegram, maka secara otomatis *prototype* akan mengirim *link Google Maps* sebagai akses *link* untuk masuk ke menu *Google Maps*. Salah satu hasil pengujian *prototype* di rumah pribadi ditunjukkan pada Gambar 13.



Gambar 13 Tampilan dari *Link Google Maps* Telegram

Pengujian akurasi merupakan proses pengujian yang menentukan seberapa akurat suatu sistem atau perangkat dalam menentukan lokasi *prototype*. Proses pengujian akurasi ini dilakukan dengan membandingkan lokasi yang telah ditentukan oleh sistem yang sebenarnya dengan lokasi pada *prototype*. Hasil pengujian ini akan berupa rata-rata selisih dari koordinat *Longitude* dan *Latitude* [18]

5. Simpulan

Penggunaan sensor PIR dan Modul GPS Neo 6Mv2 berhasil digunakan dan diimplementasikan sebagai respon sensor untuk mendeteksi pergerakan area sekitar sensor PIR. NodeMCU Esp8266 dan Modul GPS Neo 6Mv2 berhasil dikoneksikan dengan Telegram dan berhasil digunakan sebagai chat request pada *prototype*. Jarak rata-rata antara koordinat *default* dari *Google Maps* dan koordinat *prototype* sejauh 0,9363 meter dari 30 lokasi yang berbeda. Waktu respon rata-rata dari *prototype* ini selama 7,73 detik dari 30 lokasi yang berbeda.

Daftar Referensi

- [1] D. Setiadi and M. N. Abdul Muhaemin, "Penerapan Internet Of Things (Iot) Pada Sistem Monitoring Irigasi (SMART IRIGASI)," *Infotronik J. Teknol. Inf. dan Elektron.*, vol. 3, no. 2, p. 95, 2018, doi: 10.32897/infotronik.2018.3.2.108.
- [2] H. Soeroso, A. Zuhri Arfianto, N. Eka Mayangsari, and M. Taali, "Penggunaan Bot Telegram Sebagai Announcement System pada Intansi Pendidikan," *Semin. Master PPNS*, vol. 2, no. 1, pp. 45–48, 2017.
- [3] Kevin Wijaya Setiady, Jusia Amanda Ginting, "Perancangan dan Implementasi Security dan Sistem Kendali Otomatis Smart Home Menggunakan NodeMCU", *Jurnal Algoritma, Logika dan Komputasi*, Vol. VI (No.1), halaman 543 - 552, 2023.
- [4] Achmad Fariid Amali, "Sistem Deteksi Kebakaran Berbasis Internet of Things (IoT) dengan Perangkat Arduino", Tugas Akhir, Program Studi Informatika, Program Sarjana Fakultas Teknologi Industri, Universitas Islam Indonesia, 2020
- [5] M. Nega, E. Susanti, and A. Hamzah, "Internet of Things (IoT) Kontrol Lampu RUMah Menggunakan Nodemcu dan ESP-12E berbasis Telegram Chatbot," *J. Scr.*, vol. 7, no. 1, pp. 88–99, 2019.
- [6] Raka Abi Affan Arifin, "Alat Pendeteksi Lokasi Kebakaran Berbasis Internet of Things (IoT)", Skripsi, Program Studi Pendidikan Teknik Elektronika, Fakultas Teknik, Universitas Negeri Jakarta, 2024.
- [7] Ardi Dwi Prasetya, Haryanto, dan Kunto Aji Wibisono, "Rancang Bangun Sistem Monitoring dan Pendeteksi Lokasi Kebocoran Pipa Berdasarkan Analisis Debit Air Berbasis IoT", *Elektrika*, Vol. 12, No.1, halaman 39-47, 2020.
- [8] Jahja, J., Agung, H., Mulyana, T.M.S., Hakim, L., Sipayung, E., "Classification of heart disease trigger factors using Naive Bayes method to predict the risk of heart disease using IoT-based heart rate sensors", *AIP Conference Proceedings*, Volume 2693, Issue 1, 6 November 2023.
- [9] Rivalentino Arron, Angelina Pramana Thenata, "Perancangan Aplikasi Cek Radius Outlet PT. IJS Berbasis Web Menggunakan Metode Haversine Formula", *ZONAsi: Jurnal Sistem Informasi*, Vol. 6, No. 2, Mei 2024.
- [10] R. P. Prasetya and N. Vendyansyah, "Implementasi Sistem Tracking Pengendara Mobil Berbasis Iot Sebagai Keamanan Cerdas Pada Perlintasan Kereta Api," *J. Mnemon.*, vol. 5, no. 2, pp. 93–97, 2022, doi: 10.36040/mnemonic.v5i2.4668.
- [11] C. M. G. Butar-butar and Y. T. Samuel, "Perancangan Sistem Kendali Kendaraan Bermotor Jarak Jauh Menggunakan NodeMCU ESP8266," *TelKa*, vol. 9, no. 01, pp. 87–97, 2019, doi: 10.36342/teika.v9i01.793.
- [12] M. R. Hidayat, C. Christiono, and B. S. Sapudin, "Perancangan Sistem Keamanan Rumah Berbasis IoT Dengan NodeMCU ESP8266 Menggunakan Sensor PIR HC-SR501 Dan Sensor Smoke Detector," *Kilat*, vol. 7, no. 2, pp. 139–148, 2018, doi: 10.33322/kilat.v7i2.357.
- [13] A. D. Pangestu, F. Ardianto, and B. Alfaresi, "Sistem Monitoring Beban Listrik Berbasis Arduino Nodemcu Esp8266," *J. Ampere*, vol. 4, no. 1, p. 187, 2019, doi: 10.31851/ampere.v4i1.2745.
- [14] A. A. Muin, M. Syafei, and A. Qashlim, "Implementasi Formula Haversine Pada Sistem Informasi Guru Mengaji Private," *J. Ilm. Ilmu Komput.*, vol. 6, no. 2, pp. 60–66, 2020.
- [15] A. Info, "Implementasi Global Positioning System ," vol. 2, no. 1, pp. 47–58, 2023.
- [16] A. Saputra and Y. Dewanto, "Pengaman Helm Dengan Gps Melalui Sms Berbasis Mikrokontroler Arduino Uno Dengan Smartphone," *J. Teknol. Ind.*, vol. 9, pp. 4–16, 2021, [Online]. Available: <https://journal.universitassuryadarma.ac.id/index.php/jti/article/view/656%0Ah> <https://journal.universitassuryadarma.ac.id/index.php/jti/article/viewFile/656/627>
- [17] Suzanzefi, Ilyas, Sholihin, F. E. Yunita, N. Putri, and I. Sulistiana, "Kendaraan Darat Tanpa Awak Dengan Menggunakan Radio Gps Neo 6 Mberbasis Internet of Thing," *Informanika*, vol. 6, no. 2, pp. 48–54, 2019.
- [18] W. Nugraha, D. Syauqy, and A. S. Budi, "Sistem Deteksi Perpindahan Kendaraan Bermotor Berdasarkan Data GPS dan Sensor IMU Menggunakan Naive Bayes," *J. Pengemb. Teknol. Inf. dan Ilmu Komput. e-ISSN*, vol. 2548, no. 1, p. 964X, 2021.