

# PERANCANGAN DAN IMPLEMENTASI SISTEM PRESENSI BERBASIS QR CODE DINAMIS DENGAN VALIDASI LOKASI DAN JARINGAN LOKAL UNTUK MENCEGAH KECURANGAN PRESENSI

**Henri Donand Kristanto; Endang Wahyu Pamungkas, S.Kom., M.Kom., Ph.D.**  
**Program Studi Teknik Informatika, Fakultas Komunikasi dan Informatika,**  
**Universitas Muhammadiyah Surakarta**

## Abstrak

Perkembangan teknologi telah mendorong perubahan dalam sistem presensi. Sistem presensi manual memiliki kelemahan seperti kurangnya efisiensi dan tingginya potensi manipulasi kehadiran. Berbagai Solusi digital seperti *QR Code* statis pun masih mudah untuk disalahgunakan. Diperlukan sebuah inovasi yang menjadi tujuan penelitian ini, yaitu untuk membuat sistem pencatatan kehadiran berbasis *QR Code* dinamis dengan validasi geolokasi dan jaringan lokal untuk mencegah kecurangan tersebut. Sistem ini dirancang dan diujicobakan pada konteks pencatatan kehadiran mahasiswa dalam sesi perkuliahan di lingkungan kampus. Pengembangan sistem presensi ini menggunakan metode *Software Development Life Cycle* (SDLC) dengan pendekatan model *waterfall*. Metode ini meliputi beberapa tahapan, yaitu analisa kebutuhan, desain sistem, implementasi, integrasi, pengujian, dan pemeliharaan. Luaran yang dihasilkan dari penelitian ini adalah *website* yang dikembangkan menggunakan *framework* Django. Dari sisi Pengujian fungsionalitas, sistem ini menggunakan metode *Black box testing*. seluruh fitur utama sistem sistem telah berjalan sesuai dengan kebutuhan fungsional. Dari sisi *usability*, sistem diuji menggunakan sistem penilaian *System Usability Scale* (SUS) dan menghasilkan skor sebesar 73,42. Skor ini termasuk dalam kategori *Good* (Baik) yang menunjukkan bahwa sistem layak dan mudah untuk digunakan. Selain itu, hasil kuesioner evaluasi menunjukkan respons yang sangat positif, di mana 96,7% responden setuju bahwa sistem mampu mengurangi peluang tipis absen, serta 93,3% responden bersedia merekomendasikan sistem ini. Penelitian ini menghasilkan sistem presensi QR Code dinamis dengan validasi ganda yang berhasil diimplementasikan dan menunjukkan performa serta persepsi pengguna yang positif dalam upaya mencegah tindakan kecurangan presensi.

**Kata Kunci:** *QR Code* Dinamis, Sistem Presensi, Geolokasi, Jaringan Lokal, Keamanan, Sistem, Anti Kecurangan.

## Abstract

Continuous shifts in technology have transformed roll-call tracking. Traditional manual verification and even basic digital alternatives, like stationary QR codes, remain highly vulnerable to inefficiency and fraud. To address these flaws, this research develops a web-oriented attendance architecture deploying time-variant dynamic QR codes integrated with geographic positioning and local intranet verification. The engineering pipeline follows the SDLC Waterfall framework, executing sequentially through needs assessment, structural blueprinting, coding, integration, testing, and maintenance. Built upon the Django framework, the resulting website underwent functional validation via Black-box testing, confirming that all core operations satisfy design specifications. Regarding user experience, evaluation using the System Usability Scale (SUS) yielded a score of 73.42, placing the platform in the "Good" tier and affirming its deployment readiness. Survey responses further validate these findings, with 96.7% of participants agreeing the system eliminates proxy attendance and 93.3% willing to recommend it. This research successfully implemented a dynamic QR Code attendance system with dual validation, which demonstrated positive performance and user perception in supporting efforts to prevent attendance fraud.

**Keywords:** Dynamic QR Code, Attendance System, Geolocation, Local Network, Security System, Anti-Fraud

## 1. PENDAHULUAN

Perkembangan teknologi di era digital telah merubah berbagai aspek kehidupan dengan sangat cepat, termasuk dalam bidang administratif dan pendidikan (SNEINEH Anees & Wael, 2025). Salah satu bentuk perubahan yang terjadi adalah digitalisasi dalam sistem presensi yang sebelumnya masih dilakukan secara manual dan konvensional seperti tanda tangan (Alda et al., 2024). Sistem presensi ini dinilai kurang efisien dan tidak efektif, rentan terhadap kesalahan pencatatan, serta besarnya potensi kecurangan seperti manipulasi absensi atau kehadiran (Perin, 2025).

Berbagai teknologi presensi biometrik mulai dikembangkan, seperti teknologi presensi *fingerprint* dan *face recognition*. Teknologi *fingerprint* adalah sistem yang menggunakan konsep *biometric* yang mengidentifikasi seseorang melalui pola unik dari sidik jari (Pauzan & Yanti, 2022). Teknologi *face recognition* yaitu teknologi yang memanfaatkan algoritma *machine learning* untuk mengidentifikasi identitas seseorang berdasarkan fitur-fitur di wajah (Firdaus & Hidayat, 2025). Kedua teknologi tersebut memiliki beberapa keterbatasan, seperti ketergantungan akan perangkat keras khusus, biaya implementasi yang tinggi serta kompleksitas integrasi yang mengharuskan pengguna untuk langsung datang ke perangkat untuk melakukan presensi (Nguyen-Tat et al., 2024).

Sistem pencatatan kehadiran mempunyai peran yang sangat penting dalam memantau partisipasi dan kedisiplinan seseorang (Chethan & Bharath, 2025). Sistem presensi merupakan strategi penting yang diterapkan oleh institusi untuk mengelola kehadiran (Emmanuel Tamuno, 2025). Seiring perkembangan teknologi, berbagai institusi berupaya membangun sistem presensi digital *Quick Response Code (QR Code)* (Imanullah & Reswan, 2022). *QR Code* merupakan gambar 2 dimensi yang berisi simbol matriks di dalam sel berbentuk persegi (Mulya & Wulan Sari, 2024). *QR Code* dinilai dapat menjadi solusi sebagai alat otomatisasi pengambilan data presensi. *QR Code* memungkinkan proses pencatatan kehadiran dilakukan secara cepat dan tepat karena prosesnya hanya dilakukan melalui perangkat *smartphone* (Husni et al., 2024). Namun sistem ini masih memiliki kelemahan yang memungkinkan terjadinya kecurangan presensi (Mohammed & Zidan, 2023).

Sebagai pembanding, beberapa penelitian terdahulu telah mengembangkan sistem presensi dengan pendekatan berbeda. Kombinasi *QR Code* dinamis, geofencing, dan pemeriksaan *IMEI* (Nwabuwe et al., 2023). Aplikasi web absensi berbasis geolokasi dengan *Laravel* (Hasna, 2026). *QR Code* dengan kontrol berbasis IP menggunakan *CodeIgniter* dan *MySQL* (Prasetyo, 2025). Ketiga penelitian tersebut terbukti meningkatkan transparansi dan akuntabilitas presensi, namun masih menyisakan celah keamanan sebagaimana dirangkum pada Tabel 1.

Tabel 1. Tabel *gap research*

| Aspek  | Hasna (2026)     | Prasetyo (2025)  | Nwabuwe (2023) | Penelitian Ini   |
|--------|------------------|------------------|----------------|------------------|
| Metode | <i>Waterfall</i> | <i>Waterfall</i> |                | <i>Waterfall</i> |

|                 |                        |                          |                                  |                                        |
|-----------------|------------------------|--------------------------|----------------------------------|----------------------------------------|
| Teknologi       | Laravel                | CodeIgniter, MySQL       | React native, node js            | Django                                 |
| Metode Presensi | Geolokasi              | QR Code, IP              | QR Code dinamis, geolokasi, IMEI | QR Code Dinamis, Geolokasi, Jaringan   |
| Kelebihan       | Validasi lokasi        | Validasi QR dan jaringan | <i>Multi layer validation</i>    | <i>Multi layer validation, praktis</i> |
| Kekurangan      | Rentan <i>fake</i> GPS | QR Code statis           | Tidak praktis                    |                                        |

Berdasarkan permasalahan tersebut, diperlukan perancangan sistem presensi berbasis *QR Code* yang berubah secara berkala sehingga presensi sulit dimanipulasi. penelitian ini mengusulkan pengembangan sistem presensi yang memiliki validasi ganda. Absensi hanya bisa dilakukan Ketika pengguna melakukan presensi dalam wilayah dengan radius yang telah ditentukan dan pengguna telah terhubung dengan jaringan lokal yang tersedia. dengan 2 teknik validasi ini data absensi menjadi lebih akurat dan dapat mencegah terjadinya manipulasi absensi.

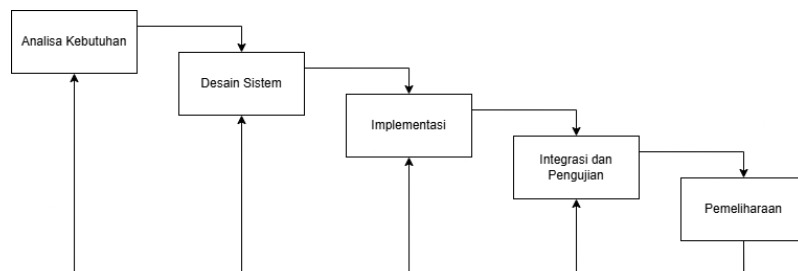
Hasil akhir dari penelitian ini diharapkan dapat mampu memberikan kontribusi secara teoritis dan praktis. Dari aspek teoritis, penelitian ini diharapkan dapat menjadi referensi dalam pengembangan sistem presensi berbasis keamanan digital. Sementara dari aspek praktis, platform yang telah dirancang dan dikembangkan diharapkan dapat membantu institusi dan organisasi dalam mencegah praktik kecurangan absensi.

## 2. METODE

Proses pengembangan dalam penelitian ini dijalankan secara terstruktur dan sistematis dengan menggunakan metode *Software Development Life Cycle* (SDLC) dengan pendekatan model *waterfall*. SDLC sangat penting untuk keberhasilan proyek pengembangan perangkat lunak mengingat kerangka kerja ini menawarkan metodologi yang terstruktur dan terencana dalam pengembangan *software* (Al-Tahat et al., 2026). Pemilihan kerangka kerja ini diterapkan karena dinilai mampu mengakomodasi spesifikasi dan alur kerja sistem pencatatan kehadiran yang sudah matang sejak fase awal, sehingga memastikan seluruh proses pengembangan berjalan secara terstruktur dan terdokumentasi. Model *waterfall* dipilih menggantikan pendekatan *agile* yang bersifat iteratif, mengingat kebutuhan sistem telah terdefinisi secara matang sejak tahap awal (Saravanos & Curinga, 2023). Model *waterfall* lebih sesuai digunakan karena penelitian ini tidak memerlukan perubahan yang signifikan selama proses pengembangan.

Sebagai sebuah cetak biru, SDLC memetakan seluruh siklus hidup rekayasa perangkat lunak secara menyeluruh, mulai dari fase perencanaan awal hingga perawatan sistem pasca rilis. Model *waterfall* merupakan model sistematis yang mensyaratkan perencanaan menyeluruh harus diselesaikan secara penuh sebelum tahap selanjutnya atau tahap eksekusi dimulai (Mishra & Ganesh, 2026). Melalui pendekatan yang terorganisasi, model ini mempermudah proses desain sekaligus

implementasi sistem. Rangkaian prosedur model *waterfall* sendiri meliputi analisis kebutuhan, desain sistem, implementasi, integrasi pengujian, dan pemeliharaan. Berikut prosedur tahapan model *waterfall* pada Gambar 1.



Gambar 1. Gambar Model Waterfall

## 2.1 Analisis kebutuhan

Fase awal dalam metodologi *waterfall* dimulai dengan mengidentifikasi kebutuhan sistem melalui tahap analisa kebutuhan. Fase ini melibatkan studi kepustakaan serta penjaringan aspirasi mengenai kebutuhan sistem pencatatan kehadiran melalui penyebaran kuesioner.

Berdasarkan hasil analisis melalui kuesioner yang telah disebarakan kepada responden, diperoleh informasi mengenai kebutuhan pengguna. Temuan dari hasil kuesioner menegaskan adanya beberapa kelemahan pada sistem pencatatan kehadiran saat ini. Berangkat dari temuan tersebut, sistem yang akan dirancang dikelompokkan ke dalam dua spesifikasi utama, yakni kebutuhan fungsional serta kebutuhan non fungsional

### 2.1.1 Kebutuhan Fungsional

Kebutuhan fungsional Mendefinisikan seluruh kapabilitas dan fitur-fitur inti yang harus tertanam dalam sistem pencatatan kehadiran ini. Kebutuhan fungsional meliputi: Sistem menghadirkan fitur login untuk *admin*. *Admin* dapat membuat sesi presensi untuk setiap pertemuan. *Admin* dapat mengatur sesi presensi untuk setiap pertemuan. Sistem menghasilkan *QR Code* dinamis yang berubah setiap interval waktu tertentu. Sistem menyediakan formulir untuk pengisian data presensi. Pengguna dapat mengisi formulir presensi. Sistem melakukan validasi lokasi pengguna berdasarkan koordinat geolokasi. Sistem melakukan validasi jaringan berdasarkan jaringan yang telah ditentukan. Sistem menyimpan data presensi pengguna ke dalam basis data. *Admin* dapat melihat daftar pengguna yang telah melakukan presensi.

### 2.1.2 Kebutuhan Non Fungsional

Kebutuhan non fungsional mendefinisikan seluruh kapabilitas sistem yang berkaitan dengan spesifikasi sistem dan lingkungan yang mendukung proses perancangan sekaligus operasional sistem. Adapun detail mengenai spesifikasi non fungsional dari platform yang akan dikembangkan yaitu :

- a. Kebutuhan Pengembang: OS Windows 10/11, *Framework* Django, Visual Studio Code, *Ngrok*, *Database Management System*, *Web browser*, *Version Control System*.

- b. Kebutuhan Pengguna: Perangkat smartphone, browser, kamera smartphone dan akses ke jaringan *WIFI* dan *GPS*.
- c. Keamanan: Sistem harus mampu mencegah berbagai kecurangan presensi.
- d. Performa: Sistem harus mampu memproses permintaan pengguna secara cepat dalam wa
- e. Usability: Sistem didesain dengan antarmuka yang mudah digunakan.

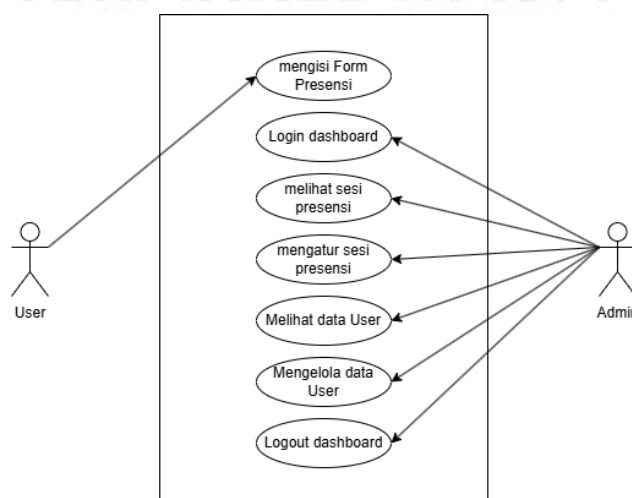
## 2.2 Desain Sistem

Fase desain sistem merupakan fase membuat rancangan desain sistem secara menyeluruh menurut hasil analisis kebutuhan sistem. Pada tahap ini dilakukan perancangan struktur sistem, perancangan alur proses sistem, serta perancangan antarmuka pengguna. Tahap perancangan arsitektur dalam penelitian ini memanfaatkan instrumen pemodelan yang terstruktur seperti *Use Case Diagram*, *Activity Diagram* dan *Entity Relationship Diagram*. Kombinasi diagram ini berfungsi untuk memvisualisasikan interaksi antara *user* dan sistem serta memetakan logika alur secara menyeluruh.

### 2.2.1 Use Case Diagram

Pemodelan melalui *use case diagram* menyajikan gambaran umum mengenai interaksi dan batasan antar aktor dengan sistem melalui fungsionalitas fitur-fitur spesifik yang disediakan *software* tersebut. *Use case diagram* bertujuan memvisualisasikan dan merapikan fitur-fitur yang terdapat pada sistem (Mornie et al., 2025). Melalui visualisasi ini, pembaca dapat dengan mudah memahami peran masing-masing aktor serta memahami secara rinci umpan balik yang dihasilkan sistem atas setiap tindakan yang dilakukan oleh pengguna.

Pada sistem presensi yang dirancang, terdapat dua aktor utama yaitu *admin* dan *user*. peran *admin* dalam sistem lebih berfokus pada pengelolaan dan penyediaan sistem presensi, sedangkan *user* berperan sebagai pihak yang melakukan proses presensi melalui pemindaian *QR Code* dan pengisian data kehadiran. Rancangan *Use Case Diagram* pada sistem presensi ini ditunjukkan pada Gambar 2.



Gambar 2. Use Case Diagram



### 2.2.2 Activity Diagram

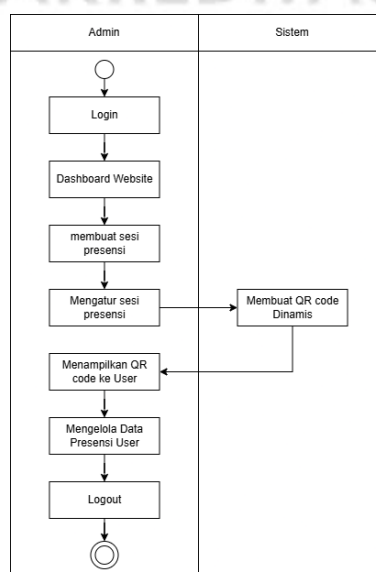
Menurut (Napitupulu et al., 2025), *activity diagram* berfungsi sebagai representasi visual yang memetakan dan menjelaskan alur kerja serta merepresentasikan proses operasional yang berjalan pada suatu sistem. Diagram ini membantu menjelaskan langkah-langkah yang dilakukan oleh aktor serta bagaimana sistem merespons setiap tindakan yang terjadi selama proses berlangsung.

Arsitektur sistem pencatatan kehadiran ini memetakan dua jenis alur *activity diagram* yang merepresentasikan interaksi dari sisi admin dan user. Alur operasionalnya dimulai saat admin mengakses sistem melalui halaman *login*. Setelah verifikasi *login* dinyatakan valid, admin dapat melihat dan mengelola menu pada halaman *dashboard*. Melalui *dashboard* tersebut, *admin* dapat membuat dan mengatur sesi presensi. Ketika sesi presensi dibuat, sistem akan menghasilkan *QR Code* dinamis sebagai media presensi. *QR Code* dinamis dihasilkan berdasarkan token yang dibentuk dari kombinasi gabungan ID sesi, waktu interval dan secret key yang dihitung menggunakan persamaan :

$$w = \left\lfloor \frac{T}{I} \right\rfloor \quad (1)$$

$$token = SHA - 256(sesi\_id || W || secret\_key) \quad (2)$$

Pada persamaan (1) W merupakan nilai waktu interval, T merupakan waktu unix saat ini dan I merupakan panjang interval dalam satuan detik. Pada persamaan (2) token terbentuk dari gabungan sesi\_id, nilai waktu interval dan secret key yang nantinya akan digabung lalu dienkripsi menggunakan *hashing* SHA-256. Token ini akan berubah dalam interval waktu tertentu sehingga *QR Code* yang dihasilkan juga ikut berubah secara berkala. *QR Code* yang telah dihasilkan kemudian ditampilkan kepada *user* untuk dipindai. *admin* memiliki otoritas untuk melakukan rekapitulasi seluruh data kehadiran. Begitu seluruh proses administrasi selesai dijalankan, pihak *admin* dapat melakukan tindakan *logout*. Rancangan *activity diagram admin* ditunjukkan pada Gambar 3.



Gambar 3. Activity Diagram Admin

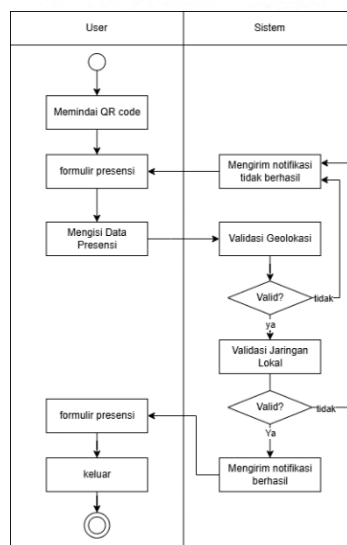
Sementara itu, *activity diagram* untuk *user* memvisualisasikan seluruh rangkaian prosedur pencatatan kehadiran yang berjalan di dalam sistem. Alur diawali dengan pengguna melakukan pemindaian *QR Code*. Setelah berhasil meminda, *user* diarahkan ke halaman formulir presensi untuk mengisi data. Setelah form presensi dikirimkan, sistem akan melakukan proses validasi token terlebih dahulu untuk memastikan bahwa *QR Code* yang digunakan masih valid. Selanjutnya sistem melakukan validasi geolokasi dengan membandingkan koordinat lokasi *user* yang diperoleh dari perangkat dengan titik lokasi yang telah ditentukan. Sistem memperoleh koordinat *user* secara real time dari perangkat melalui browser, lalu menghitung jarak terhadap titik lokasi presensi menggunakan formula Haversine:

$$\alpha = \sin^2 \left( \frac{\Delta lat}{2} \right) + \cos (lat_1) \cdot \cos (lat_2) \cdot \sin^2 \left( \frac{\Delta long}{2} \right) \quad (3)$$

$$c = 2 \cdot \arctan 2(\sqrt{a} \cdot \sqrt{1 - a}) \quad (4)$$

$$d = R \cdot c \quad (5)$$

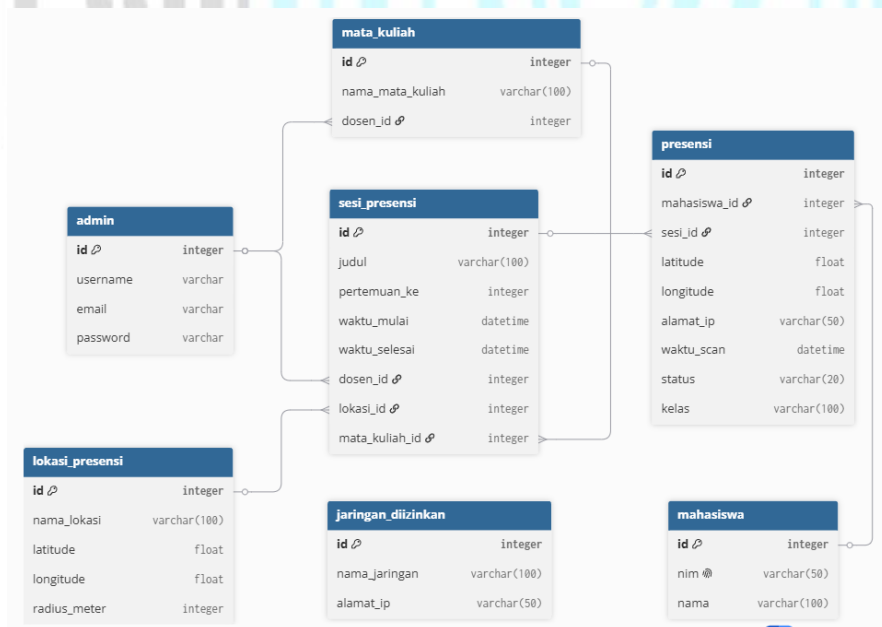
$\Delta lat$  dan  $\Delta long$  merupakan selisih antara 2 *latitude* dan 2 *longitude*. Pada persamaan (3) menghitung nilai  $\alpha$  sebagai nilai utama Haversine. Nilai  $\alpha$  digunakan untuk memperoleh sudut sentral  $c$  seperti pada persamaan (4). Pada persamaan (5) akan dihasilkan estimasi jarak yang akurat untuk sistem geolokasi dengan cara mengalikan jari-jari bumi  $R$  sebesar 6.371km dengan nilai  $c$ . *User* dianggap valid ketika radius yang diizinkan lebih besar sama dengan dari  $d$  atau jarak antara dua titik. Selain itu, sistem juga melakukan validasi jaringan lokal dengan memeriksa alamat IP atau jaringan yang digunakan oleh *user* dan membandingkannya dengan daftar jaringan yang diizinkan dalam sistem. Apabila seluruh proses validasi tersebut berhasil, sistem akan menyimpan data presensi ke dalam basis data dan menampilkan notifikasi bahwa presensi berhasil dilakukan. Rancangan *activity diagram user* ditunjukkan pada Gambar 4.



Gambar 4. Activity Diagram User

### 2.2.3 Entity relationship diagram

Untuk merancang struktur penyimpanan pada basis data serta memvisualisasikannya, digunakan *entity relationship diagram*. Diagram ini mempermudah pemahaman bagaimana hubungan antar komponen data pada sistem pencatatan kehadiran ini saling berinteraksi. Pada sistem ini terdapat beberapa entitas utama yaitu *admin*, *sesi\_presensi*, *presensi*, *mahasiswa*, *mata\_kuliah*, *lokasi\_diizinkan* dan *jaringan\_diizinkan*. Entitas *admin* berfungsi sebagai tempat penyimpanan data bagi administrator yang bertanggung jawab atas pengelolaan sistem pencatatan kehadiran ini. Entitas *sesi\_presensi* menyimpan informasi mengenai kegiatan presensi yang dilakukan dalam setiap pertemuan. Setiap sesi presensi dapat menghasilkan beberapa token qr yang digunakan untuk menghasilkan *QR Code* dinamis selama proses presensi berlangsung. Selanjutnya entitas mahasiswa digunakan untuk menyimpan data *user* yang melakukan presensi, sedangkan data kehadiran mahasiswa disimpan pada entitas *presensi* yang memiliki relasi dengan entitas mahasiswa dan *sesi\_presensi*. Entitas *mata\_kuliah* berisi daftar mata kuliah untuk memudahkan dalam mencari mata kuliah tertentu. Terdapat entitas *lokasi\_presensi* yang berisi daftar lokasi yang telah didaftarkan. Selain itu terdapat entitas *jaringan\_diizinkan* yang berfungsi sebagai referensi jaringan yang diperbolehkan dalam proses presensi untuk mendukung proses validasi jaringan. Hubungan antar entitas dalam sistem ini sebagian besar bersifat *one to many* (1:M) yang memungkinkan satu entitas memiliki keterkaitan dengan beberapa entitas lainnya. Berikut adalah tampilan ERD pada Gambar 5.



Gambar 5. Entity relationship diagram

## 2.3 Implementasi

Menurut (Rakhmadi et al., 2025), fase implementasi merupakan sebuah proses untuk mentransformasikan atau merealisasikan cetak biru rancangan arsitektur yang telah dirancang pada tahap sebelumnya menjadi sebuah bentuk aplikasi yang siap dieksekusi. Pada tahap ini dilakukan



proses pengkodean (*coding*) untuk membangun sistem presensi berbasis web menggunakan *framework* Django sebagai *backend* utama. Django digunakan untuk mengelola logika sistem, pengolahan data, serta komunikasi antara antarmuka *user* dan basis data.

Basis data berguna untuk menyimpan berbagai informasi yang berkaitan dengan sistem presensi, seperti data *user*, data sesi presensi, token *QR Code*, serta data kehadiran *user*. HTML, CSS, dan JavaScript digunakan untuk membangun sistem sehingga menghasilkan tampilan yang interaktif dan mudah digunakan.

Proses implementasi meliputi pengembangan fitur sesi presensi, pembuatan *QR Code* dinamis yang berubah secara berkala, serta proses presensi oleh *user* melalui pemindaian *QR Code*. Selain itu, sistem juga mengimplementasikan mekanisme validasi geolokasi untuk memastikan *user* berada dalam radius lokasi yang ditentukan, serta validasi jaringan lokal untuk memastikan presensi dilakukan melalui jaringan yang telah diizinkan. Dengan tahap implementasi ini, rancangan sistem yang sebelumnya masih berupa konsep dan desain mulai diwujudkan menjadi sebuah aplikasi yang dapat dijalankan dan diuji coba.

## **2.4 Integrasi dan pengujian**

Tahap pengujian harus dilakukan guna memastikan bahwa fitur dan fungsionalitas pada sistem yang telah dirancang dan dikembangkan dapat beroperasi sesuai rancangan (Imaduddin & Hayati, 2023). Pengujian sistem bertujuan untuk menemukan kesalahan pada sistem dan memastikan bahwa semua bagian dalam sistem dapat berjalan dengan lancar (Khatib Sulaiman et al., 2024). Metode pengujian yang digunakan dalam penelitian ini adalah *Black Box Testing*, *System Usability Scale*, dan pengujian lainnya. Metode *Black Box Testing* yaitu metode pengujian yang hanya berfokus pada evaluasi fungsi sistem tanpa memerlukan pengetahuan lebih dalam kode sumber sistem (Aryadi et al., 2026). Pengujian dilakukan dengan cara mencoba berbagai fitur yang tersedia dalam sistem. Hasil dari pengujian ini digunakan untuk memastikan bahwa sistem dapat berjalan sesuai dengan rancangan kebutuhan fungsional. Selain itu, dilakukan evaluasi terhadap kemudahan penggunaan sistem menggunakan metode *System Usability Scale* (SUS).

Pengujian *System Usability Scale* (SUS) ini melibatkan 30 responden yang dipilih menggunakan teknik *purposive sampling*, yaitu pemilihan responden berdasarkan kriteria tertentu yang relevan dengan konteks penggunaan sistem. Responden yang dilibatkan terdiri dari mahasiswa, siswa, pegiat organisasi, asisten dosen, serta mentor program yang memiliki pengalaman atau keterlibatan langsung dalam proses pencatatan kehadiran, baik sebagai peserta maupun penyelenggara kegiatan. Kriteria ini dipilih agar responden dapat memberikan penilaian yang relevan terhadap kemudahan penggunaan sistem dalam konteks nyata pencatatan kehadiran. Adapun jumlah 30 responden digunakan mengacu pada rekomendasi umum dalam pengujian usability, di mana

ukuran sampel tersebut dinilai cukup representatif untuk menghasilkan estimasi skor SUS yang stabil, sebagaimana banyak digunakan dalam beberapa penelitian sebelumnya. Para responden diminta untuk mengisi kuesioner yang terdiri dari 10 pernyataan dengan skala Likert 1–5, mulai dari sangat tidak setuju hingga sangat setuju. Adapun pernyataan dalam kuesioner SUS disajikan pada Tabel 2.

Tabel 2. Daftar pernyataan SUS

|     | <b>Pernyataan</b>                                                              |
|-----|--------------------------------------------------------------------------------|
| Q1  | Saya berpikir akan menggunakan sistem ini lagi.                                |
| Q2  | Saya merasa sistem ini rumit untuk digunakan.                                  |
| Q3  | Saya merasa sistem ini mudah digunakan.                                        |
| Q4  | Saya membutuhkan bantuan dari orang lain dalam menggunakan sistem ini.         |
| Q5  | Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.                 |
| Q6  | Saya merasa ada banyak hal yang tidak konsisten.                               |
| Q7  | Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat. |
| Q8  | Saya merasa sistem ini membingungkan.                                          |
| Q9  | Saya merasa tidak ada hambatan dalam menggunakan sistem ini.                   |
| Q10 | Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.    |

Perhitungan skor *system usability scale* dilakukan dengan cara mengurangi skor pada pernyataan ganjil dengan 1, dan untuk pernyataan genap dilakukan pengurangan dari 5. Hasil dari seluruh skor kemudian dijumlahkan dan dikalikan dengan 2,5 untuk mendapatkan nilai akhir dalam rentang 0–100. Terdapat pengujian evaluasi sistem yang dilakukan dengan menggunakan kuesioner berbasis skala Likert untuk mengukur persepsi *user* terhadap sistem yang telah dikembangkan. Pengujian ini dilakukan untuk mengukur tingkat keberhasilan sistem. Perlu ditegaskan bahwa pengukuran efektivitas sistem pada penelitian ini menggunakan pendekatan pengujian fungsional dan persepsi pengguna, bukan perbandingan empiris terhadap tingkat kecurangan presensi sebelum dan sesudah penerapan sistem. Pendekatan ini dipilih karena keterbatasan waktu dan akses terhadap data presensi riil pada periode sebelum penerapan sistem. Sebelum kuesioner evaluasi sistem disebarkan kepada responden, instrumen ini terlebih dahulu melalui proses validasi ahli untuk menilai validitas isi, yaitu relevansi dan kejelasan setiap item pernyataan terhadap konstruk yang diukur. Validasi ahli ini dilakukan oleh Ihsan Cahyo Utomo, S.Kom., M.Kom. dosen Sistem Jaringan Universitas Muhammadiyah Surakarta, yang dipilih karena memiliki kompetensi yang relevan dengan mekanisme validasi jaringan pada sistem yang dikembangkan. Berdasarkan hasil validasi, seluruh 11 item pernyataan dinyatakan relevan dengan konstruk yang diukur, dan instrumen secara keseluruhan dinyatakan layak digunakan tanpa revisi. pernyataan dalam kuesioner disajikan pada Tabel 3.

Tabel 3. Daftar pernyataan evaluasi sistem

| <b>Bagian</b> | <b>P</b> | <b>Pernyataan evaluasi sistem</b>                                                    |
|---------------|----------|--------------------------------------------------------------------------------------|
| A             | P1       | Praktik titip absen masih umum terjadi di lingkungan perkuliahan.                    |
|               | P2       | Sistem presensi yang ada saat ini belum cukup efektif mencegah manipulasi kehadiran. |
|               | P3       | Presensi manual kurang mampu memastikan kehadiran mahasiswa secara                   |

|   |     |                                                                                                      |
|---|-----|------------------------------------------------------------------------------------------------------|
|   |     | langsung.                                                                                            |
| B | P4  | QR Code yang berubah secara berkala mempersulit praktik titip absen.                                 |
|   | P5  | Validasi lokasi membantu memastikan mahasiswa berada di area perkuliahan.                            |
|   | P6  | Validasi jaringan WiFi lokal membantu memastikan mahasiswa benar-benar berada di lokasi perkuliahan. |
| C | P7  | Sistem ini lebih aman dibanding presensi manual.                                                     |
|   | P8  | Sistem ini lebih efektif dalam memverifikasi kehadiran mahasiswa.                                    |
|   | P9  | Sistem ini dapat mengurangi peluang terjadinya kecurangan presensi                                   |
|   | P10 | Sistem mudah digunakan melalui smartphone.                                                           |
|   | P11 | Saya merekomendasikan sistem ini untuk digunakan pada kegiatan perkuliahan.                          |

## 2.5 Pemeliharaan

Tahap pemeliharaan merupakan tahap lanjutan setelah integrasi dan pengujian yang bertujuan untuk menjaga kinerja sistem tetap berjalan dengan lancar. Pada tahap ini dilakukan pemantauan kinerja sistem secara teratur, mengidentifikasi serta menangani permasalahan teknis yang muncul dan pembaruan sistem untuk optimalisasi. Dengan adanya tahap pemeliharaan ini, diharapkan sistem presensi yang dikembangkan dapat terus digunakan secara optimal dan tetap memenuhi kebutuhan *user* dalam jangka waktu yang lebih panjang.

## 3. HASIL DAN PEMBAHASAN

Pada penelitian skripsi yang telah dilakukan, dihasilkan sebuah sistem presensi berbasis *QR Code* yang akan berubah setiap interval waktu yang telah ditentukan, serta memiliki 2 sistem validasi yaitu berdasarkan lokasi dan jaringan lokal. Sistem ini dirancang untuk mencegah terjadinya kecurangan presensi. Terdapat 2 aktor yang akan menggunakan sistem ini, yaitu *admin* dan *user*. *Admin* berperan dalam membuat dan mengatur sesi presensi, sedangkan *user* hanya mengisi presensi.

### 3.1 Antarmuka User

Antarmuka *user* terdiri dari satu halaman utama, yaitu formulir presensi, yang diakses setelah *user* memindai *QR Code* yang ditampilkan admin. Formulir ini meminta tiga data wajib (Nama, NIM, Kelas) dan menampilkan notifikasi berhasil atau gagal tergantung hasil validasi token, lokasi, dan jaringan.

### 3.2 Antarmuka Admin

Antarmuka admin terdiri dari lima halaman utama. Halaman *login* dan *register* berfungsi memverifikasi identitas admin serta mendaftarkan akun baru ke dalam sistem. Halaman *dashboard* menjadi pusat kendali sistem, tempat admin mengatur lokasi, jaringan, mata kuliah, dan sesi presensi. Halaman pengaturan lokasi dan jaringan digunakan untuk mendaftarkan titik koordinat dan alamat IP yang diizinkan sebagai syarat validasi presensi. Halaman tampilan *QR Code* menampilkan *QR Code* dinamis yang diperbarui otomatis sesuai interval waktu yang ditentukan kepada *user*. Halaman data

presensi menampilkan daftar user yang telah melakukan presensi pada suatu sesi, lengkap dengan NIM, nama, kelas, alamat IP, titik lokasi, dan waktu scan, serta menyediakan fitur *export* ke format Excel untuk keperluan rekapitulasi.

### 3.3 Pengujian

Fase evaluasi atau pengujian dieksekusi guna menjamin bahwa seluruh kapabilitas fungsional sistem sudah memenuhi standar dari sistem pencatatan kehadiran dan selaras dengan rancangan cetak biru awal. Fase ini diimplementasikan melalui tiga mekanisme utama, yaitu *Black box testing*, *system usability testing*, serta evaluasi sistem secara menyeluruh.

#### 3.3.1 Black box Testing

Penelitian ini memanfaatkan metode *Black box testing* guna menguji segala aspek fungsionalitas dari sistem yang telah dibangun. Tujuan utama dari pengujian ini untuk memastikan bahwa setiap *input* yang dimasukkan menghasilkan *output* yang sesuai. Jumlah *test case* yang diuji pada sistem pencatatan kehadiran ini berjumlah 21 skenario. Skenario uji coba sistem pencatatan kehadiran ini diterapkan secara menyeluruh, mencakup dari fitur login, fitur-fitur utama, sistem validasi kehadiran, hingga berbagai potensi kecurangan. Pada pengerjaan sistem pencatatan kehadiran ini ditemukan beberapa kesalahan kecil, namun telah dilakukan perbaikan sehingga semua fitur dapat berjalan sesuai dengan yang diharapkan. Hasil pengujian *Black Box Testing* disajikan pada Tabel 4.

Tabel 4. Hasil Pengujian *Black box Testing*

| No | Fitur yang Diuji         | Skenario Pengujian                                                             | Output yang Diharapkan                                                      | Status |
|----|--------------------------|--------------------------------------------------------------------------------|-----------------------------------------------------------------------------|--------|
| 1  | Login Admin              | Admin memasukkan <i>username</i> dan <i>password</i> yang benar                | menuju halaman <i>dashboard admin</i>                                       | valid  |
| 2  | Login Admin              | Admin memasukkan <i>username</i> atau <i>password</i> yang salah               | Sistem memberikan notifikasi error                                          | valid  |
| 3  | Logout Admin             | Admin menekan tombol <i>logout</i>                                             | kembali ke halaman <i>login</i>                                             | valid  |
| 4  | Membuat Sesi Presensi    | Admin membuat sesi presensi baru dengan data lengkap                           | data tersimpan di basis data dan ditampilkan                                | valid  |
| 5  | Membuat Sesi Presensi    | Admin membuat sesi presensi dengan data tidak lengkap                          | Sistem memberikan notifikasi error                                          | valid  |
| 6  | Membuat Sesi Presensi    | Admin membuat sesi presensi dengan format yang salah                           | Sistem memberikan notifikasi error                                          | valid  |
| 7  | Pengaturan Jaringan (IP) | Admin menambahkan dan merubah alamat IP jaringan baru dengan format yang benar | data tersimpan di basis data dan ditampilkan                                | valid  |
| 8  | Pengaturan Jaringan (IP) | Admin menambahkan dan merubah alamat IP jaringan baru dengan format yang salah | Sistem memberikan notifikasi error                                          | valid  |
| 9  | Pengaturan Jaringan (IP) | Admin menghapus alamat IP dari daftar jaringan                                 | Data terhapus dari basis data                                               | valid  |
| 10 | Pengaturan Lokasi        | Admin Menambahkan dan merubah titik koordinat lokasi dengan format yang benar  | Titik koordinat lokasi berhasil ditambahkan dan tersimpan ke dalam database | valid  |

|    |                                       |                                                                                                                |                                                             |       |
|----|---------------------------------------|----------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------|-------|
| 11 | Pengaturan Lokasi                     | <i>Admin</i> menambahkan titik koordinat lokasi dengan format yang tidak valid                                 | Sistem memberikan notifikasi error                          | valid |
| 12 | Pengaturan Lokasi                     | <i>Admin</i> menghapus titik koordinat lokasi                                                                  | Titik koordinat berhasil dihapus dari database              | valid |
| 13 | Tampilan QR Code Dinamis              | <i>Admin</i> membuka halaman QR Code pada sesi yang aktif                                                      | Sistem menampilkan QR Code dinamis yang aktif               | valid |
| 14 | Scan QR Code                          | <i>User</i> memindai QR Code yang masih aktif dan valid                                                        | Sistem mengarahkan <i>user</i> ke halaman formulir presensi | valid |
| 15 | Scan QR Code                          | <i>User</i> memindai QR Code yang sudah kadaluarsa                                                             | Sistem memberikan notifikasi error                          | valid |
| 16 | Pengisian Form presensi               | <i>User</i> mengisi form presensi dengan data lengkap                                                          | data tersimpan di basis data dan ditampilkan                | valid |
| 17 | Pengisian Form presensi               | <i>User</i> mengisi form presensi dengan data yang tidak lengkap                                               | Sistem memberikan notifikasi error                          | valid |
| 18 | Validasi Geolokasi                    | <i>User</i> melakukan presensi dari luar radius lokasi yang ditentukan                                         | Sistem memberikan notifikasi error                          | valid |
| 19 | Validasi Jaringan Lokal               | <i>User</i> melakukan presensi menggunakan jaringan yang tidak terdaftar                                       | Sistem memberikan notifikasi error                          | valid |
| 20 | Validasi geolokasi dan Jaringan Lokal | <i>User</i> melakukan presensi dari dalam radius yang telah ditentukan dan menggunakan jaringan yang terdaftar | Validasi jaringan berhasil dan proses presensi dilanjutkan  | valid |
| 21 | Pencegahan Presensi Ganda             | <i>User</i> melakukan presensi kembali pada sesi yang sama                                                     | Sistem memberikan notifikasi error                          | valid |

### 3.3.2 Pengujian System Usability Scale (SUS)

Pengujian ini melibatkan 30 responden yang merupakan *user* sistem. Responden diminta untuk mencoba langsung sistem presensi. Setelah mencoba sistem presensi, responden diminta untuk mengisi kuesioner yang terdiri dari 10 pertanyaan SUS menggunakan skala Likert 1 sampai 5. Berikut adalah hasil evaluasi menggunakan SUS yang disajikan pada Tabel 5.

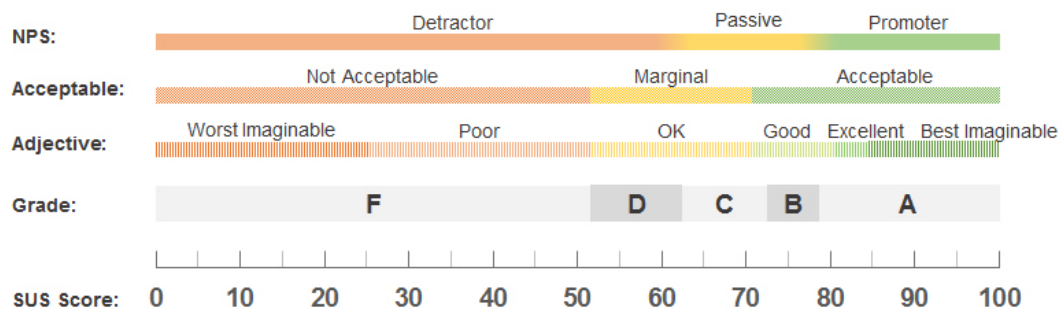
Tabel 5. Evaluasi menggunakan SUS

| Responden | Pertanyaan |    |    |    |    |    |    |    |    |     | Jumlah | Skor<br>(Jumlah x 2,5) |
|-----------|------------|----|----|----|----|----|----|----|----|-----|--------|------------------------|
|           | Q1         | Q2 | Q3 | Q4 | Q5 | Q6 | Q7 | Q8 | Q9 | Q10 |        |                        |
| R 1       | 4          | 4  | 4  | 0  | 4  | 0  | 4  | 4  | 4  | 0   | 28     | 70                     |
| R 2       | 4          | 3  | 4  | 4  | 3  | 3  | 2  | 0  | 2  | 3   | 28     | 70                     |
| R 3       | 2          | 2  | 2  | 2  | 2  | 2  | 2  | 2  | 2  | 2   | 20     | 50                     |
| R 4       | 2          | 2  | 3  | 2  | 2  | 2  | 3  | 2  | 2  | 2   | 22     | 55                     |
| R 5       | 4          | 4  | 4  | 4  | 4  | 4  | 4  | 4  | 4  | 2   | 38     | 95                     |
| R 6       | 3          | 1  | 3  | 2  | 3  | 3  | 3  | 3  | 3  | 3   | 27     | 67,5                   |
| R 7       | 4          | 4  | 4  | 2  | 4  | 2  | 4  | 4  | 4  | 4   | 36     | 90                     |
| R 8       | 3          | 3  | 3  | 3  | 3  | 3  | 3  | 3  | 3  | 2   | 29     | 72,5                   |
| R 9       | 2          | 3  | 3  | 3  | 4  | 3  | 3  | 2  | 3  | 1   | 27     | 67,5                   |



|                    |   |   |   |   |   |   |   |   |   |   |              |      |
|--------------------|---|---|---|---|---|---|---|---|---|---|--------------|------|
| R 10               | 3 | 3 | 3 | 3 | 3 | 4 | 3 | 4 | 3 | 1 | 30           | 75   |
| R 11               | 3 | 3 | 3 | 1 | 3 | 1 | 4 | 2 | 2 | 1 | 23           | 57,5 |
| R 12               | 3 | 4 | 4 | 4 | 3 | 2 | 3 | 4 | 4 | 2 | 33           | 82,5 |
| R 13               | 3 | 3 | 2 | 1 | 3 | 3 | 3 | 3 | 2 | 2 | 25           | 62,5 |
| R 14               | 3 | 2 | 3 | 2 | 4 | 2 | 4 | 3 | 3 | 1 | 27           | 67,5 |
| R 15               | 4 | 4 | 4 | 2 | 4 | 4 | 4 | 4 | 4 | 3 | 37           | 92,5 |
| R 16               | 4 | 3 | 3 | 3 | 4 | 3 | 4 | 4 | 4 | 2 | 34           | 85   |
| R 17               | 3 | 4 | 4 | 1 | 3 | 2 | 3 | 4 | 2 | 0 | 26           | 65   |
| R 18               | 4 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 4 | 4 | 39           | 97,5 |
| R 19               | 3 | 2 | 3 | 1 | 3 | 2 | 1 | 2 | 2 | 1 | 20           | 50   |
| R 20               | 4 | 3 | 4 | 2 | 4 | 3 | 3 | 3 | 3 | 1 | 30           | 75   |
| R 21               | 3 | 4 | 4 | 4 | 4 | 3 | 4 | 4 | 1 | 2 | 33           | 82,5 |
| R 22               | 3 | 4 | 4 | 3 | 4 | 4 | 4 | 4 | 0 | 3 | 33           | 82,5 |
| R 23               | 2 | 2 | 3 | 4 | 3 | 3 | 3 | 4 | 3 | 2 | 29           | 72,5 |
| R 24               | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 3 | 33           | 82,5 |
| R 25               | 3 | 2 | 3 | 2 | 3 | 3 | 3 | 2 | 3 | 1 | 25           | 62,5 |
| R 26               | 4 | 3 | 4 | 1 | 4 | 4 | 4 | 4 | 4 | 4 | 36           | 90   |
| R 27               | 4 | 3 | 4 | 3 | 3 | 3 | 3 | 3 | 2 | 3 | 31           | 77,5 |
| R 28               | 4 | 3 | 3 | 3 | 3 | 3 | 4 | 4 | 4 | 1 | 32           | 80   |
| R 29               | 4 | 0 | 4 | 0 | 4 | 0 | 4 | 0 | 4 | 0 | 20           | 50   |
| R 30               | 4 | 3 | 3 | 3 | 3 | 3 | 3 | 4 | 3 | 1 | 30           | 75   |
| <b>Rata - rata</b> |   |   |   |   |   |   |   |   |   |   | <b>73,42</b> |      |

Berdasarkan hasil perhitungan yang telah dilakukan, diperoleh nilai rata-rata SUS sebesar **73,42**. Berikut adalah skala hasil penilaian yang disajikan pada Gambar 14.



Gambar 14. Skala penilaian SUS

Hasil evaluasi menggunakan SUS menghasilkan nilai SUS sebesar 73,42. Nilai tersebut berada pada interval 70–80 yang tergolong dalam kategori *Good* (Baik). Representasi angka ini menegaskan bahwa sistem pencatatan kehadiran dengan menggunakan QR Code dinamis serta validasi ganda yang telah dikembangkan ini dapat diterima dengan baik oleh *user*, mudah dipahami, serta tidak memerlukan banyak bantuan dalam pengoperasiannya. Dengan demikian, sistem pencatatan kehadiran ini dinilai telah memenuhi standar *usability* yang memadai.

### 3.3.3 Pengujian Kuesioner Evaluasi Sistem

Pengujian evaluasi sistem dilakukan dengan menggunakan kuesioner berbasis skala Likert untuk mengukur persepsi *user* terhadap sistem yang telah dikembangkan. Responden diminta untuk mengisi kuesioner yang terdiri dari 11 pernyataan menggunakan skala Likert 1 sampai 5, dengan keterangan: 1 = Sangat Tidak Setuju, 2 = Tidak Setuju, 3 = Netral, 4 = Setuju, dan 5 = Sangat Setuju. Analisis hasil kuesioner dilakukan dengan menghitung persentase responden yang menjawab Setuju (4) atau Sangat Setuju (5) pada setiap pernyataan menggunakan rumus berikut:

$$\%S + SS = \frac{\text{Jumlah Jawaban } S(4) + SS(5)}{\text{Total Responden}} \times 100\% \quad (6)$$

$$\bar{x} = \frac{\sum_{i=1}^n (\text{Nilai Jawaban ke-}i)}{n} \quad (7)$$

Persamaan (6) digunakan untuk menghitung persentase responden yang menjawab setuju dan sangat setuju. Pada persamaan (7)  $n$  adalah total jumlah responden. Nilai rata-rata yang mendekati 5 menunjukkan kecenderungan persetujuan yang semakin tinggi dari responden. Berikut adalah hasil evaluasi kuesioner yang telah diisi oleh 30 responden yang disajikan pada Tabel 6.

Tabel 6. Hasil Kuesioner Evaluasi Sistem Presensi QR Code Dinamis

| Responden | Bagian A |    |    | Bagian B |    |    | Bagian C |    |    |     |     |
|-----------|----------|----|----|----------|----|----|----------|----|----|-----|-----|
|           | P1       | P2 | P3 | P4       | P5 | P6 | P7       | P8 | P9 | P10 | P11 |
| R 1       | 3        | 5  | 5  | 5        | 5  | 5  | 5        | 5  | 4  | 5   | 5   |
| R 2       | 5        | 5  | 5  | 2        | 5  | 5  | 5        | 5  | 5  | 5   | 5   |
| R 3       | 3        | 3  | 3  | 4        | 3  | 3  | 3        | 3  | 3  | 4   | 3   |
| R 4       | 3        | 4  | 5  | 4        | 3  | 3  | 3        | 3  | 4  | 3   | 4   |
| R 5       | 5        | 3  | 4  | 3        | 5  | 5  | 5        | 5  | 5  | 5   | 5   |
| R 6       | 4        | 4  | 4  | 4        | 4  | 4  | 5        | 4  | 4  | 5   | 4   |
| R 7       | 5        | 5  | 5  | 5        | 5  | 5  | 5        | 5  | 5  | 5   | 5   |
| R 8       | 4        | 4  | 4  | 3        | 4  | 4  | 4        | 4  | 4  | 4   | 4   |
| R 9       | 5        | 4  | 5  | 2        | 4  | 4  | 4        | 4  | 5  | 4   | 3   |
| R 10      | 4        | 4  | 4  | 3        | 4  | 4  | 4        | 3  | 4  | 4   | 4   |
| R 11      | 5        | 4  | 4  | 4        | 4  | 4  | 4        | 5  | 5  | 4   | 5   |
| R 12      | 5        | 4  | 5  | 5        | 4  | 5  | 3        | 5  | 5  | 5   | 5   |
| R 13      | 3        | 2  | 1  | 5        | 5  | 5  | 4        | 4  | 5  | 4   | 4   |
| R 14      | 4        | 5  | 5  | 4        | 5  | 4  | 5        | 4  | 5  | 5   | 5   |
| R 15      | 5        | 5  | 5  | 5        | 5  | 5  | 5        | 5  | 5  | 5   | 5   |
| R 16      | 5        | 5  | 5  | 5        | 3  | 5  | 5        | 5  | 4  | 4   | 5   |
| R 17      | 5        | 5  | 4  | 5        | 5  | 4  | 5        | 4  | 5  | 4   | 5   |
| R 18      | 5        | 5  | 4  | 5        | 5  | 5  | 5        | 5  | 5  | 5   | 5   |
| R 19      | 1        | 3  | 5  | 3        | 4  | 4  | 3        | 3  | 4  | 4   | 4   |
| R 20      | 5        | 2  | 5  | 2        | 4  | 4  | 5        | 5  | 5  | 5   | 5   |
| R 21      | 4        | 5  | 5  | 2        | 3  | 5  | 5        | 3  | 4  | 5   | 5   |

|           |           |      |           |      |           |           |     |           |           |           |           |
|-----------|-----------|------|-----------|------|-----------|-----------|-----|-----------|-----------|-----------|-----------|
| R 22      | 5         | 4    | 3         | 3    | 5         | 5         | 3   | 5         | 4         | 5         | 4         |
| R 23      | 3         | 4    | 4         | 4    | 4         | 5         | 3   | 5         | 5         | 5         | 4         |
| R 24      | 5         | 5    | 5         | 5    | 5         | 4         | 4   | 5         | 4         | 5         | 5         |
| R 25      | 4         | 4    | 3         | 4    | 3         | 4         | 4   | 4         | 5         | 5         | 4         |
| R 26      | 4         | 5    | 5         | 5    | 5         | 5         | 5   | 5         | 5         | 5         | 5         |
| R 27      | 3         | 1    | 5         | 4    | 4         | 4         | 4   | 4         | 4         | 5         | 5         |
| R 28      | 5         | 5    | 4         | 4    | 4         | 4         | 4   | 5         | 4         | 4         | 5         |
| R 29      | 5         | 5    | 5         | 5    | 5         | 5         | 5   | 5         | 5         | 5         | 5         |
| R 30      | 5         | 5    | 5         | 5    | 4         | 5         | 5   | 4         | 4         | 5         | 5         |
| total     | 127       | 124  | 131       | 119  | 128       | 133       | 129 | 131       | 135       | 138       | 137       |
| Rata-rata | 4,23      | 4,13 | 4,37      | 3,97 | 4,27      | 4,43      | 4,3 | 4,37      | 4,5       | 4,6       | 4,57      |
| % S+SS    | 76,7<br>% | 80%  | 86,7<br>% | 70%  | 83,3<br>% | 93,3<br>% | 80% | 83,3<br>% | 96,7<br>% | 96,7<br>% | 93,3<br>% |

Berdasarkan Tabel 6, pada Bagian A diketahui bahwa 76,7% responden setuju praktik titip absen masih umum terjadi, dan 80% menilai sistem presensi saat ini belum efektif mencegah manipulasi kehadiran. Hasil ini mengkonfirmasi adanya kebutuhan nyata terhadap sistem presensi yang lebih andal. Pada Bagian B, validasi jaringan lokal mendapatkan tingkat persetujuan tertinggi sebesar 93,3% dengan rata-rata 4,43, diikuti validasi lokasi sebesar 83,3%. Kedua fitur tersebut dinilai efektif dalam memastikan mahasiswa benar-benar hadir di lokasi perkuliahan. Pada Bagian C, kemudahan penggunaan melalui smartphone memperoleh persentase tertinggi sebesar 96,7% dengan rata-rata 4,6. Pernyataan bahwa sistem dapat mengurangi peluang titip absen serta kesediaan merekomendasikan sistem masing-masing memperoleh 96,7% dan 93,3%. Klaim keamanan (P7) dan efektivitas verifikasi (P8) yang dipersepsikan responden ini sejalan dengan hasil pengujian fungsional pada Tabel 4, di mana sistem berhasil menolak seluruh skenario simulasi kecurangan, meliputi penggunaan QR Code kadaluarsa, presensi di luar radius lokasi, presensi melalui jaringan tidak terdaftar, serta percobaan presensi ganda. Dengan demikian, persepsi keamanan dan efektivitas responden didukung oleh bukti pengujian teknis, bukan semata klaim subjektif. Secara keseluruhan, hasil kuesioner menunjukkan bahwa sistem mendapatkan respons positif dari sisi keamanan, efektivitas, maupun kemudahan penggunaan. Perlu digarisbawahi bahwa hasil evaluasi ini bersifat persepsional, yaitu menggambarkan penilaian dan keyakinan subjektif responden terhadap efektivitas sistem, bukan merupakan hasil pengukuran empiris langsung terhadap penurunan praktik titip absen di lapangan. Pembuktian efektivitas secara empiris, misalnya melalui perbandingan tingkat kecurangan presensi sebelum dan sesudah penerapan sistem dalam periode penggunaan nyata, belum dilakukan dalam penelitian ini dan dapat menjadi arah pengujian lanjutan.

### 3.3.4 Pengujian kinerja dan performa sistem

Pengujian performa dilakukan dua tahap. Tahap pertama mengukur waktu eksekusi fungsi inti (pembuatan token, validasi lokasi/IP, pembuatan QR) menggunakan *time.perf\_counter()* dengan

1.000 iterasi pada Tabel 7. Tahap kedua adalah uji beban menggunakan Locust pada skenario 50, 100, dan 300 pengguna simulasi (*ramp up* 3/detik, durasi 3 menit, via *tunneling* ngrok) dengan alur *GET* /absen dan *POST* /absen/submit yang disajikan pada table 8, Tabel 9, dan Tabel 10.

Tabel 7. Hasil Pengujian waktu eksekusi fungsi inti sistem

| Fungsi                 | Mean (ms) | Median (ms) | Std Dev (ms) | Min (ms) | Max (ms) | P95 (ms) |
|------------------------|-----------|-------------|--------------|----------|----------|----------|
| Generate token dinamis | 0,0012    | 0,0011      | 0,0011       | 0,0011   | 0,0353   | 0,0013   |
| Validasi lokasi        | 0,0008    | 0,0008      | 0,0004       | 0,0007   | 0,0138   | 0,0009   |
| Validasi IP            | 0,0007    | 0,0007      | 0,0001       | 0,0006   | 0,0033   | 0,0008   |
| Generate QR            | 7,3155    | 6,9744      | 4,6779       | 6,7082   | 153,4883 | 8,2533   |

Validasi lokasi, validasi jaringan, dan pembuatan token dinamis hanya membutuhkan waktu pada skala mikrodetik sehingga kontribusinya terhadap total waktu respons dapat diabaikan. Pembuatan gambar QR Code menjadi komponen paling berat dengan rata-rata 7,3155 ms.

Tabel 8. Hasil Pengujian performa sistem

| User | Total Request | RPS  | Median (ms) | P95 (ms) | P99 (ms) | Rata-rata (ms) | Failure rate |
|------|---------------|------|-------------|----------|----------|----------------|--------------|
| 50   | 10.639        | 62,4 | 58          | 88       | 140      | 61,61          | 50,0%        |
| 100  | 14.130        | 63,2 | 54          | 220      | 830      | 90,79          | 70,6%        |
| 300  | 6.897         | 21,6 | 55          | 350      | 105.000  | 3.515,43       | 69,3%        |

Tabel 9. Hasil Pengujian *GET* /absen (form)

| User | Total Request | Berhasil | Gagal (CSRF tidak ditemukan) |
|------|---------------|----------|------------------------------|
| 50   | 5.320         | 5.320    | 0                            |
| 100  | 10.081        | 4.049    | 6.032                        |
| 300  | 4.882         | 2.015    | 2.867                        |

Tabel 10. Hasil Pengujian *POST* /absen/submit

| User | Total Request | Berhasil (Presensi Tercatat) | Ditolak (Anti Duplikasi) | Gagal (Error Lain 403/500) |
|------|---------------|------------------------------|--------------------------|----------------------------|
| 50   | 5.319         | 50                           | 5.269                    | 0                          |
| 100  | 4.049         | 100                          | 3.942                    | 75                         |
| 300  | 2.015         | 99                           | 1.914                    | 2                          |

Hasil menunjukkan validasi lokasi, jaringan, dan pembuatan token hanya membutuhkan waktu skala mikrodetik, sementara pembuatan QR Code menjadi komponen terberat (rata-rata 7,3 ms). Waktu respons median tetap stabil (54-58 ms) di seluruh skenario, namun P95/P99 meningkat tajam pada beban tinggi (hingga 105.000 ms pada 300 pengguna), mengindikasikan titik jenuh infrastruktur pengujian. Kegagalan yang tercatat berasal dari dua sumber, yaitu mekanisme anti duplikasi presensi yang justru bekerja konsisten (dibuktikan jumlah presensi berhasil yang mendekati jumlah pengguna simulasi), dan keterbatasan server pengembangan Django serta tunneling gratis dalam menangani

volume koneksi tinggi. Secara keseluruhan, sistem menunjukkan performa baik pada beban yang merepresentasikan kondisi nyata. Penurunan performa pada beban ekstrem lebih mencerminkan keterbatasan infrastruktur pengujian, bukan kapasitas aplikasi.

### **3.3.5 Keterbatasan penelitian**

Sistem yang dikembangkan memiliki beberapa keterbatasan, yaitu akurasi validasi geolokasi bergantung pada kualitas sinyal GPS, proses presensi memerlukan koneksi internet dan jaringan WiFi lokal yang stabil, serta pengujian performa menggunakan server pengembangan Django dan *tunneling* ngrok gratis turut membatasi hasil pada beban tinggi. Meski demikian, pengujian pada level fungsi menunjukkan logika inti sistem tetap berjalan pada skala mikrodetik, sehingga penurunan performa tersebut lebih mencerminkan keterbatasan infrastruktur pengujian daripada kompleksitas komputasi aplikasi.

## **4. PENUTUP**

Merujuk pada hasil perancangan, implementasi, dan pengujian, sistem pencatatan kehadiran berbasis *QR Code* dinamis dengan validasi lokasi dan jaringan lokal berhasil dikembangkan sesuai rancangan kebutuhan fungsional awal. Hal ini divalidasi melalui *Black Box Testing*, di mana seluruh 21 skenario dinyatakan valid. Konsistensi mekanisme anti duplikasi ini turut teramati pada pengujian performa, di mana percobaan presensi berulang tetap ditolak meski dalam kondisi beban tinggi dan bersamaan. Dari sisi performa, sistem menunjukkan waktu respons yang baik pada beban yang merepresentasikan skenario nyata. Penurunan performa pada beban sangat tinggi lebih disebabkan oleh keterbatasan infrastruktur pengujian daripada kapasitas aplikasi maupun basis data.

Berdasarkan kuesioner evaluasi 30 responden, 96,7% setuju sistem berpotensi mengurangi peluang titip absen, 80% menilai sistem lebih aman dibanding presensi manual, dan 93,3% bersedia merekomendasikan sistem ini. Skor SUS sebesar 73,42 (kategori *Good*) turut mengindikasikan bahwa sistem mudah digunakan. Perlu digarisbawahi bahwa temuan ini bersifat persepsional, bukan pengukuran empiris langsung terhadap penurunan praktik kecurangan presensi di lapangan.

Dengan demikian, sistem presensi berbasis *QR Code* dinamis yang dikembangkan menunjukkan indikasi positif sebagai solusi pencegahan kecurangan presensi. Pembuktian efektivitas secara empiris dalam jangka panjang, misalnya melalui implementasi dan perbandingan langsung dengan metode presensi konvensional, masih diperlukan sebagai penelitian lanjutan.

## **DAFTAR PUSTAKA**

Alda, M., Juarsyah, M., Nugraha, A., & Alfachry, L. R. (2024). Aplikasi Absensi Mahasiswa Kerja Praktik Menggunakan QR Code Berbasis Android. *Jurnal Manajemen Informatika (JAMIKA)*,



- 14(1), 27–41. <https://doi.org/10.34010/jamika.v14i1.11775>
- Al-Tahat, K. S., Magableh, A. A., Albahri, A. S., Garfan, S., Deveci, M., Albahri, O. S., Alamoodi, A. H., & Sharaf, I. M. (2026). Evaluation of software development life cycle using multi-criteria decision-making approach. *Kuwait Journal of Science*, 53(3), 100595. <https://doi.org/10.1016/j.kjs.2026.100595>
- Aryadi, T., Ridho, M., Hafizh, A., & Dani, I. (2026). Pengujian Keamanan Website Menggunakan Metode Black Box dengan OWASP ZAP pada Kali Linux. *Jurnal Penelitian Multidisiplin Bangsa*, 2(8), 1482–1491. <https://doi.org/10.59837/jpnmb.v2i8.736>
- Chethan, M., & Bharath, C. (2025). Issue 4 | [www.ijedr.org](http://www.ijedr.org) IJEDR2504598. *International Journal of Engineering Development and Research*, 13, 753. [www.ijedr.org](http://www.ijedr.org)
- Emmanuel Tamuno, O. (2025). DESIGN AND IMPLEMENTATION OF A SMART STAFF ATTENDANCE SYSTEM USING QR CODE TECHNOLOGY. *Ayden Journal of Intelligent System and Computing*, 13(2), 2997–187. <https://doi.org/10.5281/zenodo.15431630>
- Firdaus, F. M., & Hidayat, H. (2025). Perancangan dan Implementasi Sistem Absensi Siswa Berbasis Web Menggunakan Face Recognition dan SMS Gateway. *Jurnal Manajemen Informatika (JAMIKA)*, 15(1), 32–46. <https://doi.org/10.34010/jamika.v15i1.13601>
- Husni, I., Amin, A., Lusiana, V., Hartono, B., Wahyu, D., Studi, P., Informatika, T., Jl, S., & Lomba Juang, T. (2024). QR Code-Based Attendance System for Contact Tracking Post-Pandemic. *COGITO Smart Journal*, 10(1).
- Imaduddin, H., & Hayati, L. N. (2023). Perancangan Sistem Informasi Buku Tamu Di Pengadilan Agama Sragen Kelas 1A. In *Jurnal Abdimas PHB* (Vol. 6).
- Imanullah, M., & Reswan, Y. (2022). Randomized QR-code scanning for a low-cost secured attendance system. *International Journal of Electrical and Computer Engineering*, 12(4), 3762–3769. <https://doi.org/10.11591/ijece.v12i4.pp3762-3769>
- Khatib Sulaiman, J., Reiswa Aditama Farhan Mudhoffar, R., Widayat, W., & Muhammadiyah Surakarta, U. (2024). Perancangan Sistem Informasi Rekam Medis Berbasis Website. *Indonesian Journal of Computer Science*.
- Mishra, U., & Ganesh, N. (2026). A Data-Driven Comparative Analysis of Agile and Waterfall Methodologies: Predicting Cost and schedule Variances Using Statistical and Machine Learning Approaches. *Array*, 100665. <https://doi.org/10.1016/j.array.2025.100665>
- Mohammed, M. S., & Zidan, K. A. (2023). Enhancing attendance tracking using animated QR codes: a case study. *Indonesian Journal of Electrical Engineering and Computer Science*, 31(3), 1716–1723. <https://doi.org/10.11591/ijeecs.v31.i3.pp1716-1723>
- Mornie, M. N., Jali, N., Cheah, W. S., Mit, E., Saeed, S., Jali, S. K., & Greer, D. (2025). Visualisation

- of User Stories to UML use Case Diagram. *Journal of Advanced Research in Applied Sciences and Engineering Technology*, 63(3), 68–80. <https://doi.org/10.37934/araset.63.3.6880>
- Mulya, N., & Wulan Sari, I. (2024). Web-Based Library Information System Application with QR Code Support (Case Study at SMK Negeri 1 Tambusai). In *Jurnal Komputer dan Teknologi (JUKOTI)* (Vol. 1, Number 1).
- Napitupulu, R. J., Silvia, H., Rhamadan, A. I., & Niqotaini, Z. (2025). ANALISIS SISTEM PERPUSTAKAAN UPN VETERAN JAKARTA : STUDI KASUS PEMINJAMAN RUANG DISKUSI MENGGUNAKAN UML. *Jurnal Sistem Informasi (TEKNOFILE)*, 3, 379–396.
- Nguyen-Tat, B. T., Bui, M. Q., & Ngo, V. M. (2024). Automating attendance management in human resources: A design science approach using computer vision and facial recognition. *International Journal of Information Management Data Insights*, 4(2). <https://doi.org/10.1016/j.jjime.2024.100253>
- Pauzan, M., & Yanti, I. (2022). Sistem Absensi Fingerprint Berbasis Arduino dengan Data Penyimpanan di Micro SD. In *Jawa Barat Gema Wiralodra* (Vol. 12, Number 2).
- Perin, M. A. D. (2025). Technology-Assisted Attendance Monitoring: A Case Study on QR Code System Usability and Performance. *Journal of Technology-Assisted Learning*, 1(2), 115–123. <https://doi.org/10.70232/jtal.v1i2.6>
- Prasetyo, A. (2025). 94824+Completed+72-81. *Information System and Emerging Technology Journal*, 6, 72–82.
- Rakhmadi, A., Firdaus, M. I., & Nugroho, S. (2025). *COmputing and INformation Systems Journal A Systematic Implementation of the Waterfall Model in E-Commerce System Development for Small Businesses* (Vol. 1, Number 1). [www.coins-indocompt.org](http://www.coins-indocompt.org)
- Saravanos, A., & Curinga, M. X. (2023). Simulating the Software Development Lifecycle: The Waterfall Model. *Applied System Innovation*, 6(6). <https://doi.org/10.3390/asi6060108>
- SNEINEH Anees, A., & Wael, S. A. (2025). *Smart Attendance System for University Students Based on QR Code*.