

PENERAPAN FORMULA HAVERSINE PADA SISTEM CROWDSOURCING MOBILE UNTUK EFISIENSI PEMETAAN LOKASI BARANG HILANG

Muhammad Azhar Ibrahim¹, Joko Aryanto²

^{1,2}Universitas Teknologi Yogyakarta, Indonesia

Email: aimazhar63@gmail.com

Abstrak

Kehilangan barang di ruang publik kerap memicu krisis penumpukan informasi (information overload) pada grup percakapan konvensional akibat ketiadaan filter geografis yang memadai. Merespons urgensi tersebut, penelitian ini menggagas sistem crowdsourcing Lost and Found cerdas menggunakan pendekatan metodologi Prototyping yang dirancang khusus untuk ekosistem seluler (mobile). Arsitektur sistem dibangun di atas kerangka kerja Flutter untuk menghadirkan pengalaman antarmuka pengguna yang responsif, sementara seluruh komputasi spasial yang berat didelegasikan secara terpusat ke backend Laravel dan basis data MySQL guna menjaga efisiensi memori perangkat. Inovasi fundamental dari penelitian ini terletak pada pemanfaatan Location-Based Service (LBS) yang dipertajam oleh kalkulasi algoritma Haversine Formula pada sisi peladen, memungkinkan mesin untuk menghitung jarak lengkung bumi secara presisi dan menyaring data pelaporan secara seketika murni berdasarkan kedekatan radius aktual pengguna. Hasil pengujian membuktikan bahwa sentralisasi komputasi jarak ini tidak hanya meringankan beban pemrosesan smartphone secara drastis, tetapi juga mencatatkan tingkat presisi yang tinggi. Komputasi Haversine menghasilkan tingkat akurasi rata-rata sebesar 98,89% dengan rentang simpangan deviasi metrik yang sangat minim, yakni 0,013 km hingga 0,040 km saat dibandingkan dengan OpenStreetMap. Pada akhirnya, integrasi teknologi ini sukses menciptakan ekosistem pencarian barang yang jauh lebih terarah, efisien, dan terbebas dari paparan informasi yang tidak relevan.

Kata Kunci: *Crowdsourcing, Formula Haversine, Location-Based Service, Mobile, Prototyping.*

Abstract

Losing items in public spaces frequently triggers a crisis of information overload within conventional chat groups due to the absence of adequate geographical filters. Responding to this urgency, this study initiates a smart crowdsourcing Lost and Found system utilizing a Prototyping methodology explicitly tailored for the mobile ecosystem. The system's architecture is built upon the Flutter framework to deliver a highly responsive user interface, while all heavy spatial computations are centrally delegated to a Laravel backend and MySQL database to preserve device memory efficiency. The fundamental innovation of this research lies in the utilization of Location-Based Service (LBS) sharpened by the Haversine Formula algorithm on the server side, enabling the engine to precisely calculate the earth's curvature distance and instantaneously filter report data purely based on the user's actual proximity radius. Testing results substantiate that centralizing this spatial computation not only drastically alleviates the client smartphone's processing load but also achieves high precision. The Haversine computation produced an average accuracy rate of 98.89% with a highly minimal metric deviation ranging from 0.013 km to 0.040 km when compared to OpenStreetMap. Ultimately, the integration of these technologies successfully forges a search

ecosystem that is significantly more targeted, efficient, and free from irrelevant informational noise.

Keywords: *Crowdsourcing, Haversine Formula, Location-Based Service, Mobile, Prototyping.*

A. PENDAHULUAN

Tingginya mobilitas masyarakat di ruang publik berbanding lurus dengan peningkatan risiko kehilangan barang pribadi. Area kampus, pusat perbelanjaan, hingga fasilitas transportasi massal menjadi titik rawan insiden harian yang menuntut respons penanganan cepat. Praktik pencarian dan pelaporan barang selama ini masih sering bertumpu pada metode konvensional. Pola pengelolaan konvensional yang tidak terintegrasi ini terbukti rapuh dalam memfasilitasi pelaporan darurat, memicu proses pencarian yang tidak efisien, kerumitan koordinasi antarpihak, serta tingginya risiko kehilangan data pelaporan barang itu sendiri (Fadila & Safitri, 2026; Chan et al., 2025). Publik sangat membutuhkan ruang terpusat untuk mewadahi krisis manajemen informasi tersebut.

Memecah kebuntuan sistem konvensional menuntut pemanfaatan kekuatan kolektif komunitas melalui pendekatan *crowdsourcing*. Konsep kerumunan ini memberdayakan individu-individu di sekitar lokasi kejadian untuk secara aktif berpartisipasi sebagai agen pelapor maupun penemu barang. Aksesibilitas menjadi kunci utama agar partisipasi publik dapat terjalin maksimal tanpa hambatan fungsional. Membangun sebuah ekosistem pelaporan terpusat dalam wujud aplikasi seluler (*mobile*) memberikan tingkat mobilitas yang tinggi dan fleksibilitas akses tanpa membutuhkan infrastruktur tambahan bagi penggunanya (Pamungkas, Suratno, & A, 2025). Lingkungan seluler ini menghapus birokrasi pelaporan fisik yang selama ini sering menyurutkan niat baik seseorang saat hendak mengembalikan barang temuan di lapangan.

Ketersediaan platform pelaporan digital semata belum cukup tangguh jika informasi yang disajikan mengabaikan batasan kedekatan geografis. Menampilkan daftar barang hilang yang berjarak belasan kilometer dari posisi aktual seorang pengguna hanya mereplikasi masalah kelebihan informasi pada tingkat antarmuka yang baru. Beberapa penelitian terdahulu telah berupaya mengembangkan sistem *Lost and Found*, seperti penerapan pencocokan data otomatis berbasis *web* pada layanan transportasi public (Fadila & Safitri, 2026). Sayangnya, pengembangan tersebut masih berfokus pada digitalisasi administratif semata dan mengabaikan parameter proksimitas spasial. Di sisi lain, integrasi *Location-Based Service* (LBS) dan algoritma komputasi jarak telah banyak dibuktikan keandalannya dalam menyaring radius spasial. (Desmile et al., 2021) membuktikan efisiensi navigasi LBS pada ekosistem seluler Android, sementara (Prihantoro & Wahyuddin, 2022) mencatat tingkat deviasi metrik jarak yang sangat minim (dengan rata-rata selisih 0,038 km) pada penerapan Haversine Formula untuk pencarian lokasi cerdas berbasis Android. Efektivitas komputasi koordinat ini menjadi sangat optimal ketika dikolaborasikan dengan antarmuka aplikasi seluler untuk memproses dan menampilkan urutan lokasi terdekat secara seketika berdasarkan posisi aktual penggunanya (Hidayati & Mutiah, 2022). Perhitungan matematis ini mengkalkulasi jarak terpendek dengan memperhitungkan derajat kelengkungan permukaan sferis bumi (Adil, Dwiputri, & Triwijoyo, 2022). Dengan pemrosesan tersebut, aplikasi dimungkinkan untuk memangkas beban memori perangkat dengan hanya menyaring dan menayangkan laporan yang murni berada pada radius terdekat dari pengguna (Alma'aarij & Sancoko, 2026). Sayangnya, mayoritas implementasi LBS dan Formula Haversine terdahulu masih terisolasi pada pencarian entitas statis (seperti bengkel, fasilitas umum, atau kedai kopi). Eksplorasi implementasi Haversine yang diuji pada objek hilangnya barang yang bersifat dinamis dan membutuhkan pembaruan data spasial dari kontribusi massa (*crowdsourcing*) masih sangat terbatas. Berpijak

pada urgensi dan celah arsitektur tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan *Location-Based Service* beserta kalkulasi Haversine Formula pada sistem *crowdsourcing Lost and Found* berbasis *mobile*. Tujuan spesifik dari pengembangan ini adalah memvalidasi tingkat keakuratan penyaringan jarak spasial pada sisi peladen untuk mencegah disorientasi data visual pengguna. Kontribusi utama dari penelitian ini mencakup reduksi signifikan terhadap beban penumpukan informasi dalam pencarian barang, efisiensi kinerja memori komputasi perangkat, serta terciptanya sebuah standar baru dalam pengembangan sistem informasi layanan publik yang lebih tertarget, presisi, dan saling menguntungkan.

B. TINJAUAN PUSTAKA

1. Konsep *Crowdsourcing*

Crowdsourcing merupakan metode pelibatan sekelompok orang atau masyarakat luas (*crowd*) untuk secara aktif berkontribusi dalam suatu proses pengumpulan data atau penyelesaian masalah melalui infrastruktur teknologi informasi (Rambe & Ikhwan, 2026). Dalam konteks sistem pelaporan barang hilang (*Lost and Found*), pendekatan ini mengadopsi aktivitas daring yang partisipatif, di mana masyarakat tidak sekadar menjadi konsumen informasi. Pengguna secara sukarela saling berpartisipasi dan berkontribusi secara massal melalui komunitas untuk membagikan data pelaporan sekaligus membantu proses validasi temuan di lapangan (Harahap & Wijayanti, 2022). Pentingnya partisipasi publik dalam ekosistem ini terletak pada skalabilitas dan kecepatan validasi data. Laporan kehilangan atau penemuan barang yang diunggah oleh satu individu dapat divalidasi silang oleh individu lain di sekitarnya secara *real-time*. Partisipasi aktif kerumunan ini meminimalisir *blind-spot* pelaporan dan meningkatkan probabilitas pencocokan data (*data matching*), yang mana tingkat keberhasilannya berbanding lurus dengan densitas pengguna aktif di suatu wilayah.

2. *Location-Based Service* (LBS)

Untuk mengikat data *crowdsourcing* ke dalam konteks spasial yang valid, sistem memerlukan teknologi *Location-Based Service* (LBS). LBS adalah layanan informasi dinamis yang dapat diakses melalui perangkat bergerak dengan memanfaatkan kapabilitas geolokasi untuk menyajikan data berdasarkan posisi geografis pengguna secara actual (Desmille et al., 2021). rsitektur LBS beroperasi melalui integrasi empat komponen fundamental: (1) Mobile Device, yakni perangkat antarmuka keras dan lunak di sisi pengguna yang menangkap titik koordinat; (2) Communication Network, yang berfungsi sebagai jembatan transmisi data spasial (seperti jaringan seluler atau Wi-Fi); (3) Positioning Component, yaitu modul satelit navigasi (seperti GPS atau GLONASS) yang mengkalkulasi lokasi aktual; dan (4) Service Provider, yakni peladen (server) yang memproses koordinat tersebut menjadi layanan atau informasi yang relevan (Desta, 2022). Pada aplikasi pelacakan seluler (*mobile tracking*), LBS memegang peranan esensial dalam mengubah titik koordinat mentah menjadi intelijen lokasi. Data ini krusial untuk memetakan distribusi laporan barang hilang sehingga relevansi informasi dapat dibatasi secara keruangan, mencegah terjadinya kelebihan informasi (*information overload*) bagi pengguna (Alma'aarij & Sancoko, 2026).

3. Formula Haversine

Pengolahan data spasial yang dihasilkan oleh LBS membutuhkan algoritma pengukuran jarak yang presisi. Dalam komputasi keruangan, Formula Haversine merupakan metode yang sangat vital untuk mengetahui jarak antar dua titik dengan memperhitungkan derajat kelengkungan bumi berdasarkan garis bujur dan lintang (Adil et al., 2022). Algoritma ini secara spesifik beroperasi dengan mengambil dua titik koordinat pada kelengkungan permukaan bumi untuk menghasilkan jarak lingkaran besar yang aktual (Prihantoro & Wahyuddin, 2022).

Secara komparatif, algoritma ini terbukti memiliki nilai akurasi pencarian jarak spasial yang lebih unggul dibandingkan dengan pendekatan garis lurus biasa seperti metode *Euclidean Distance* (Rahayu, Fanni, & Bima, 2022).

Secara matematis, perhitungan ini menggunakan titik koordinat lintang (*latitude*) dan bujur (*longitude*) sebagai variabel input. Representasi rumus Haversine dikalkulasi dengan mengonversi nilai derajat koordinat menjadi radian terlebih dahulu, yang dapat diimplementasikan secara presisi ke dalam fungsi kode pemrograman pada aplikasi seluler (Hasanah, Carudin, & Purnamasari, 2023). Rumus matematis dari algoritma ini dapat dituliskan dengan persamaan sebagai berikut (Adil et al., 2022):

$$x = (\text{lon2} - \text{lon1}) * \cos\left(\frac{\text{lat1} + \text{lat2}}{2}\right) \quad (1)$$

$$y = (\text{lat2} - \text{lat1}) \quad (2)$$

$$d = \sqrt{(x * x + y * y)} * R \quad (3)$$

Keterangan:

lat1 = *latitude* atau garis lintang tempat pertama

lon1 = *longitude* atau garis bujur tempat pertama

lat2 = *latitude* atau garis lintang tempat kedua

lon2 = *longitude* atau garis bujur tempat kedua

x = *longitude* (bujur)

y = *latitude* (lintang)

d = Jarak

R = Radius Bumi (6.371 km)

1 derajat = 0,0174532925 Radian

4. *Mobile Development* dan Geolokasi

Integrasi dari keseluruhan konsep di atas bermuara pada lapisan antarmuka pengguna melalui pengembangan perangkat lunak seluler (*mobile development*). Integrasi teknologi GPS pada perangkat seluler memberikan mobilitas tinggi untuk memverifikasi dan mengekstraksi titik koordinat pengguna secara *real-time* (Pamungkas et al., 2025). Pada tingkat arsitektur perangkat lunak lintas platform (*cross-platform*), data lintang dan bujur yang diekstraksi dari sensor perangkat (*hardware layer*) diteruskan menuju antarmuka pemrograman aplikasi (API) pemetaan guna memproses pencarian dan pengurutan lokasi secara otomatis (Hidayati & Mutiah, 2022). Layanan API ini menerjemahkan deretan angka desimal geolokasi menjadi representasi visual peta yang interaktif dan dapat dipahami oleh manusia (seperti peletakan penanda dan *geofencing*). Dengan demikian, integrasi komprehensif antara GPS pada *smartphone*, visualisasi Maps API, dan logika kalkulasi Haversine inilah yang kelak menjadi landasan arsitektural bagi sistem untuk menayangkan data pelaporan secara *real-time*, relevan, dan terukur

C. METODE

Penelitian ini mengadaptasi tahapan *System Development Life Cycle* (SDLC) dengan menggunakan model pengembangan *Prototyping*. Pemilihan metode ini didasarkan pada karakteristik pengembangan perangkat lunak seluler berbasis *Location-Based Service* (LBS) yang menuntut pengujian teknis secara iteratif terhadap ekstraksi sensor GPS dan kalkulasi jarak algoritma. Tahapan pelaksanaan perancangan sistem pelaporan barang hilang (*Lost and Found*) diuraikan sebagai berikut:

1. Analisis Kebutuhan (*Requirement Gathering*)

Fokus utama pada tahap ini adalah mengidentifikasi spesifikasi fungsional untuk mengatasi masalah penumpukan informasi pelaporan tanpa filter geografis. Sistem dianalisis

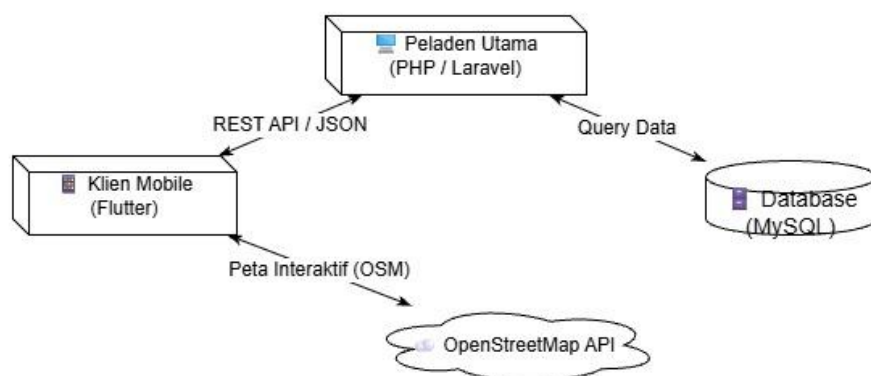
untuk dapat mengekstraksi titik koordinat satelit (lintang dan bujur) dari perangkat keras secara otomatis saat pelaporan dilakukan, serta kebutuhan integrasi basis data untuk menyimpan parameter keruangan tersebut.

2. Perancangan Sistem (*Quick Design*)

Proses penyusunan cetak biru perangkat lunak secara cepat agar logika aplikasi dapat segera divisualisasikan dan dievaluasi. Untuk memberikan gambaran teknis yang komprehensif, perancangan sistem ini dibagi menjadi tiga model visual utama:

a. Arsitektur Sistem

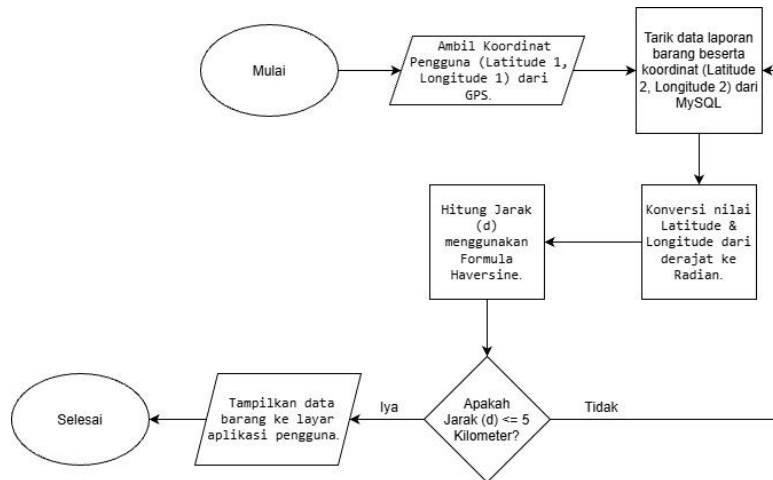
Arsitektur sistem ini dikembangkan dengan mengadopsi model infrastruktur klien-peladen (*client-server*). Pada lapisan antarmuka pengguna di sisi klien (*front-end*), sistem dibangun menggunakan kerangka kerja Flutter (Mufid NR & Rizaldy, 2025) untuk memastikan proses ekstraksi koordinat lokasi dari sensor perangkat keras seluler dapat berjalan secara responsif. Selanjutnya, aliran data spasial tersebut ditransmisikan menuju peladen sentral (*back-end*) yang dikembangkan melalui bahasa pemrograman PHP dan kerangka kerja Laravel (Arianto & Susetyo, 2022). Sentralisasi pada peladen ini berfungsi sebagai inti komputasi logika keruangan, yang kemudian diintegrasikan secara langsung dengan antarmuka pemrograman aplikasi (API) OpenStreetMap (OSM) (Saputro, 2026). Pemanfaatan platform data spasial *open-source* ini dinilai krusial guna menghasilkan visualisasi titik geolokasi secara interaktif, efisien, dan terbebas dari hambatan lisensi komersial..



Gambar 1. Arsitektur Sistem

b. Diagram Alir (*Flowchart*) Algoritma Haversine

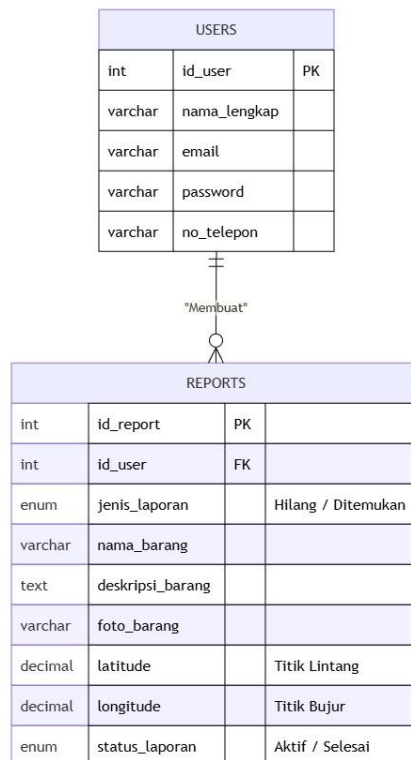
Flowchart ini memvisualisasikan alur kerja komputasi pada peladen dalam menyaring informasi keruangan. Proses dimulai saat peladen menerima titik koordinat pelapor aktual. Sistem kemudian memanggil daftar laporan barang dari basis data, mengonversi nilai derajat koordinat menjadi radian, lalu menghitungnya menggunakan rumus trigonometri Formula Haversine. Algoritma akan melakukan validasi; jika hasil kalkulasi jarak linier berada dalam radius ≤ 5 kilometer, data akan dikirim ke layar pengguna. Jika di luar radius, data diabaikan.



Gambar 2. Diagram Alir (Flowchart) Algoritma Haversine

c. *Entity Relationship Diagram (ERD)*

Fondasi penyimpanan data relasional pada sistem ini dibangun menggunakan sistem manajemen basis data MySQL guna memastikan efisiensi dan optimasi penyimpanan secara fisik maupun logis (Šušter & Ranisavljević, 2023). Skema *Entity Relationship Diagram (ERD)* dirancang untuk memetakan hubungan fungsional antara entitas primer USERS dan entitas sekunder REPORTS melalui relasi kardinalitas *One-to-Many* (di mana satu pengguna memiliki hak akses untuk membuat banyak laporan). Modifikasi esensial pada perancangan basis data ini terletak pada penambahan atribut keruangan di dalam tabel REPORTS, yaitu kolom *latitude* dan *longitude* yang menggunakan tipe data desimal. Kehadiran atribut spasial ini menjadi prasyarat mutlak bagi peladen agar dapat mengeksekusi komputasi jarak lengkung sferis bumi secara presisi.



Gambar 3. Entity Relationship Diagram (ERD) Pangkalan Data

3. Pembangunan Prototipe (*Build Prototype*)

Transformasi rancangan arsitektur ke dalam wujud perangkat lunak. Sisi antarmuka klien (*front-end*) dibangun menggunakan kerangka kerja Flutter untuk memastikan responsivitas interaksi pengguna pada ekosistem seluler. Seluruh logika algoritma pencarian jarak dieksekusi secara terpusat pada sisi peladen (*back-end*) menggunakan bahasa pemrograman PHP dengan kerangka kerja Laravel dan sistem manajemen basis data MySQL.

4. Evaluasi Prototipe (*System Evaluation*)

Purwarupa sistem diuji secara fungsional melalui simulasi pergerakan (*independent moving test*) untuk mengevaluasi responsivitas sensor GPS secara *real-time*. Evaluasi teknis difokuskan pada pengujian algoritma Haversine dengan menyimulasikan pergerakan koordinat perangkat aktual menjauhi atau mendekati titik pelaporan fiktif di pangkalan data. Iterasi kode program dan penyesuaian sinkronisasi API Pemetaan dilakukan pada tahap ini hingga kalkulasi radius pada peladen terbukti konsisten dengan pergerakan di lapangan.

5. Implementasi dan Pengujian Akhir (*Testing*)

Sistem akhir dievaluasi melalui dua instrumen validasi utama. Pengujian pertama menggunakan *Black Box Testing* untuk memverifikasi fungsionalitas seluruh antarmuka dan modul aplikasi tanpa membedah struktur kode sumber. Pengujian kedua difokuskan pada validasi akurasi spasial, di mana nilai metrik jarak (dalam kilometer atau meter) hasil komputasi Formula Haversine pada sistem dibandingkan langsung dengan hasil pengukuran linier dari instrumen navigasi standar Open Street Map (OSM) untuk menghitung persentase presisi algoritma.

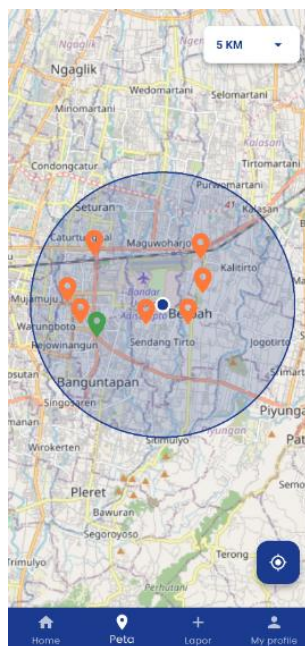
D. HASIL DAN PEMBAHASAN

1. Implementasi Antarmuka Sistem

Implementasi sisi klien berwujud aplikasi seluler berbasis Android yang dikembangkan menggunakan framework Flutter. Setelah pengguna melewati tahap autentikasi (*login* atau registrasi) guna memvalidasi identitas pelapor ke dalam basis data, fokus utama pada antarmuka sistem ini langsung diarahkan pada visualisasi *Location-Based Service* (LBS) dan ekstraksi koordinat pengguna.

a. Implementasi Visualisasi Peta dan Radius LBS

Halaman peta merupakan representasi visual dari titik koordinat keruangan. Sistem berhasil memvisualisasikan data spasial menggunakan peta interaktif dengan indikator radius jangkauan berbentuk lingkaran biru. Pengguna dapat memilih filter batas jarak (misalnya 5 KM) melalui menu *dropdown* di sudut kanan atas. Titik pusat berwarna biru menunjukkan lokasi aktual pengguna (GPS perangkat), sementara *pin* berwarna oranye dan hijau menunjukkan distribusi laporan barang yang berada di dalam radius proksimitas tersebut.



Gambar 4. Antarmuka Peta Interaktif dan Filter Radius LBS

b. Implementasi Halaman Beranda (Daftar Penemuan)

Halaman ini berfungsi menayangkan daftar kurasi laporan barang. Bukti keberhasilan komputasi peladen terlihat pada penayangan teks estimasi jarak yang bersifat dinamis pada setiap kartu laporan, seperti "Estimasi 687 meter" untuk laporan jam Garmin dan "Estimasi 997 meter" untuk dompet kulit. Hal ini membuktikan bahwa algoritma Haversine sukses menyaring dan menghitung jarak objek secara *real-time*.



Gambar 5. Antarmuka Beranda dan Estimasi Jarak

c. Implementasi Form Pelaporan Barang

Halaman ini membuktikan keberhasilan sistem dalam menangkap titik geolokasi perangkat keras pengguna. Pada kolom "Lokasi Barang", nilai *Latitude* (-7.80251438) dan *Longitude* (110.43329090) berhasil dieksekusi dan terisi secara otomatis mengikuti pergeseran penanda (*pin*) pada peta.

Gambar 6. Antarmuka Ekstraksi Koordinat Pelaporan

2. Pengujian Fungsionalitas Sistem

Tahap pengujian fungsionalitas ini dilaksanakan menggunakan pendekatan *Black Box Testing* dengan mengadopsi teknik *Equivalence Partitions* (Amalia, Wanda, Hamidah, & Kristanto, 2021) untuk mengevaluasi kinerja fitur pada aplikasi *Lost and Found*. Pengujian ini bertujuan untuk memastikan bahwa sistem mampu merespons setiap interaksi pengguna sesuai dengan skenario perancangan tanpa perlu membedah struktur kode sumbernya. Untuk menyimulasikan penggunaan secara nyata pada ekosistem seluler, proses pengujian perangkat lunak ini dijalankan secara langsung pada *smartphone* dengan sistem operasi Android 13. Rincian dari serangkaian skenario pengujian beserta hasilnya dapat dilihat pada Tabel 1 berikut.

Tabel 1. Pengujian Fungsionalitas Sistem

No	Kondisi Awal	Hasil Harapan
1	User mengisi form registrasi dan menekan tombol daftar	Menyimpan data pengguna baru di database dan menampilkan notifikasi berhasil
2	User memasukkan email dan password lalu menekan tombol masuk	Memvalidasi kredensial akun dan mengarahkan user ke halaman beranda
3	User membuka halaman lapor barang dan menggeser penanda pada peta	Menampilkan dan mengisi otomatis titik koordinat latitude dan longitude sesuai lokasi penanda
4	User menekan tombol simpan laporan dengan form terisi lengkap	Menyimpan data laporan barang ke database dan menampilkan notifikasi berhasil
5	User memilih batas jarak pada menu filter radius di halaman peta	Menampilkan penanda lokasi barang yang hanya berada di dalam radius lingkaran yang dipilih
6	User membuka halaman beranda (daftar penemuan di sekitar)	Mengurutkan dan menampilkan daftar barang beserta estimasi jarak dari lokasi user saat ini
7	User memilih atau menekan salah satu kartu laporan barang	Menampilkan halaman detail yang berisi informasi lengkap dan lokasi spesifik barang

Merujuk pada pemaparan hasil pengujian di atas, dapat ditarik kesimpulan bahwa keseluruhan komponen antarmuka dan fungsionalitas aplikasi *Lost and Found* mampu beroperasi secara optimal pada lingkungan sistem operasi Android 13. Seluruh skenario pengujian dinyatakan valid dan sukses merespons masukan pengguna tanpa adanya kegagalan sistem (*error*). Namun demikian, perlu digarisbawahi bahwa kelancaran beberapa fitur inti

terutama pada saat penarikan titik koordinat pelaporan dan penyaringan radius peta LBS sangat bergantung pada satu prasyarat teknis mutlak. Pengguna diwajibkan untuk memberikan izin akses fitur lokasi (GPS) pada pengaturan perangkat seluler mereka, karena tanpa akses tersebut, algoritma komputasi spasial pada peladen tidak akan dapat mengeksekusi perhitungan jarak.

3. Hasil Pengujian Akurasi Algoritma Haversine

Pengujian akurasi ini merupakan tahapan krusial untuk memvalidasi tingkat presisi komputasi pada sisi peladen (*server*). Evaluasi dilakukan dengan membandingkan nilai jarak hasil perhitungan algoritma Haversine di dalam sistem terhadap metrik pengukuran linier dari instrumen navigasi OpenStreetMap (OSM). Proses validasi ini menggunakan sepuluh sampel data koordinat pelaporan riil (*latitude* dan *longitude*) yang diekstraksi secara langsung dari basis data MySQL. Tujuan utama dari komparasi ini adalah untuk memastikan bahwa rumus trigonometri yang diimplementasikan mampu mengkompensasi kelengkungan sferis bumi secara presisi, sehingga penyaringan radius *Location-Based Service* (LBS) dapat mereduksi penumpukan informasi secara optimal. Rincian hasil komputasi dan perbandingan jarak dari kesepuluh titik sampel pengujian tersebut dijabarkan lebih lanjut pada Tabel 2.

Tabel 2. Hasil perhitungan jarak menggunakan algoritma Haversine

No	Nama Barang	Latitude	Longitude	Jarak Haversine (km)	OpenStreetMap (km)	Selisih (km)
1	Jam garmin	-7.80490319	11.042753878	0.68	0.70	0.013
2	Dompot kulit	-7.80477999	11.044204952	0.99	1.01	0.013
3	Flash disk	-7.79428126	11.044714865	1.78	1.80	0.020
4	TWS	-7.80931877	11.041054227	2.61	2.65	0.033
5	Kunci motor	-7.78251563	11.044639450	2.65	2.68	0.029
6	Kacamata	-7.77650800	11.042853120	2.93	2.96	0.022
7	Sepeda fixie	-7.80469138	11.040486159	3.14	3.17	0.029
8	Tas asus	-7.78139776	11.040961421	3.50	3.54	0.031
9	Handphone	-7.78139776	11.040961421	3.50	3.54	0.031
10	Power bank	-7.79744309	11.040037109	3.67	3.71	0.040

Mengacu pada hasil komputasi sepuluh titik lokasi riil barang hilang, algoritma Haversine menunjukkan presisi yang sangat tangguh dalam menghitung jarak keruangan jika dikomparasikan dengan OpenStreetMap. Rentang selisih perhitungan yang dihasilkan tergolong sangat sempit, yakni berkisar dari simpangan terkecil sebesar 0,013 km hingga simpangan maksimal sebesar 0,040 km. Perolehan deviasi tersebut bermuara pada tingginya persentase tingkat akurasi rata-rata keseluruhan yang menyentuh angka 98,89%.

Tabel 3. Tingkat Akurasi Perhitungan Haversine

No	Nama Barang	Jarak Haversine (km)	OpenStreetMap (km)	Akurasi (%)
1	Jam garmin	0.68	0.70	98.14
2	Dompet kulit	0.99	1.01	98.71
3	Flash disk	1.78	1.80	98.88
4	TWS	2.61	2.65	98.75
5	Kunci motor supra	2.65	2.68	98.91
6	Kacamata	2.93	2.96	99.25
7	Sepeda fixie	3.14	3.17	99.08
8	Tas asus	3.50	3.54	99.12
9	Handphone	3.50	3.54	99.12
10	Power bank	3.67	3.71	98.92
Rata-Rata				98.89

Berdasarkan sepuluh sampel data riil pelaporan barang yang diekstraksi dari basis data MySQL pada Tabel 1 dan Tabel 2, komputasi algoritma Haversine pada peladen mencatatkan performa presisi yang sangat konsisten dengan kondisi geospasial aktual. Sistem mampu merespons titik koordinat dengan margin *error* yang sangat kecil (berkisar antara 13 hingga 40 meter), menghasilkan persentase akurasi rata-rata keseluruhan sebesar 98,89%. Tingginya nilai akurasi ini membuktikan bahwa kelengkungan sferis bumi telah dikompensasi dengan sangat baik.

E. KESIMPULAN

Berdasarkan hasil analisis dan pengujian sistem yang telah dilakukan, dapat disimpulkan bahwa penerapan *Location-Based Service* (LBS) dan algoritma *Haversine* pada aplikasi *Lost and Found* terbukti efektif dalam menjawab permasalahan penumpukan informasi pelaporan barang. Sistem secara presisi mampu mengekstraksi titik koordinat pengguna secara *real-time* dan memfilter tayangan kurasi data *crowdsourcing* murni berdasarkan kedekatan radius geografis. Secara komputasional, algoritma *Haversine* mencatatkan tingkat akurasi rata-rata yang sangat tinggi sebesar 98,89% saat dikomparasikan dengan metrik ukur OpenStreetMap. Rentang simpangan deviasi yang dihasilkan sangat minim, yakni berada pada angka 0,013 km hingga 0,040 km. Hal ini membuktikan bahwa rumusan matematis pada peladen mampu mengkompensasi faktor kelengkungan sferis bumi secara andal untuk menyaring laporan barang di sekitar pengguna secara presisi.

Meskipun sistem telah beroperasi secara optimal, penelitian ini masih memiliki keterbatasan. Keterbatasan utama terletak pada karakteristik perhitungan algoritma *Haversine* yang mengkalkulasi proksimitas berdasarkan jarak lurus udara (*great-circle distance*), sehingga estimasi jarak yang ditampilkan belum mempertimbangkan topografi wilayah maupun rute jalan raya yang sebenarnya (*routing distance*). Selain itu, sistem belum dilengkapi dengan algoritma kompresi citra tingkat lanjut pada sisi klien, sehingga unggahan foto barang dengan resolusi tinggi berpotensi membebani kapasitas penyimpanan peladen seiring bertambahnya volume data pelaporan. Keterbatasan teknis ini dapat menjadi pijakan evaluasi untuk pengembangan sistem aplikasi pada penelitian selanjutnya.

DAFTAR PUSTAKA

- Adil, A., Dwiputri, R. A., & Triwijoyo, B. K. (2022). *Aplikasi Spasial Rekomendasi Wisata Terdekat dengan Metode Haversine Berbasis Mobile The Nearest Tourism Recommendation Application with The Mobile-Based Haversine Method*. 4(1), 95–106. <https://doi.org/10.30812/bite.v4i1.1948>
- Alma'aarij, M. A., & Sancoko, S. D. (2026). *Optimalisasi Pencarian Coffeeshop Terdekat*

- Menggunakan Algoritma Haversine pada Sistem Mobile Recommendation*. 09(2), 206–212. <https://doi.org/10.36085/jsai.v9i2.10479>
- Amalia, A., Wanda, S., Hamidah, P., & Kristanto, T. (2021). *Pengujian Black Box Menggunakan Teknik Equivalence Partitions Pada Aplikasi E-Learning Berbasis Web*. 3(3), 269–274. <https://doi.org/10.47065/bits.v3i3.1062>
- Arianto, O. D., & Susetyo, Y. A. (2022). *Penerapan restful web service dengan framework laravel untuk pembangunan sistem informasi manajemen sumber daya manusia*. 07, 522–532. <https://doi.org/10.29100/jipi.v7i2.2870>
- Chan, S. C. S., Chea, S. C. S., & Heng, D. H. D. (2025). Power Grid Protection: Quantitative Engineering of Dependability, Security, and Cascading Risk Under Modern Operating Conditions. *Techno: Journal of Engineering, Technology and Industrial Applications*, 1(2), 101–115.
- Desmile, J., Orisa, M., & Wahyuni, F. S. (2021). *Aplikasi Pencarian Bengkel Sepeda Motor Menggunakan Location Based Service pada Wilayah Kota Tebing Tinggi Berbasis Android*. 5(1). Retrieved from 10.36040/jati.v5i1.3235
- Desta. (2022). *Literatur Review : Penerapan Teknologi Location-Based Services dalam Mobile Application Berbasis Android*. (1), 1–13.
- Fadila, A., & Safitri, N. (2026). *Sistem Informasi Lost and Found Barang Berbasis Web di PT Kereta Commuter Indonesia*. 11(1), 76–87. <https://doi.org/10.51211/imbi.v11i1.3948>
- Harahap, A. R., & Wijayanti, L. (2022). *Penerapan Crowdsourcing dalam Interaksi Komunitas Informasi di Indonesia : Systematic Literature Review*. 8(2), 95–108. <https://doi.org/10.14710/lenpust.v8i2.46649>
- Hasanah, M., Carudin, & Purnamasari, I. (2023). *Implementasi Algoritma Haversine Pada Fitur Geolocation Aplikasi Pencarian Terapis Baby Spa*. 9(15), 376–390. <https://doi.org/10.5281/zenodo.8213622>
- Hidayati, R., & Mutiah, N. (2022). *Penerapan Metode Haversine Formula Pada Pencarian Lokasi Fasilitas Kesehatan Terdekat*. 6, 278–286. <https://doi.org/10.30865/mib.v6i1.3445>
- Mufid NR, N. F., & Rizaldy, A. (2025). *Analisis Perbandingan antara Framework Flutter Dengan Native*. 3(1), 33–41. <https://doi.org/10.33772/anoatik.v3i1.103>
- Pamungkas, K., Suratno, T., & A, M. R. (2025). *Perancangan Sistem Presensi Berbasis GPS Pada Layanan Kesehatan Menggunakan Metode Prototype*. 15(2), 221–228. <https://doi.org/10.37859/jf.v15i2.9316>
- Prihantoro, D. D., & Wahyuddin, M. I. (2022). *Implementasi Algoritma Haversine Formula dan Location Based Service Pada Aplikasi Pencarian Lokasi Bird Contest Berbasis Android*. *Jurnal Media Informatika Budidarma*, 6(1), 663. <https://doi.org/10.30865/mib.v6i1.3546>
- Rahayu, S., Fanni, R., & Bima, K. (2022). *Perbandingan Haversine Formula dan Euclidean Distance dalam Pencarian Jarak Terdekat Rumah Penampungan Hewan (Shelter)*. XIV(1), 25–34. <https://doi.org/10.22441/fifo.2022.v14i1.003>
- Rambe, F. R. A., & Ikhwan, A. (2026). *Penerapan Metode Crowdsourcing pada Sistem Informasi Pelaporkan dan Verifikasi Data Penerima Bantuan Sosial: Studi Kasus Dinas Sosial Kabupaten*. 4(7). <https://doi.org/10.62281/8gbrcd29>
- Saputro, H. (2026). *Jurnal Informatika dan Komputer (JIK)*. 17(2), 41–55. Retrieved from <https://journal.unmaha.ac.id/index.php/jik/article/view/764>
- Šušter, I., & Ranisavljević, T. (2023). *OPTIMIZATION OF MYSQL DATABASE*. 11(1), 141–151. <https://doi.org/10.5937/jouproman2301141Q>