Rancang Bangun Sistem Manajemen Aktivitas Mobil Poolcar Menggunakan ESP32 dan SIM808 di PT. X

Shaquille Arriza Hidayat[1]
[1]Program Studi Teknik Multimedia dan Jaringan, Jurusan Teknik Informatika dan Komputer, Politeknik Negeri Jakarta
Depok, Jawa Barat
shaquille.arriza.hidayat.tik22@mhsw.pnj.ac.id

Abstract - *Digital technology adoption in Indonesia is crucial for realizing the Indonesia Emas 2045 vision. According to 2025 data from IMDI, the utilization of digital technology can boost productivity by up to 67%. However, the management of poolcar operations within PT. X's mobility and logistic division remains conventional. Recording vehicle activity, vehicle condition, and tracking driver locations are still carried out manually, which relies heavily on honesty and discipline. This research aims to develop an IoT-based poolcar management system using ESP32 and SIM808. The system obtains vehicle coordinates using GPS and transmits them to the backend system via GPRS network and MQTT protocol. The backend system then processes the data, before forwarding the vehicle location via WebSocket to the frontend system. The location is also compared with the activity destination using Haversine algorithm. Results indicate the system successfully tracks vehicles and monitors activities in real-time, validated trough testing on system functionality, GPS accuracy, GSM signal strength, and Quality of Service (QoS) to evaluate the performance of the developed system.*

Keywords: *Haversine Algorithm; Internet of Things (IoT); SIM808; Tracking System*

Abstrak - *Adaptasi teknologi digital di Indonesia merupakan hal yang penting untuk mewujudkan visi Indonesia Emas 2045. Berdasarkan data IMDI tahun 2025, pemanfaatan teknologi digital mampu meningkatkan produktivitas hingga 67%. Namun, manajemen aktivitas mobil poolcar pada divisi mobilitas dan logistik di PT. X masih dilakukan secara konvensional. Pencatatan aktivitas kendaraan, kondisi kendaraan, serta pelacakan lokasi pengemudi masih dilakukan secara manual, yang sangat bergantung pada kejujuran dan kedisiplinan. Penelitian ini bertujuan untuk mengembangkan sistem manajemen poolcar berbasis IoT menggunakan ESP32 dan SIM808. Sistem ini memperoleh koordinat kendaraan menggunakan GPS dan mengirimkannya ke sistem backend melalui jaringan GPRS dan protokol MQTT. Sistem backend kemudian memproses data tersebut, sebelum meneruskan lokasi kendaraan melalui WebSocket ke sistem frontend. Lokasi tersebut juga dibandingkan dengan tujuan aktivitas menggunakan algoritma Haversine. Hasil penelitian menunjukkan bahwa sistem berhasil melacak kendaraan dan memonitor aktivitas secara real-time, yang divalidasi melalui pengujian pada fungsionalitas sistem, akurasi GPS, kekuatan sinyal GSM, dan Quality of Service (QoS) untuk mengevaluasi performa sistem yang dibangun.*

Kata kunci: *Algoritma Haversine, Internet of Things (IoT), SIM808, Sistem Pelacakan*

I. PENDAHULUAN

Digitalisasi telah menjadi prioritas nasional dalam beberapa tahun terakhir, yang berfungsi sebagai kunci utama untuk meningkatkan produktivitas dan efisiensi. Berdasarkan data IMDI tahun 2025, karyawan yang dibekali dengan keterampilan digital mampu meningkatkan produktivitas sebesar 67% pada usaha menengah besar dan 46% pada usaha menengah kecil. Peningkatan transformasi digital akan menjadi hal yang esensial untuk mewujudkan visi Indonesia Emas 2045 [1].

Salah satu area yang dapat ditingkatkan melalui digitalisasi adalah divisi mobilitas dan logistik, khususnya pada manajemen operasi mobil poolcar. Saat ini, manajemen poolcar yang digunakan untuk pengiriman barang maupun transportasi rapat masih sangat bergantung pada metode konvensional. Pencatatan aktivitas dan kondisi kendaraan masih dilakukan menggunakan kertas, sementara pelacakan lokasi pengemudi masih dilakukan melalui panggilan telepon dan pesan singkat untuk memverifikasi posisi pengemudi. Metode ini dinilai sangat tidak efisien karena sangat bergantung pada kejujuran dan kedisiplinan staf. Di sisi lain, solusi pelacakan digital komersial yang

tersedia di pasaran sering kali menuntut biaya langganan yang mahal dan kurang fleksibel untuk disesuaikan dengan kebutuhan operasional spesifik perusahaan.

Perkembangan Internet of Things (IoT) bisa menjadi solusi alternatif yang lebih fleksibel, terjangkau, juga dapat dikustomisasi. Beberapa penelitian telah dilakukan untuk membangun sistem pelacakan kendaraan berbasis Internet of Things. Penelitian yang dilakukan sebelumnya membangun sistem pelacakan menggunakan modul GPS NEO-6M untuk mendapatkan koordinat lokasi dan menggunakan modul GSM SIM800L untuk komunikasi dengan Internet [2]. Kemudian penelitian lain membangun pelacakan kendaraan menggunakan modul GPS NEO-M8N dan modul LTE SIM7600 [3]. Selain itu, penelitian tersebut juga menggunakan protokol MQTT sebagai komunikasi antara sistem IoT dengan server backend juga protokol WebSocket untuk menampilkan lokasi secara real-time pada frontend.

Penelitian ini mengusulkan rancang bangun sistem manajemen aktivitas mobil poolcar berbasis *Internet of Things* (IoT) menggunakan mikrokontroler ESP32 dan modul GSM/GPRS/GPS SIM808. Dalam penelitian ini, sistem akan dirancang untuk menggunakan teknologi GPS guna mendapatkan lokasi kendaraan dan jaringan GSM untuk mengirimkan data ke sistem pemantauan menggunakan protokol MQTT. Selain itu, setiap koordinat kendaraan yang mendekati koordinat tujuan aktivitas akan dihitung dan dianggap selesai menggunakan algoritma Haversine. Sistem ini juga menyediakan pemantauan lokasi kendaraan dan lokasi tujuan menggunakan WebSocket untuk menampilkan lokasi kendaraan secara real-time. Sistem yang dikembangkan ini diharapkan dapat membantu dalam mengelola kegiatan operasional kendaraan.

II. METODOLOGI PENELITIAN

Bagian ini menguraikan metodologi penelitian, yang terdiri dari analisis kebutuhan perangkat lunak dan perangkat keras, perancangan arsitektur, implementasi tiga sistem utama, pengujian sistem, dan evaluasi. Tahapan penelitian dapat dilihat pada Gambar 1.

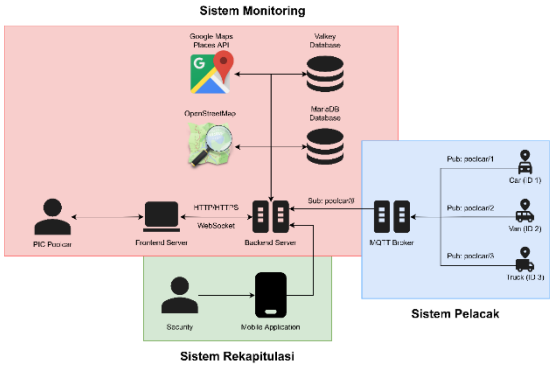


Gambar 1. Tahapan Penelitian

A. *Arsitektur Sistem*

Sistem manajemen aktivitas poolcar terdiri dari tiga sistem kecil yang saling terhubung, yaitu sistem pelacak, sistem *monitoring*, dan sistem rekapitulasi. Sistem pelacak dirancang untuk memperoleh koordinat poolcar dan mengirimkannya ke server *backend* melalui protokol MQTT. Sistem *monitoring* terdiri dari sebuah server yang bertindak sebagai server *backend* dan *frontend*. Server tersebut terhubung langsung ke MariaDB dan Valkey untuk menyimpan data aktivitas dan data koordinat. Selain itu, *frontend* menggunakan OpenStreetMap untuk menampilkan penanda kendaraan dan aktivitas, serta Google Places API untuk melakukan pencarian dalam membuat kontak baru. Sistem rekapitulasi digunakan untuk mencatat kondisi poolcar sebelum dan sesudah aktivitas.

Pengguna dapat mengakses sistem baik melalui web *frontend* dari sistem *monitoring* maupun melalui aplikasi *mobile* dari sistem pelaporan. Server *frontend* hanya dapat diakses oleh karyawan, sedangkan aplikasi *mobile* hanya dapat diakses oleh sekuriti. Sebagai tambahan, admin dapat mengakses kedua sistem tersebut sekaligus memiliki fitur tambahan seperti membuat, memperbarui, dan menghapus data poolcar, kontak, serta pelacak. Keseluruhan arsitektur dapat dilihat pada Gambar 2.

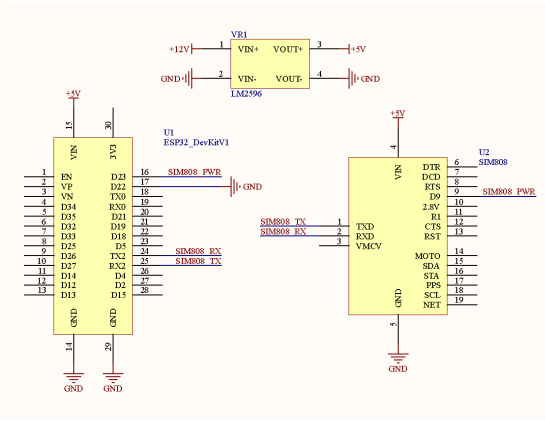


Gambar 2. Arsitektur Sistem

B. *Perakitan Perangkat Keras*

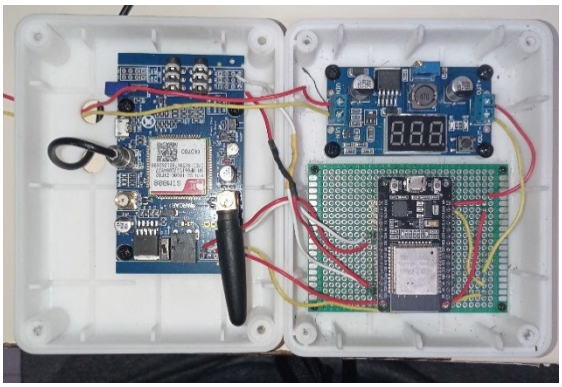
Sistem pelacak dibangun menggunakan mikrokontroler ESP32, modul SIM808 untuk akuisisi GPS dan GSM, serta LM2596 *step-down*. ESP32 juga terhubung langsung ke SIM808, berkomunikasi melalui protokol UART. Keseluruhan sistem dijalankan menggunakan aki kendaraan, yang diturunkan tegangannya menjadi 5V untuk dapat menerima penarikan arus dari

SIM808 secara aman. Rancangan skematis dari sistem pelacak dapat dilihat pada Gambar 3.



Gambar 3. Rancangan Skematis

ESP32 diprogram dalam bahasa C++ menggunakan PlatformIO untuk mengakuisisi data GPS, memformatnya menjadi *payload* JSON, dan *publish* ke broker MQTT. Komponen tersebut disolder pada PCB *dot-matrix* dan dimasukkan ke dalam *casing* PVC 3D khusus untuk melindungi perangkat keras. Hasil akhir perakitan sistem pelacak ditunjukkan pada Gambar 4.



Gambar 4. Sistem Pelacak beserta *Casing*

C. *Konfigurasi Server Backend*

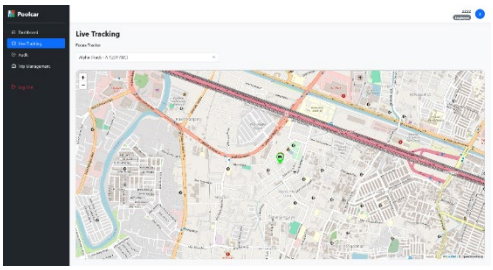
Sistem pemantauan untuk sistem *backend* dan *frontend* *dideploy* pada AlmaLinux 10 menggunakan Podman Quadlet. Server ini menggabungkan MariaDB untuk penyimpanan data, Valkey untuk menyimpan koordinat terbaru, dan Mosquitto sebagai MQTT broker. *Backend* dikembangkan menggunakan bahasa Rust yang menjalankan layanan REST API dan WebSocket sekaligus memproses *payload* MQTT yang diterima. Struktur *database* ditunjukkan pada Gambar 5.



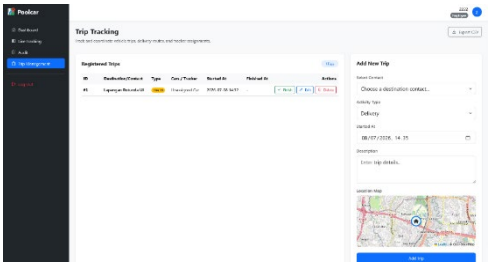
Gambar 5 Struktur *Database*

D. *Aplikasi Web Frontend*

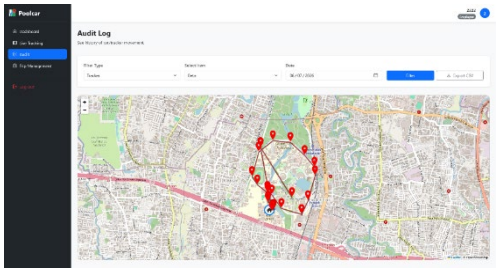
Web *frontend* bertindak sebagai antarmuka pengguna untuk mengakses sistem *monitoring*. *Frontend* berkomunikasi dengan *backend* melalui REST API dan WebSocket untuk menampilkan halaman. *Frontend* dibangun menggunakan JavaScript dengan *library* OpenStreetMap. Antarmuka ini memungkinkan pengguna untuk melacak lokasi poolcar secara *real-time*, mengelola aktivitas, dan melihat riwayat pergerakan kendaraan. Ketiga antarmuka ini ditunjukkan pada Gambar 6 hingga 8.



Gambar 6. Halaman Pelacakan Lokasi Poolcar secara Real-Time



Gambar 7. Halaman Pengelolaan Aktivitas



Gambar 8. Halaman Riwayat Lokasi Poolcar

E. *Aplikasi Mobile Rekapitulasi*

Sebuah aplikasi *mobile* dikembangkan menggunakan Flutter untuk sistem rekapitulasi. Aplikasi Android ini memungkinkan sekuriti untuk memindai kode QR pada poolcar untuk mencatat bahan bakar dan nilai odometer pada poolcar yang dilakukan saat keberangkatan (*departure*) dan kepulangan (*return*).

F. *Prosedur Pengujian Fungsionalitas*

Pengujian fungsionalitas sistem dilakukan menggunakan metode *black-box* dilakukan untuk memverifikasi kemampuan inti dari sistem pelacak, pemantauan, dan pelaporan. Pengujian ini mencakup skenario berhasil (publikasi MQTT, penerusan data WebSocket, dan pembaruan *database*) serta skenario gagal (kehilangan daya modul, pemutusan jaringan GSM, dan *downtime* server).

G. *Prosedur Pengujian Akurasi GPS*

Pengujian Akurasi GPS dilakukan dengan membandingkan koordinat yang diperoleh dengan koordinat dari referensi GPS yang mempunyai presisi tinggi [7]. Jarak *error* didapatkan menggunakan rumus Haversine pada tiga skenario lingkungan yang berbeda, yaitu: *Open Area*, *Semi-Urban*, dan *Semi-Indoor* [8]. Nilai yang diambil adalah *Circular Error Probable* (CEP), *Carrier-to-Noise ratio* (C/N0), dan *Horizontal Dilution of Precision* (HDOP) yang didapat dengan mengambil 100 titik data di setiap lokasi [9], [10]. Spesifikasi GPS referensi dirinci pada Tabel 1.

TABEL I. SPESIFIKASI REFERENSI GPS

Tipe	Spesifikasi
Module Name	CYCLONE M10 GPS Module M1018C
GPS Chip	UBX-M10050-KB
GNSS	BeiDou, Galileo, GLONASS, GPS/QZSS, SBAS
Horizontal Accuracy	1.5 m CEP
Satellite Acquisition	Cold start 28 s and hot start 1 s

H. *Prosedur Pengujian Kekuatan Sinyal GSM*

Sistem pelacak dipasang pada kendaraan lalu dijalankan sepanjang rute yang telah ditentukan. Nilai *Received Signal Strength Indicator* (RSSI) diukur secara terus-menerus untuk mengamati kekuatan GPRS. Kekuatan sinyal dapat dikategorikan menggunakan nilai RSSI yang dirinci pada Tabel II.

TABEL II. KATEGORI NILAI RSSI

Nilai RSSI	Kategori Sinyal
< -100	Sangat Buruk
-100-(-90) dBm	Buruk
-90-(-80) dBm	Cukup
-80-(-70) dBm	Baik
>-70 dBm	Sangat Baik

I. *Prosedur Pengujian Quality of Service (QoS)*

Untuk menilai kualitas transmisi MQTT melalui jaringan 2G/GPRS, data dikumpulkan selama periode operasional 24 jam. Evaluasi ini berfokus pada pengukuran *throughput*, latensi, dan *packet loss*. [9], [10].

III. HASIL DAN PEMBAHASAN

Untuk menentukan apakah sistem manajemen aktivitas mobil poolcar yang dikembangkan memenuhi kebutuhan operasionalnya, evaluasi dilakukan pada ketiga pengujian sistem, yaitu fungsionalitas sistem, akurasi GPS, kekuatan sinyal GSM, dan *Quality of Service* (QoS).

A. *Hasil Pengujian Fungsionalitas Sistem*

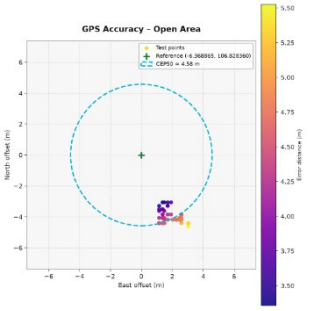
Pengujian fungsionalitas dilakukan menggunakan pendekatan *black-box* untuk mengevaluasi interaksi antara perangkat keras, API *backend*, dan antarmuka *frontend*. Sistem menunjukkan nilai *error rate* 0% di seluruh skenario yang berhasil. Hal ini menunjukkan kemampuan sistem untuk mengakuisisi data GPS, mentransmisikan *payload* MQTT, dan memproses algoritma Haversine untuk memperbarui *database* secara mulus.

Sementara pada skenario kegagalan, sistem menunjukkan nilai *error rate* sebesar 40%. ESP32 berhasil menyala kembali dari simulasi gangguan layanan MQTT dan kehilangan daya sementara.

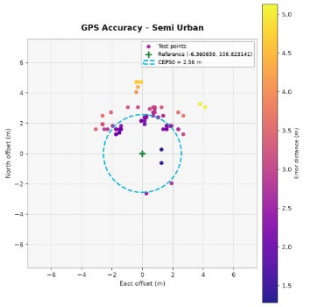
Namun, sistem gagal pulih ketika terjadi pencabutan kartu SIM dan penghentian pada layanan MariaDB.

B. Hasil Pengujian Akurasi GPS

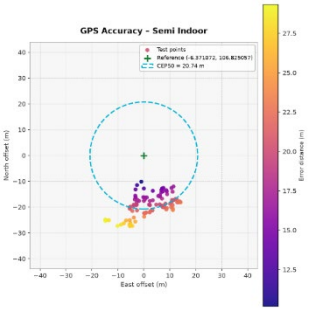
Akurasi modul SIM808 dievaluasi dengan menghitung *Circular Error Probable* (CEP) terhadap GPS referensi. Pengujian tersebut menghasilkan CEP sebesar 4,58 meter di skenario *Open Area*, 2,56 meter di lingkungan *Semi-Urban*, dan 20,74 meter di lingkungan *Semi-Indoor*. Ruangan. Akurasi GPS dapat dilihat pada Gambar 9 sampai gambar 11..



Gambar 9. Performa Akurasi GPS di Skenario *Open Area*



Gambar 10. Performa Akurasi GPS di Skenario *Semi-Urban*

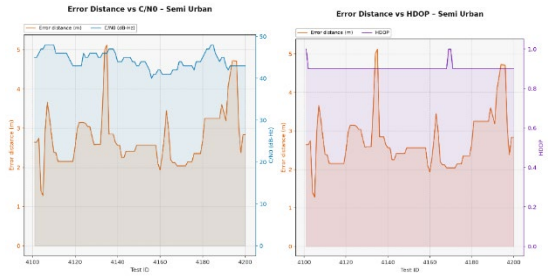


Gambar 11. Performa Akurasi GPS di Skenario *Semi Indoor*

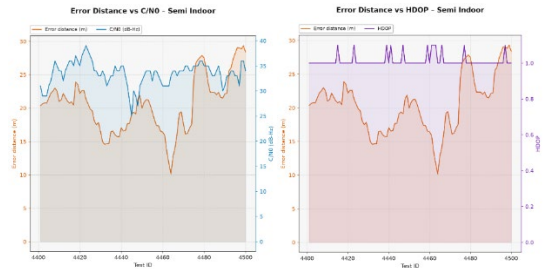
Lingkungan *Semi-Urban* menjadi skenario dengan kinerja paling baik, yang menunjukkan bahwa bangunan bertingkat rendah dan tutupan pohon tidak menurunkan kualitas sinyal. Sebaliknya, pada skenario *Semi-Indoor*. rasio C/N_0 turun ke rata-rata 33,44 dB-Hz, yang secara langsung mengakibatkan nilai *error* sebesar 20,74 meter. Tiga gambar dari Gambar 12 hingga Gambar 14 menunjukkan visualisasi nilai *error* terhadap C/N_0 dan HDOP pada masing-masing ketiga skenario.



Gambar 12. Performa Akurasi GPS di Skenario *Open Area*



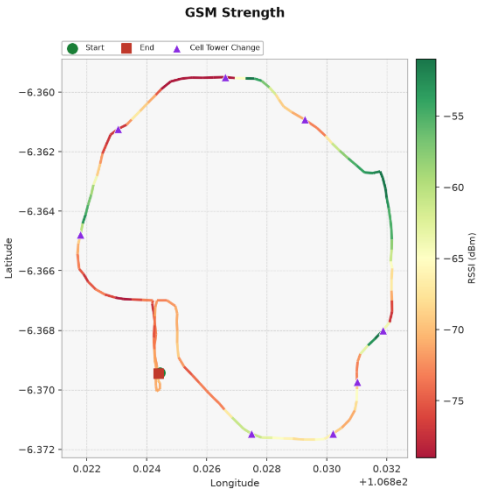
Gambar 13. Performa Akurasi GPS di Skenario *Semi Urban*.



Gambar 14. Performa Akurasi GPS di Skenario *Semi Urban*.

C. Hasil Pengujian Kekuatan Sinyal GSM

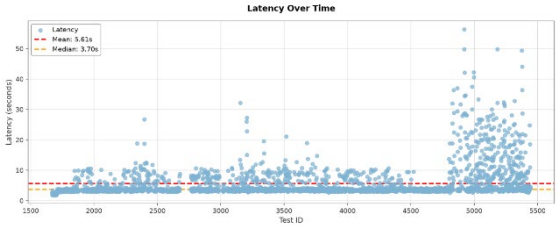
Pengujian menghasilkan 3.776 koordinat data menghasilkan rata-rata RSSI sebesar -67.86 dBm, yang mengklasifikasikan jaringan sebagai "Sangat Baik". Secara kebetulan, ketika RSSI mencapai sekitar -70 dBm dan lebih rendah, SIM808 akan mengubah koneksi menara selulernya. Grafik latensi GSM ditunjukkan pada Gambar 15.



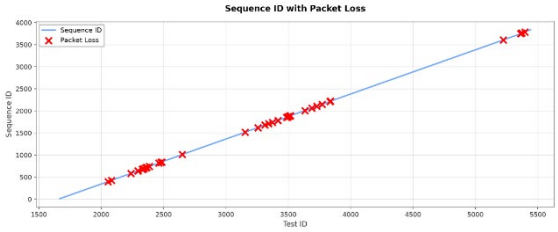
Gambar 15. Plot Latensi GSM dengan nilai RSSI

D. Hasil Pengujian Quality of Service (QoS)

Hasil evaluasi pengujian QoS pada transmisi MQTT melalui jaringan GPRS mengungkapkan latensi rata-rata sebesar 5,61 detik dan *throughput* sebesar 0,609 kbps. Walaupun nilai tersebut masuk ke dalam kategori "Buruk" berdasarkan standar TIPHON, hal tersebut merupakan batasan dari jaringan GPRS. Sedangkan, *packet loss* yang didapat tetap sangat rendah pada angka 1,32%. Hal ini mengonfirmasi bahwa meskipun transmisi sistem lambat, jaringan ini sangat andal. Nilai tersebut sangat cocok dengan sistem manajemen aktivitas mobil poolcar yang tidak membutuhkan batasan *real-time*. Visualisasi latensi berbanding waktu dan *packet loss* berbasis waktu ditunjukkan pada Gambar 16 dan Gambar 17.



Gambar 16. Plot dari QoS Latensi berbanding Waktu



Gambar 16. Plot dari QoS *Packet Loss* berbanding Waktu

IV. KESIMPULAN DAN SARAN

Penelitian ini telah berhasil merancang, mengimplementasikan, dan mengevaluasi sistem manajemen aktivitas mobil poolcar berbasis IoT menggunakan mikrokontroler ESP32 dan modul SIM808. Perangkat keras tersebut secara andal memperoleh koordinat melalui GPS dan mentransmisikannya secara *real-time* ke server *backend* melalui jaringan GPRS menggunakan protokol MQTT. Lebih lanjut, arsitektur sistem ini secara efektif mengintegrasikan *frontend* pemantauan berbasis web untuk manajemen aktivitas dan aplikasi *mobile* berbasis Android bagi petugas keamanan untuk mencatat kondisi kendaraan, yang semuanya didukung oleh *backend* yang memanfaatkan MariaDB, Mosquitto, dan Valkey.

Pengujian telah memvalidasi kesiapan operasional sistem, dengan mencapai tingkat kesalahan 0% pada skenario fungsional standar dan menunjukkan tingkat kesalahan 40% selama pengujian yang ekstrem. Pelacakan GPS berkinerja secara optimal di lingkungan Semi-Perkotaan dengan *Circular Error Probable* (CEP) sebesar 2,56 meter, sementara lingkungan Area Terbuka dan Semi-Dalam Ruangan masing-masing menghasilkan CEP sebesar 4,58 meter dan 20,74 meter. Evaluasi jaringan mengungkapkan koneksi seluler yang sangat stabil dengan rata-rata RSSI sebesar -67,86 dBm. Meskipun dibatasi oleh kendala arsitektur bawaan GPRS, *packet loss* yang sangat rendah yaitu 1,32% mengonfirmasi keandalan sistem.

V. REFERENSI

[1] Kementerian Komunikasi dan Digital Republik Indonesia, “Indeks Masyarakat Digital Indonesia 2025,” Kementerian Komunikasi dan Digital Republik Indonesia, Jakarta, Laporan Tahunan IMDI 2025, 2025. [Online]. Available: https://imdi.sdmdigital.id/

[2] M. Y. Ashadi, S. Ariyani, B. S. Rintyarna, and N. K. Wardati, “Desain Sistem Keamanan Sepeda Motor Dengan Memanfaatkan GPS Tracker Berbasis IoT,” *J. Tek. Elektro Dan Komputasi ELKOM*, vol. 4, no. 2, pp. 152–159, Aug. 2022, doi: 10.32528/elkom.v4i2.7428.

[3] S. D. Mufti, “Rancang Bangun Sistem Pelacakan Kendaraan Pengangkut Sampah Berbasis IoT Menggunakan ESP32, SIM7600, dan NEO-M8,” Skripsi, Politeknik Negeri Jakarta, Depok,

2025. [Online]. Available: https://repository.pnj.ac.id/id/eprint/27615/