

Sistem Pemberitahuan Kedatangan/Keberangkatan Bus Berdasarkan Waktu Nyata Dan Pelacakan Bus

1st Muhamad Haris Dharmawan
Pratama

Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

mhdharmawan@student.telkomuniversity.ac.id

2nd Giva Andriana Mutiara

Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

givamz@telkomuniversity.ac.id

3rd Lisda Meisaroh

Fakultas Ilmu Terapan
Universitas Telkom
Bandung, Indonesia

lisdameisaroh@telkomuniversity.ac.id

Menurut temuan survei INRIX, kemacetan lalu lintas sering terjadi di kota-kota terbesar di Indonesia. Banyaknya masyarakat yang menggunakan kendaraan pribadi daripada angkutan umum menjadi salah satu faktor penyebab kemacetan. Bus merupakan salah satu transportasi umum yang paling banyak digunakan. Bus sering dijumpai di kota-kota besar di Indonesia sebagai transportasi umum untuk bepergian. Bus dapat digunakan untuk meminimalisir kemacetan karena dapat mengangkut banyak penumpang dalam satu waktu. Namun masih banyak masyarakat yang tidak mau menggunakan bus. Salah satu faktornya adalah seringnya keterlambatan kedatangan bus, yang menyebabkan ketidakpastian jadwal. Oleh karena itu, pada penelitian ini bertujuan untuk membuat sistem yang dapat melacak bus berbasis GPS yang ditempatkan di dalam bus, membuat sistem pemberitahuan jarak dan perkiraan waktu kedatangan bus yang ditempatkan di titik pemberhentian, dan mengukur keakuratan modul GPS yang digunakan. Sistem ini menggunakan perangkat keras yaitu Raspberry Pi sebagai mikroprosesor, modul Neo-M8M, dan L76x GPS Hat sebagai modul GPS yang digunakan yang sangat berpengaruh untuk menjalankan fitur *tracking*. Metode yang digunakan dalam penelitian ini menggunakan metode Haversine, Metode Haversine membutuhkan garis lintang, garis bujur dan jari jari bumi untuk mengukur jarak. Metode ini digunakan untuk mengukur jarak bus dan titik pemberhentian bus, metode Haversine digunakan juga untuk mengukur tingkat akurasi dari modul GPS yang digunakan.

Kata Kunci — transportasi, lalu lintas, GPS, *tracking*, macet

I. PENDAHULUAN

Teknologi berkembang secara pesat telah memberikan perubahan pada aspek transportasi modern di dunia, pada saat ini produsen otomotif berlomba menciptakan kendaraan yang canggih, salah satunya fitur yang disematkan adalah kendaraan dapat dihubungkan dengan smartphone dan memiliki akses internet. Sehingga produsen dan konsumen dapat memantau posisi kendaraan. Standar umum kendaraan saat ini adalah sistem GPS *tracking* (deteksi posisi kendaraan). Tidak hanya transportasi pribadi, transportasi umum juga mendapatkan modernisasi teknologi.

Berdasarkan hasil survei INRIX, kemacetan lalu lintas sering terjadi di kota - kota besar Indonesia. INRIX yang

berbasis di Inggris adalah sebuah perusahaan dan lembaga riset yang bergerak di bidang transportasi. Beberapa kota termacet di Indonesia diantaranya Pontianak, Medan, Padang, Yogyakarta, Malang, Jakarta, dan Bandung [1]. Salah satu faktor yang menyebabkan terjadinya kemacetan adalah banyaknya masyarakat yang membawa transportasi pribadi dibandingkan transportasi umum.

Bus merupakan salah satu transportasi umum yang banyak digunakan. Bus sering ditemukan di kota - kota besar di Indonesia sebagai angkutan umum untuk bepergian. Bus dapat digunakan untuk meminimalisir kemacetan karena bus dapat mengangkut banyak penumpang dalam satu kali jalan. tetapi masih banyak masyarakat yang tidak ingin menaiki bus, salah satu faktornya adalah sering terjadinya keterlambatan kedatangan bus yang menyebabkan ketidakpastian jadwal. Pengalihan menggunakan transportasi pribadi menjadi transportasi umum khususnya bus merupakan salah satu upaya yang perlu dilakukan untuk mengurangi kemacetan yang sering terjadi di kota-kota besar yang ada di Indonesia. Untuk melakukan hal tersebut, perlu adanya perubahan strategi oleh pemerintah untuk meningkatkan minat masyarakat agar transportasi umum menjadi transportasi utama. Salah satu masalah yang sering terjadi pada transportasi umum yaitu bus, sering terjadinya ketidakjelasan kedatangan bus yang menyebabkan masyarakat tidak tahu kapan akan kedatangan bus, perlu adanya pembuatan sistem yang dapat melakukan *tracking* bus berbasis GPS yang diletakkan di dalam bus, dan membuat sistem notifikasi bila bus sebentar lagi akan sampai.

Oleh karena itu dalam penelitian ini akan membahas tentang sistem *tracking* bus untuk memberi tahu posisi bus, sistem perkiraan bus akan datang pada titik pemberhentian yang akan di tampilkan di setiap titik pemberhentian dan mengukur tingkat akurasi modul GPS yang digunakan.

II. KAJIAN TEORI

Pada penelitian yang dilakukan oleh Yetti Yuniati, Melvi Ulvan, Mardiyah Azzahra yang berjudul "Implementasi Modul *Global Positioning System* pada Sistem *Tracking* Bus Rapid Transit (BRT) Lampung Menuju Smart Transportation". Pada penelitian ini sistem

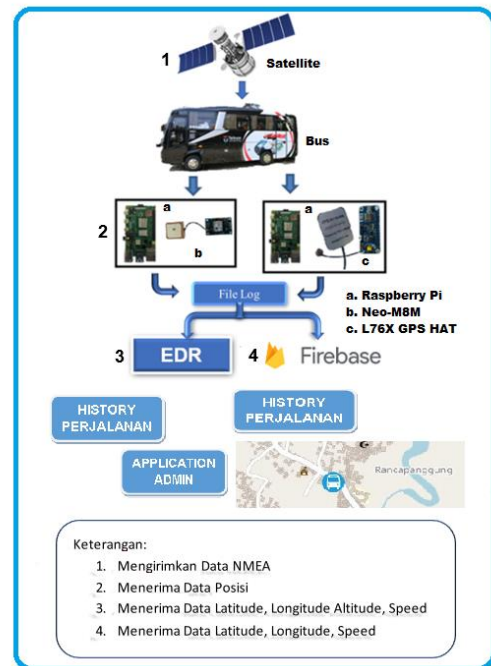
tracking pada bus menggunakan modul *Global Positioning System* (GPS) dengan mengirimkan data posisi bus melalui *Short Message servis* (SMS), yang akan disimpan dan diolah oleh data base, sehingga dapat diketahui posisi bus tersebut[2].

Pada penelitian yang dilakukan oleh Felia Citra Dwiyani Putri Rosyadi, Yudha Saintika, Yoso adi Setyoko yang berjudul “*Prototype Real-time Monitoring System Bus Trans Jateng Berbasis Android untuk Informasi Waktu Kedatangan Bus di Halte*”. Pada penelitian ini menggunakan modul SIM808 sebagai modul GSM dan modul GPS, yang digunakan untuk menerima data GPS[3].

Pada penelitian yang dilakukan oleh Imam Ajie Pradana yang berjudul “*Sistem Pemantauan bus Sekolah dengan Rfid Dan Web Aplikasi Berbasis Internet Of Things*”. Pada penelitian ini menggunakan sistem GPS yang terdiri dari Modul ESP8266 Nodemcu dan GPS Neo 6M yang akan memberikan koordinat posisi bus sekolah berupa link yang dapat dibuka menggunakan aplikasi google maps[4].

Pada penelitian yang dilakukan oleh Alfi Safil Ahsan yang berjudul “*Sistem Pemantau Posisi Bus Karyawan*”. Pada penelitian ini menggunakan modul SIM800L sebagai modul GSM, dan sensor GPS uBlox NEO-6M akan memberikan koordinat posisi bus berupa link yang dapat dibuka menggunakan aplikasi Google Maps[5].

Pada penelitian yang dilakukan oleh Rizal Wahyulianto yang berjudul “*Sistem Tracking Kendaraan Dengan Mikrokontroler Berbasis Web*”. Pada penelitian ini menggunakan modul SIM800L yang dihubungkan dengan GPS untuk mengirimkan koordinat lokasi yang dikirimkan melalui Short Message servis (SMS)[6].



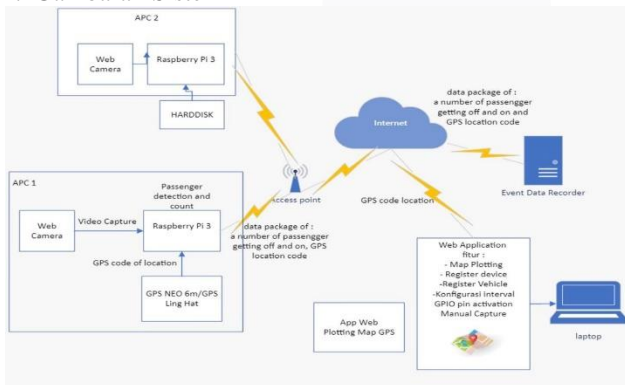
GAMBAR 2 Blok Diagram GPS

Cara kerja dari rancangan sistem yang di usulkan adalah, satellite akan mengirimkan sinyanya berupa data NMEA, modul GPS menangkap sinyal radio dengan data digital yang menghasilkan data NMEA posisi bus, modul GPS yang digunakan adalah modul GPS Neo-M8M dan L76X GPS HAT. Data koordinat akan dikonversi menjadi data posisi oleh suatu program Python berbasis microprocessor Raspberry Pi 4 yang akan menghasilkan data *latitude*, *longitude*, *altitude* dan *speed*, data tersebut akan dibuat menjadi dua file log, file log untuk EDR dan database. Data yang disimpan untuk EDR terdapat 2 data, data history perjalanan yang disimpan didalam Raspberry Pi, dan data koordinat yang dikirimkan untuk EDR. Data koordinat akan dikirimkan ke database dan disimpan, database yang digunakan adalah Firebase. Data yang disimpan di dalam Firebase terdapat 2 data, yaitu data history perjalanan modul GPS, dan data titik koordinat untuk menampilkan posisi modul GPS di website.

Untuk memahami perancangan sistem *tracking* bus yang akan dibuat, berikut *flowchart* dari sistem perancangan yang dibuat dapat dilihat pada GAMBAR .

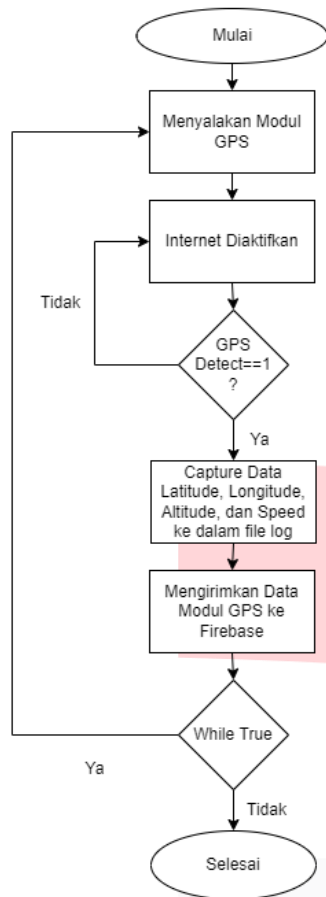
III. METODE

A. Gambaran Sistem



GAMBAR 1 Blok Diagram

Perancangan Sistem dari keseluruhan yang dibuat saat ini, terbagi menjadi empat bagian, dimulai dari APC 1 sebagai sebagai sistem penghitung penumpang di pintu masuk, APC 2 sebagai penghitung penumpang yang duduk di kursi penumpang, GPS sebagai sistem *tracking* bus dan pengiriman data ke server dan web application. Pada sistem *tracking* bus menggunakan mikroprosesor dan modul GPS akan menghasilkan data posisi bus yang nantinya akan dikirimkan ke server.



GAMBAR 3 Flowchart

B. Kebutuhan Perangkat Keras dan Lunak

Berikut adalah kebutuhan perangkat keras dan perangkat lunak yang digunakan pada sistem yang dibuat.

TABEL 1 Perangkat keras yang digunakan

No	Nama Perangkat	Fungsi
1	Raspberry Pi 4	Sebagai Mikrokontroler utama yang digunakan untuk membuat dan menyimpan program
2	Modul Neo-M8M	Sebagai modul GPS tracker untuk menerima titik koordinat
3	Modul L76X GPS HAT	Sebagai modul GPS tracker untuk menerima titik koordinat
4	Monitor Raspberry Pi	Sebagai display output dari Raspberry Pi untuk mengolah program

Tabel 2 Perangkat lunak yang digunakan

No	Nama Perangkat	Fungsi
1	Sistem Operasi Raspbian	Sistem Operasi untuk menjalankan Raspberry Pi
2	Thonny Python IDE	Aplikasi untuk membuat program berbahasa python dan menjalankan program
3	Firebase	Sebagai media pengiriman data titik koordinat

C. Perancangan Sistem

a. Sistem Tracking Modul GPS

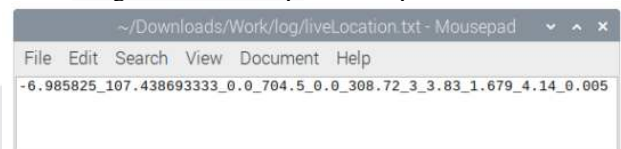
Sistem ini menggunakan mikroprosesor berupa Raspberry Pi dan tiga modul GPS yaitu, Modul GPS Ublox Neo-M8M, Modul GPS Teltonika FMB-130, dan

Waveshare L76X GPS HAT. Sistem ini berfungsi sebagai alat yang digunakan untuk melakukan tracking GPS yang diletakkan di dalam bus. Dengan cara menghubungkan antara Raspberry Pi dengan modul GPS melalui port serial *Universal Asynchronous Receiver Transmitter (UART)*, modul GPS akan mendapatkan sinyal untuk menerima data NMEA yang dikirimkan kepada Raspberry Pi, untuk dilakukan perubahan melalui program yang dibuat menggunakan Python. Program ini menghasilkan data posisi modul GPS berupa data seperti berikut.

- Latitude = Garis lintang
- Longitude = Garis bujur
- Altitude = Ketinggian dari permukaan laut (m)
- Speed = Kecepatan (m/s)
- Climb = *Climb or Sink rate in m/s of upwards or downwards movement*
- Track = *Course over ground in degrees from true north*
- Mode = *NMEA mode*
- EPS = Estimasi kesalahan dalam meter/detik
- EPX = Estimasi kesalahan longitude dalam meter
- EPV = Estimasi kesalahan vertical dalam meter
- EPT = Perkiraan kesalahan stempel waktu

b. Sistem Penyimpanan Data Modul GPS

Sistem ini berfungsi sebagai media penyimpanan data modul GPS yang dibuat berupa file log. Data yang telah disimpan pada file log akan dikirimkan kepada EDR dan database untuk di proses, database yang digunakan adalah database Firebase. File log dibuat menjadi beberapa bagian yaitu, file log untuk EDR, file log Firebase, dan file log History. File log untuk EDR dibuat untuk mengirimkan semua data posisi modul GPS untuk diproses oleh tim modul Pengiriman Data ke Server dan Web Application, isi dari file log untuk EDR dapat dilihat pada GAMBAR 4.



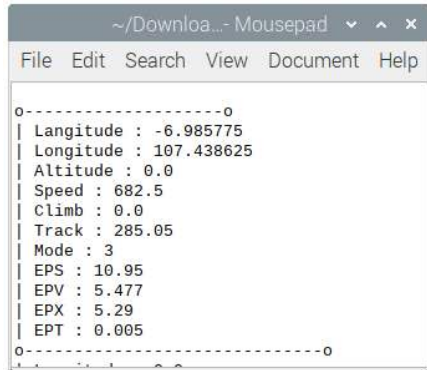
GAMBAR 4 Menentukan Titik Berdasarkan Rute Jalan

File log untuk Firebase dibuat menjadi dua bagian, yaitu file log history dan file log bus, file log history akan menyimpan semua data yang dikirimkan modul GPS dari awal sampai akhir, file log bus dibuat untuk menyimpan data posisi terbaru dari modul GPS yang dimana akan selalu memperbarui tanpa menyimpan data posisi yang lama, kedua file log ini akan dikirimkan kepada database Firebase, isi dari database Firebase dapat dilihat pada GAMBAR 5.



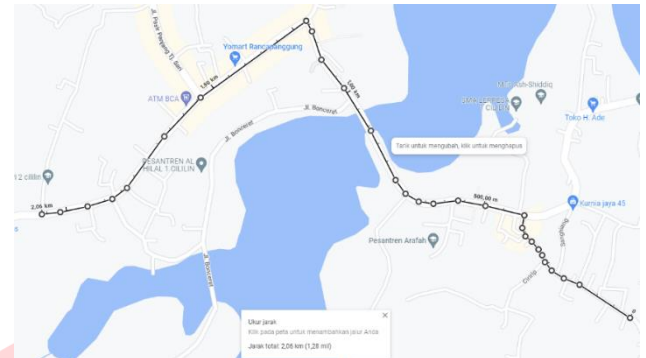
GAMBAR 5 Menentukan Titik Berdasarkan Rute Jalan

History merupakan hasil pengiriman data modul GPS dari file log history, user1 adalah data dari file log bus. Untuk pengiriman data posisi dari modul GPS ke Firebase dibutuhkan koneksi internet, jika koneksi internet terputus maka pengiriman data posisi modul GPS akan berhenti. File log history akan menyimpan semua data posisi modul GPS yang nantinya akan disimpan di dalam Raspberry Pi, isi dari file log history dapat dilihat pada GAMBAR 6.



GAMBAR 6 Menentukan Titik Berdasarkan Rute Jalan

Pada sistem ini metode Haversine digunakan untuk menghitung secara garis lurus. Garis lurus disesuaikan dengan rute perjalanan dan menghitung dari titik ke titik selanjutnya untuk menghitung jarak seperti pada GAMBAR 7.



GAMBAR 7 Menentukan Titik Berdasarkan Rute Jalan

c. Sistem Peringatan LED

Terdapat dua LED yang dipasang di dalam box alat modul GPS, yang digunakan sebagai peringatan untuk modul GPS, LED Hijau digunakan sebagai tanda modul GPS sudah mendapatkan data posisi, jika LED Hijau menyala, dan LED Hijau mati maka modul GPS sudah tidak mendapatkan data posisi.

LED Merah digunakan sebagai tanda cek internet, jika LED Merah menyala maka Raspberry Pi tidak terhubung dengan internet, jika LED Merah mati maka Raspberry Pi sudah terhubung Kembali dengan internet.

d. Sistem Penghitung jarak dan Waktu Kedatangan Bus

Perancangan sistem penghitungan jarak dan waktu kedatangan bus dilakukan dengan perhitungan menggunakan rumus Haversine. Haversine adalah metode yang digunakan untuk menghitung jarak antar dua titik dengan memperhitungkan bahwa bumi adalah sebuah bidang yang memiliki derajat kelengkungan, bukanlah sebuah bidang yang datar. Metode Haversine menghitung jarak antara 2 titik dengan berdasarkan Panjang garis lurus antara 2 titik pada garis lintang (*latitude*) dan garis bujur (*longitude*)[13]. Pada metode ini menggunakan juga ruas atau jari – jari bumi, yang memiliki Panjang 6371 km. Berikut bentuk rumus Haversine:

$$\Delta Lat = lat2 - lat1$$

$$\Delta Long = long2 - long1$$

$$a = \sin^2\left(\frac{\Delta lat}{2}\right) + \cos(lat1) \cdot \cos(lat2) \cdot \sin^2\left(\frac{\Delta long}{2}\right)$$

$$c = 2 \cdot \text{asin}(\sqrt{a})$$

$$d = R \cdot c$$

Keterangan :

- ΔLat = Besaran perubahan *latitude*
- $\Delta Long$ = Besaran perubahan *longitude*
- c = Kalkulasi perpotongan sumbu
- d = Jarak
- R = Ruas bumi

IV. HASIL DAN PEMBAHASAN

A. Pengujian Akurasi Modul GPS dengan Gmaps

Pengujian ini dilakukan dengan cara menyimpan modul GPS didalam motor. Dan menentukan sepuluh titik pemberhentian untuk diambil data titik koordinat. Lalu menghitung selisih dari modul GPS yang digunakan dengan titik koordinat yang dihasilkan oleh Google Maps dengan menggunakan metode Haversine.

TABEL 3 Penghitungan Selisih Modul GPS

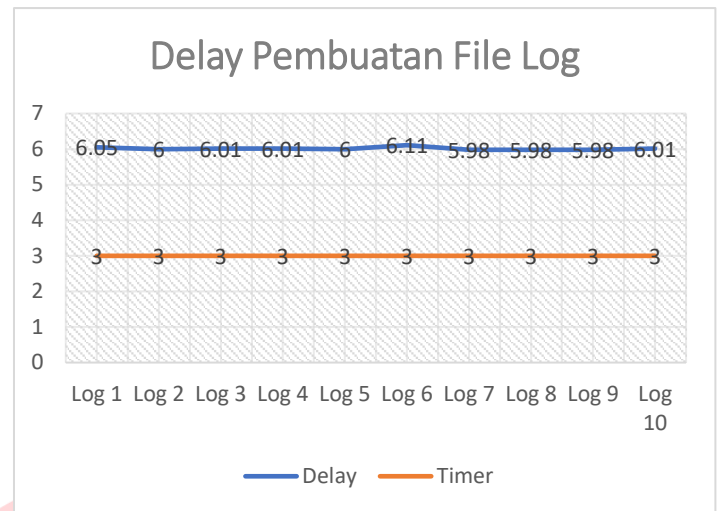
Titik	GPS Module	Latitude	Longitude	Selisih Jarak [M]
Titik 1	Gmaps	-6.989591	107.447696	
	Neo-M8M	-6.9895246	107.4476762	7.7
	FMB-130	-6.989733	107.447321	44.3
	L76X	-6.98959667	107.44751	20.54
Titik 2	Gmaps	-6.990439	107.4424144	
	Neo-M8M	-6.990415	107.442353	7.28
	FMB-130	-6.990425	107.44249	8.49
	L76X	-6.99054	107.442465	12.54
Titik 3	Gmaps	-6.987146	107.439563	
	Neo-M8M	-6.9871881	107.4395064	7.81
	FMB-130	-6.987501	107.439545	39.52
	L76X	-6.98719167	107.4396183	7.94
Titik 4	Gmaps	-6.985741	107.438628	
	Neo-M8M	-6.9856692	107.438637	8.05
	FMB-130	-6.985531	107.438621	23.36
	L76X	-6.98571833	107.4386217	2.62
	Gmaps	-6.983507	107.436302	

Titik 5	Neo-M8M	-6.9835885	107.4363281	9.51
	FMB-130	-6.983531	107.436515	23.66
	L76X	-6.98354333	107.4362517	6.87
Titik 6	Gmaps	-6.983016	107.437175	
	Neo-M8M	-6.9832228	107.4371393	23.33
	FMB-130	-6.983099	107.437226	10.81
	L76X	-6.98311	107.4371783	10.46
Titik 7	Gmaps	-6.980465	107.439592	
	Neo-M8M	-6.9804496	107.4395531	4.62
	FMB-130	-6.980463	107.43955	4.64
	L76X	-6.98042833	107.4395933	4.08
Titik 8	Gmaps	-6.980597	107.432516	
	Neo-M8M	-6.9806005	107.4323326	20.25
	FMB-130	-6.980635	107.432311	23.02
	L76X	-6.98056833	107.4323333	20.41
Titik 9	Gmaps	-6.982797	107.42716	
	Neo-M8M	-6.9827543	107.427094	8.7
	FMB-130	-6.982805	107.427156	0.99
	L76X	-6.98280833	107.4271417	2.38
Titik 10	Gmaps	-6.983512	107.42902	
	Neo-M8M	-6.9834054	107.4287056	36.67
	FMB-130	-6.982516	107.42905	110.8
	L76X	-6.983605	107.4290533	10.98

Setelah menghitung selisih antara modul GPS dan Google Maps, modul GPS Neo-M8M mendapat skor 4 dari 10, modul GPS FMB-130 mendapat skor 1 dari 10, dan modul GPS L76X GPS HAT mendapat skor tertinggi 5 dari 10

B. Pengujian Delay Pengiriman Data GPS Ke EDR

Pengujian *delay* pengiriman ke file log, telah di atur selama tiga detik untuk setiap pengiriman data modul GPS. Pengujian dilakukan dengan perulangan pengiriman data sebanyak sepuluh kali.

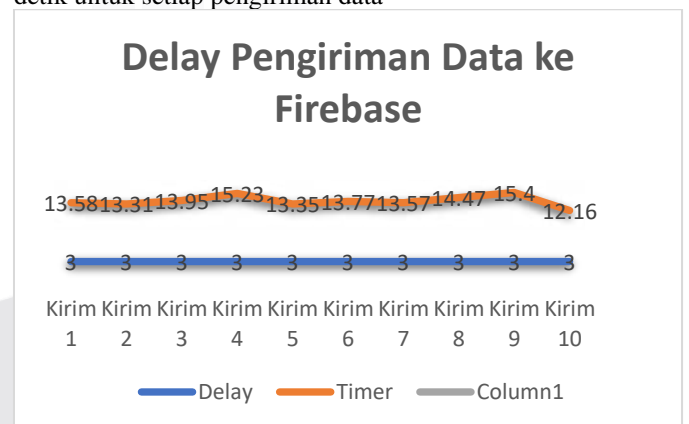


GAMBAR 8 Grafik Delay Pengiriman ke File Log

Berdasarkan grafik delay pengiriman ke file log pada GAMBAR 8, pengiriman Data Koordinat berhasil dikirimkan kedalam file log, dengan *delay* rata-rata pengiriman data ke file log 6,01 detik yang seharusnya sudah diatur menjadi 3 detik

C. Pengujian Delay Pengiriman Data GPS ke Firebase

Pengujian ini dijalankan menggunakan Raspberry Pi yang terhubung ke internet untuk mengirim data modul GPS ke Firebase. Jalankan modul GPS untuk memeriksa apakah modul GPS mengirimkan data koordinat ke Firebase dan menampilkannya dengan sukses di peta sesuai dengan data yang dikirim, data ke Firebase disetel ke 3 detik untuk setiap pengiriman data



GAMBAR 9 Grafik Delay Pengiriman ke Firebase

Berdasarkan grafik delay pengiriman ke Firebase pada GAMBAR 9, Pengiriman data modul GPS ke *Real-time* Database Firebase terkirim dengan baik akan tetapi pengiriman terpengaruhi oleh koneksi internet, dengan *delay* pengiriman rata rata 13.88 detik yang tidak sesuai dengan *delay* yang sudah diatur selama 3 detik untuk setiap pengiriman ke Firebase.

D. Pengujian Jarak dan Waktu Kedatangan Bus

Pada Pengujian Ini ditentukan 3 titik sebagai halte atau titik pemberhentian bus, dan melakukan pengujian pengukuran waktu dan jarak sebanyak 3 kali perulangan pada setiap halte atau titik pemberhentian, dengan melakukan pengukuran akurasi waktu dibandingkan

dengan *stopwatch*, dan pengukuran jarak dibandingkan dengan jarak yang dihasilkan oleh Google Maps.

TABEL 4 Pengujian Perhitungan Jarak dan Waktu

Halte	No.	Program Jarak [M]	Google Maps [M]	Akurasi Jarak
Halte 1	1.	288.85	400	72%
	2.	380.5	400	95%
	3.	309.8	400	77%
Halte 2	1.	1023.9	1200	85%
	2.	1117	1200	93%
	3.	946	1200	79%
Halte 3	1.	1423.51	2100	68%
	2.	1511.1	2100	72%
	3.	1468	2100	70%

TABEL 4 Pengujian Perhitungan Jarak dan Waktu

Halte	No	Stopwatch [s]	program Waktu [s]	Akurasi Waktu
Halte 1	1.	67	59.56	89%
	2.	87	135	64%
	3.	76.2	112	68%
Halte 2	1.	144	211.51	68%
	2.	196	287	68%
	3.	188	264	71%
Halte 3	1.	302	294.05	97%
	2.	307.8	384.6	80%
	3.	308.4	384.66	80%

Berdasarkan TABEL 4 dan TABEL 5, Penghitungan Jarak dilakukan dengan cara menghitung jarak pada bus dengan halte 1, halte 2, dan halte 3, perhitungan akurasi jarak bus dilakukan dengan membandingkan hasil program jarak dengan yang dihasilkan oleh Google Maps. Perhitungan waktu dilakukan dengan membagi jarak dengan kecepatan yang dihasilkan oleh modul GPS, perhitungan akurasi waktu dilakukan dengan membandingkan hasil program waktu dengan yang dihasilkan *Stopwatch*.

Berdasarkan hasil pengujian pada Tabel 4-2, waktu dan jarak yang dihasilkan masih kurang akurat dengan presentasi hasil pengukuran akurasi waktu yang dilakukan pengujian sebanyak 3 kali, pada halte 1 sebesar 74%, halte 2 sebesar 69%, dan halte 3 sebesar 86% dibandingkan dengan *Stopwatch*, dan persentasi hasil pengukuran akurasi jarak yang dilakukan pengujian sebanyak 3 kali, pada halte 1 sebesar 81%, halte 2 sebesar 86%, dan halte 3 sebesar 70% dibandingkan dengan Google maps.

V. KESIMPULAN

Pada Perancangan sistem dan hasil pengujian pada proyek akhir ini, dapat disimpulkan pembuatan sistem *tracking* GPS yang di pasang di dalam bus, dapat simpulkan dalam pengiriman data ke dalam Firebase dikategorikan sebagai *soft real-time*, berdasarkan hasil pengujian yang didapatkan dengan rata-rata *delay* 13,88 detik dari batas waktu yang ditentukan, yaitu 3 detik.

Pengiriman data ke dalam file log dapat dikategorikan *soft real-time*, berdasarkan hasil pengujian yang didapatkan dengan rata-rata *delay* 6,01 detik dari batas waktu yang di atur selama 3 detik pengiriman.

Pengujian akurasi modul GPS berdasarkan dari tiga modul GPS yang digunakan, di dapatkan modul GPS L76X GPS HAT sebagai modul GPS terbaik dari hasil penghitungan selisih jarak dengan Google Maps dari 10 titik mendapatkan 5 terbaik. Sedangkan modul GPS Neo-M8M mendapatkan hasil 4 titik terbaik dari 10 titik, dan FMB-130 mendapatkan hasil 1 titik terbaik dari 10 titik.

Pembuatan sistem pemberitahuan jarak dan waktu kedatangan bus, dapat disimpulkan bahwa sistem kurang akurat berdasarkan pengujian yang dilakukan sebanyak 3 kali, dengan hasil pengujian akurasi perhitungan waktu dari 3 titik pemberhentian bus yang didapatkan hasil, pada titik 1 sebesar 74%, titik 2 sebesar 69%, dan titik 3 sebesar 86%, pengujian akurasi perhitungan jarak dengan hasil pada titik 1 sebesar 81%, titik 2 sebesar 86%, dan titik 3 sebesar 70%. sistem ini juga dibuat dengan cara yang kurang efektif dimana tidak dapat dilakukan di jalur perjalanan yang sulit. Karena proses penghitungan jarak dilakukan dengan menghitung secara garis lurus, tidak berdasarkan rute jalan.

VI. REFERENSI

- [1] Ibrahim, M. (2021, May 31). 7 Kota Termacet di Indonesia, Mengejutkan, Jakarta Posisi Kedua, Pertama Kota Mana? - *Galamedia News*. GALAMEDIA GRUP.
- [2] Yuniati, Y., Ulvan, M., Azzahra, M., Teknik Elektro Fakultas Teknik Universitas Lampung Jalan Soemantri Brojonegoro No, J., & Lampung, B. (2017). IMPLEMENTASI MODUL GLOBAL POSITIONING SYSTEM (GPS) PADA SISTEM TRACKING BUS RAPID TRANSIT (BRT) LAMPUNG MENUJU SMART TRANSPORTATION. *SITEKIN: Jurnal Sains, Teknologi Dan Industri*, 14(2), 150–156.
- [3] F. C. D. Putri Rosyadi, Y. Sainika, and Y. Adi Setyoko, "Prototype Real-Time Monitoring System Bus Trans Jateng Berbasis Android untuk Informasi Waktu Kedatangan Bus di Halte", *INISTA*, vol. 3, no. 2, pp. 30-42, Jun. 2021
- [4] Pradana, I. A., Hapsari, G. I., & Gunawan, T. (2020). Sistem Pemantauan Bus Sekolah Dengan Rfid Dan Web Aplikasi Berbasis Internet Of Things. *EProceedings of Applied Science*, 6(1).
- [5] Ahsan, A. S., Wibowo, A. S., & Cahyadi, W. A. (2020). Sistem Pemantau Posisi Bus Karyawan. *EProceedings of Engineering*, 7(1).
- [6] Wahyulianto, R. (2019). Sistem Tracking Kendaraan dengan Mikrokontroler Berbasis Web. *Eprint.Uty.Ac.Id*.
- [7] Gintoro, et.al.2010, "Analisis dan Perancangan Sistem Pencarian Taksi Terdekat dengan Pelanggan Menggunakan Layanan Berbasis Lokasi" Seminar Nasional Aplikasi Teknologi Informasi 2010 (SNATI 2010). Yogyakarta