

## IMPLEMENTASI SISTEM ABSENSI MAGANG BERBASIS *GEOLOCATION* MENGGUNAKAN METODE *PROTOTYPING*

Tiara Amanda Lestari<sup>1</sup>, Haidar Al-Mutawalli<sup>2</sup>, Yusuf Ramadhan Nasution<sup>3</sup>

<sup>1,2,3</sup> Ilmu Komputer, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sumatera Utara

<sup>1</sup>tiaraamanda7888@gmail.com, <sup>2</sup>haidar0701232073@uinsu.ac.id, <sup>3</sup>ramadhannst@uinsu.ac.id

### ABSTRAK

Proses absensi pada kegiatan magang sering kali masih dilakukan secara manual atau menggunakan aplikasi berbasis web yang belum dilengkapi dengan validasi lokasi secara akurat, sehingga berpotensi menimbulkan kecurangan, seperti melakukan absensi dari luar area kerja. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem absensi magang berbasis *geolocation* dengan menggunakan metode pengembangan *Prototyping*. Sistem ini memanfaatkan *Geolocation API* untuk memperoleh koordinat lokasi pengguna serta menggunakan arsitektur *serverless Firebase Cloud Functions* sebagai *backend* dalam memproses data kehadiran. Validasi lokasi dilakukan dengan membandingkan posisi pengguna terhadap titik lokasi kantor menggunakan perhitungan jarak berdasarkan rumus *Haversine*. Hasil pengujian menunjukkan bahwa sistem mampu melakukan validasi lokasi dengan presisi, di mana sistem berhasil menerima absensi pada jarak valid ( $\leq 100$  meter) dan secara otomatis menolak absensi dengan notifikasi peringatan pada jarak yang melebihi radius, seperti yang terdeteksi pada pengujian jarak 13.789 meter dari titik kantor. Dengan demikian, sistem ini efektif meningkatkan keaslian data kehadiran serta mengurangi potensi kecurangan dalam absensi magang.

**Kata Kunci:** Absensi, Geolocation, Prototyping, Serverless, Firebase

### ABSTRACT

*The attendance process in internship activities is often still carried out manually or using web-based applications that are not yet equipped with accurate location validation, which can potentially lead to fraud, such as taking attendance from outside the work area. This study aims to design and implement a geolocation-based internship attendance system using the Prototyping development method. The system utilizes the Geolocation API to obtain the user's location coordinates and uses a serverless Firebase Cloud Functions architecture as the backend to process attendance data. Location validation is carried out by comparing the user's position with the office location point using distance calculations based on the Haversine formula. The test results demonstrate that the system performs precise location validation, successfully accepting attendance within the valid radius ( $\leq 100$  meters) and automatically rejecting attendance with a warning notification for distances exceeding the radius, as evidenced by a test detection at 13,789 meters from the office location. Thus, this system effectively enhances the authenticity of attendance data and reduces the potential for fraud in internship attendance.*

**Keywords:** Attendance, Geolocation, Prototyping, Serverless, Firebase

## I. PENDAHULUAN

Magang merupakan salah satu program penting yang diselenggarakan oleh institusi pendidikan untuk meningkatkan kompetensi, pengalaman kerja, serta keterampilan *soft skill* mahasiswa sebelum mereka terjun ke dunia industri secara langsung [1]. Program ini menjadi bagian dari kurikulum akademik yang bertujuan untuk mengintegrasikan teori yang diperoleh di bangku perkuliahan dengan praktik di lapangan. Dalam perkembangan teknologi yang semakin canggih, permasalahan pemantauan kehadiran mahasiswa pada saat melakukan praktik kerja industri dapat diatasi dengan sebuah aplikasi digital [2]. Aplikasi merupakan perangkat lunak yang digunakan untuk memenuhi tujuan tertentu, seperti pengolahan data, komunikasi, maupun sistem informasi. Salah satu bentuk penerapannya adalah aplikasi presensi yang berfungsi untuk mencatat dan memantau kehadiran anggota dalam suatu perusahaan, instansi, atau organisasi. Seiring dengan perkembangan teknologi, sistem absensi telah mengalami banyak transformasi, mulai dari sistem manual berbasis kertas, *fingerprint*, kartu identitas, hingga berbasis lokasi (*geolocation*) yang kini menjadi tren karena tingkat kepraktisan dan keamanannya [3].

Pemanfaatan teknologi *geolocation* memungkinkan sistem presensi untuk melakukan validasi kehadiran berdasarkan posisi geografis pengguna [4]. Melalui penggunaan GPS, sistem dapat mendeteksi lokasi pengguna secara *real-time* dan menentukan apakah pengguna berada di dalam radius yang telah ditentukan. Jika pengguna berada di luar radius tersebut, sistem secara otomatis akan menolak proses absensi. Pendekatan ini dinilai efektif dalam mengurangi potensi kecurangan serta meningkatkan akurasi data kehadiran. Beberapa penelitian sebelumnya menunjukkan bahwa penerapan teknologi *geolocation* dan pengenalan wajah mampu meminimalkan manipulasi absensi serta meningkatkan efisiensi dalam pengelolaan data kehadiran [5].

Penelitian ini dilaksanakan di lingkungan Dewan Perwakilan Rakyat Daerah Provinsi Sumatera Utara (DPRD Provsu), di mana proses absensi peserta magang sebelumnya masih dilakukan secara manual sehingga berpotensi menimbulkan ketidak akuratan data dan kecurangan. Untuk mengatasi permasalahan tersebut, dirancang dan diimplementasikan sistem absensi magang berbasis *geolocation* dengan menggunakan metode pengembangan *Prototyping*.

Sistem ini memanfaatkan *Geolocation API* untuk memperoleh koordinat pengguna serta *Firebase Cloud Functions* sebagai arsitektur *serverless* dalam memproses dan menyimpan data kehadiran. Melalui penerapan metode *Prototyping*, sistem dikembangkan secara iteratif dengan melibatkan pengguna untuk memastikan kesesuaian terhadap kebutuhan lapangan. Hasil implementasi sistem ini diharapkan dapat memberikan solusi efektif bagi instansi dalam meningkatkan keaslian data kehadiran, meminimalkan potensi kecurangan, serta mendukung digitalisasi proses administrasi magang di DPRD Provinsi Sumatera Utara.

## II. LANDASAN TEORI

### A. Sistem Absensi Berbasis Web

Sistem absensi merupakan suatu mekanisme yang digunakan untuk mencatat dan memantau kehadiran seseorang dalam suatu kegiatan, baik di lingkungan pendidikan maupun dunia kerja [4]. Sistem informasi absensi adalah rangkaian komponen yang saling berinteraksi untuk mengumpulkan, mengolah, dan menyajikan data kehadiran agar dapat digunakan sebagai dasar pengambilan keputusan [6]. Pada perkembangannya, sistem absensi mengalami transformasi dari metode manual seperti pencatatan kehadiran menggunakan kertas, menjadi sistem digital berbasis perangkat elektronik dan aplikasi daring. Sistem absensi berbasis digital memiliki keunggulan dalam hal efisiensi, akurasi data, serta kemudahan dalam proses rekapitulasi [2]. Dalam konteks penelitian ini, sistem absensi dirancang untuk mencatat kehadiran peserta magang berdasarkan lokasi aktual pengguna menggunakan teknologi *geolocation*.

*Website* adalah sekelompok halaman atau sekumpulan halaman web yang saling berhubungan dan dapat diakses dari mana saja di dunia, selama ada koneksi internet [7]. Setiap halaman *website* memiliki alamat unik disebut (*Uniform Resource Locator*) URL. Sebuah situs web dapat berisi berbagai jenis informasi, termasuk teks, gambar, video, dan audio. Selain itu, situt web fitur interaktif seperti formulir, kontak, komentar, atau obrolan. Situt web terdiri dari dua elemen utama : situt web *client-side* dan *server-side*. *Client-side website* adalah yang di tampilkan kepada pengguna melalui *browser*, seperti Google Chrome atau Internet Explorer. Bagian ini terdiri dari HTML, CSS, dan JavaScript untuk mendesign dan menampilkan halaman web [8].

#### B. Firebase

Firebase *Cloud Functions* merupakan layanan *serverless* yang disediakan oleh Google Firebase untuk menjalankan kode *backend* tanpa memerlukan pengelolaan server secara manual [9]. *Cloud Functions* memungkinkan pengembang untuk mengeksekusi logika bisnis secara otomatis berdasarkan peristiwa tertentu (*event-driven*), seperti penyimpanan data, autentikasi, atau perubahan pada basis data. Arsitektur *serverless* ini memberikan keuntungan berupa skalabilitas otomatis, efisiensi biaya, serta kemudahan integrasi dengan layanan Firebase lainnya, seperti Firestore dan *Authentication*. Dalam penelitian ini, Firebase *Cloud Functions* digunakan sebagai komponen utama dalam proses validasi lokasi dan penyimpanan data absensi. Dengan pendekatan ini, sistem dapat beroperasi dengan cepat dan stabil tanpa memerlukan infrastruktur server konvensional, sehingga lebih efisien untuk diterapkan pada aplikasi absensi berbasis web.

#### C. React js + vite

React adalah kerangka kerja (*framework*) JavaScript. React awalnya diciptakan oleh insinyur di Facebook untuk mengatasi tantangan yang terlibat saat mengembangkan antarmuka pengguna yang kompleks dengan kumpulan data yang berubah dari waktu ke waktu [10]. React, juga dikenal sebagai React.js atau ReactJS, merupakan sebuah pustaka JavaScript yang digunakan untuk membangun antarmuka pengguna (UI) dan komponen-komponennya. Pustaka ini dikembangkan oleh Facebook dan komunitas pengembang lainnya. React digunakan untuk mengelola tampilan halaman web atau aplikasi dengan fokus pada pengelolaan *state* dan *rendering* ke DOM. Namun, untuk fitur-fitur tambahan seperti perutean dan fungsionalitas klien tertentu, aplikasi React membutuhkan penggunaan library tambahan. Dalam arsitektur *Model View Controller* (MVC), React berperan sebagai lapisan *view* yang mengatur tampilan aplikasi. Dirilis pada tahun 2013, React telah menjadi pilihan utama dalam pengembangan aplikasi web, digunakan oleh perusahaan-perusahaan besar seperti Apple, PayPal, dan Netflix, serta digunakan untuk membangun lebih dari 32.000 situs web.

Vite.js adalah sebuah alat pembangunan yang memberikan pengalaman pengembangan yang lebih cepat dan lebih ringkas untuk proyek web modern. Vite.js dikembangkan oleh Evan You, pencipta Vue.js, dan digunakan secara default oleh Vue dan untuk proyek React. Vite.js terdiri dari dua bagian utama: sebuah server pengembangan yang memberikan fitur-fitur yang lebih baik daripada modul ES *native*, seperti

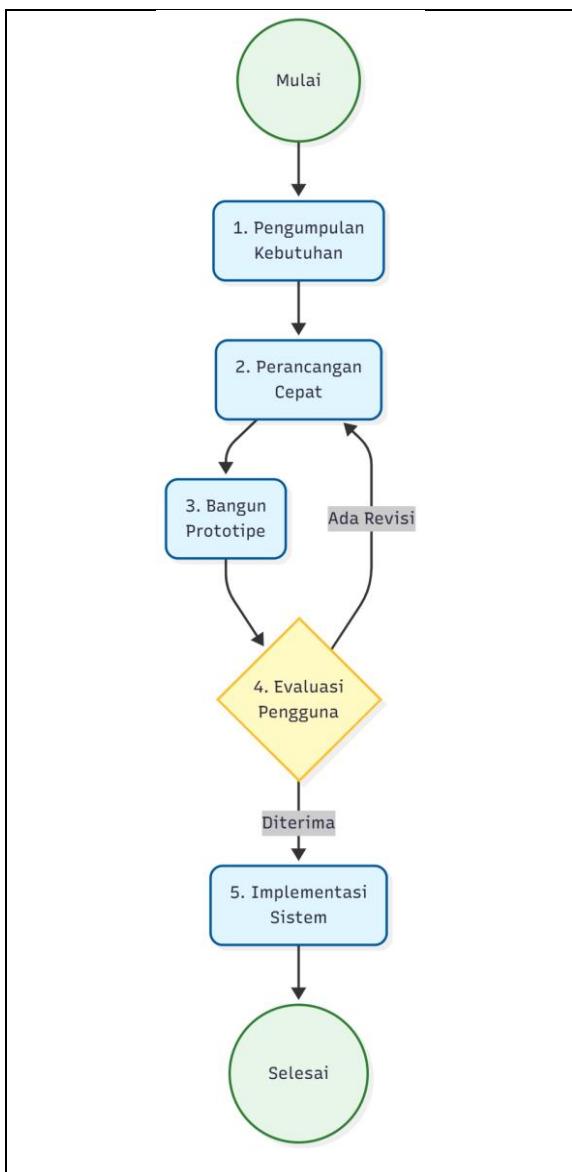
*Hot Module Replacement* (HMR) yang sangat cepat, dan perintah *build* yang mengumpulkan kode dengan *Rollup*, *pre-configured* untuk menghasilkan asset statis yang sangat optimitas untuk produksi. Vite.js juga sangat dapat diperluas melalui API Plugin dan API JavaScript dengan dukungan penuh untuk tipe TypeScript.

#### D. Geolocation

*Geolocation* adalah teknologi yang digunakan untuk menentukan lokasi geografis suatu perangkat secara *real-time* dengan memanfaatkan sinyal dari *Global Positioning System* (GPS), jaringan Wi-Fi, atau menara seluler [11]. *Geolocation* API memungkinkan aplikasi berbasis web atau mobile untuk mengakses informasi lokasi pengguna melalui koordinat lintang dan bujur. Data ini dapat digunakan untuk berbagai keperluan, seperti pelacakan, navigasi, serta validasi lokasi [12]. Dalam sistem absensi berbasis *geolocation*, koordinat lokasi pengguna dibandingkan dengan titik lokasi yang telah ditentukan untuk memverifikasi kehadiran. Validasi dilakukan menggunakan rumus *Haversine*, yaitu rumus yang menghitung jarak antara dua titik berdasarkan koordinat geografis di permukaan bumi. Apabila jarak pengguna berada dalam radius yang telah ditentukan, maka absensi dinyatakan valid. Teknologi ini dinilai efektif untuk mencegah kecurangan seperti absensi jarak jauh (*fake GPS*) karena sistem dapat mengenali posisi aktual pengguna dengan tingkat akurasi tinggi [13].

### III. METODOLOGI PENELITIAN

Metode *Prototyping* merupakan salah satu pendekatan dalam pengembangan perangkat lunak yang menekankan pada pembuatan model awal (*prototype*) dari sistem untuk memperoleh umpan balik langsung dari pengguna sebelum sistem dikembangkan secara penuh [14]. Metode ini bertujuan untuk menjembatani komunikasi antara pengembang dan pengguna agar kebutuhan sistem dapat dipahami secara lebih baik sejak tahap awal. *Prototype* berfungsi sebagai representasi awal dari sistem yang menggambarkan antarmuka, alur, dan fungsionalitas dasar aplikasi. Melalui pendekatan ini, pengguna dapat memberikan tanggapan terhadap rancangan sistem sehingga pengembang dapat melakukan revisi dan penyempurnaan secara iteratif.



Gambar 1. Metode Prototype

Keunggulan metode ini terletak pada fleksibilitas dan efisiensi waktu, karena memungkinkan perbaikan dilakukan secara langsung berdasarkan masukan pengguna [15]. Dalam penelitian ini, metode *Prototyping* dipilih karena sesuai dengan kebutuhan pengembangan sistem absensi magang di DPRD Provinsi Sumatera Utara yang memerlukan proses adaptif dan kolaboratif antara pengembang dan pengguna.

#### A. Implementasi Tahapan *Prototyping*

Berbeda dengan tahapan teoretis yang kaku, implementasi *prototyping* dalam penelitian ini berjalan secara adaptif dan terfokus pada fitur. Berikut adalah tahapan iterasi yang dilakukan dalam membangun sistem:

1) Komunikasi dan Pengumpulan Kebutuhan: Tahap ini adalah analisis kebutuhan fungsional dan non-fungsional [1]. Kebutuhan fungsional utama yang teridentifikasi adalah: (a) sistem harus mampu melakukan autentikasi dua peran (Admin dan Peserta); (b) sistem harus dapat mencatat absensi peserta menggunakan validasi *geolocation*; (c) Admin harus dapat mengelola data peserta dan mengkonfigurasi aturan absensi. Kebutuhan non-fungsional adalah sistem harus berbasis web, responsif, dan aman.

2) Perancangan Cepat dan Pembangunan Prototipe (Iteratif): Tahap ini diulang dalam beberapa iterasi untuk membangun sistem secara bertahap. Iterasi 1 berfokus pada fitur inti peserta (autentikasi dan absensi *geolocation*). Iterasi 2 membangun panel admin untuk pemantauan *real-time* dan verifikasi *role* pengguna. Iterasi 3 menambahkan fitur konfigurasi dinamis untuk aturan absensi (jam telat, hari libur). Iterasi 4 menyempurnakan sistem dengan fitur manajemen data lengkap seperti koreksi absensi dan data detail mahasiswa.

3) Evaluasi Prototipe oleh Pengguna: Setelah setiap iterasi, *prototype* dievaluasi. Umpan balik dari pengguna (Admin) digunakan untuk memperbaiki dan mengarahkan pengembangan iterasi berikutnya [12]. Contohnya, kebutuhan akan halaman "Manajemen Kehadiran" muncul setelah mengevaluasi prototipe "Dashboard Admin".

#### B. Arsitektur Sistem

Sistem ini dirancang menggunakan arsitektur *client-server* modern dengan pendekatan *serverless*. Sisi *client* (antarmuka pengguna) dibangun sebagai *Single Page Application* (SPA) sementara sisi *server* mengandalkan layanan *Backend-as-a-Service* (BaaS) dari Firebase [9].

1) *Client-Side (Frontend)*: Dibangun menggunakan pustaka JavaScript React (React.js) dan di-*bundle* menggunakan *build tool* Vite [10]. Seluruh komponen antarmuka (UI) menggunakan *library* Material-UI (MUI). Navigasi antar halaman dikelola oleh React Router DOM.

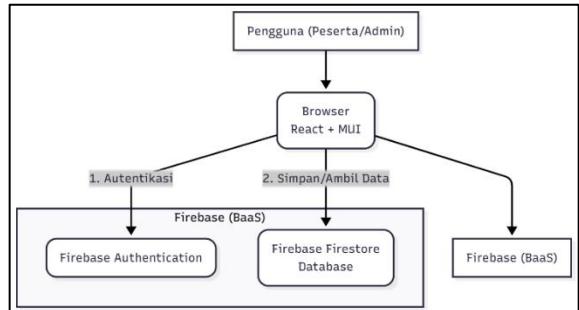
2) *Server-Side (Backend)*: Sistem ini memanfaatkan layanan Firebase secara penuh. Dua layanan inti yang digunakan adalah Firebase Authentication untuk mengelola autentikasi pengguna, dan Firebase Firestore sebagai *database* NoSQL *real-time* untuk menyimpan seluruh data aplikasi, termasuk *users*, *lokasi\_logs*, dan *settings*.

#### IV. HASIL DAN PEMBAHASAN

##### A. Hasil Perancangan Sistem

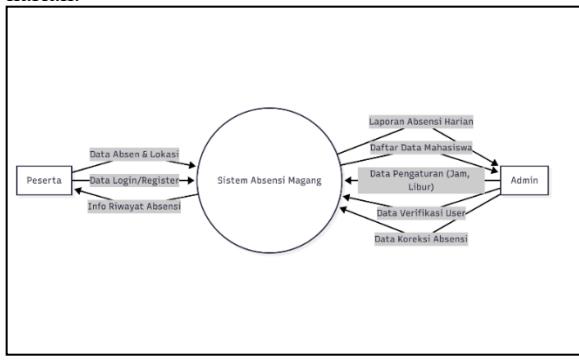
Berdasarkan metodologi *prototyping* yang dijelaskan pada Bagian III, dihasilkan beberapa artefak perancangan yang menjadi cetak biru (*blueprint*) bagi pengembangan sistem.

1) Arsitektur Sistem: Sistem ini dirancang menggunakan arsitektur *client-server* modern dengan pendekatan *serverless* (BaaS). Sisi client (antarmuka pengguna) bertugas menangani tampilan dan interaksi pengguna, sementara sisi server (Firebase) bertugas menangani logika autentikasi dan penyimpanan data. Arsitektur sistem secara visual diilustrasikan pada Gambar 2.



Gambar 2. Arsitektur Sistem

2) Diagram Konteks (DFD Level 0): Untuk memvisualisasikan alur data utama pada sistem, dibuatlah Diagram Konteks. Diagram ini mengidentifikasi dua aktor utama, yaitu Admin dan Peserta. Diagram Konteks yang diusulkan dapat dilihat pada Gambar 3. Peserta mengirimkan data absensi dan menerima informasi riwayat, sementara Admin mengirimkan data pengaturan (seperti hari libur dan jam telat) serta menerima laporan absensi harian.



Gambar 3. DFD Level 0

3) Skema Database (Rancangan Koleksi): Karena sistem ini menggunakan *database* NoSQL (Firebase Firestore), perancangan data difokuskan pada struktur koleksi. Terdapat tiga koleksi utama

yang saling berelasi seperti ditunjukkan pada Gambar 4 dan 5, yaitu: *users* (menyimpan data dan peran pengguna), *lokasi\_logs* (menyimpan setiap catatan absensi), dan *settings* (menyimpan aturan bisnis aplikasi).

The diagram illustrates three database collections: *users*, *lokasi\_logs*, and *settings*. The *users* collection is shown as a table with columns: string uid (PK), string namaLengkap, string email, string role, string nim, string universitas, string fakultas, string jurusan, string periodeMagang, and string fotoProfilURL. A relationship is shown between the *users* collection and a user icon, labeled 'memiliki' (possesses). The *lokasi\_logs* collection is shown as a table with columns: string logId (PK), string userId (FK), string email, datetime timestamp, float latitude, float longitude, float jarakDariKantor, string status, and string alasan. The *settings* collection is shown as a table with columns: string settingId (PK), string settingName, string settingValue, and string settingType.

users			
string	uid	PK	ID Pengguna (PK)
string	namaLengkap		
string	email		
string	role		
string	nim		
string	universitas		
string	fakultas		
string	jurusan		
string	periodeMagang		
string	fotoProfilURL		

lokasi_logs			
string	logId	PK	ID Log (PK)
string	userId	FK	ID Pengguna (FK)
string	email		
datetime	timestamp		
float	latitude		
float	longitude		
float	jarakDariKantor		
string	status		
string	alasan		

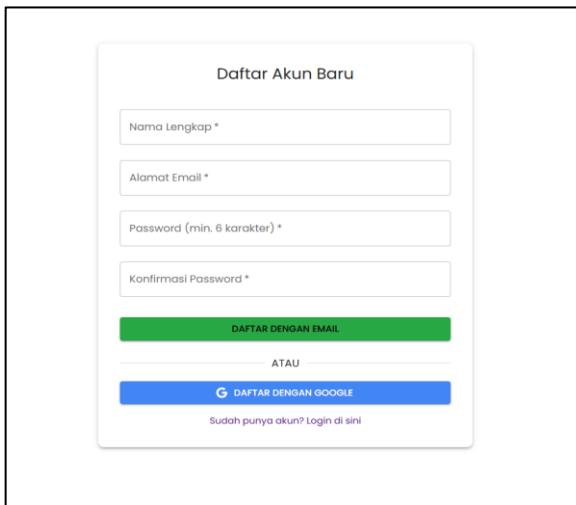
Gambar 4. Koleksi *Users* dan *Lokasi*

settings			
string	docId	PK	ID Dokumen (app_settings)
datetime	latenessThreshold		
object	weeklyHolidays		
array	specialHolidays		
float	officeLat		
float	officeLong		

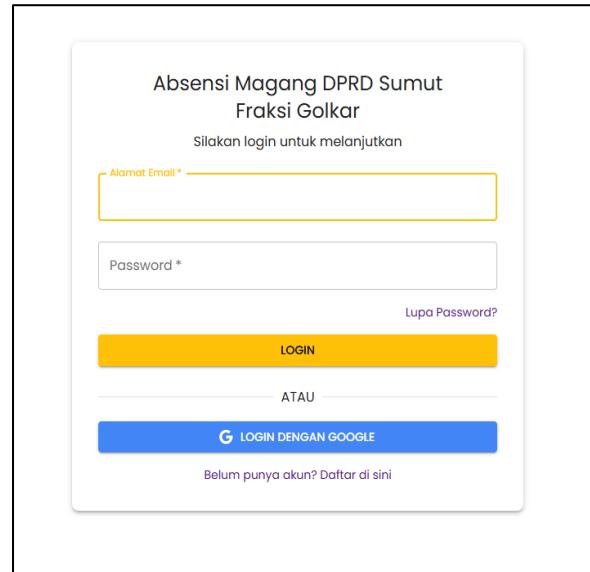
Gambar 5. Koleksi *Settings*

#### B. Implementasi Antarmuka Sistem

1) Antarmuka Autentikasi Pengguna: Halaman autentikasi mencakup proses *Login* dan *Registrasi*. Fitur keamanan utama pada tahap ini adalah implementasi sistem *role*, di mana setiap pengguna baru yang mendaftar akan secara otomatis mendapatkan *role* "pending". Status ini mengharuskan Admin untuk melakukan verifikasi manual (melalui *ManajemenUserPage.jsx*) sebelum peserta dapat mengakses dasbor absensi. Berikut adalah tampilan antarmuka yang dibangun. Gambar 6 menampilkan halaman *Registrasi* yang memungkinkan pengguna baru mendaftar menggunakan email atau akun Google. Setelah terdaftar, pengguna dapat masuk melalui halaman *Login* yang ditampilkan pada Gambar 7.

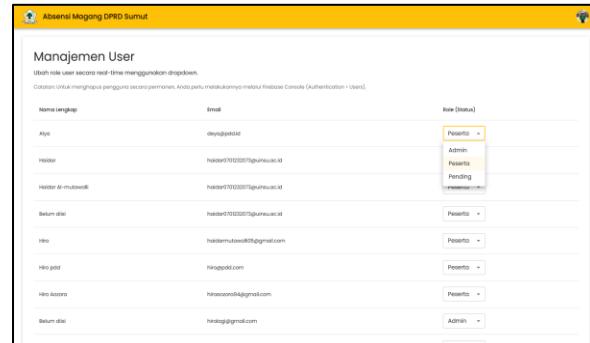


Gambar 6. Tampilan Halaman *Register*



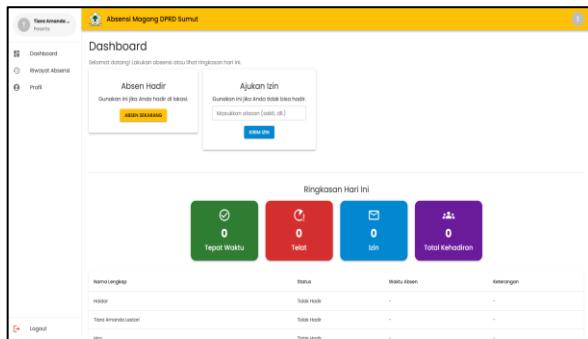
Gambar 7. Tampilan Halaman *Login*

Fitur keamanan utama pada tahap autentikasi adalah implementasi sistem *role*. Setiap pengguna baru secara otomatis mendapatkan status "pending". Gambar 8 menunjukkan halaman *Manajemen User* di sisi Admin, di mana Admin harus melakukan verifikasi manual dengan mengubah status pengguna dari "pending" menjadi "peserta" melalui menu *dropdown* agar pengguna tersebut dapat mengakses fitur utama.



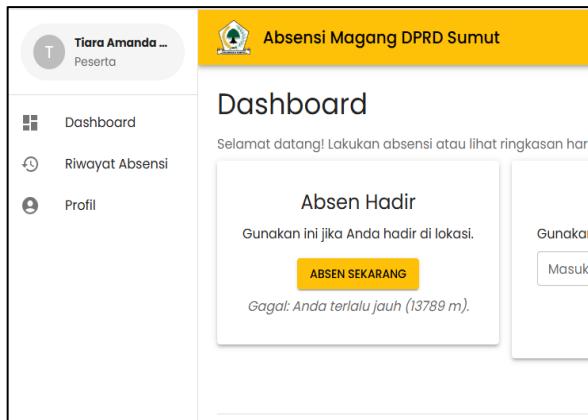
Gambar 8. Tampilan *Manajemen User* oleh Admin

2) Antarmuka Fitur Absensi (Peserta): Ini adalah fitur inti dari sistem. Halaman *Dasbor Peserta* (*DashboardPage.jsx*) menyediakan dua aksi utama: "Absen Sekarang" dan "Ajukan Izin". Ketika tombol "Absen Sekarang" ditekan, sistem memicu *Geolocation API* (*navigator.geolocation*) pada *browser* untuk mendapatkan koordinat presisi tinggi. Koordinat ini kemudian divalidasi terhadap radius kantor yang telah ditentukan. Sistem secara otomatis menentukan status "Tepat Waktu" atau "Telat" berdasarkan batas waktu yang diambil dari koleksi *settings*. Berikut adalah antarmuka yang dibangun.



Gambar 9. Tampilan Halaman Dashboard User

Sistem dilengkapi dengan validasi *geolocation* yang ketat. Gambar 10 menunjukkan respons sistem ketika pengguna mencoba melakukan absensi dari lokasi yang tidak valid. Sistem menampilkan pesan kesalahan yang spesifik, menginformasikan bahwa jarak pengguna saat ini melebihi batas radius yang diizinkan dari titik kantor.



Gambar 10. Tampilan Halaman Jika User Belum Masuk Radius

Jika pengguna berada dalam radius yang valid, sistem akan memproses data kehadiran. Gambar 11 menampilkan status "Tepat Waktu" dengan indikator visual berwarna hijau apabila absensi dilakukan sebelum batas waktu yang ditentukan. Sebaliknya, jika absensi dilakukan melewati batas waktu, sistem akan mencatatnya sebagai "Telat" dengan indikator merah seperti terlihat pada Gambar 13.



Gambar 11. Tampilan Halaman Jika User Tepat Waktu

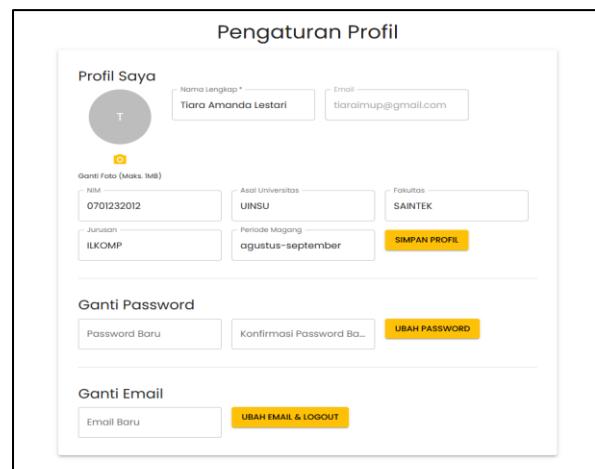


Gambar 12. Tampilan Input Izin

Selain kehadiran fisik, peserta dapat mengajukan izin ketidakhadiran. Gambar 12 menampilkan modal pop-up input izin di mana peserta wajib mengisi alasan ketidakhadiran sebelum mengirimkan data. Peserta juga dapat mengelola informasi pribadi mereka melalui halaman Pengaturan Profil yang ditunjukkan pada Gambar 14.

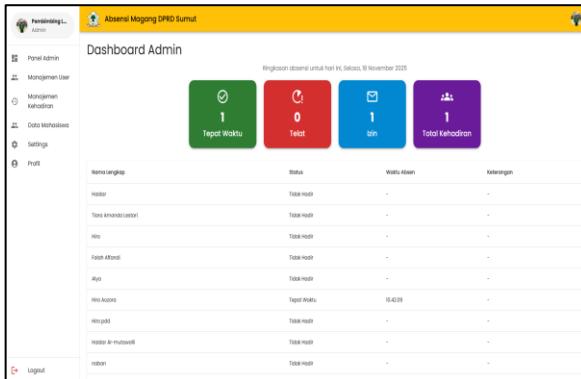


Gambar 13. Tampilan Halaman Jika User Terlambat



Gambar 14. Tampilan Pengaturan Profil User

3) Antarmuka Panel Admin: Admin memiliki akses ke panel khusus untuk mengelola dan memantau sistem. Halaman utama admin (AdminPage.jsx) menyajikan ringkasan kehadiran harian. Gambar 15 menampilkan *Dashboard* Admin yang menyajikan ringkasan data kehadiran seluruh peserta secara *real-time* dalam bentuk kartu statistik.



The dashboard displays the following statistics:

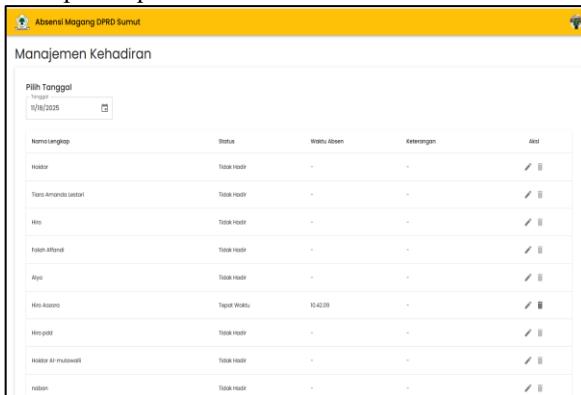
- Tepat Waktu: 1
- Tidak: 0
- Izin: 1
- Total Kehadiran: 1

Attendance list:

Nama Lengkap	Status	Waktu Absen	Keterangan
Holder	Tidak Hadir	-	-
Tiara Amanda Lestari	Tidak Hadir	-	-
Him	Tidak Hadir	-	-
Fath Affandi	Tidak Hadir	-	-
Aisy	Tidak Hadir	-	-
Hin Azizra	Tepat Waktu	12.42.09	-
Hin gold	Tidak Hadir	-	-
Holder Al-mutawill	Tidak Hadir	-	-
notin	Tidak Hadir	-	-

Gambar 15. Tampilan Dashboard Admin

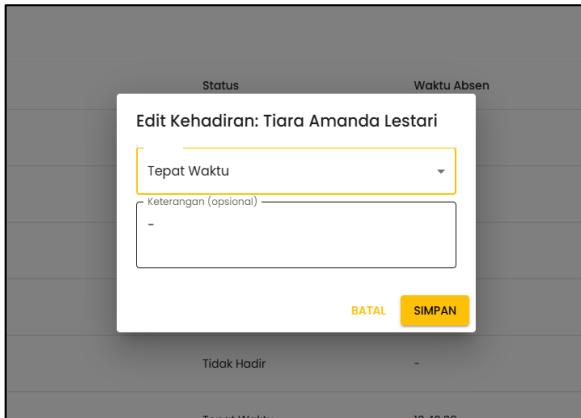
Untuk pengelolaan data harian, Admin menggunakan halaman Manajemen Kehadiran seperti pada Gambar 16. Halaman ini memungkinkan Admin melihat daftar absensi per tanggal. Jika terjadi kesalahan *input* atau status, Admin dapat melakukan koreksi data melalui fitur Edit Kehadiran yang ditampilkan pada Gambar 17.



Pilih Tanggal: 1/11/2025

Nama Lengkap	Status	Waktu Absen	Keterangan	Aksi
Holder	Tidak Hadir	-	-	✓
Tiara Amanda Lestari	Tidak Hadir	-	-	✓
Him	Tidak Hadir	-	-	✓
Fath Affandi	Tidak Hadir	-	-	✓
Aisy	Tidak Hadir	-	-	✓
Hin Azizra	Tepat Waktu	12.42.09	-	✓
Hin gold	Tidak Hadir	-	-	✓
Holder Al-mutawill	Tidak Hadir	-	-	✓
notin	Tidak Hadir	-	-	✓

Gambar 16. Tampilan Halaman Manajemen Kehadiran



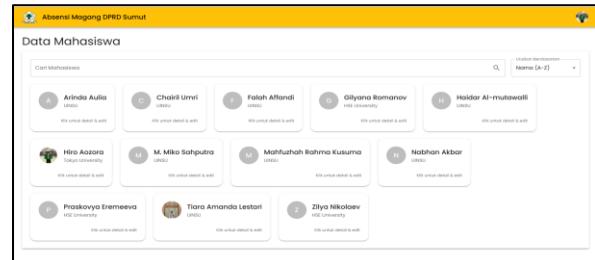
Status: Tepat Waktu

Keterangan (opsional):

BATAL SIMPAN

Gambar 17. Tampilan Edit data Kehadiran

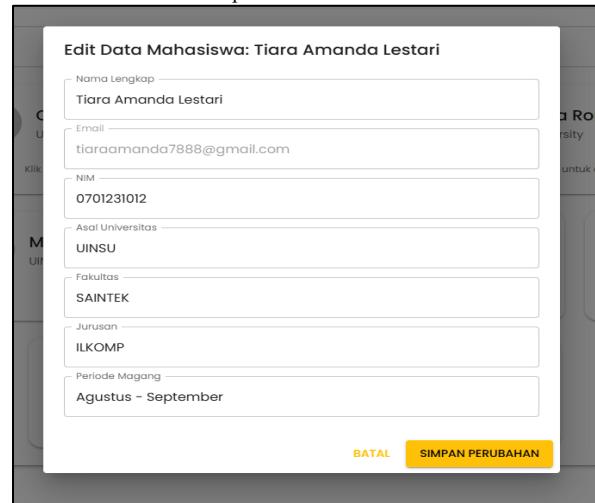
Admin juga dapat mengelola data induk mahasiswa melalui halaman Data Mahasiswa (Gambar 18) dan melakukan pembaruan informasi melalui formulir edit yang terlihat pada Gambar 19.



Daftar Mahasiswa:

Profile	Nama	Keterangan
Arinda Aulia	Arinda Aulia	Tidak Hadir
Chairil Umri	Chairil Umri	Tidak Hadir
Fath Affandi	Fath Affandi	Tidak Hadir
Gilyana Romanov	Gilyana Romanov	Tidak Hadir
Holder Al-mutawill	Holder Al-mutawill	Tidak Hadir
Hiro Azizra	Hiro Azizra	Tidak Hadir
M. Miko Saptutra	M. Miko Saptutra	Tidak Hadir
Mohfuzhah Bahma Kusuma	Mohfuzhah Bahma Kusuma	Tidak Hadir
Nafisah Akbar	Nafisah Akbar	Tidak Hadir
Prasiknya Ermeeva	Prasiknya Ermeeva	Tidak Hadir
Tiara Amanda Lestari	Tiara Amanda Lestari	Tidak Hadir
Zhy Nikolev	Zhy Nikolev	Tidak Hadir

Gambar 18. Tampilan Halaman Data Mahasiswa



Edit Data Mahasiswa: Tiara Amanda Lestari

Nama Lengkap: Tiara Amanda Lestari

Email: tiaraamanda7888@gmail.com

NIM: 0701231012

Asal Universitas: UINSU

Fakultas: SAINTEK

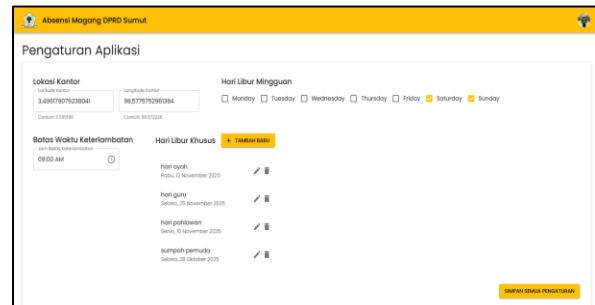
Jurusan: ILKOMP

Periode Magang: Agustus - September

BATAL SIMPAN PERUBAHAN

Gambar 19. Tampilan Edit Data Mahasiswa

Salah satu fitur unggulan dari metode *prototyping* adalah fleksibilitas konfigurasi. Gambar 20 menampilkan halaman Pengaturan (Settings) di mana Admin dapat menentukan titik koordinat kantor, batas waktu keterlambatan, dan hari libur mingguan. Selain itu, Gambar 21 menunjukkan fitur untuk menambahkan hari libur khusus (nasional) secara dinamis tanpa perlu mengubah kode program. Admin juga memiliki halaman profil sendiri seperti pada Gambar 22.



Lokasi Kantor: Jl. Pemuda No. 12, Bandar Lampung, 80111, Indonesia

Hari Libur Mingguan:

- Monday:
- Tuesday:
- Wednesday:
- Thursday:
- Friday:
- Saturday:
- Sunday:

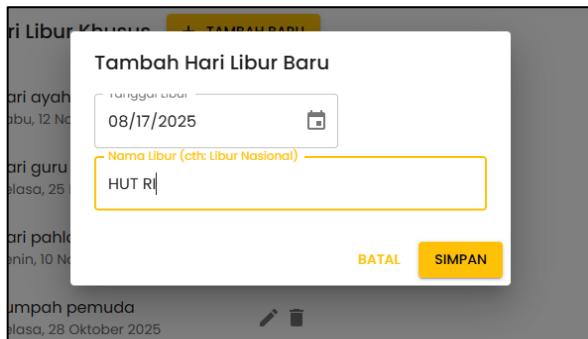
Batas Waktu Keterlambatan: 09.00 AM

Hari Libur Khusus:

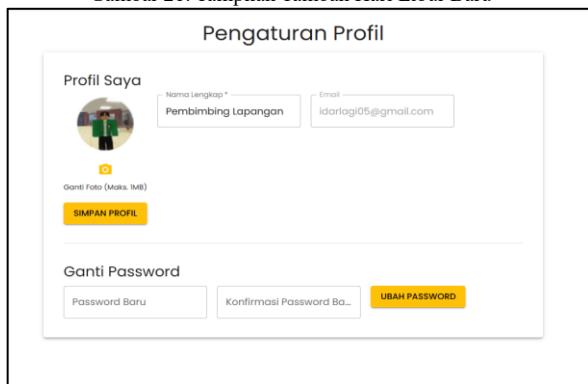
- Hari Raya: 01/11/2025
- Hari Guru: 25/11/2025
- Hari Pahlawan: 10/11/2025
- Surpiah Perduka: 28/11/2025

Simpan Perubahan

Gambar 20. Tampilan Halaman Settings



Gambar 21. Tampilan Tambah Hari Libur Baru



Gambar 22. Tampilan Pengaturan Profil

### C. Hasil Pengujian Sistem

Pengujian sistem dilakukan menggunakan metode *Black Box Testing* untuk memvalidasi fungsionalitas utama dari sistem yang telah diimplementasikan. Pengujian ini berfokus pada kesesuaian antara *input* yang diberikan dan *output* yang diharapkan dari berbagai skenario penggunaan, baik dari sisi Peserta maupun Admin. Hasil pengujian skenario fungsional disajikan pada Tabel 1.

Tabel 1. Hasil Pengujian Sistem

No.	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Autentikasi: Admin mengubah <i>role</i> pengguna dari "pending" menjadi "peserta".	Pengguna dengan <i>role</i> "peserta" berhasil <i>login</i> dan masuk ke dasbor peserta.	Pengguna berhasil <i>login</i> dan diarahkan ke /app/dashboard.	Valid
2.	Absensi (Valid): Peserta menekan tombol "Absen Sekarang" di dalam radius	Sistem menyimpan data absensi ke lokasi_log s dengan status "Tepat Waktu".	Data absensi berhasil disimpan. Status "Tepat Waktu".	Valid

	lokasi yang ditentukan (< 100m).	Waktu" (jika sebelum jam telat).		
3.	Absensi (Invalid - Jarak): Peserta menekan tombol "Absen Sekarang" di luar radius lokasi yang ditentukan (> 100m).	Sistem menolak absensi dan menampilkan pesan error "Gagal: Anda terlalu jauh...".	Sistem menolak dan menampilkan pesan error jarak.	Valid
4.	Absensi (Valid - Telat): Peserta absen di dalam radius, namun setelah batas jam keterlambatan (09:00).	Sistem menyimpan data absensi ke lokasi_log s dengan status "Telat".	Data absensi berhasil disimpan. Status "Telat".	Valid
5.	Absensi (Izin): Peserta menekan tombol "Kirim Izin" dengan alasan yang diisi.	Sistem menyimpan data ke lokasi_log s dengan status "Izin" dan alasan terisi.	Data berhasil disimpan. Status "Izin".	Valid
6.	Admin (Koreksi): Admin mengubah status absensi peserta (dari "Telat" menjadi "Izin").	Data di lokasi_log s berhasil diperbarui.	Data absensi peserta tersebut berhasil di-update di database.	Valid
7.	Admin (Settings): Admin mengubah hari "Sabtu" dari libur ( <i>true</i> ) menjadi tidak libur ( <i>false</i> ).	Pengaturan tersimpan. Peserta bisa melakukan absensi di hari Sabtu.	Pengaturan tersimpan. Halaman dasbor peserta tidak lagi menampilkan "Hari Ini Libur" pada hari Sabtu.	Valid

#### D. Pembahasan

Implementasi sistem ini berhasil memvalidasi dua aspek utama penelitian: penggunaan teknologi *geolocation* untuk akuntabilitas kehadiran dan efektivitas metode *prototyping* dalam pengembangan sistem yang adaptif.

Pembahasan pertama berfokus pada akurasi validasi kehadiran fisik. Sistem manual memiliki kelemahan signifikan dalam verifikasi lokasi yang sering kali dimanipulasi. Dengan mengimplementasikan *Geolocation API* dan rumus *Haversine*, sistem ini mampu menghitung jarak pengguna terhadap titik koordinat kantor secara presisi. Berdasarkan hasil pengujian lapangan pada Tabel 1 (Skenario 2 dan 3), sistem menunjukkan konsistensi tinggi. Secara spesifik, sistem berhasil mendeteksi pengguna yang berada dalam radius aman (5-15 meter) dan mengizinkan proses absensi. Sebaliknya, pada pengujian di luar lokasi (Gambar 10), sistem secara akurat mendeteksi posisi pengguna yang berada pada jarak 13.789 meter dari titik kantor. Karena nilai ini jauh melampaui toleransi radius 100 meter, sistem secara otomatis memblokir akses tombol absensi dan memberikan umpan balik visual. Data kuantitatif ini membuktikan bahwa mekanisme validasi lokasi berfungsi efektif mencegah kecurangan absensi jarak jauh (*remote attendance*).

Pembahasan kedua menyoroti relevansi metode *prototyping*. Kebutuhan admin yang spesifik, seperti kemampuan untuk mengubah jam batas keterlambatan atau menambah hari libur nasional secara mandiri, tidak sepenuhnya terlihat pada tahap analisis awal. Metode *prototyping* memungkinkan fitur-fitur konfigurasi ini (seperti pada Gambar 20) untuk ditambahkan pada iterasi ketiga sebagai respons langsung terhadap evaluasi pengguna. Tanpa siklus iteratif ini, sistem yang dihasilkan akan bersifat kaku dan memerlukan intervensi *programmer* setiap kali ada perubahan aturan jam kerja atau hari libur. Hasil pengujian Skenario 7 memvalidasi bahwa perubahan pengaturan oleh admin berdampak langsung (*real-time*) pada logika bisnis sistem di sisi pengguna.

Terakhir, dari sisi arsitektur teknis, pemilihan teknologi *serverless* menggunakan Firebase terbukti meningkatkan efisiensi pengembangan. Penggunaan *Backend-as-a-Service* (BaaS) memungkinkan pengembang untuk fokus sepenuhnya pada logika bisnis vital—yaitu validasi *geolocation* dan aturan absensi—tanpa perlu mengalokasikan sumber daya untuk mengelola infrastruktur server, *database*, dan *endpoint API* secara manual. Hal ini mempercepat siklus pengembangan setiap iterasi *prototype* secara signifikan dibandingkan pendekatan pengembangan *backend* tradisional.

#### KESIMPULAN

Penelitian ini telah berhasil merancang dan mengimplementasikan sistem absensi magang berbasis *geolocation* dengan menggunakan metode *Prototyping*. Penerapan metode *Prototyping* terbukti efektif dalam pengembangan sistem ini, memungkinkan adanya adaptasi fitur secara iteratif berdasarkan evaluasi dan umpan balik. Hal ini terlihat dari penambahan fitur-fitur krusial seperti panel konfigurasi admin dan manajemen *role* pengguna yang dikembangkan pada iterasi lanjutan sebagai respons atas kebutuhan pengguna.

Berdasarkan hasil pengujian fungsional (*Black Box Testing*), sistem yang diimplementasikan terbukti berjalan valid sesuai dengan rancangan. Sistem ini secara fungsional mampu memvalidasi lokasi peserta magang secara akurat, membedakan antara absensi di dalam dan di luar radius yang ditentukan, serta menerapkan aturan keterlambatan secara otomatis. Dengan demikian, implementasi sistem ini berhasil menjawab permasalahan utama penelitian, yaitu meningkatkan akuntabilitas, menjamin keaslian data kehadiran, dan mengurangi potensi kecurangan yang terdapat pada sistem absensi manual.

Untuk pengembangan di masa mendatang, sistem ini memiliki beberapa area yang dapat disempurnakan. Penelitian selanjutnya dapat berfokus pada implementasi deteksi *mock location* (*fake GPS*) untuk lebih memperkuat keamanan data.

#### REFERENSI

- [1] L. Rahmania, “Pengembangan Sistem Pengelolaan Kegiatan Magang Mahasiswa Polman Negeri Babel Berbasis Android dengan Metode Prototype,” 2025.
- [2] H. A. Rahmadhani dan I. S. S. Putri, “Rancang Bangun Sistem Absensi Peserta Magang Berbasis Web di PT Jasamarga RO3,” *Jutisi: Jurnal Ilmiah Teknik Informatika dan Sistem Informasi*, vol. 14, no. 2, hlm. 1087–1098, Agu 2025, doi: 10.35889/jutisi.v14i2.2812.
- [3] L. Hidayati, M. D. Irawan, dan N. R. Nasution, “Implementation of the Prototype Method in Designing an Android-Based Pramubakti Attendance Application with Geolocation Features,” *Jurnal IPTEK Bagi Masyarakat*, vol. 3, no. 3, hlm. 92–106, Apr 2024.
- [4] D. Purwanto, R. E. Putri, Y. Fadly, dan D. C. Pratiwi, “Sistem Absensi Online Berbasis Web Dengan Penggunaan Teknologi GPS,” *Jurnal Minfo Polgan*, vol. 13, no. 2, hlm. 1800–1811, Nov 2024, doi: 10.33395/jmp.v13i2.14258.
- [5] T. Marlein Tamtelahitu, J. Sambono, dan J. E. Unenor, “Perancangan Sistem Absensi Pintar Mahasiswa Menggunakan Teknik QR Code Dan Geolocation,” *JIPI Jurnal Ilmiah Penelitian dan Pembelajaran Informatika*, vol. 6, no. 1, hlm. 114–125, Jun 2021.
- [6] T. A. Pertiwi, N. T. Luchia, P. Sinta, R. Aprinastya, A. Dahlia, dan I. R. Fachrezi, “Perancangan dan Implementasi Sistem Informasi Absensi Berbasis Web Menggunakan Metode Agile Software Development,”

*Jurnal Testing dan Implementasi Sistem Informasi*, vol. 1, no. 1, hlm. 53–66, 2023.

[7] A. H. Manullang, M. Aritonang, dan M. J. Purba, “Sistem Informasi Bimbingan Belajar Number One Medan Berbasis Web,” *TAMKA: Jurnal Tugas Akhir Manajemen Informatika & Komputerisasi Akuntansi*, vol. 1, no. 1, hlm. 44–49, Jun 2021, doi: 10.46880/tamika.Vol1No1.pp44-49.

[8] T. Limbong dan Sriadi, *Pemrograman Web Dasar*. Medan: Yayasan Kita Menulis, 2021.

[9] E. G. Harahap, “Perancangan dan Prototype Aplikasi Konsultan Skripsi Berbasis Android dengan Google Api/Firebase,” Skripsi, Universitas Teknologi Digital Indonesia, Yogyakarta, 2024.

[10] Marcellino, “Analisa dan Perancangan Sistem Informasi Berbasis Web Pada Inventaris Barang Menggunakan Framework React.Js,” Skripsi, Universitas Buddhi Dharma, Tangerang, 2024.

[11] A. Prayogo, “Analisis dan Perancangan Sistem Pengawasan Teknisi Lapangan Menggunakan Teknologi Geolocation Berbasis Web Dan Mobile,” Skripsi, Universitas Satya Negara Indonesia, 2022.

[12] I. Muthmainnah, F. Wahyu Christanto, J. Manurung, dan M. Sidiq, “Implementasi Metode Geolocation Menggunakan Teknologi Webcam pada Sistem Absensi Pegawai,” *JSIG: JURNAL SISTEM INFORMASI GALUH*, vol. 2, no. 2, hlm. 2024, Jul 2024, [Daring]. Tersedia pada: <https://ojs.unigal.ac.id/index.php/jsig/index>

[13] A. C. Wardhana, A. R. Hasan, dan T. Rijanandi, “Pengembangan dan Evaluasi Sistem Presensi Pegawai dengan Data Geolocation Menggunakan Metode Prototipe,” *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 5, hlm. 1386–1392, Okt 2022, doi: 10.30865/jurikom.v9i5.4909.

[14] A. S. Wahyuni, A. I. Arifin, Nurdiasih, dan W. Haryono, “Sistem Aplikasi Berbasis Web SMK Solusi Efisien untuk Monitoring Kehadiran Siswa SMK Menggunakan Model Prototype,” *Modem: Jurnal Informatika dan Sains Teknologi*, vol. 3, no. 3, hlm. 37–46, Jul 2025, doi: 10.62951/modem.v3i3.514.

[15] R. Sanuri, M. Muzakar, A. B. Thoha, dan H. A. Raihan, “Pengembangan Aplikasi Live Video Streaming Berbayar Untuk Konser Musik Indonesia Menggunakan Progressive Web Apps,” dalam *Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST) 2024*, Yogyakarta: Universitas AKPRIND Indonesia, Nov 2024, hlm. 321–331.