



Tersedia Online : <http://e-journals.unmul.ac.id/>

ADOPSI TEKNOLOGI DAN SISTEM INFORMASI (ATASI)

Alamat Jurnal : <http://e-journals2.unmul.ac.id/index.php/atasi/index>



Implementasi Algoritma Haversine pada Aplikasi Reminder Service Sepeda Motor Berbasis GPS Smartphone

Ridwan Kurniawan^{1)*}, Dini Hamidin²⁾, Widia Resdiana³⁾

^{1,2,3)} Diploma III Teknik Informatika, Sekolah Vokasi, Universitas Logistik dan Bisnis Internasional

E-Mail : ridwankurniawan419@gmail.com¹⁾; dinihamidin@ulbi.ac.id²⁾; widiarresdiana@ulbi.ac.id³⁾;

ARTICLE INFO

Article history:

Received : 08/08/2025

Revised : 02/06/2026

Accepted : 02/07/2026

Available online : 31/08/2026

Keywords:

*Motorcycle
Service Reminder
Haversine Algorithm
WebSocket
GPS*

ABSTRACT

Motorcycles are the primary mode of transportation for the Indonesian population, with high usage levels across various social groups. Routine vehicle maintenance, such as oil changes and component inspections, greatly depends on the distance traveled and the time of the last service. However, many motorcycle users experience delays in performing maintenance, which can lead to engine damage. The Motorcycle Service Reminder application is designed to help users remember service schedules accurately based on actual travel distance. The Haversine algorithm is used to calculate the vehicle's travel distance based on the user's location obtained through the smartphone's GPS sensor. WebSocket technology is utilized as a real-time communication channel between the user's device and the server, enabling continuous and direct transmission of location data to the server for processing. This research results in the development of a Motorcycle Service Reminder application featuring automatic service notifications based on distance, periodic location tracking, and service history management. The application is expected to help maintain the vehicle in optimal condition and ensure its safe use.

ABSTRAK

Kendaraan sepeda motor merupakan sarana transportasi utama masyarakat Indonesia, dengan tingkat penggunaan yang tinggi dari berbagai kalangan. Perawatan rutin kendaraan, seperti penggantian oli dan pengecekan komponen, sangat bergantung pada jarak tempuh dan waktu terakhir servis. Namun, banyak pengguna sepeda motor yang mengalami keterlambatan dalam melakukan servis yang dapat mengakibatkan kendaraanya menyebabkan kerusakan pada mesin. Aplikasi Reminder Service Sepeda Motor digunakan untuk membantu pengguna dalam mengingat jadwal servis kendaraan secara tepat waktu berdasarkan jarak tempuh aktual. Algoritma Haversine digunakan untuk menghitung jarak tempuh kendaraan berdasarkan lokasi pengguna yang diperoleh melalui sensor GPS pada smartphone. Teknologi WebSocket dimanfaatkan sebagai jalur komunikasi real-time antara perangkat pengguna dan server, sehingga data pergerakan lokasi dapat dikirim secara langsung dan berkelanjutan ke server untuk dilakukan perhitungan. Penelitian ini menghasilkan aplikasi Reminder Service Sepeda Motor yang memiliki fitur utama berupa notifikasi servis otomatis berdasarkan jarak tempuh, pelacakan lokasi secara berkala, dan pengelolaan riwayat servis. Aplikasi ini diharapkan dapat menjaga kondisi kendaraan tetap optimal dan aman digunakan.

2026 Adopsi Teknologi dan Sistem Informasi (ATASI) with CC BY SA license.

*) Corresponding Author

<https://doi.org/10.30872/atasi.v5i2.3602>

2026 Adopsi Teknologi dan Sistem Informasi (ATASI) with CC BY SA license.

1. PENDAHULUAN

Menurut (Badan Pusat Statistik, 2022), Kendaraan sepeda motor merupakan salah satu alat transportasi utama yang banyak digunakan oleh masyarakat Indonesia. Tercatat sebanyak 125.305.332 unit kendaraan bermotor, dan sepeda motor menjadi sarana transportasi sehari-hari bagi berbagai kalangan, mulai dari pekerja kantoran, buruh pabrik, pengemudi ojek online, hingga pelajar. Menurut (Dayera et al., 2023) seiring dengan intensitas penggunaan kendaraan yang tinggi, diperlukan perawatan berkala agar performa kendaraan tetap optimal dan umur mesin tetap terjaga, beberapa bentuk perawatan rutin yang penting dilakukan antara lain pergantian oli, servis mesin, pengecekan rem, serta pengecekan kondisi ban. Perawatan tersebut umumnya bergantung pada jarak tempuh kendaraan dan waktu terakhir melakukan servis. Namun banyak pengguna sepeda motor yang tidak mencatat atau mengingat kapan terakhir kali mereka melakukan servis. Hal ini menyebabkan servis kendaraan sering terlambat dilakukan, yang dapat berakibat pada penurunan performa hingga kerusakan komponen kendaraan (Sumantri et al., 2025).

Berdasarkan permasalahan tersebut, diperlukan sistem pengingat servis sepeda motor yang dapat memberikan notifikasi otomatis kepada pengguna berdasarkan jarak tempuh kendaraannya. Oleh karena itu, penulis mengusulkan pembuatan aplikasi *Reminder Service Sepeda Motor* yang dapat membantu pengguna mengingat waktu servis secara tepat waktu. Aplikasi ini memanfaatkan sensor GPS pada perangkat smartphone untuk mendeteksi pergerakan pengguna dan menghitung jarak tempuh secara real-time menggunakan algoritma Haversine, yang digunakan untuk menghitung jarak antara dua titik koordinat lintang dan bujur di permukaan bumi. Selain itu, aplikasi ini dirancang menggunakan teknologi WebSocket agar informasi yang ditampilkan bersifat dinamis dan dapat diperbarui secara real-time. Dengan adanya aplikasi ini, diharapkan pengguna tidak lagi lupa atau terlambat dalam melakukan perawatan kendaraan, sehingga kondisi kendaraan tetap terjaga dan aman digunakan.

2. TINJAUAN PUSAKA

Penelitian yang dilakukan oleh Yosia Kehat Driesa dan Ramos Somya (Driesa & Somya, 2023) bertujuan untuk merancang aplikasi pengingat servis rutin sparepart sepeda motor berbasis Android. Aplikasi tersebut dikembangkan menggunakan framework React Native dan Firebase Realtime Database, serta menerapkan metode pengembangan *Extreme Programming* (XP). Fitur utama dari aplikasi ini adalah pengingat jadwal servis melalui notifikasi otomatis berdasarkan waktu. Pengujian dilakukan menggunakan metode *black-box testing*. Hasil penelitian menunjukkan bahwa aplikasi ini efektif dalam membantu pengguna menjaga jadwal servis kendaraan mereka.

Sementara itu, penelitian oleh Fajar Kurniawan (Kurniawan, 2022) berfokus pada pengembangan aplikasi Petugas *Realtime Tracking* berbasis Android, yang memanfaatkan algoritma Haversine untuk menghitung jarak, serta menggunakan *Polyline* untuk visualisasi lintasan pergerakan. Proses pengembangan dilakukan dengan metode *prototype* yang terdiri dari enam tahapan, dan diuji melalui *alpha testing* oleh pengembang serta *beta testing* oleh pengguna. Evaluasi sistem dilakukan menggunakan metode PIECES, yang mencakup aspek performa (*performance*), efisiensi, kontrol, dan layanan (*service*). Hasil evaluasi menunjukkan skor rata-rata sebesar 91,2%, yang mengindikasikan bahwa aplikasi berjalan dengan baik dan sesuai harapan.

Jika dibandingkan dengan kedua penelitian tersebut, penelitian yang dilakukan oleh penulis juga menggunakan algoritma Haversine untuk menghitung jarak tempuh kendaraan. Namun, penerapannya difokuskan pada sistem *reminder* servis kendaraan yang dikombinasikan dengan komunikasi real-time menggunakan teknologi WebSocket. Berbeda dengan penelitian sebelumnya yang bergantung pada Firebase atau hanya fokus pada pelacakan lokasi, sistem ini dirancang untuk mengirimkan data pergerakan secara dinamis ke server, yang kemudian digunakan untuk menghitung jarak tempuh sebagai dasar pemberian pengingat servis kendaraan. Penelitian ini dikembangkan menggunakan metode Waterfall dan bertujuan untuk menghadirkan solusi pengingat servis berbasis pelacakan jarak tempuh aktual (*real-distance tracking*).

A. Reminder Service Sepeda Motor

Reminder merupakan sistem aplikasi yang dirancang untuk menyampaikan informasi mengenai jadwal suatu aktivitas tertentu. Fungsinya tidak hanya sebagai pengingat waktu, tetapi juga sebagai penyedia informasi kontekstual yang relevan dengan detail kegiatan yang direncanakan (Ilhadi et al., 2023). Reminder service sepeda motor hadir sebagai solusi untuk membantu pengguna mengingat jadwal perawatan kendaraannya secara tepat waktu (Driesa & Somya, 2023).

B. GPS Smartphone

Global Positioning System (GPS) adalah sistem navigasi berbasis satelit yang memungkinkan perangkat menerima sinyal dari satelit GPS untuk menentukan lokasi geografis secara presisi. Sistem GPS terdiri atas 24 satelit yang mengorbit Bumi pada ketinggian sekitar 20.200 kilometer dan dirancang untuk menyediakan informasi posisi, kecepatan tiga dimensi, serta waktu secara terus-menerus di seluruh dunia tanpa bergantung pada kondisi cuaca maupun waktu (Rahman et al., 2023). Smartphone modern menggunakan A-GPS yang menggabungkan sinyal GPS dan jaringan seluler untuk penentuan lokasi lebih cepat, namun dengan akurasi lebih rendah dibanding GPS tradisional. Fitur LBS memanfaatkan jaringan seluler untuk memberikan informasi lokasi saat aplikasi seperti peta digunakan. Meski praktis untuk kebutuhan harian, akurasi GPS smartphone perlu dipertimbangkan untuk

pengumpulan data presisi. Berbagai aplikasi kini memudahkan pencatatan titik lokasi (waypoints) secara instan (Krista & Bettinger, 2019). Informasi lokasi yang diperoleh melalui GPS dapat dimanfaatkan dalam berbagai layanan berbasis lokasi atau Location Based Service (LBS). Location Based Service (LBS) adalah layanan yang menggunakan data lokasi pengguna sebagai dasar dalam penyediaan informasi berbasis geografis. Teknologi ini memungkinkan aplikasi untuk menampilkan informasi lokasi secara real-time dan memberikan layanan yang disesuaikan dengan posisi pengguna saat itu (Hardiansyah et al., 2018). Dalam penelitian ini, LBS digunakan untuk memperoleh koordinat lokasi pengguna yang kemudian diproses menggunakan Algoritma Haversine untuk menghitung jarak antara pengguna dan lokasi tujuan.

C. Algoritma Haversine

Metode Haversine adalah rumus navigasi yang digunakan untuk menghitung jarak terpendek antara dua titik di permukaan bumi, yang ditentukan berdasarkan garis bujur (longitude) dan garis lintang (latitude). Rumus ini pertama kali ditemukan oleh James Andrew pada tahun 1805 dan digunakan oleh Josef de Mendoza y Ríos pada tahun 1801. Istilah "haversine" diperkenalkan oleh Prof. James Inman pada tahun 1835, yang awalnya digunakan dalam penelitian terkait Masalah Utama Astronomi Nautikal. Selain untuk navigasi di bumi, rumus Haversine juga diterapkan untuk mengukur jarak antar bintang. Rumus Haversine dirancang untuk menghitung jarak antara dua titik di permukaan bumi, yang berbentuk lengkung, sehingga memberikan hasil yang lebih akurat dibandingkan metode yang mengasumsikan bumi sebagai bidang datar (Prasetya et al., 2020).

$$a = \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos(\varphi_1) \cdot \cos(\varphi_2) \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right) \dots\dots\dots(1)$$

$$c = 2 \cdot \text{atan}^2(\sqrt{a} \sqrt{1-a}) \dots\dots\dots(2)$$

$$d = R \cdot c \dots\dots\dots(3)$$

Dimana :

D = jarak antara dua titik di permukaan bumi (hasil akhir)

R = jari jari bumi (6371)

$\Delta\varphi$ = selisih lintang = $\varphi_1 - \varphi_2$

$\Delta\lambda$ = selisih bujur = λ_1, λ_2

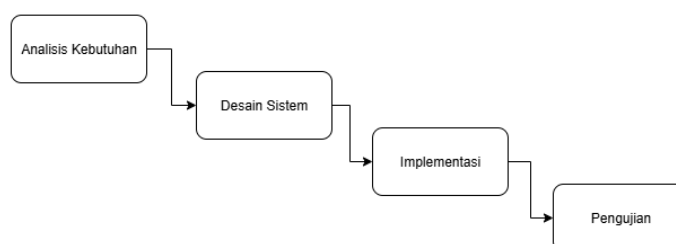
φ_1, φ_2 = lintang (latitude) titik 1 dan 2 dalam radian

D. Websocket

WebSocket merupakan protokol komunikasi yang memungkinkan koneksi dua arah (full-duplex) antara klien dan server secara persistif. Tidak seperti model HTTP tradisional yang bersifat satu arah dan terputus setelah respons diberikan, WebSocket memungkinkan data dikirim secara real-time tanpa perlu melakukan permintaan berulang. Protokol ini menyediakan Application Programming Interface (API) yang dapat digunakan oleh halaman web untuk menjalin koneksi langsung dengan server jarak jauh, sehingga sangat mendukung implementasi sistem real-time secara efisien dan ringan dibandingkan metode sebelumnya seperti long polling atau Comet (Ogundeyi & Yinka-Banjo, 2019).

3. METODE PENELITIAN

Metode pengembangan sistem menggunakan metode *Waterfall* yang meliputi tahap analisis kebutuhan, desain sistem, implementasi dan pengujian. Metode Waterfall adalah salah satu model dalam *Software Development Life Cycle (SDLC)* yang menggunakan pendekatan berurutan (sekuensial), di mana setiap tahapan pengembangan sistem harus diselesaikan terlebih dahulu sebelum melanjutkan ke tahap berikutnya (Hidayati, 2019).



Gambar 1. Metode *Waterfall*

A. Analisis Kebutuhan

Tahapan ini bertujuan untuk mengidentifikasi dan merumuskan kebutuhan sistem, baik kebutuhan fungsional maupun non-fungsional.

- Kebutuhan fungsional mencakup fitur-fitur yang harus dimiliki aplikasi, seperti pencatatan data servis, pelacakan lokasi, perhitungan jarak, dan pengiriman notifikasi.
- Kebutuhan non-fungsional mencakup performa sistem, keamanan data, antarmuka pengguna yang responsif, dan dukungan real-time communication.

B. Desain Sistem

Tahapan ini bertujuan untuk menerjemahkan kebutuhan yang sudah dikumpulkan ke dalam bentuk rancangan sistem.

- Desain arsitektur sistem: Menentukan bagaimana komponen frontend (React Native), backend (REST API + WebSocket), dan database (MongoDB) saling terhubung.
- Desain model bisnis: Menggunakan BPMN untuk menggambarkan alur proses.
- Desain sistem terperinci: Menggunakan UML seperti Use Case Diagram, Sequence Diagram, dan Activity Diagram untuk menggambarkan hubungan antar entitas dan alur logika program.
- Algoritma: Perancangan alur perhitungan jarak menggunakan Flowchart Haversine.

C. Implementasi

Tahap implementasi merupakan proses pembangunan sistem berdasarkan desain yang telah dibuat.

- Pengembangan dilakukan menggunakan React Native untuk antarmuka pengguna (user interface).
- WebSocket digunakan untuk komunikasi real-time antara aplikasi dan server, khususnya dalam pengiriman data lokasi GPS.
- REST API digunakan untuk kebutuhan seperti login, registrasi, dan manajemen data.
- MongoDB digunakan sebagai database untuk menyimpan data pengguna, data kendaraan, dan histori servis.
- Logika perhitungan jarak ditempatkan di server menggunakan algoritma Haversine.

D. Pengujian

Tahapan ini dilakukan untuk memastikan bahwa sistem berjalan sesuai dengan kebutuhan dan tidak memiliki kesalahan fungsional.

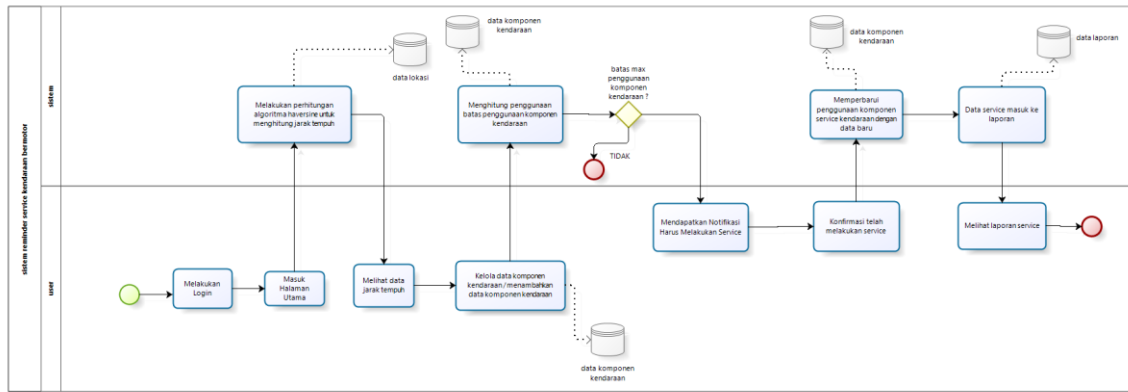
- Pengujian black-box: Untuk menguji fungsionalitas sistem dari sisi pengguna, seperti apakah tombol berfungsi dengan benar, sistem memberikan notifikasi jika jarak tempuh telah melampaui batas, dll.
- Pengujian unit test: Pengujian terhadap fungsi-fungsi tertentu pada kode secara terpisah menggunakan *Jest*.

4. HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil dari proses perancangan dan pengembangan aplikasi *Reminder Service Sepeda Motor*. Proses pengembangan dilakukan berdasarkan metode *Waterfall*, yang terdiri dari tahapan analisis kebutuhan, desain sistem, implementasi, serta pengujian. Pada tahap ini, disajikan model proses bisnis menggunakan *Business Process Model and Notation (BPMN)*, perancangan sistem menggunakan *Unified Modeling Language (UML)* seperti *Use Case Diagram*, serta *Flowchart* untuk menggambarkan algoritma *Haversine*. Selain itu, disajikan pula dokumentasi hasil implementasi dan pengujian fungsionalitas aplikasi. Hasil yang diperoleh kemudian dianalisis dan diinterpretasikan untuk mengevaluasi sejauh mana aplikasi dapat berfungsi sesuai dengan tujuan yang telah ditetapkan.

A. BPMN

BPMN memodelkan proses bisnis yang digunakan untuk menggambarkan alur kerja dimulai dari kejadian awal, proses yang dijalankan, hingga hasil akhirnya (Ismanto et al., 2020). Dalam rancangan ini menggambarkan alur kegiatan dari pihak pengguna dalam melakukan interaksi dengan aplikasi, mulai dari pengambilan data lokasi hingga proses notifikasi pengingat servis.

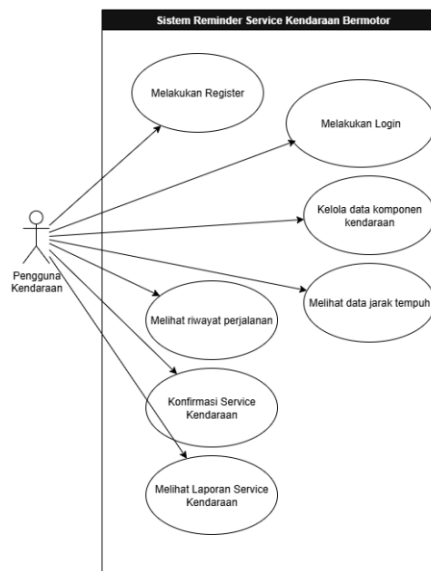


Gambar 2. BPMN Aplikasi Reminder Service Motor

Dalam proses bisnis aplikasi *Reminder Service Sepeda Motor*, sistem akan menerima titik lokasi pengguna sebagai acuan dalam menghitung jarak tempuh kendaraan. Perhitungan jarak dilakukan menggunakan algoritma Haversine. Hasil dari perhitungan tersebut digunakan oleh sistem untuk menentukan komponen kendaraan yang perlu dilakukan servis. Selanjutnya, sistem akan mengirimkan notifikasi kepada pengguna mengenai komponen yang harus diservis. Setelah pengguna melakukan konfirmasi bahwa servis telah dilakukan, sistem akan memperbarui data kondisi komponen kendaraan dan menyimpan transaksi tersebut.

B. Use Case Digram

Use case diagram merupakan salah satu bagian dari *UML* yaitu diagram yang menunjukkan hubungan antara actors dan use cases (Arianti et al., 2022). Pada bagian ini, dijelaskan berbagai interaksi antara aktor (user) dengan sistem, serta fungsi-fungsi utama yang dapat dijalankan oleh pengguna dalam aplikasi.

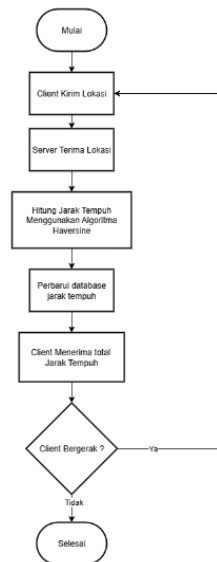


Gambar 3. Use case diagram Aplikasi *Reminder Service Sepeda Motor*

Gambar 3 menunjukkan interaksi pengguna sepeda motor dengan beberapa fungsi utama pada sistem atau aplikasi, seperti melakukan registrasi dan login, mengelola data kendaraan, melihat data jarak tempuh dan riwayat perjalanan, serta melakukan konfirmasi dan melihat laporan servis kendaraan.

C. Flowchart

Flowchart atau sering disebut dengan diagram alir merupakan suatu jenis diagram yang merepresentasikan algoritma atau langkah-langkah instruksi yang berurutan dalam sistem (Arianti et al., 2022). Diagram alir (flowchart) pada aplikasi ini menggambarkan proses perhitungan jarak tempuh kendaraan menggunakan algoritma Haversine yang diterapkan dalam sistem.



Gambar 4. Flowchart Perhitungan Jarak Tempuh

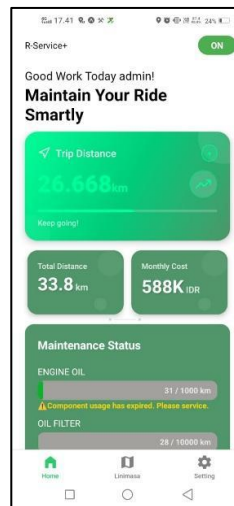
Gambar 4 menunjukkan flowchart yang menggambarkan proses sistem dalam melakukan perhitungan jarak tempuh, yang menjadi dasar dalam pemberian pengingat (*reminder*) service kendaraan pada aplikasi ini. Pengguna (client) akan mengirimkan titik lokasi ke server, kemudian server akan menghitung jarak menggunakan algoritma *Haversine* untuk menghasilkan nilai jarak tempuh. Ketika pengguna bergerak, proses ini akan diulang secara berkala sehingga total jarak tempuh dapat diperbarui secara real-time dan digunakan sebagai acuan dalam mengirimkan notifikasi servis kendaraan.

D. Analisis Penerapan Algoritma Haversine

Algoritma Haversine digunakan untuk menghitung jarak perpindahan pengguna berdasarkan koordinat GPS yang diterima secara berkala oleh sistem. Ketika aplikasi dijalankan, sistem akan menerima data koordinat lokasi pengguna dari GPS smartphone dan menyimpannya sebagai titik awal. Saat pengguna berpindah lokasi, sistem akan menerima koordinat baru dan menghitung jarak antara titik sebelumnya dengan titik terbaru menggunakan algoritma Haversine. Sebagai contoh, ketika pengguna berpindah sejauh 100 meter dari lokasi sebelumnya, sistem akan menghitung jarak perpindahan tersebut menggunakan algoritma Haversine dan menambahkan hasilnya ke total jarak tempuh kendaraan. Proses ini dilakukan secara berulang setiap kali terdapat pembaruan lokasi dari GPS sehingga diperoleh akumulasi jarak tempuh kendaraan secara real-time. Penerapan algoritma Haversine pada aplikasi memungkinkan sistem untuk memperkirakan penggunaan kendaraan berdasarkan pergerakan aktual pengguna. Akumulasi jarak tempuh yang dihasilkan kemudian digunakan sebagai dasar dalam menentukan waktu servis kendaraan dan pengiriman notifikasi reminder service kepada pengguna.

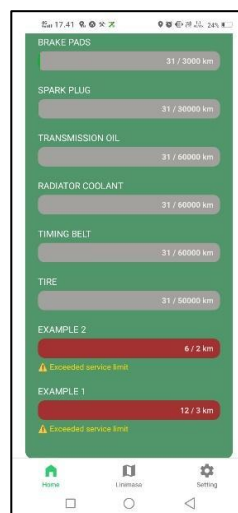
E. Hasil Implementasi

Pada bagian ini disajikan hasil implementasi dari aplikasi *Reminder Service Sepeda Motor* yang telah dikembangkan. Implementasi ditunjukkan melalui berbagai tampilan antarmuka (user interface) aplikasi, mulai dari halaman utama, pelacakan lokasi, hingga notifikasi pengingat servis. Setiap tampilan mencerminkan fungsionalitas utama aplikasi yang berkaitan dengan pemantauan jarak tempuh kendaraan secara real-time. Selain itu, dilakukan pula pengujian aplikasi untuk memastikan seluruh fitur berjalan sesuai dengan yang diharapkan. Pengujian dilakukan untuk mengevaluasi keakuratan perhitungan jarak, kestabilan koneksi real-time menggunakan WebSocket, serta keandalan sistem dalam mengirimkan notifikasi servis berdasarkan jarak tempuh. Hasil implementasi dan pengujian ini menjadi dasar dalam menilai keberhasilan sistem dalam memenuhi kebutuhan pengguna terkait pengingat servis kendaraan bermotor secara otomatis.



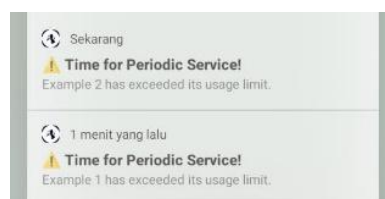
Gambar 5. Halaman Utama Aplikasi

Pada Gambar 5 ditampilkan halaman utama dari aplikasi, yang menyajikan informasi mengenai jarak tempuh kendaraan, laporan servis, serta data komponen kendaraan. Jarak tempuh kendaraan diperoleh melalui perhitungan menggunakan algoritma *Haversine*, yang bekerja berdasarkan perpindahan lokasi pengguna dari satu titik ke titik lainnya. Nilai jarak tempuh ini kemudian digunakan sebagai acuan utama dalam mengaktifkan fitur *reminder* servis kendaraan, sehingga pengguna dapat memperoleh notifikasi secara otomatis ketika komponen kendaraan telah mencapai batas penggunaan maksimal dan perlu dilakukan perawatan



Gambar 6. Data Penggunaan Komponen Kendaraan

Gambar 6. Menunjukkan daftar jenis komponen kendaraan yang terdaftar dalam sistem. Setiap komponen memiliki atribut batas maksimum penggunaan (*maximum usage*) yang ditentukan berdasarkan satuan jarak tempuh (kilometer). Peningkatan nilai penggunaan (*usage*) pada masing-masing komponen dilakukan secara otomatis oleh sistem berdasarkan akumulasi jarak tempuh pengguna yang dihitung menggunakan algoritma *Haversine*. Proses ini memungkinkan pemantauan kondisi komponen secara dinamis dan menjadi dasar dalam pemberian notifikasi servis ketika suatu komponen telah mencapai ambang batas penggunaannya.



Gambar 7. Notifikasi Service Kendaraan

Pada Gambar 7. menunjukan notifikasi yang dikirimkan oleh sistem sebagai pengingat untuk melakukan *service* pada komponen kendaraan tertentu. Notifikasi ini secara otomatis akan dikirimkan apabila penggunaan suatu komponen telah mencapai batas maksimal atau masa berlakunya telah kedaluwarsa. Fitur ini berfungsi sebagai sistem peringatan dini agar pengguna dapat segera melakukan perawatan, sehingga dapat mencegah kerusakan lebih lanjut pada kendaraan.

Penelitian ini menghasilkan sebuah aplikasi *reminder* servis sepeda motor berbasis *Android* yang memanfaatkan *GPS* pada *smartphone*. Aplikasi ini ditujukan bagi pengguna sepeda motor untuk membantu mereka mengingat jadwal servis rutin kendaraan. Saat aplikasi dijalankan, halaman utama yang ditampilkan adalah halaman login. Pengguna yang telah terdaftar dapat langsung masuk dengan memasukkan username dan password yang telah didaftarkan. Jika belum memiliki akun, pengguna dapat melakukan pendaftaran terlebih dahulu. Sistem akan memantau pergerakan pengguna untuk menghitung jarak tempuh perjalanan. Ketika suatu komponen kendaraan telah mencapai batas maksimal penggunaan, sistem akan memberikan notifikasi pengingat servis. Aplikasi ini juga berfungsi sebagai catatan pribadi pengguna; saat notifikasi servis diterima, pengguna dapat melakukan konfirmasi servis untuk membuat data komponen kendaraan yang baru. Selain itu, sistem juga menyediakan laporan servis sepeda motor yang dapat digunakan sebagai referensi dalam mengelola keuangan servis kendaraan. Aplikasi mobile ini diuji menggunakan dua pendekatan pengujian, yaitu *black-box testing* dan *unit testing*. *Black-box testing* merupakan salah satu teknik pengujian perangkat lunak yang umum digunakan untuk menguji fungsionalitas aplikasi tanpa mengetahui struktur internal sistem. Penggunaan metode ini bertujuan untuk meminimalkan kesalahan pada keluaran yang dihasilkan, agar sesuai dengan tujuan utama dari perancangan aplikasi (Driesa & Somya, 2023). Sementara itu, *unit testing* digunakan untuk menguji bagian-bagian kecil dari kode secara terpisah (unit) guna memastikan bahwa setiap fungsi atau komponen dalam aplikasi berjalan sesuai dengan yang diharapkan. Dengan pengujian unit ini, pengembang dapat mendeteksi dan memperbaiki kesalahan lebih awal dalam proses pengembangan (Hasibuan, 2021). Hasil dari pengujian *black-box* ditampilkan pada Tabel 1 dan *unit test* Tabel 2.

Tabel 1. Pengujian Blackbox Testing

Skenario	Deskripsi	Hasil
Login dan Register	Pengguna dapat masuk dan mendaftar akun baru dengan sukses.	Berhasil
Navigasi	Pengguna dapat dengan mudah menavigasi antara halaman-halaman utama.	Berhasil
Jarak Tempuh	Pengguna dapat menerima data jarak tempuh secara realtime	Berhasil
Notifikasi	Pengguna mendapat notifikasi ketika komponen kendaraan mencapai batas	Berhasil
Riwayat Perjalanan	Pengguna dapat mengakses riwayat perjalanan yang ditampilkan menggunakan map dan polyline	Berhasil
Laporan Service	Pengguna dapat melihat laporan servis	Berhasil

Tabel 2. Pengujian Unit Testing Menggunakan Jest

Nama fungsi	Deskripsi	Hasil
calculateDistance	Menghitung jarak tempuh menggunakan algoritma Haversine	Berhasil
sendServiceNotification	Mengirim notifikasi jika komponen melebihi batas	Berhasil
getUsageTrackByUserId	Mengambil data jarak tempuh pengguna	Berhasil
addServiceLog	Menambahkan log servis ke database	Berhasil
getAllTravelHistory	Mengambil seluruh data history perjalanan pengguna	Berhasil

5. KESIMPULAN

A. Kesimpulan

Penelitian ini menghasilkan sebuah aplikasi *Reminder* servis sepeda motor berbasis *Android* yang memberikan notifikasi secara otomatis kepada pengguna berdasarkan jarak tempuh kendaraan. Sistem ini memanfaatkan sensor *GPS* dari perangkat *smartphone* dan algoritma Haversine untuk menghitung jarak antar lokasi secara real-time. Teknologi *WebSocket* diterapkan untuk menjaga komunikasi dua arah antara aplikasi dan server agar data lokasi dan jarak tempuh dapat dikirim serta diperbarui secara dinamis. Hasil pengujian menunjukkan bahwa sistem dapat berfungsi sesuai dengan tujuan dan memberikan informasi pengingat servis berdasarkan penggunaan kendaraan.

B. Saran

Untuk pengembangan lebih lanjut, disarankan agar aplikasi ini dapat terhubung langsung dengan sensor kilometer yang ada pada kendaraan bermotor. Hal ini dapat dicapai dengan memanfaatkan teknologi *Internet of Things* (IoT), seperti penggunaan mikrokontroler dan sensor digital odometer. Dengan demikian, perhitungan jarak tempuh akan menjadi lebih akurat karena diambil langsung dari kendaraan, bukan hanya estimasi dari koordinat *GPS*, yang rentan terhadap fluktuasi sinyal atau gangguan lingkungan.

6. DAFTAR PUSTAKA

rianti, T., Fa'izi, A., Adam, S., & Wulandari, M. (2022). Perancangan sistem informasi perpustakaan menggunakan diagram UML (Unified Modelling Language). *JIKTI: Jurnal Ilmiah Komputer Terapan dan*

- Informasi*, 1(1), 19–25. <https://garuda.kemdiktisaintek.go.id/documents/detail/3053203>
- Dayera, D., Toumahuw, H. A., Kusumawardana, A. R. N., Aryanto, I., & Christina, B. (2023). Sosialisasi Perawatan Kendaraan Sepeda Motor Di Kampung Malaumkarta Distrik Makbon. *Jurnal Pengabdian Masyarakat*, 1(2), 64–73.
- Driesa, Y. K., & Somya, R. (2023). Perancangan Aplikasi Service Reminder Sepeda Motor Berbasis Android Mobile. *Jurnal Teknik Informatika Dan Sistem Informasi*, 10(1), 287–297. <https://jurnal.mdp.ac.id/index.php/jatisi/article/view/2705/1139>
- Hardiansyah, Laxmi, G. F., & Kusumah, F. S. F. (2018). Perancangan Location Based Service (LBS) pada pencarian event aplikasi Sahabat Jasa berbasis Android. *Prosiding SEMNATI*, 1, 133–140. <https://prosiding.uika-bogor.ac.id/index.php/semnati/article/view/74>
- Hasibuan, A. N. (2021). Pengujian dengan Unit Testing dan Test case pada Proyek Pengembangan Modul Manajemen Pengguna. *AUTOMATA*, 2(1). <https://journal.uui.ac.id/AUTOMATA/article/view/17367>
- Hidayati, N. (2019). Penggunaan Metode Waterfall Dalam Rancang Bangun Sistem Informasi Penjualan. *Generation Journal*, 3(1), 1–10.
- Ilhadi, V., Ardiansyah, D., & Muthmainnah. (2023). Aplikasi Reminder Jadwal Kegiatan Berbasis Mobile. *Sisfo: Jurnal Ilmiah Sistem Informasi*, 6(1), 78–85.
- Ismanto, Firman, H., & Kristinanti. (2020). Pemodelan Proses Bisnis Menggunakan Business Process Modelling Notation (BPMN). *Briliant: Jurnal Riset Dan Konseptual*, 5(1), 69–76.
- Krista, M., & Bettinger, P. (2019). Smartphone GPS accuracy study in an urban environment. *PLoS ONE*, 14(7), e0219890. <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0219890>
- Kurniawan, F. (2022). *Pengembangan Aplikasi Petugas Realtime Tracking Menggunakan Formula Haversine Dan Polyline Berbasis Android* [Universitas Negeri Jakarta]. <http://repository.unj.ac.id/32885/>
- Ogundeyi, K. E., & Yinka-Banjo, C. (2019). Websocket In Real Time Application. *Nigerian Journal of Technology (NIJOTECH)*, 38(4), 1010–1020.
- Prasetya, D. A., Nguyen, P. T., Faizullin, R., Iswanto, I., & Armay, E. (2020). Resolving the Shortest Path Problem using the Haversine Algorithm. *Journal of Critical Reviews*, 7(1), 62–64.
- Rahman, Z., Aisuwarya, R., & Suwandi, R. (2023). Rancang Bangun Sistem Pelacakan pada Penyewaan Sepeda Motor Menggunakan GPS Berbasis Internet of Things. *CHIPSET*, 4(1), 47–60.
- Badan Pusat Statistik. (2022). *Perkembangan Jumlah Kendaraan Bermotor Menurut Jenis (Unit)*. <https://www.bps.go.id/id/statistics-table/2/NTcjMg==/perkembangan-jumlah-kendaraan-bermotor-menurut-jenis--unit-.html>
- Sumantri, A., Oktaviani, Z., & Kurniawan, D. (2025). Sistem Informasi Reminder Service Kendaraan Roda Empat dengan Teknologi Whatsapp Gateway Berbasis Website. *Jurnal Penelitian Rumpun Ilmu Teknik*, 4(2), 29–35.