JURNAL ILMIAH SISTEM INFORMASI (JUSI)
VOL 5, No. 3, November 2026, pp. 175-194
p-ISSN : 2803-1507 (media cetak) | e-ISSN : 2803-1531 (media online)
Link Publikasi: https://ejurnal.provisi.ac.id/index.php/JUISI/issue/view/175

Implementation of Haversine, Mock Location Detection, and Face Recognition in an Integrated Attendance and Payroll Information System

Syed M Hafiz Firdaus*[1], Robby Maududy[2],
[1,2] Fakultas Sains dan Teknologi, Universitas Cipasung
Jl. Borolong , 46466 Kabupaten Tasikmalaya, Jawa Barat, Indonesia
e-mail*[1]: afizmuhammad132@gmail.com

ARTICLE INFO

Article History:

Received: 15-05-2026
Revised: 15-06-2026
Accepted: 21--07-2026
Published: 28-10-2026

ABSTRACT

The advancement of information technology has encouraged organizations to adopt digitally integrated systems in managing human resources. However, several companies still implement manual attendance and payroll processes that are vulnerable to recording errors, inefficiency, and data manipulation. At PT Munggaran Jaya Utama, employee attendance is conducted through photo-based reporting and manually recapitulated using spreadsheet applications, resulting in potential inaccuracies in attendance and payroll calculations. This research aims to implement the Haversine Algorithm, Mock Location Detection, and Face Recognition within a web and mobile-based attendance system to improve accuracy, security, and operational efficiency. This study employed the Research and Development (R&D) method with the Agile Scrum framework. Data collection techniques included observation, interviews, documentation, and literature studies. The Haversine Algorithm was utilized to calculate the distance between employee coordinates and attendance locations. Mock Location Detection was applied to identify GPS manipulation attempts, while Face Recognition was implemented to verify employee identity through selfie authentication during attendance. The results indicate that the proposed system improves attendance validation accuracy, minimizes fraudulent activities such as fake GPS and proxy attendance, and enhances payroll data processing efficiency. Furthermore, the system supports real-time attendance monitoring and integrated data management.

Keywords: *Haversine Algorithm, Mock Location Detection, Face Recognition, Employee Attendance System, Payroll System.*

Abstrak

Perkembangan teknologi informasi yang semakin pesat mendorong berbagai organisasi untuk menerapkan sistem digital terintegrasi dalam pengelolaan sumber daya manusia. Meskipun demikian, masih terdapat perusahaan yang menggunakan proses presensi dan penggajian secara manual sehingga berisiko menimbulkan kesalahan pencatatan, ketidakefisienan kerja, serta manipulasi data. Pada PT Munggaran Jaya Utama, proses presensi karyawan dilakukan melalui dokumentasi foto dan direkap secara manual menggunakan aplikasi spreadsheet, sehingga berpotensi menyebabkan ketidakakuratan data presensi dan penggajian. Penelitian ini bertujuan untuk mengimplementasikan algoritma *Haversine*, *Mock Location Detection*, dan *Face Recognition* pada sistem presensi karyawan berbasis web dan *mobile* guna meningkatkan akurasi, keamanan, dan efisiensi operasional sistem. Penelitian ini menggunakan metode *Research and Development* (R&D) dengan pendekatan *Agile Scrum* sebagai metode pengembangan sistem. Teknik pengumpulan data dilakukan melalui observasi, wawancara, dokumentasi, dan studi literatur. Algoritma

DOI: https://doi.org/10.51903/7tpfp877

p-ISSN : 2803-1507 e-ISSN : 2803-1531

Haversine diterapkan untuk menghitung jarak antara koordinat pengguna dengan lokasi presensi yang telah ditentukan. *Mock Location Detection* digunakan untuk mendeteksi indikasi manipulasi lokasi *GPS*, sedangkan *Face Recognition* dimanfaatkan untuk melakukan verifikasi identitas pengguna melalui autentikasi berbasis swafoto saat proses presensi berlangsung. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan mampu meningkatkan akurasi validasi presensi, meminimalkan potensi kecurangan seperti penggunaan *fake GPS* dan titip absen, serta meningkatkan efisiensi pengolahan data penggajian. Selain itu, sistem mendukung pemantauan presensi secara *real-time* dan pengelolaan data yang terintegrasi.

Kata Kunci: Algoritma *Haversine*, Deteksi Lokasi Palsu, Pengenalan Wajah, Sistem Presensi Karyawan, Sistem Penggajian.

1. PENDAHULUAN

Perkembangan teknologi digital telah membawa perubahan dalam berbagai aspek pengelolaan perusahaan, khususnya pada bidang administrasi sumber daya manusia. Pemanfaatan teknologi informasi dilakukan untuk mendukung proses pengolahan data agar menjadi lebih sistematis, efektif, dan akurat. Namun demikian, masih terdapat perusahaan yang menjalankan proses presensi dan penggajian secara konvensional, baik melalui pencatatan manual maupun penggunaan aplikasi yang belum saling terhubung. Keadaan tersebut dapat menimbulkan sejumlah kendala, seperti terjadinya kesalahan input data, lambatnya proses pengolahan informasi, serta kurang optimalnya ketepatan perhitungan penggajian karyawan [1]. Oleh karena itu, penggunaan sistem informasi terintegrasi dinilai mampu membantu perusahaan dalam meningkatkan kualitas pengelolaan data sehingga proses operasional dapat berjalan lebih efisien dan terorganisasi dengan baik [1], [2].

Permasalahan serupa ditemukan pada PT Munggaran Jaya Utama, di mana proses presensi dan penggajian karyawan masih dilakukan secara manual dan sebagian menggunakan bantuan aplikasi sederhana. Berdasarkan hasil observasi dan wawancara yang dilakukan, pencatatan kehadiran karyawan dilakukan melalui pengiriman foto dari lokasi kerja maupun kantor, kemudian data tersebut direkap kembali secara manual menggunakan *Microsoft Excel*. Mekanisme tersebut dinilai kurang efektif karena proses penginputan data membutuhkan waktu yang cukup lama, terutama ketika jumlah data karyawan terus bertambah. Selain itu, sistem yang belum terintegrasi juga berpotensi menyebabkan berbagai permasalahan, seperti ketidaksesuaian data, keterlambatan proses rekapitulasi, serta meningkatnya kemungkinan terjadinya manipulasi presensi. Penelitian yang dilakukan oleh Zahroh et al. [3], menjelaskan bahwa penggunaan metode absensi manual dan pengolahan data berbasis *Microsoft Excel* masih memiliki kelemahan, terutama dari sisi efisiensi waktu, tingkat akurasi, dan potensi terjadinya kesalahan dalam pengelolaan data.

Permasalahan lain yang sering ditemukan dalam sistem presensi ialah manipulasi lokasi menggunakan *fake GPS*, praktik titip absen, serta kualitas dokumentasi kehadiran yang kurang baik, seperti foto yang buram atau tidak jelas. Kondisi tersebut dapat menghambat proses validasi kehadiran dan menurunkan tingkat akurasi data presensi maupun penggajian. Beberapa penelitian terdahulu telah mengembangkan sistem presensi berbasis teknologi, seperti *QR Code*, *barcode*, dan *Radio Frequency Identification* (RFID), untuk meningkatkan efisiensi pengelolaan data. Akan tetapi, sistem berbasis *QR Code* masih bergantung pada kualitas pemindaian dan kondisi pencahayaan sehingga berpotensi mengalami kegagalan verifikasi [4]. Sementara itu, sistem presensi berbasis RFID dinilai kurang fleksibel karena bergantung pada perangkat tertentu dan area tertentu sehingga kurang mendukung mobilitas kerja di lapangan [5]. Selain itu, Penelitian terdahulu menjelaskan bahwa sistem presensi konvensional masih rentan terhadap manipulasi, seperti titip absen dan penggunaan fingerprint ganda, sehingga diperlukan mekanisme validasi lokasi yang lebih akurat dan terintegrasi [6].

Sejumlah penelitian terdahulu telah mengembangkan sistem presensi berbasis lokasi untuk meningkatkan keakuratan verifikasi kehadiran karyawan. Pemanfaatan teknologi *geolocation* pada sistem administrasi presensi dan penggajian berbasis web dinilai mampu membantu perusahaan dalam mengurangi kemungkinan terjadinya manipulasi data kehadiran [1]. Penelitian lain menerapkan algoritma *Haversine* sebagai metode perhitungan jarak antara posisi pengguna dengan titik lokasi presensi yang telah ditentukan [7]. Selain itu, penggunaan teknologi *Face Recognition* juga dinilai efektif dalam meningkatkan keamanan sistem melalui proses autentikasi biometrik berbasis wajah [8]. Integrasi teknologi geolokasi dan pengenalan wajah pada aplikasi presensi *mobile* terbukti mampu meningkatkan validitas data kehadiran sekaligus menekan potensi terjadinya kecurangan presensi [9]. Meskipun demikian, berdasarkan hasil kajian terhadap beberapa penelitian sebelumnya, sebagian besar pengembangan sistem masih berorientasi pada platform web dan belum menggabungkan validasi lokasi, pendeteksian *fake GPS*, serta verifikasi identitas pengguna dalam satu

p-ISSN : 2803-1507 e-ISSN : 2803-1531

sistem yang terintegrasi. Selain itu, beberapa penelitian terdahulu masih menggunakan metode pengembangan yang kurang fleksibel terhadap perubahan kebutuhan pengguna selama proses pengembangan sistem berlangsung. Kondisi tersebut menunjukkan bahwa masih terdapat celah penelitian dalam pengembangan sistem presensi yang mampu memberikan tingkat keamanan, ketepatan, dan integrasi sistem yang lebih optimal.

Berdasarkan kondisi tersebut, penelitian ini mengembangkan sistem informasi presensi dan penggajian karyawan berbasis web dan *mobile* dengan menerapkan algoritma *Haversine*, *Mock Location Detection*, dan *Face Recognition*. Algoritma *Haversine* digunakan untuk menghitung jarak antara koordinat pengguna dengan titik lokasi presensi yang telah ditentukan [7], Selanjutnya, *Mock Location Detection* diterapkan untuk mengidentifikasi indikasi penggunaan aplikasi manipulasi lokasi atau *fake GPS* pada perangkat pengguna. Adapun teknologi *Face Recognition* dimanfaatkan sebagai proses verifikasi identitas melalui autentikasi berbasis swafoto pengguna [8]. Penerapan teknologi geolocation pada sistem presensi juga dinilai mampu meningkatkan ketepatan dan validitas data kehadiran secara langsung atau *real-time* [10]. Pengembangan sistem pada penelitian ini menggunakan metode *Agile Scrum* karena dinilai lebih adaptif dan mampu mendukung proses pengembangan secara bertahap sesuai kebutuhan pengguna [11]. Melalui penelitian ini diharapkan dapat dihasilkan sistem presensi yang lebih aman, akurat, dan terintegrasi melalui kombinasi validasi lokasi dan autentikasi biometrik, serta mampu mendukung perusahaan dalam meningkatkan efektivitas pengelolaan presensi dan penggajian karyawan.

2. TINJAUAN PUSTAKA

2.1. Sistem Informasi

Pengembangan sistem informasi dalam berbagai literatur ilmiah diarahkan untuk mengintegrasikan berbagai elemen organisasi demi mendukung proses pengambilan keputusan. Penelitian terdahulu menekankan bahwa interaksi yang terorganisasi antara sumber daya manusia, perangkat lunak, dan jaringan komputer dalam sebuah sistem terbukti mampu meningkatkan efisiensi operasional secara signifikan [3], [12]. Selain itu, pengelolaan data secara terpadu melalui sistem informasi berkontribusi terhadap penyediaan data yang tepat waktu dan relevan bagi pemangku kepentingan [13]. Melalui pendekatan sistem terintegrasi ini, proses administrasi dapat berjalan secara lebih sistematis dan mudah diakses, yang pada akhirnya berdampak positif pada mutu pelayanan serta kinerja organisasi secara menyeluruh.

2.2. Sistem Informasi Presensi dan Penggajian

Kajian mengenai otomatisasi administrasi sumber daya manusia menunjukkan bahwa digitalisasi pencatatan kehadiran mampu meminimalkan kesalahan rekapitulasi data jika dibandingkan dengan metode konvensional [1]. Di sisi lain, penelitian terkait sistem penggajian berfokus pada akurasi penghitungan remunerasi berdasarkan variabel kehadiran, jabatan, tunjangan, dan potongan secara otomatis [14]. Integrasi antara subsistem presensi dan penggajian dalam penelitian terdahulu terbukti efektif dalam memangkas birokrasi administratif serta meminimalisasi potensi manipulasi data keuangan perusahaan.

2.3. Presensi Berbasis *Geolocation* (GPS)

Pemanfaatan teknologi berbasis lokasi (*geolocation*) dalam sistem kehadiran karyawan telah banyak diteliti untuk memastikan kesesuaian posisi fisik pengguna secara *real-time* [15]. Implementasi koordinat geografis berupa *latitude* dan *longitude* dalam studi-studi terdahulu menunjukkan tingkat fleksibilitas yang lebih tinggi serta efisiensi biaya jika dibandingkan dengan infrastruktur fisik seperti *Radio Frequency Identification* (RFID) atau pencatatan berbasis *QR Code*, khususnya untuk lingkungan kerja dinamis atau lapangan.

2.4. Algoritma *Haversine*

Dalam konteks validasi lokasi, beberapa penelitian terdahulu mengandalkan algoritma *Haversine* untuk menghitung jarak spasial antara dua titik koordinat di permukaan bumi dengan mempertimbangkan kelengkungan bumi [7]. Pengujian dalam literatur menunjukkan bahwa metode ini memberikan tingkat ketelitian yang cukup baik untuk penentuan batas wilayah (*geofencing*) pada aplikasi berbasis lokasi. Formulasi matematis yang diadopsi dari kajian terdahulu untuk menghitung jarak tersebut dijabarkan melalui Persamaan (1), (2), dan (3).

$$a = \sin^2(\Delta\phi/2) + \cos(\phi_1) \cdot \cos(\phi_2) \cdot \sin^2(\Delta\lambda/2) \quad (1)$$

$$c = 2 \cdot \text{atan2}(\sqrt{a}, \sqrt{(1-a)}) \quad (2)$$

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

p-ISSN : 2803-1507 e-ISSN : 2803-1531

Di mana, d merupakan jarak antara dua titik dalam satuan kilometer (km), sedangkan R adalah jari-jari bumi sebesar 6.371 km. Simbol ϕ_1 dan ϕ_2 masing-masing menunjukkan nilai latitude titik pertama dan titik kedua dalam satuan radian, sementara $\Delta\phi$ merupakan selisih nilai latitude ($\phi_2 - \phi_1$). Selanjutnya, $\Delta\lambda$ menunjukkan selisih nilai longitude ($\lambda_2 - \lambda_1$), sedangkan a dan c merupakan besaran yang digunakan dalam perhitungan sudut antara kedua titik.

2.5. *ock Location Detection*

Keamanan data spasial menjadi perhatian utama dalam implementasi sistem presensi berbasis lokasi. Berdasarkan penelitian terdahulu, penggunaan aplikasi manipulasi GPS (*fake GPS*) menjadi ancaman serius yang dapat merusak validitas data kehadiran [16]. Oleh karena itu, pengembangan mekanisme deteksi lokasi palsu (*Mock Location Detection*) dalam literatur ilmiah diposisikan sebagai instrumen krusial untuk memverifikasi keaslian koordinat perangkat sebelum data diproses oleh sistem guna menjaga integritas data presensi.

2.6. *Face Recognition*

Otentikasi biometrik berbasis pengenalan wajah (*face recognition*) telah banyak diimplementasikan dalam penelitian terdahulu untuk meningkatkan lapis keamanan sistem absensi [8]. Studi-studi terdahulu menunjukkan bahwa verifikasi identitas melalui pencocokan karakteristik wajah secara digital efektif dalam mereduksi celah kecurangan, seperti praktik penitipan absen (*proxy attendance*). Kombinasi antara pengenalan wajah dan validasi berbasis GPS dalam berbagai riset terbukti menghasilkan sistem pengawasan yang komprehensif karena karakteristik biometrik wajah bersifat unik dan sulit dipalsukan.

2.7. Metode *Agile Scrum*

Pengembangan perangkat lunak yang adaptif terhadap perubahan kebutuhan sering kali mengadopsi metodologi *Agile*. Literasi terkait menunjukkan bahwa kerangka kerja *Scrum* mampu meningkatkan produktivitas pengembangan melalui pembagian siklus kerja yang terstruktur atau disebut *sprint* [11]. Tahapan baku dalam *Scrum* yang meliputi *Product Backlog, Sprint Planning, Sprint Execution, Daily Scrum, Sprint Review,* dan *Sprint Retrospective* telah divalidasi oleh berbagai riset sebagai pendekatan yang efektif untuk menghasilkan produk perangkat lunak yang adaptif, transparan, dan minim risiko kegagalan sistem [17].

2.8. *Framework Laravel*

Pada bagian arsitektur *backend*, riset-riset terdahulu mengenai pengembangan aplikasi berbasis web banyak memanfaatkan *framework* Laravel karena implementasi pola *Model-View-Controller* (MVC) yang dimilikinya [18]. Dokumentasi dan penelitian sebelumnya membuktikan bahwa fitur-fitur bawaan seperti *routing, middleware, dan Object Relational Mapping (ORM)* secara signifikan mempercepat waktu pengembangan serta mempermudah pemeliharaan struktur kode program dalam jangka panjang [19].

2.9. *MySQL*

Pengelolaan data relasional yang stabil dan efisien dalam sistem informasi berskala web umumnya mengandalkan sistem manajemen basis data MySQL. Penelitian terdahulu menunjukkan bahwa penggunaan *Structured Query Language* (SQL) pada MySQL menawarkan performa yang optimal dalam menangani operasi transaksi data yang saling berhubungan, termasuk data administratif organisasi yang membutuhkan konsistensi tinggi [20], [21].

2.10. *Kotlin*

Untuk pengembangan aplikasi pada platform *mobile*, tren riset terkini menunjukkan pergeseran ke arah penggunaan bahasa pemrograman Kotlin. Berdasarkan kajian terdahulu, karakteristik Kotlin yang ringkas (*concise*) dan aman dari kesalahan penunjukan objek kosong (*null safety*) terbukti meningkatkan efisiensi penulisan kode serta mengoptimalkan performa aplikasi Android secara keseluruhan saat mengeksekusi fitur fungsional secara *real-time* [22], [23].

2.11. *Blackbox Testing*

Validasi terhadap fungsionalitas sistem dalam berbagai penelitian rekayasa perangkat lunak sering kali menerapkan metode *Black Box Testing*. Pendekatan ini berfokus pada kesesuaian antara input yang diberikan dengan output yang dihasilkan tanpa melibatkan analisis kode internal, sehingga mampu mendeteksi deviasi fitur dari spesifikasi kebutuhan pengguna secara objektif [24].

p-ISSN : 2803-1507 e-ISSN : 2803-1531

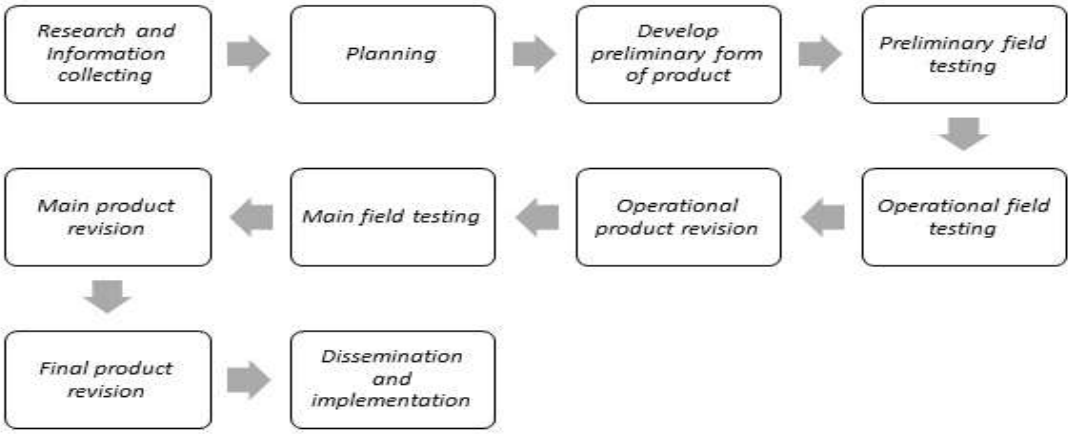
2.12. *Usability Testing*

Evaluasi terhadap aspek pengalaman pengguna (*user experience*) menjadi indikator keberhasilan implementasi sistem baru di lingkungan organisasi. Penelitian terdahulu kerap menerapkan *Usability Testing* menggunakan instrumen *System Usability Scale* (SUS) sebagai standar kuesioner yang valid dan reliabel [25], [26]. Penggunaan skala Likert dalam SUS terbukti secara ilmiah mampu memberikan gambaran terukur mengenai tingkat kemudahan penggunaan (*usability*), efisiensi, dan kepuasan pengguna akhir yang dapat dijadikan landasan objektif untuk penyempurnaan sistem pada tahap selanjutnya.

3. METODOLOGI PENELITIAN

3.1. Metode Penelitian

Penelitian ini menerapkan metode *Research and Development* (R&D) dengan mengadopsi sepuluh tahapan standar model Borg & Gall sebagai kerangka kerja makro penelitian. Pemilihan Metode R&D didasarkan pada orientasi penelitian yang tidak hanya terbatas pada kajian teoritis terhadap permasalahan administratif, melainkan juga bertujuan menghasilkan produk teknologi terapan berupa sistem informasi presensi dan penggajian karyawan berbasis *web-mobile* pada PT Munggaran Jaya Utama, sebagaimana diilustrasikan pada Gambar 1.



Gambar 1. Research and Development (R&D)

Tabel 1. Matriks Integrasi Tahapan R&D Borg & Gall dan Siklus *Agile Scrum*

Tahapan R&D	Agile Scrum	Implementasi Pada Penelitian
Research & Information collecting & Planning	*Product Backlog Gathering*	Melakukan observasi dan wawancara terstruktur guna menyusun daftar kebutuhan fungsional sistem (fitur presensi berbasis GPS, *Face Recognition*, dan *payroll*).
Develop preliminary form of product	*Sprint Planning & Execution*	Merancang permodelan UML serta melaksanakan pengkodean (*coding*) fitur fungsional memanfaatkan *framework* Laravel dan aplikasi *mobile* Kotlin.
Preliminary field testing hingga *Main product revision*	*Sprint Review & Retrospective*	Melakukan pengujian fungsional secara iteratif melalui *Black-Box Testing* (skenario positif dan negatif) serta evaluasi ketergunaan menggunakan metode *System Usability Scale* (SUS).
Final product revision & Dissemination and implementation	*Product Increment / Deployment*	Melakukan instalasi dan peluncuran versi final aplikasi SIPRESGA untuk diimplementasikan secara nyata pada kantor cabang percontohan PT Munggaran Jaya Utama.

Guna mengoptimalkan efisiensi dan adaptabilitas proses eksekusi rekayasa perangkat lunak oleh Pengembang tunggal, tahapan linear R&D pada Gambar 1 diintegrasikan dengan kerangka kerja *Agile Scrum* yang bertindak sebagai metode mikro pengembangan sistem. Korelasi operasional antara kotak tahapan prosedural R&D asli dengan siklus *Sprint* pada *Scrum* dijabarkan secara terintegrasi pada Tabel 1.

3.2. Metode Pengumpulan Data

Tahap pengumpulan data dilakukan untuk memperoleh berbagai informasi yang dibutuhkan dalam proses analisis kebutuhan, perancangan, hingga pengembangan sistem. Pada penelitian ini, teknik pengumpulan data

p-ISSN : 2803-1507 e-ISSN : 2803-1531

yang diterapkan terdiri atas observasi, wawancara, dokumentasi, dan studi literatur.

3.2.1. Observasi

Observasi dilakukan melalui pengamatan langsung terhadap mekanisme presensi dan penggajian karyawan di PT Munggaran Jaya Utama. Melalui metode ini, peneliti memperoleh data mengenai prosedur kerja, kendala administrasi, serta kebutuhan sistem yang belum terpenuhi. Informasi yang dihimpun meliputi tata cara pencatatan kehadiran, pengolahan upah, dan rekapitulasi data presensi pegawai.

3.2.2. Wawancara

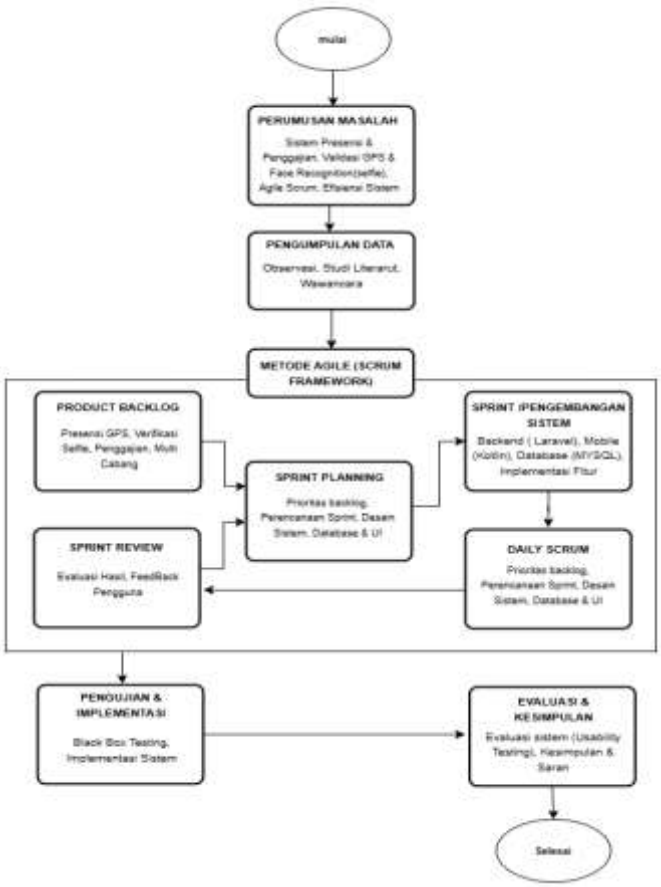
Wawancara dilakukan bersama *superadmin*, kepala kantor, admin kantor, dan karyawan untuk menghimpun data mengenai kebutuhan pengguna, kendala sistem berjalan, serta masukan bagi pengembangan sistem baru. Hasil wawancara tersebut digunakan sebagai instrumen analisis kebutuhan agar perancangan sistem dapat mengakomodasi kebutuhan pengguna secara tepat.

3.2.3. Dokumentasi

Teknik dokumentasi dilakukan dengan menghimpun dokumen terkait objek penelitian, seperti data karyawan, presensi, penggajian, struktur organisasi, dan dokumen pendukung lainnya. Data ini dimanfaatkan untuk melengkapi hasil observasi dan wawancara agar analisis kebutuhan berjalan lebih terarah, sistematis, dan akurat.

3.2.4. Studi Literatur

Studi literatur dilakukan dengan menelaah berbagai referensi ilmiah, seperti jurnal, buku, dan penelitian terdahulu terkait sistem informasi presensi, penggajian, *geolocation*, algoritma *Haversine*, *Mock Location Detection*, *Face Recognition*, serta metode *Agile Scrum*. Kajian pustaka ini berfungsi sebagai landasan teoretis sekaligus penunjang dalam perancangan dan pengembangan sistem.



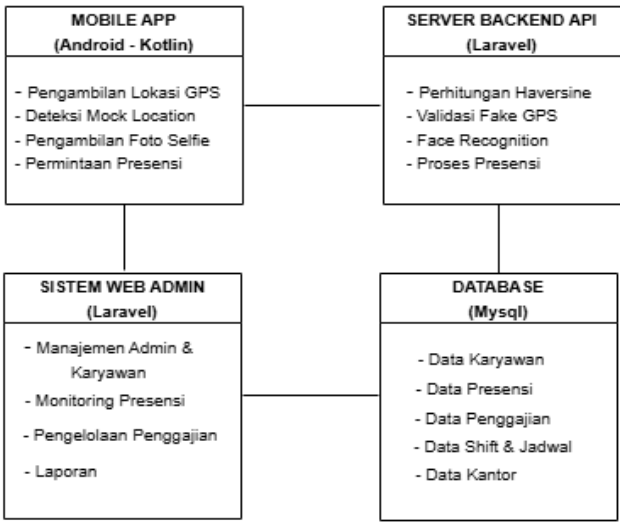
Gambar 2. Tahapan Penelitian

3.3. Metode Pengembangan SIstem

Metode pengembangan sistem yang digunakan dalam penelitian ini adalah *Agile Scrum* untuk mengelola

p-ISSN : 2803-1507 e-ISSN : 2803-1531

pembangunan sistem secara bertahap melalui beberapa *sprint* agar lebih fleksibel dan adaptif. Tahapannya meliputi *Product Backlog* untuk menyusun kebutuhan sistem, *Sprint Planning* untuk menentukan prioritas, serta *sprint development* untuk membangun sistem menggunakan Laravel, Kotlin, dan MySQL. Pengawasan progres dilakukan melalui *Daily Scrum*, sedangkan evaluasi produk dan penjaringan umpan balik dilakukan pada *Sprint Review*. Pada fase pengembangan, sistem mengimplementasikan algoritma *Haversine* untuk validasi lokasi, *Mock Location Detection* untuk deteksi *fake* GPS, dan *Face Recognition* untuk verifikasi identitas.



Gambar 3. Arsitektur Sistem

3.4. Tahapan Penelitian

Penelitian ini dilaksanakan melalui tahapan sistematis yang terarah, mulai dari identifikasi masalah hingga pengujian dan evaluasi sistem untuk mencapai tujuan penelitian. Alur pelaksanaan keseluruhan dari perumusan masalah hingga penarikan kesimpulan digambarkan pada Gambar 2.

Adapun tahapan penelitian yang dilakukan meliputi:

1. Perumusan Masalah, tahap identifikasi kendala operasional pada sistem presensi dan penggajian di PT Munggaran Jaya Utama, termasuk analisis celah keamanan terkait manipulasi kehadiran, ketiadaan validasi lokasi dan identitas, serta manajemen data *payroll* manual.
2. Pengumpulan Data, tahap penggalian informasi dasar melalui metode observasi lapangan, wawancara terstruktur, dan studi literatur yang digunakan sebagai landasan perancangan sistem.
3. Metode *Agile Scrum*, fase rekayasa perangkat lunak yang dieksekusi secara iteratif dalam siklus *Sprint*. Aktivitas teknis pada tahap ini meliputi penyusunan *product backlog*, perencanaan desain (UML, database, UI), serta pengkodean fitur fungsional (Laravel, Kotlin, MySQL) yang mengintegrasikan algoritma *Haversine*, *Mock Location Detection*, dan *Face Recognition* secara terpadu.
4. Pengujian dan Implementasi, tahap validasi fungsionalitas sistem menggunakan metode *Black-Box Testing* guna memastikan seluruh modul berjalan sesuai skenario logis (positif dan negatif) sebelum diaplikasikan pada lingkungan kerja perusahaan.
5. Evaluasi dan Kesimpulan, tahap akhir penelitian berupa pengukuran aspek ketergunaan (*usability*) melalui instrumen *System Usability Scale* (SUS) guna menilai kepuasan pengguna akhir, diikuti dengan penarikan kesimpulan serta perumusan saran pengembangan.

3.5. Arsitektur Sistem

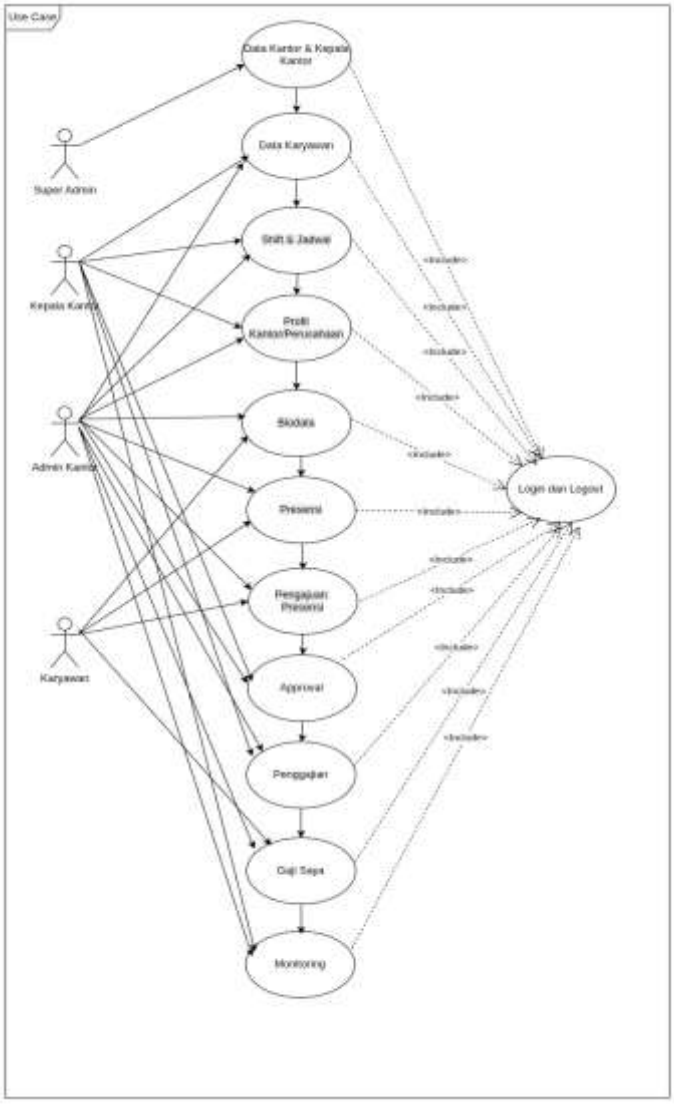
Arsitektur sistem pada penelitian ini terdiri dari aplikasi *mobile, backend* API, database, dan sistem web admin yang saling terintegrasi. Aplikasi *mobile* digunakan oleh karyawan untuk melakukan presensi dengan mengambil lokasi GPS dan foto selfie secara real-time. Data tersebut kemudian dikirimkan ke backend server untuk dilakukan validasi lokasi menggunakan algoritma *Haversine*, deteksi manipulasi lokasi menggunakan *Mock Location Detection*, serta verifikasi identitas menggunakan *Face Recognition*. Data yang telah melalui proses validasi selanjutnya disimpan ke dalam basis data MySQL dan dimanfaatkan oleh sistem web admin untuk kebutuhan monitoring presensi, pengelolaan data karyawan, serta pengolahan data penggajian. Struktur integrasi komponen infrastruktur arsitektur sistem ini secara menyeluruh diperlihatkan pada Gambar 3.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

Arsitektur sistem dirancang secara sistematis untuk mendukung proses pengelolaan data yang terintegrasi, aman, dan lebih efisien dalam lingkungan operasional perusahaan.

3.6. Implementasi Sistem

Implementasi sistem pada penelitian ini menggunakan beberapa teknologi yang saling terintegrasi untuk mendukung pengembangan sistem presensi dan penggajian berbasis web dan *mobile*. *Framework* Laravel digunakan sebagai *backend* utama dalam mengelola proses bisnis, autentikasi pengguna, pengolahan data presensi dan penggajian, serta penyediaan *Application Programming Interface* (API) untuk aplikasi *mobile*. Pengelolaan data dilakukan menggunakan MySQL sebagai sistem manajemen basis data yang menyimpan data karyawan, presensi, lokasi, dan penggajian secara terpusat. Pada sisi antarmuka web, sistem dikembangkan menggunakan *Tailwind CSS* dan *JavaScript* untuk menghasilkan tampilan yang responsif, modern, dan interaktif. Adapun pengembangan aplikasi *mobile* pada penelitian ini dilakukan menggunakan bahasa pemrograman Kotlin dengan bantuan Android Studio sebagai perangkat lunak pengembangan atau *Integrated Development Environment* (IDE). Aplikasi *mobile* tersebut mendukung implementasi fitur presensi, seperti pengambilan lokasi GPS, deteksi *mock location*, pengambilan foto swafoto, serta pertukaran data dengan *backend* API secara *real-time*.



Gambar 4. Use Case Diagram

3.7. Pemodelan Sistem

Perancangan sistem dalam penelitian ini menggunakan *Unified Modeling Language* (UML) sebagai media pemodelan alur proses bisnis dan mekanisme kerja sistem. Penggunaan UML bertujuan memfasilitasi analisis

p-ISSN : 2803-1507 e-ISSN : 2803-1531

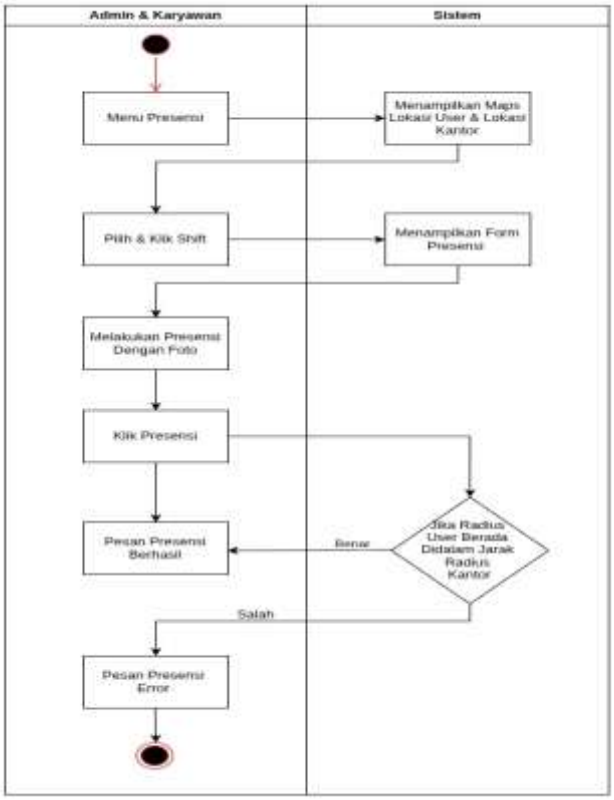
kebutuhan, perancangan, dan implementasi aplikasi agar lebih terstruktur serta mudah dipahami.

3.7.1. *Use Case Diagram*

Use Case Diagram digunakan sebagai media pemodelan untuk menunjukkan hubungan interaksi antara pengguna dan sistem pada proses bisnis yang dijalankan. Pemetaan interaksi fungsional dan pembagian hak akses dari seluruh aktor tersebut disajikan secara terstruktur dalam *Use Case* Diagram pada Gambar 4.

Pengguna sistem dalam penelitian ini terdiri atas *superadmin*, kepala kantor, admin kantor, dan karyawan, yang masing-masing memiliki hak akses berbeda. *Superadmin* mengelola data kantor, pimpinan, dan informasi internal. Kepala kantor berwenang mengelola data pegawai, pengaturan *shift* kerja, penggajian, pemantauan presensi, serta persetujuan pengajuan. Admin kantor bertugas mengelola biodata staf, presensi, penggajian, pemantauan sistem, dan verifikasi pengajuan. Sementara itu, karyawan memiliki akses terbatas untuk melakukan presensi, mengajukan permohonan izin, serta melihat profil, gaji, dan riwayat kehadiran. Seluruh aktor wajib melakukan proses *login* dan *logout* untuk mengakses sistem.

Gambar 5. *Activity Diagram*



3.7.2. *Activity Diagram*

Activity Diagram digunakan untuk menggambarkan alur kerja presensi yang mengintegrasikan validasi lokasi berbasis GPS dan verifikasi swafoto (Gambar 5). Proses dimulai saat pengguna mengakses menu presensi, diikuti penayangan posisi pengguna dan koordinat kantor pada peta digital. Setelah menentukan *shift* kerja dan melakukan swafoto, data dikirim untuk divalidasi menggunakan algoritma *Haversine*. Jika pengguna berada di dalam radius, presensi dinyatakan berhasil dan data disimpan ke basis data. Sebaliknya, jika posisi berada di luar radius, sistem akan menolak presensi serta menampilkan pesan kesalahan.

3.7.3. *Sequence Diagram*

Sequence Diagram digunakan untuk menggambarkan interaksi kronologis antarkomponen sistem selama proses presensi (Gambar 6). Alur dimulai saat pengguna membuka menu presensi pada aplikasi seluler, lalu sistem mengambil data lokasi pengguna dan kantor melalui *backend* yang terhubung dengan basis data. Setelah pengguna memilih penugasan kerja (*shift*), melakukan swafoto, dan menekan tombol presensi, data dikirim untuk divalidasi menggunakan algoritma Haversine. Jika pengguna berada dalam radius kantor, sistem menampilkan notifikasi keberhasilan. jika tidak, muncul pesan kesalahan. Pemodelan ini menunjukkan bahwa proses presensi berjalan secara otomatis dan *real-time* melalui komunikasi

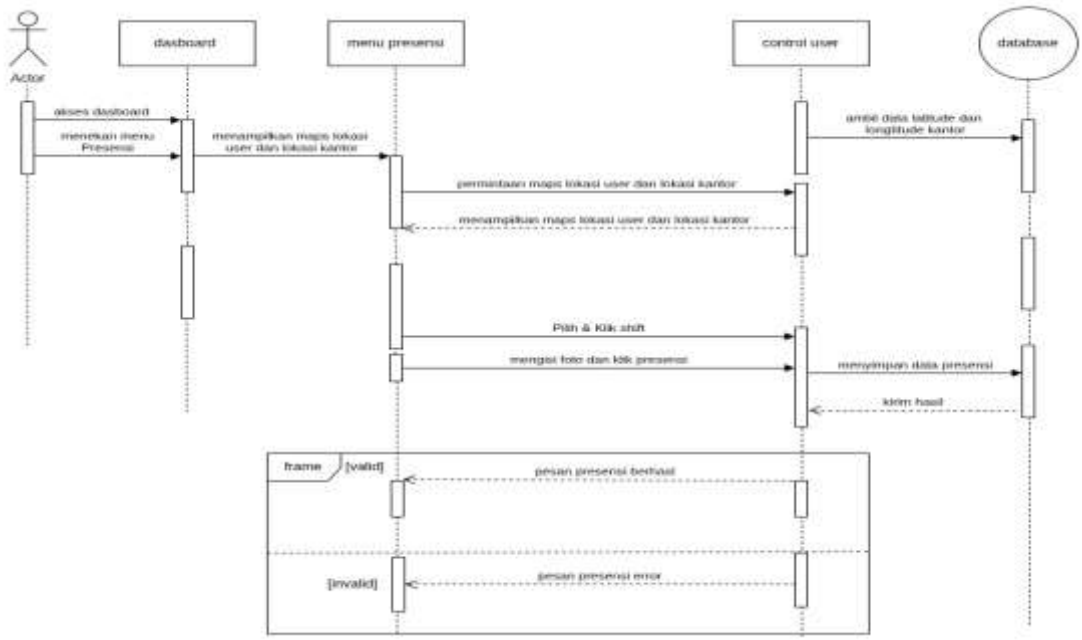
p-ISSN : 2803-1507 e-ISSN : 2803-1531

antarkomponen yang terintegrasi.

Gambar 6. Sequence Diagram

3.8. Pengujian Sistem

Pengujian sistem dilakukan dengan menggunakan metode *Black Box Testing* untuk mengetahui kesesuaian



fungsi sistem terhadap kebutuhan fungsional yang telah dirancang sebelumnya. Proses pengujian dilakukan pada beberapa fitur utama sistem, di antaranya fitur *login*, presensi berbasis lokasi, pendeteksian *fake GPS*, verifikasi foto swafoto, pengelolaan data karyawan, serta pengolahan data penggajian. Melalui pengujian tersebut, setiap fitur dievaluasi guna memastikan sistem dapat berjalan sesuai dengan proses dan keluaran yang diharapkan. Selain pengujian fungsional, penelitian ini juga menerapkan *Usability Testing* menggunakan metode *System Usability Scale* (SUS) untuk mengetahui tingkat kemudahan penggunaan sistem berdasarkan pengalaman pengguna secara langsung. Pengujian dilakukan dengan melibatkan pengguna sistem sebagai responden untuk memberikan penilaian terhadap aspek kemudahan penggunaan, efisiensi proses, serta tingkat kepuasan pengguna terhadap sistem presensi dan penggajian yang telah dikembangkan.

4. HASIL DAN PEMBAHASAN

4.1. *Implementasi Algoritma Haversine*

Implementasi algoritma *Haversine* pada penelitian ini digunakan untuk menghitung jarak antara posisi pengguna dan titik lokasi presensi yang telah ditentukan perusahaan. Proses perhitungan dilakukan berdasarkan koordinat *latitude* dan *longitude* yang diperoleh secara *real-time* dari perangkat *mobile* pengguna. Validasi lokasi dilakukan secara otomatis ketika pengguna melakukan proses presensi melalui aplikasi Android. Sistem secara dinamis akan membandingkan posisi fisik pengguna dengan radius titik koordinat kantor yang telah ditetapkan untuk menentukan apakah proses presensi tersebut dapat diterima atau ditolak dapat di lihat pada Gambar 7.

Formulasi matematis algoritma *Haversine* yang diadopsi dalam pengembangan sistem ini ditunjukkan melalui Persamaan (4) sampai (8).

$$\Delta lat = lat_2 - lat_1 \quad (4)$$

$$\Delta long = long_2 - long_1 \quad (5)$$

$$a = \sin^2(\Delta lat/2) + \cos(lat_1) \cdot \cos(lat_2) \cdot \sin^2(\Delta long/2) \quad (6)$$

$$c = 2 \cdot atan2(\sqrt{a}, \sqrt{(1-a)}) \quad (7)$$

$$d = R \cdot c \quad (8)$$

Di mana, d merupakan jarak antara dua titik lokasi, sedangkan R adalah jari-jari bumi sebesar 6.371 km. Selanjutnya, Δlat menunjukkan selisih koordinat latitude (garis lintang), sementara Δlong merupakan selisih

p-ISSN : 2803-1507 e-ISSN : 2803-1531

koordinat longitude (garis bujur). Adapun a dan c merupakan variabel perantara dalam perhitungan geometris bola yang digunakan untuk menentukan jarak antara kedua titik lokasi.

```php
private function hitungJarakHaversine($latUser, $lngUser, $latKantor, $lngKantor)
{
    $earthRadius = 6371000; // meter

    $lat1 = deg2rad($latUser);
    $lat2 = deg2rad($latKantor);

    $deltaLat = deg2rad($latKantor - $latUser);
    $deltaLng = deg2rad($lngKantor - $lngUser);

    $a = sin($deltaLat / 2) * sin($deltaLat / 2) +
        cos($lat1) * cos($lat2) *
        sin($deltaLng / 2) * sin($deltaLng / 2);

    $c = 2 * atan2(sqrt($a), sqrt(1 - $a));

    return round($earthRadius * $c);
}
```

Gambar 7. Algoritma Haversine

Implementasi struktural dari seluruh rangkaian rumus matematis ini dilakukan pada sisi *backend* server dengan menggunakan *framework* Laravel. Data koordinat pengguna yang dikirimkan oleh aplikasi *mobile* akan ditangkap oleh API untuk dieksekusi oleh fungsi khusus di dalam *controller*. Potongan kode program (*source code*) dari implementasi fungsi perhitungan algoritma Haversine pada server backend tersebut secara mendetail dapat dilihat pada Gambar 7. Guna menguji akurasi fungsional dari logika program pada Gambar 7, simulasi komputasi dilakukan menggunakan sampel koordinat empiris di lapangan. Pengujian ini menetapkan koordinat instansi PT Munggaran Jaya Utama sebagai titik pusat (*center*) pada posisi -7,350412 (Lintang) dan 108,142534 (Bujur), dengan batas radius toleransi kehadiran maksimal sejauh 50 meter.

Sebagai studi kasus, sistem mengalkulasi data koordinat perangkat karyawan yang berada pada posisi -7,350250 (Lintang) dan 108,142410 (Bujur). Proses konversi derajat ke radian dan substitusi ke dalam persamaan *Haversine* menghasilkan nilai variabel *a* sebesar $1{,}22 \times 10^{-11}$ dan variabel *c* sebesar $7{,}01 \times 10^{-6}$. Melalui kalkulasi akhir, jarak spasial antara karyawan dan kantor cabang terhitung sejauh 22,3 meter. Karena hasil pembulatan jarak berada di bawah atau sama dengan ambang batas toleransi 50 meter, sistem menampilkan status valid sehingga proses presensi berhasil diterima. Untuk membuktikan keandalan fungsi sistem dalam mendeteksi skenario keberhasilan dan kegagalan verifikasi lokasi secara menyeluruh, pengujian terhadap beberapa variasi koordinat perangkat karyawan disajikan secara komprehensif pada Tabel 2.

Tabel 2. Matriks Skenario Pengujian Validasi Jarak Lokasi Algoritma *Haversine*

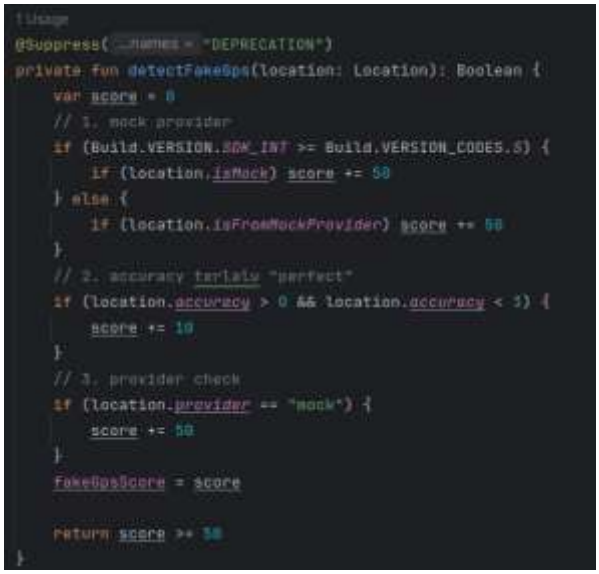
No	Koordinat Pusat Kantor (Target)	Koordinat Perangkat Karyawan (Aktual)	Jarak Kalkulasi Sistem	Radius Toleransi	Status Skenario	Respons Fungsional Sistem	Status Uji
1	-7.350412, 108.142534	-7.350250, 108.142410	22,3 Meter	≤ 50 Meter	Dalam Radius	Presensi Berhasil Diterima	Berhasil
2	-7.350412, 108.142534	-7.350390, 108.142510	3,6 Meter	≤ 50 Meter	Dalam Radius	Presensi Berhasil Diterima	Berhasil
3	-7.350412, 108.142534	-7.351200, 108.143100	108,5 Meter	≤ 50 Meter	Luar Radius	Presensi Ditolak (Pesan *Error*)	Berhasil
4	-7.350412, 108.142534	-7.353150, 108.145620	452,1 Meter	≤ 50 Meter	Luar Radius	Presensi Ditolak (Pesan *Error*)	Berhasil

4.2. Implementasi *Mock Location Detection*

Fitur *Mock Location Detection* diintegrasikan pada aplikasi *mobile* Android untuk menangkal manipulasi lokasi melalui aplikasi pihak ketiga (*fake GPS*). Proteksi ini berjalan pada tahap awal sebelum algoritma *Haversine* menghitung jarak koordinat. Proses pemeriksaan status lokasi palsu dilakukan oleh fungsi `cekMockLocation` yang menerima parameter objek `Location` dan mengembalikan nilai *Boolean*. Struktur logika fungsi ini memanfaatkan percabangan berbasis versi SDK perangkat (`Build.VERSION.SDK_INT`) seperti yang ditunjukkan pada Gambar 8. Jika gawai menggunakan Android 12 ke atas (`Build.VERSION_CODES.S`), deteksi mengevaluasi properti `location.isMock`. Sementara itu, perangkat dengan versi Android di bawahnya diperiksa melalui properti `location.isFromMockProvider`. Jika fungsi tersebut menghasilkan nilai `true`, sistem otomatis

p-ISSN : 2803-1507 e-ISSN : 2803-1531

menolak presensi dan mengirimkan notifikasi peringatan.



```
1 usage
@Suppress("DEPRECATION")
private fun detectFakeGps(location: Location): Boolean {
    var score = 0
    // 1. mock provider
    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.S) {
        if (location.isMock) score += 50
    } else {
        if (location.isFromMockProvider) score += 50
    }
    // 2. accuracy terlalu "perfect"
    if (location.accuracy > 0 && location.accuracy < 3) {
        score += 10
    }
    // 3. provider check
    if (location.provider == "mock") {
        score += 50
    }
    fakeGpsScore = score

    return score >= 50
}
```

Gambar 8. *Mock Location Detection*

Keandalan arsitektur kode diuji secara komprehensif menggunakan empat variasi karakteristik gawai dan versi OS Android yang terhubung ke server *backend* (Tabel 3). Simulasi kecurangan dilakukan dengan mengaktifkan koordinat palsu di sekitar area radius kantor PT Munggaran Jaya Utama lewat menu *Developer Options*. Fungsi `cekMockLocation` bertugas menangkap *flag* manipulasi tersebut untuk dikirimkan sebagai parameter `is_mock_location` menuju fungsi `storePresensi` pada *backend* Laravel yang dimana dapat dilihat pada tabel 3.

Tabel 3. Spesifikasi Perangkat dan Aplikasi dalam Pengujian Deteksi *Fake GPS*

ID Perangkat	Model Perangkat (Gawai)	Versi OS Android (API Level)	Aplikasi Fake GPS yang Diuji	Pustaka/Metode Android Kotlin
Perangkat 1	Samsung Galaxy A53	Android 12 (API 31: S)	*Fake* GPS (ByteRev)	`location.isMock`
Perangkat 2	POCO X5 5G	Android 13 (API 33: T)	GPS JoyStick (AppNinjas)	`location.isMock`
Perangkat 3	Xiaomi Redmi Note 10	Android 11 (API 30: R)	*Fake* GPS (ByteRev)	`location.isFromMockProvider`
Perangkat 4	Asus Zenfone 7	Android 10 (API 29: Q)	Tidak Menggunakan (Normal)	`location.isFromMockProvider`

Tabel 4. Matriks Hasil Pengujian Fungsionalitas *Mock Location Detection*

No	Perangkat	Aplikasi Penguji	Status Mock pada Perangkat	Nilai Parameter ke API Server	Respons JSON Backend Laravel	Status Uji
1	Perangkat 1	*Fake* GPS (ByteRev)	Aktif (*True*)	`is_mock_location: true`	HTTP 422 (Presensi ditolak karena terdeteksi menggunakan *fake GPS*)	Berhasil
2	Perangkat 2	GPS JoyStick	Aktif (*True*)	`is_mock_location: true`	HTTP 422 (Presensi ditolak karena terdeteksi menggunakan *fake GPS*)	Berhasil
3	Perangkat 3	*Fake* GPS (ByteRev)	Aktif (*True*)	`is_mock_location: true`	HTTP 422 (Presensi ditolak karena terdeteksi menggunakan *fake GPS*)	Berhasil
4	Perangkat 4	Tidak Aktif (Normal)	Nonaktif (*False*)	`is_mock_location: false`	HTTP 201 (Presensi berhasil disimpan)	Berhasil

Aliran parameter data beserta respons sistem dari hasil simulasi tersebut disajikan secara mendetail pada Tabel 4. Berdasarkan hasil uji fungsional pada Tabel 4, sistem SIPRESGA menunjukkan konsistensi akurasi sebesar 100% dalam memblokir akses presensi yang tidak sah di seluruh versi Android. Perangkat yang memanipulasi lokasi secara otomatis menerima respons status HTTP 422, sedangkan perangkat normal mendapatkan respons HTTP 201.

Meskipun proteksi ini bekerja optimal pada kondisi standar, fungsionalitas sistem memiliki keterbatasan

p-ISSN : 2803-1507 e-ISSN : 2803-1531

teknis karena hanya mampu mendeteksi penyedia lokasi palsu resmi dari *Developer Options*. Sistem berpotensi mengalami kegagalan deteksi apabila gawai pengguna telah dimodifikasi hak akses utamanya (*rooting*) menggunakan modul tingkat lanjut level kernel seperti *Smali Patcher* atau LSPosed yang mampu menyembunyikan status *mock* langsung dari sistem operasi. Keterbatasan operasional ini menjadi peluang rekayasa penting bagi pengembangan sistem keamanan pada penelitian masa depan.

4.3. Implementasi *Face Recognition*

Pada tahap ini, sistem melakukan verifikasi wajah dengan membandingkan dua gambar, yaitu foto utama (*master path*) dan foto swafoto (*selfie path*). Proses verifikasi wajah dilakukan menggunakan potongan kode Python berbasis *library face_recognition* seperti yang tertera pada Gambar 9.

```python
import face_recognition

THRESHOLD = 0.50

def verify_face(master_path, selfie_path):

    master_image = face_recognition.load_image_file(master_path)
    selfie_image = face_recognition.load_image_file(selfie_path)

    master_encodings = face_recognition.face_encodings(master_image)
    selfie_encodings = face_recognition.face_encodings(selfie_image)

    master_encoding = master_encodings[0]
    selfie_encoding = selfie_encodings[0]

    distance = face_recognition.face_distance(
        [master_encoding],
        selfie_encoding
    )[0]

    verified = bool(distance <= THRESHOLD)

    print(json.dumps({
        "verified": verified,
        "distance": float(round(distance, 6)),
        "threshold": float(THRESHOLD)
    }))
```

```php
$pathMaster = public_path('storage/' . $profile->pas_foto);
$pathSelfie = storage_path('app/public/' . $fotoPath);

if (!file_exists($pathMaster)) {
    return response()->json([
        'success' => false,
        'message' => 'Pas foto tidak ditemukan di storage.'
    ], 422);
}

if (!file_exists($pathSelfie)) {
    return response()->json([
        'success' => false,
        'message' => 'File selfie tidak ditemukan.'
    ], 422);
}
```

Gambar 9. *Face Recognition*

Penelitian ini menerapkan teknologi *face recognition* berbasis *deep learning* melalui model dlib (*ResNet 128-dimensional face embedding*) menggunakan Python untuk memverifikasi identitas pegawai saat swafoto presensi. Langkah ini bertujuan untuk meminimalkan kecurangan seperti titip absen (*proxy attendance*). Proses verifikasi dilakukan secara spesifik (*one-to-one matching*) dengan mengambil citra wajah pengguna saat presensi, lalu membandingkannya langsung dengan data referensi pada kolom *pas_foto* yang dipanggil berdasarkan *id_pegawai* bersangkutan dari basis data internal. Sistem menetapkan nilai ambang batas (*threshold*) sebesar 0,5, di mana hasil pencocokan dinyatakan valid apabila nilai jarak (*face distance*) ≤ 0,5. Metode evaluasi yang digunakan dalam penelitian ini adalah *verification success rate*, yaitu pengukuran tingkat keberhasilan sistem dalam mencocokkan wajah pengguna (*selfie*) dengan data referensi (pas foto) yang tersimpan pada database pegawai. Sistem melakukan perhitungan *face distance* menggunakan model embedding wajah, kemudian hasilnya dibandingkan dengan nilai ambang batas tersebut untuk menentukan apakah wajah sesuai atau tidak.

Pengujian sistem dilakukan melalui 5 (lima) skenario dengan total 100 kali uji coba untuk mengukur keandalan sistem. Berdasarkan hasil pengujian yang dirangkum pada Tabel 5, sistem berhasil mencapai rata-rata tingkat akurasi sebesar 94%. Autentikasi pada skenario pengguna tidak valid serta pengujian manipulasi foto (*spoofing*) menghasilkan tingkat keberhasilan 100%. Meskipun terjadi sedikit penurunan performa pada kondisi pencahayaan rendah (85%) dan variasi sudut wajah (90%), integrasi antara verifikasi biometrik wajah dan validasi lokasi berbasis GPS terbukti menghasilkan mekanisme presensi yang lebih aman, akurat, dan minim manipulasi dibandingkan dengan metode konvensional.

Tabel 5. Hasil Pengujian *Face Recognition*

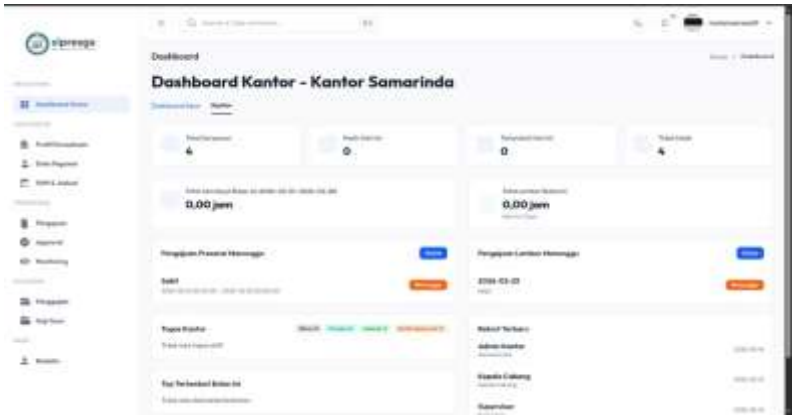
p-ISSN : 2803-1507 e-ISSN : 2803-1531

No	Skenario Uji	Jumlah Percobaan	Berhasil	Gagal	Akurasi
1	Pengguna valid (normal)	20	19	1	95%
2	Sudut wajah berbeda	20	18	2	90%
3	Orang lain (tidak valid)	20	20	0	100%
4	Pencahayaan rendah	20	17	3	85%
5	Foto/gambar *spoofing*	20	20	0	100%

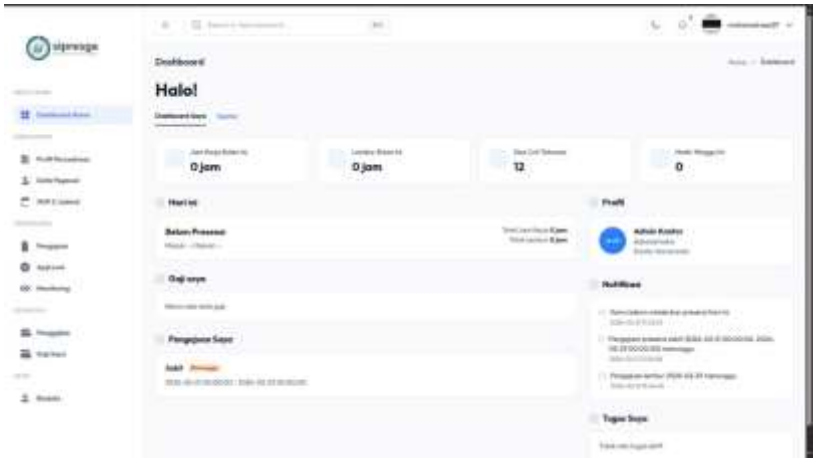
4.4. Implementasi Antarmuka Sistem

Implementasi antarmuka sistem pada penelitian ini menampilkan halaman-halaman utama yang digunakan dalam sistem presensi dan penggajian berbasis web dan *mobile*. Aplikasi *mobile* digunakan oleh karyawan untuk melakukan presensi, melihat data kehadiran, serta mengakses informasi penggajian, sedangkan sistem web digunakan oleh admin untuk monitoring presensi, pengelolaan data karyawan, dan penggajian. Setiap halaman dirancang untuk mendukung proses pengelolaan sistem secara lebih efektif, terintegrasi, dan mudah digunakan oleh pengguna.

4.4.1. Halaman *Dashboard*



Gambar 10. Halaman *Dashboard* Kantor

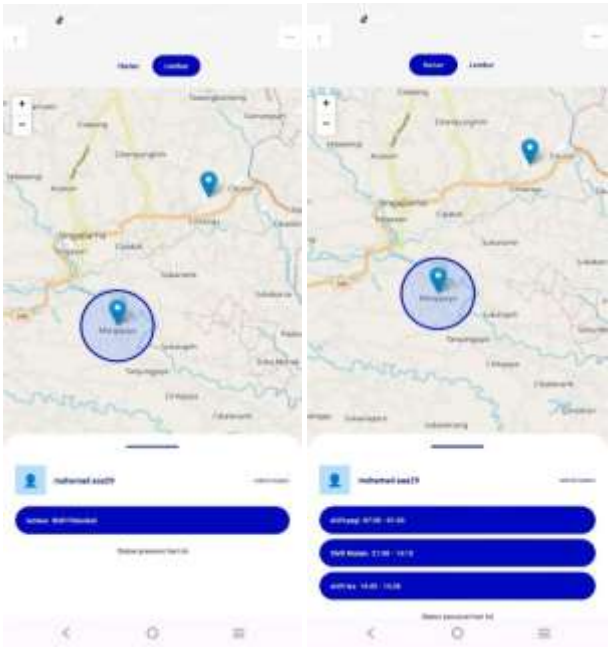


Gambar 11. Halaman *Dashboard* Saya

Halaman *dashboard* berfungsi sebagai pusat informasi *real-time* sistem yang terbagi menjadi dua hak akses. *Dashboard* admin dan kepala kantor (Gambar 10) memuat pemantauan operasional menyeluruh, seperti data kehadiran, keterlambatan, jam kerja, lembur, dan pengajuan karyawan. Sebaliknya, *dashboard* karyawan (Gambar 11) menampilkan informasi personal meliputi status presensi, jam kerja, gaji, izin, dan profil. Implementasi kedua model ini terbukti mempermudah pemantauan klinis-administratif serta mengoptimalkan pengelolaan data perusahaan secara terintegrasi.

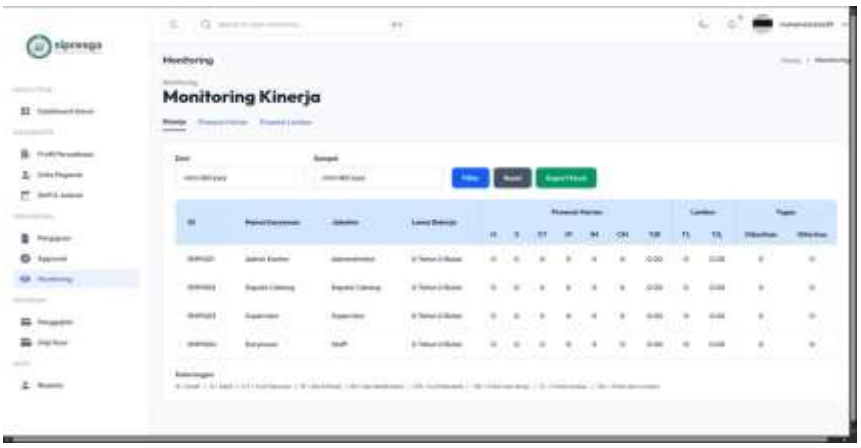
p-ISSN : 2803-1507 e-ISSN : 2803-1531

4.4.2. Halaman Presensi Mobile



Gambar 12. Halaman Presensi *Mobile*

Presensi *Mobile* diimplementasikan melalui aplikasi Android yang memungkinkan karyawan dan admin melakukan presensi berbasis GPS dan swafoto secara langsung. Tampilan antarmuka halaman ini menyajikan koordinat posisi pengguna serta peta radius kehadiran (Gambar 12). Melalui halaman tersebut, parameter keamanan seperti lokasi aktual dan autentikasi biometrik wajah diambil secara instan, lalu dikirim ke *backend* server untuk validasi kehadiran secara *real-time*.



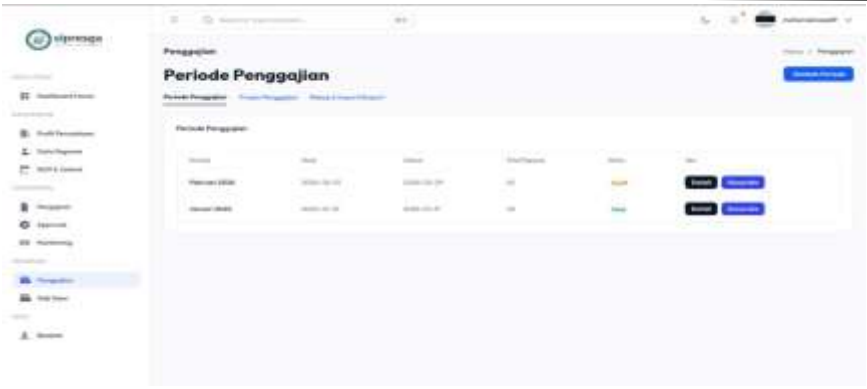
4.4.3. Halaman Monitoring Presensi

Gambar 13. Halaman Penggajian

Halaman monitoring presensi berbasis web memungkinkan admin mengawasi data kehadiran karyawan secara *real-time*. Antarmuka halaman ini menyajikan struktur tabel, rekapitulasi kehadiran, dan rincian performa kerja harian (Gambar 13). Melalui fitur tersebut, administrator dapat memantau kedisiplinan pegawai secara berkala dan terpusat, sehingga mempermudah evaluasi serta koordinasi administratif perusahaan.

4.4.4. Halaman Penggajian

p-ISSN : 2803-1507 e-ISSN : 2803-1531



Gambar 14. Halaman Penggajian

Fitur penggajian berfungsi menghitung gaji karyawan secara otomatis berdasarkan data presensi sistem. Halaman ini memuat informasi periode, jumlah pegawai, dan status penggajian (Gambar 14). Melalui antarmuka tersebut, admin dapat menggunakan tombol *detail* dan *generate* untuk memproses upah secara otomatis, sehingga administrasi keuangan menjadi lebih cepat, terstruktur, dan akurat.

Tabel 6. Black Box Testing

No	Fitur	Jenis Ske-nario	Skenario Pengujian	Hasil yang Diharapkan	Status
1	Login	Positif	Pengguna menginput *username* dan *password* yang benar	Sistem berhasil memverifikasi dan menampilkan halaman dashboard sesuai hak akses	Berhasil
2	Login	Negatif	Pengguna menginput *password* salah atau mengosongkan kolom isian	Sistem menolak proses autentikasi dan menampilkan pesan "*Username* atau *password* salah"	Berhasil
3	Presensi Fungsional	Positif	Karyawan melakukan presensi dengan swafoto, lokasi aktif, dan jaringan internet stabil	Sistem berhasil menyimpan data ke-hadiran, citra foto, koordinat lokasi, dan waktu ke basis data	Berhasil
4	Presensi Fungsional	Negatif (*edge case*)	Sistem membatalkan pengiriman data dan menampilkan pesan kesalahan jaringan	Sistem mendeteksi dan menolak proses presensi	Berhasil
5	Validasi GPS	Positif	Karyawan melakukan presensi di dalam radius toleransi kantor (≤ 50 meter)	Sistem menerima data kehadiran dan mengkalkulasi jarak aktual karyawan secara tepat	Berhasil
6	Validasi GPS	Negatif	Karyawan melakukan presensi di luar radius toleransi kantor (misal: 150 meter)	Sistem menolak presensi dan menampilkan peringatan "Anda berada di luar radius lokasi kantor"	Berhasil
7	Validasi GPS	Negatif (*edge case*)	Karyawan mengakses menu presensi dengan kondisi fitur GPS perangkat dinonaktifkan	Aplikasi mendeteksi GPS nonaktif, menampilkan dialog instruksi, dan mengunci tombol kirim	Berhasil
8	Deteksi *Fake* GPS	Negatif (Proteksi)	Pengguna mencoba melakukan presensi menggunakan aplikasi pihak ketiga (*Mock Location*)	Sistem mendeteksi mock provider, memblokir aksi presensi, dan menampilkan notifikasi peringatan	Berhasil
9	Verifikasi Swafoto	Positif	Pengguna melakukan swafoto dengan karakteristik wajah yang sesuai dengan data master	Sistem mengenali wajah, mencocokkan koordinat embedding, dan memverifikasi identitas pengguna	Berhasil
10	Verifikasi Swafoto	Negatif	Pengguna melakukan swafoto menggunakan karakteristik wajah orang lain yang tidak terdaftar	Sistem menolak verifikasi identitas dan menampilkan notifikasi kesalahan data biometrik	Berhasil
11	Kelola Data Karyawan	Positif	Admin menambahkan data identitas karyawan baru secara lengkap ke dalam sistem	Data administrasi karyawan baru berhasil tersimpan secara permanen ke dalam database.	Berhasil
12	Kelola Data Karyawan	Negatif	Admin menginput alamat email karyawan yang telah terdaftar sebelumnya (duplikasi)	Sistem menolak penyimpanan data dan menampilkan pesan peringatan duplikasi identitas	Berhasil
13	Penggajian (*Payroll*)	Positif	Admin melakukan proses kalkulasi remunerasi bulanan karyawan	Sistem otomatis menghitung gaji berdasarkan rekap kehadiran secara presisi	Berhasil
14	Penggajian (*Payroll*)	Negatif	Admin memproses penggajian bagi karyawan yang tidak memiliki catatan kehadiran sama sekali	Sistem menampilkan nominal gaji pokok dengan penerapan penalti potongan absen otomatis	Berhasil

p-ISSN : 2803-1507 e-ISSN : 2803-1531

4.5. Pengujian Sistem

Pengujian sistem merupakan tahap krusial untuk menjamin bahwa aplikasi yang telah dibangun dapat beroperasi sesuai dengan spesifikasi fungsional yang telah dirancang sebelumnya. Tujuan utama dari pengujian ini adalah untuk menemukan kesalahan *(error*), memastikan integritas data, serta memvalidasi bahwa sistem dapat menangani berbagai skenario penggunaan, baik skenario normal maupun skenario kesalahan.

4.5.1. *Black Box Testing*

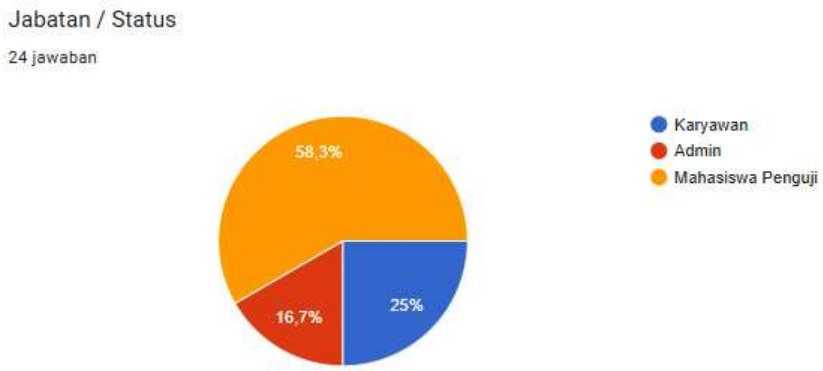
Pengujian sistem dilakukan menggunakan metode *Black Box Testing* untuk memastikan setiap fitur sistem berjalan sesuai dengan kebutuhan dan fungsi yang telah dirancang. Hasil pengujian *Black Box Testing* dapat dilihat pada Tabel 6. Pengujian sistem menggunakan metode *Black Box Testing* untuk mengevaluasi kesesuaian fungsi terhadap kebutuhan fungsional. Pengujian mencakup fitur autentikasi, presensi, validasi GPS, deteksi *fake GPS*, verifikasi swafoto, serta pengelolaan data karyawan, penggajian, dan laporan. Hasil pengujian menunjukkan tingkat keberhasilan 100% sesuai skenario yang dirancang. Sistem terbukti mampu memvalidasi lokasi dengan algoritma *Haversine*, mendeteksi lokasi palsu melalui *Mock Location Detection*, serta memverifikasi identitas via *Face Recognition*. Integrasi presensi dan penggajian ini berjalan secara *real-time* untuk mendukung akurasi manajemen data.

4.5.2. *Usability Testing*

Penelitian ini juga menerapkan *Usability Testing* untuk mengetahui tingkat kemudahan penggunaan sistem presensi dan penggajian SIPRESGA berdasarkan pengalaman pengguna secara langsung. Proses pengujian dilakukan dengan membagikan kuesioner kepada responden setelah mereka menggunakan aplikasi yang telah dikembangkan. Kuesioner disusun menggunakan skala Likert dengan rentang nilai 1 sampai 5 untuk memperoleh penilaian pengguna terhadap sistem. Beberapa aspek yang dievaluasi meliputi tingkat kemudahan penggunaan aplikasi, tampilan antarmuka sistem, proses presensi berbasis GPS dan swafoto, serta tingkat kepuasan pengguna dalam menggunakan sistem presensi dan penggajian yang telah dikembangkan.

Tabel 7. Skala Likert Pengujian *Usability*

Skor	Keterangan
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Netral
4	Setuju
5	Sangat Setuju



Gambar 15. Persentase Responden Berdasarkan Jabatan/Status

Berdasarkan hasil pengujian yang melibatkan 24 responden, sistem memperoleh nilai rata-rata *usability* sebesar 75,25% dengan kategori "Baik". Hasil tersebut menunjukkan bahwa aplikasi SIPRESGA dapat mendukung proses presensi dan penggajian secara efektif serta mudah digunakan oleh pengguna. Pengujian ini melibatkan beberapa kelompok responden dengan latar belakang yang berbeda, yaitu terdiri atas mahasiswa penguji sebesar 58,3%, karyawan sebesar 25%, dan admin sebesar 16,7%. Persentase sebaran dan demografi responden yang terlibat di dalam pengujian *usability* ini dikelompokkan secara visual berdasarkan jabatan atau status mereka pada Gambar 15. Melalui keterlibatan berbagai elemen pengguna tersebut, penilaian terhadap tingkat kegunaan aplikasi dapat diperoleh secara lebih objektif dan komprehensif dari

p-ISSN : 2803-1507 e-ISSN : 2803-1531

berbagai sudut pandang pengguna akhir.

4.6. Diskusi Hasil Penelitian

Penelitian ini menunjukkan bahwa integrasi algoritma *Haversine*, *Mock Location Detection*, dan *Face Recognition* berhasil meningkatkan keamanan serta akurasi sistem presensi berbasis web dan seluler. Algoritma *Haversine* memvalidasi lokasi pengguna melalui koordinat *Global Positioning System* (GPS), sementara *Mock Location Detection* memitigasi manipulasi lokasi oleh aplikasi *fake GPS*. Selanjutnya, *Face Recognition* memperkuat validasi identitas via verifikasi swafoto secara *real-time*. Berbeda dengan riset terdahulu yang menerapkan komponen keamanan secara parsial [15], penelitian ini menggabungkan ketiga lapisan proteksi tersebut ke dalam satu sistem terpadu guna mereduksi risiko manipulasi data kehadiran pegawai. Bukti empiris dari peningkatan performa operasional ini disajikan pada Tabel 8.

Tabel 8. Matriks Perbandingan Efisiensi Proses Sebelum dan Sesudah Implementasi SIPRESGA

Parameter Analisis	Proses Lama (Manual & Spreadsheet)	Sistem Baru (SIPRESGA)	Dampak dan Efisiensi Operasional
Waktu Rekapitulasi Kehadiran	Membutuhkan waktu 2–3 hari kerja pada akhir periode bulan berjalan.	Berjalan secara otomatis, terpusat, dan dapat dipantau secara *real-time*.	Memangkas waktu birokrasi administratif kehadiran hingga ±95%.
Validasi Lokasi Kerja	Dilakukan secara manual melalui pemeriksaan visual pada lampiran dokumentasi foto.	Divalidasi secara otomatis melalui kalkulasi koordinat menggunakan algoritma *Haversine*.	Mengeliminasi kesalahan input data jarak dan meningkatkan akurasi sistem *geofencing*.
Keamanan Data Presensi	Rentan terhadap manipulasi lokasi (*fake* GPS) dan praktik titip absen (*proxy attendance*).	Dilindungi oleh mekanisme *Mock Location Detection* dan *Face Recognition*.	Menjamin integritas, keaslian, dan validitas data kehadiran karyawan secara berkelanjutan.
Waktu Perhitungan Penggajian	Membutuhkan waktu 1–2 hari kerja untuk sinkronisasi dan perhitungan data secara manual.	Proses kalkulasi berbasis sistem otomatis (*auto-generate*) selesai dalam waktu kurang dari 5 menit.	Mempercepat siklus pembuatan laporan remunerasi dan penggajian bulanan.

Berdasarkan Tabel 8, implementasi SIPRESGA secara signifikan mengoptimalkan tata kelola presensi dan penggajian secara *real-time*. Otomatisasi sistem ini menggantikan rekapitulasi konvensional berbasis *spreadsheet*, sehingga meminimalkan risiko kesalahan manusia (*human error*). Selain itu, aplikasi seluler memberikan fleksibilitas bagi karyawan lapangan. Kendati demikian, sistem ini memiliki beberapa keterbatasan teknis, seperti ketergantungan akurasi algoritma *Haversine* pada stabilitas sinyal GPS dan kerentanan performa *Face Recognition* pada pencahayaan minim serta risiko manipulasi wajah (*face spoofing*) 2D. Variasi spesifikasi perangkat Android juga berpotensi memicu latensi pemrosesan, serta keamanan basis data MySQL masih memerlukan penguatan enkripsi. Keterbatasan tersebut menjadi ruang pengembangan bagi penelitian selanjutnya.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa implementasi algoritma *Haversine*, *Mock Location Detection*, dan *Face Recognition* pada sistem informasi presensi dan penggajian berbasis web dan *mobile* mampu meningkatkan akurasi, keamanan, dan efisiensi proses presensi karyawan di PT Munggaran Jaya Utama. Algoritma *Haversine* berhasil diterapkan untuk melakukan validasi lokasi berdasarkan perhitungan jarak antara posisi pengguna dan titik lokasi presensi yang telah ditentukan. Penerapan *Mock Location Detection* juga mampu membantu mendeteksi indikasi manipulasi lokasi menggunakan *fake* GPS sehingga dapat meminimalkan potensi kecurangan dalam proses presensi. Selain itu, teknologi *Face Recognition* berhasil digunakan untuk memverifikasi identitas pengguna melalui proses swafoto sehingga dapat mengurangi risiko titip absen dan penyalahgunaan identitas. Sistem yang dikembangkan pada penelitian ini mampu mendukung proses pengelolaan data presensi dan penggajian secara terhubung dan berlangsung secara *real-time*. Penerapan aplikasi *mobile* memberikan kemudahan bagi karyawan lapangan dalam melakukan proses presensi, sedangkan aplikasi berbasis web membantu pihak administrator dalam melakukan pemantauan serta pengelolaan data kehadiran dan penggajian secara lebih terorganisasi dan efisien. Berdasarkan hasil pengujian menggunakan metode *Black Box Testing* dan *Usability Testing*, seluruh fitur pada sistem dapat beroperasi sesuai dengan kebutuhan fungsional yang telah ditentukan serta memberikan tingkat kemudahan penggunaan yang baik bagi pengguna. Oleh karena itu, penelitian ini memberikan kontribusi dalam pengembangan sistem presensi digital yang lebih aman, adaptif, dan terintegrasi melalui penerapan validasi lokasi serta autentikasi biometrik pada proses presensi karyawan.

p-ISSN : 2803-1507 e-ISSN : 2803-1531

5.2 Saran

Berdasarkan hasil penelitian yang telah dilakukan, terdapat beberapa rekomendasi yang dapat dijadikan sebagai bahan pengembangan pada penelitian selanjutnya. Pengembangan sistem dapat dilakukan dengan menerapkan teknologi *Artificial Intelligence* (AI) yang lebih lanjut untuk meningkatkan tingkat akurasi pada proses verifikasi wajah serta pendeteksian manipulasi lokasi. Selain itu, penelitian berikutnya juga dapat mengembangkan sistem dengan integrasi teknologi *cloud computing* dan fitur notifikasi secara *real-time* guna mendukung peningkatan performa sistem serta mempermudah proses pemantauan data presensi secara lebih optimal. Penelitian ini masih memiliki beberapa keterbatasan, khususnya pada proses validasi wajah yang dipengaruhi oleh kondisi pencahayaan dan kualitas kamera pada perangkat pengguna. Oleh sebab itu, diperlukan pengembangan lebih lanjut untuk meningkatkan kemampuan teknologi *Face Recognition* agar menghasilkan tingkat akurasi yang lebih baik. Selain itu, pengembangan aplikasi pada platform selain Android juga perlu dilakukan agar sistem dapat digunakan secara lebih luas serta mendukung berbagai jenis perangkat yang digunakan oleh pengguna.

DAFTAR PUSTAKA

[1] S. Ragil, A. B. Putra, and A. S. Fitri, "Sistem Informasi Administrasi Presensi dengan Geolocation dan Penggajian Pada Bukit Darmo Golf," JUSIFOR J. Sist. Inf. dan Inform., vol. 3, no. 1, pp. 10–18, 2024, doi: 10.70609/jusifor.v3i1.3948.

[2] A. R. A. Syafaruddin, N. Natsir, and S. Syafaruddin, "Implementasi Sistem Informasi Akuntansi (SIA) Berbasis Cloud dalam Meningkatkan Efisiensi Operasional Bisnis Kecil," J. Minfo Polgan, vol. 13, no. 2, pp. 1618–1626, 2024, doi: 10.33395/jmp.v13i2.14183.

[3] S. Zahroh, A. Homaidi, and A. Baijuri, "Sistem Informasi Presensi dan Penggajian menggunakan Geolokasi Berbasis Web di Madrasah Tsanawiyah Al-Furqon Curahdami," J. Apl. Teknol. Inf. dan Manaj., vol. 5, no. 1, pp. 1–9, 2024, doi: 10.31102/jatim.v5i1.2601.

[4] Y. Nuryamin et al., "Perancangan Sistem Informasi Absensi Siswa Menggunakan Teknologi QR Code Berbasis Website," J. Sist. Inf. Univ. Suryadarma, vol. 12, no. 1, pp. 85–95, 2014, doi: 10.35968/jsi.v12i1.1342.

[5] Dedy Yusuf, Khoirur Rozikin, and Nuris Dwi Setiawan, "Pengembangan Sistem Presensi Pegawai Desa Berbasis Iot dan RFID untuk Meningkatkan Efisiensi Administrasi," Elkom J. Elektron. dan Komput., vol. 18, no. 2, pp. 18–26, 2025, doi: 10.51903/elkom.v18i2.3234.

[6] P. B. Utomo, D. Wahyudi, and M. Mujiono, "Jurnal Informatika Terpadu Positioning Systems dan Location-Based Service," vol. 11, no. 1, pp. 20–28, 2025.

[7] A. H. Ahadi, T. Djauhari, F. H. Aminuddin, and U. N. Hamzah, "Haversine, N.," Oxford English Dict., vol. 5, no. 1, 2023, doi: 10.1093/oed/4239222416.

[8] M. A. Khair, P. Aldiyuda, N. Enjelina P, M. Z. Zukhrufa, and M. Adrezo, "Perancangan Sistem Absensi Mahasiswa Berbasis Face Recognition di Lingkungan UPN Veteran Jakarta," Inform. J. Ilmu Komput., vol. 20, no. 1, pp. 35–42, 2024, doi: 10.52958/iftk.v20i1.6696.

[9] M. D. Prasetia, A. T. Gultom, L. Leticia, F. N. S. Damanik, and S. J. Pipin, "Pengembangan Aplikasi Presensi Online Berbasis Mobile dengan Penerapan Geolocator dan Face Recognition pada CV. Global Mandiri," J. SIFO Mikroskil, vol. 25, no. 1, pp. 49–66, 2024, doi: 10.55601/jsm.v25i1.1223.

[10] M. Rafli and A. Fauzi, "Perekaman Kehadiran Karyawan dengan Akses Geolokasi: Pendahuluan dalam Era Digitalisasi yang Terus Berkembang, Manajemen Kehadiran Karyawan Memegang Peran Sentral dalam Dunia Kerja" vol. 9, no. 1, pp. 91–102, 2024.

[11] Sofyan Alwi Fadillah, Nico Chandra, and Cyntia Rivatunisa, "Implementasi Agile Scrum pada Pembuatan Website Sistem Informasi Manajemen Kuliner," Decod. J. Pendidik. Teknol. Inf., vol. 4, no. 1, pp. 301–315, 2024, doi: 10.51454/decode.v4i1.357.

[12] N. N. Ahmad, C. A. P, S. O. R. M, A. S. R, and S. J. Z, "Peran Sistem Informasi Manajemen dalam Meningkatkan Efisiensi Operasional dan Pengambilan Keputusan Pada Gojek," Media Ris. Bisnis Ekon. Sains dan Terap., vol. 4, no. 2, pp. 29–37, 2026, doi: 10.71312/mrbest.v4i2.680.

[13] Indriani Azzahra and Rayyan Firdaus, "Penerapan Sistem Informasi Manajemen dalam Meningkatkan Efisiensi Perusahaan," J. Manuhara Pus. Penelit. Ilmu Manaj. dan Bisnis, vol. 2, no. 3, pp. 239–242,

p-ISSN : 2803-1507 e-ISSN : 2803-1531

2024, doi: 10.61132/manuhara.v2i3.1040.

[14] R. F. Az-Zahra Khairunisa, "Perancangan Sistem Presensi dan Penggajian Karyawan Berbasis Web Menggunakan Metode Agile (Studi Kasus: Pt. Sumber Sarana Grafika)," vol. 10, no. 5, 2025.

[15] Valentino Lim, "Perancangan Sistem Presensi Online dan Penggajian Karyawan Berbasis Mobile," J. Komput. Teknol. Inf. Sist. Inf., vol. 4, no. 3, pp. 1669–1679, 2025, doi: 10.62712/juktisi.v4i3.732.

[16] I. Arfianto and I. Ahmad Ashari, "Implementation of Geolocator For Location Manipulation Detection Iin GPS-Based Attendance Application at Watumas Clinic," Heal. Biol. Sci. ICHBS, no. December, pp. 124–140, 2024, [Online]. Available: https://ichbs.uhb.ac.id/index.php/proceeding/article/view/55

[17] Paisal, "Implementasi SCRUM Agile pada Rancang Bangun E-Commerce," vol. 8, no. 1, pp. 212–217, 2025.

[18] R. Indah Melyani, R. Rosita, and S. Aji, "Pengembangan Sistem Informasi Penggajian Berbasis Web menggunakan Framework Laravel dengan Metode Agile Software Development," J. Sist. Inf. Akunt., vol. 3, no. 1, pp. 31–36, 2023, doi: 10.31294/jasika.v3i01.2195.

[19] L. Rahmawati and S. Sumarsono, "Desain Pengembangan Website dengan Arsitektur Model View Controller pada Framework Laravel," J. Teknol. Dan Sist. Inf. Bisnis, vol. 6, no. 4, pp. 785–790, 2024, doi: 10.47233/jteksis.v6i4.1497.

[20] F. Sinlae, I. Maulana, F. Setiyansyah, and M. Ihsan, "Pengenalan Pemrograman Web: Pembuatan Aplikasi Web Sederhana dengan PHP dan MYSQL," J. Siber Multi Disiplin, vol. 2, no. 2, pp. 68–82, 2024, doi: 10.38035/jsmd.v2i2.156.

[21] A. Riyansyah, "Perancangan Sistem Informasi Penggajian Berbasis Web Bandung Laporan Tugas Akhir Perancangan dan Penerapan Sistem Informasi Penggajian Ini Dibuat dengan Tujuan Memecahkan Masalah yang Berhubungan Tentang Proses Pengolahan Data Penggajian dan Menghasilkan," INFOTECH J., vol. 7, no. 2, pp. 29–35, 2021, [Online]. Available: https://doi.org/10.31949/infotech.v7i2.1373

[22] Y. Hudatama Wibowo, "Migrasi Pengembangan Aplikasi Android (Travel Aja) dari React Native ke Kotlin Native menggunakan Metode Scrum," 2023.

[23] K. Andesa, H. Herwin, and T. Nasution, "Go-Gallon App With A Star (A*) Algorithm Implementation Using Android Kotlin," J. Teknol. Dan Open Source, vol. 5, no. 1, pp. 24–34, 2022, doi: 10.36378/jtos.v5i1.2066.

[24] Arya Sultansyah, Astri Sri Rahayu, Iqbal Yudiana, Padjrin Fauzi, Elsa Nur Aripin, and Subhanjaya Angga Atmaja, "Pengujian Black Box Testing pada Fitur Permohonan Informasi Publik melalui Website Pemerintah Jawa Barat," J. Pengabdi. Masy. dan Ris. Pendidik., vol. 3, no. 4, pp. 5912–5919, 2025, doi: 10.31004/jerkin.v3i4.1520.

[25] Feishal Azriel Arya Putra, Bibit Waluyo, Risqi Faturohman, Wahyu Dwi Purwoprasetyo, and Ito Setiawan, "Analisis Usability Testing menggunakan Metode System Usability Scale terhadap Kepuasan Pengguna Website Kemahasiswaan Universitas Amikom Purwokerto," Uranus J. Ilm. Tek. Elektro, Sains dan Inform., vol. 3, no. 1, pp. 121–130, 2025, doi: 10.61132/uranus.v3i1.673.

[26] R. Maududy and D. Rizal Nursyamsi, "Pengembangan Real-Time Monitoring dan Data Logging Berbasis Web pada Proses Robot Painting untuk Meningkatkan Efisiensi Produksi," Informatics Digit. Expert, vol. 5, no. 2, pp. 89–94, 2024, doi: 10.36423/index.v5i1.1586.