

BAB II

KAJIAN PUSTAKA

A. Kajian Teoritis

1. Sistem Presensi

Sistem Presensi merupakan mekanisme pencatatan kehadiran yang krusial bagi suatu instansi atau perusahaan, di mana data yang terkumpul dikelola agar mudah diakses kembali untuk kebutuhan pelaporan aktivitas (Wijaya et al, 2023). Selain berfungsi sebagai alat penghitung kehadiran fisik karyawan. (Gunawan et al., 2023).

Disisilain (Sunarya & Hardyanto, 2021) presensi juga berfungsi untuk memantau angka ketidakhadiran pegawai, baik karena izin, sakit, maupun alpa. (Dewi et al,2025) Sistem presensi merupakan komponen penting dalam institusi pendidikan untuk memastikan kehadiran secara efisien dan akurat.

Dengan demikian, dapat disimpulkan bahwa presensi bukan sekadar alat pencatat administratif, melainkan instrumen strategis yang digunakan manajemen untuk mengukur, mengevaluasi, dan menjaga tingkat kedisiplinan serta produktivitas pegawai dalam suatu organisasi

2. *Face Recognition* (Pengenalan wajah)

(Geralda et al., 2026) *Face Recognition* menjelaskan bahwa *face recognition* merupakan metode biometrik yang bekerja dengan cara mengenali identitas seseorang melalui karakteristik unik pada wajahnya,

yakni dengan membandingkan gambar wajah yang ditangkap dengan data yang sudah tersimpan di basis data.

Hal ini sejalan dengan (Achmad & Supatman, 2024) yang menyebutkan bahwa pengenalan wajah termasuk ke dalam salah satu jenis biometrik yang digunakan sebagai proses validasi untuk mengakses suatu sistem.

Dari beberapa definisi di atas, bahwa *face recognition* adalah teknologi biometrik berbasis wajah yang bekerja dengan mencocokkan citra wajah pengguna dengan data yang ada di dalam sistem, tujuannya untuk memverifikasi atau mengidentifikasi identitas seseorang secara otomatis, cepat.

3. *Liveness detection*

Liveness detection adalah sebuah teknik yang dirancang untuk memastikan bahwa wajah yang diterima oleh sistem benar-benar berasal dari wajah asli, bukan hasil manipulasi atau tiruan dari pihak lain yang mencoba membobol sistem (Achmad & Supatman, 2024).

(Santoso et al., 2024) juga berpendapat senada, bahwa *liveness detection* merupakan komponen penting dalam sistem keamanan biometrik, terutama pada aplikasi pengenalan wajah, yang tugasnya membedakan antara wajah manusia secara langsung (*real-time*) dengan foto atau gambar palsu yang sengaja digunakan untuk menipu sistem.

Berdasarkan beberapa definisi di atas, dapat disimpulkan bahwa *liveness detection* adalah teknik keamanan dalam sistem biometrik yang

berfungsi untuk membedakan wajah asli secara langsung dengan gambar tiruan statis. Pada penelitian ini, fungsi tersebut diimplementasikan melalui pendekatan *Active Liveness Detection* dengan skema *Random Challenge*. Metode ini bekerja dengan cara memverifikasi respon dinamis pengguna seperti gerakan berkedip atau menengok terhadap instruksi acak yang diberikan oleh sistem secara *real-time*, sehingga sistem dapat memastikan kehadiran fisik pengguna yang sah tanpa melalui proses *training* data *spoofing*.

a. *Eyeblink* (Kedipan Mata)

(Syahputra et al., 2022) menjelaskan bahwa kedipan mata bisa dideteksi menggunakan metode *Eye Aspect Ratio* (EAR), yaitu sebuah perhitungan rasio antara jarak vertikal dan horizontal pada mata. Ketika mata dalam kondisi tertutup, nilai EAR akan mengalami penurunan yang cukup signifikan.

Hal ini diperkuat oleh (Putra et al., 2024) yang menyebutkan bahwa jika nilai EAR mengalami penurunan secara konsisten dalam beberapa frame berturut-turut, hal itu bisa dijadikan indikator yang valid bahwa objek yang terdeteksi merupakan manusia hidup, bukan gambar atau foto tiruan.

Dari pendapat diatas, dapat disimpulkan bahwa *Eye Aspect Ratio* (EAR) merupakan metode yang efektif untuk mendeteksi kedipan mata sebagai bukti kehadiran manusia secara nyata, di mana penurunan

nilai EAR yang terjadi secara berulang dalam beberapa *frame* menjadi tolak ukur utama dalam membedakan objek hidup dengan objek tiruan.

b. *Head Movement* (Gerakan Kepala)

Deteksi gerakan kepala merupakan salah satu pendekatan dalam *liveness detection* yang dipakai untuk memverifikasi bahwa objek yang tertangkap kamera memang manusia hidup. (Ooi et al. 2023) menjelaskan bahwa metode ini bekerja dengan memanfaatkan *facial landmark* dan metode *Perspective-n-Point* (PnP), yang berfungsi untuk menghubungkan koordinat wajah dalam ruang tiga dimensi (3D) dengan proyeksinya pada citra dua dimensi (2D).

Sejalan dengan hal tersebut (Yahya et al., 2025) menambahkan bahwa orientasi kepala bisa direpresentasikan melalui *Euler Angle* yang terdiri dari tiga komponen, yaitu *pitch*, *yaw*, dan *roll*, di mana masing-masing komponen menggambarkan arah rotasi kepala pada sumbu yang berbeda. Dalam konteks penelitian ini, parameter yang digunakan secara spesifik adalah *yaw*, yakni untuk menangkap gerakan kepala ke arah kiri maupun kanan.

Dari uraian diatas, dapat disimpulkan bahwa deteksi gerakan kepala menjadi salah satu cara yang cukup kuat dalam *liveness detection*, di mana perhitungan *Euler Angle*, khususnya nilai *yaw*, digunakan sebagai acuan untuk memastikan bahwa objek yang terdeteksi benar-benar melakukan gerakan alami layaknya manusia hidup.

4. Computer Vision

(Adim & Nurhidayat, n.d.; Ashari et al., 2025) menyatakan *Computer Vision* adalah cabang ilmu komputer yang dirancang untuk memberi kemampuan kepada komputer dalam memahami dan memproses informasi visual seperti gambar maupun video.

Sehingga dapat disimpulkan, *computer vision* adalah cabang ilmu komputer yang memungkinkan komputer untuk memahami dan menginterpretasikan informasi visual secara mandiri, baik berupa gambar maupun video, dengan tujuan agar komputer dapat mengenali objek dan mengonversi informasi tersebut menjadi perintah yang dapat diproses lebih lanjut

a. Yolo

(Sugandi & Hartono, 2022; Sugianto et al., 2022) yang sama-sama menegaskan bahwa YOLO mampu mendeteksi dan memproses objek secara langsung tanpa jeda waktu yang berarti.

Maka dapat disimpulkan bahwa YOLO adalah algoritma deteksi objek yang dirancang untuk bekerja secara *real-time*, dengan pendekatan unik membagi gambar ke dalam *grid* dan melakukan prediksi *bounding box* serta klasifikasi objek hanya dalam sekali proses pengolahan gambar.

b. ArcFace

ArcFace adalah algoritma pengenalan wajah yang menggunakan *Additive Angular Margin Loss* untuk menghasilkan *embedding* wajah yang lebih terpisah dan andal (Nurlita et al., 2024).

(Husni & Munir, 2025) menjelaskan bahwa *ArcFace* memperkenalkan konsep *additive margin loss* berbasis sudut antar vektor, yang bertujuan menghasilkan representasi wajah yang lebih diskriminatif dalam ruang vektor.

Berdasarkan beberapa definisi tersebut, dapat disimpulkan bahwa *ArcFace* adalah algoritma yang bekerja dengan menambahkan margin pada ruang sudut antar vektor *embedding*, sehingga fitur wajah dari orang yang sama semakin rapat dan fitur wajah dari orang yang berbeda semakin jauh jaraknya, menghasilkan representasi wajah yang lebih akurat dan diskriminatif.

c. Cosine Similarity

Cosine Similarity adalah matriks yang pakai untuk mengukur tingkat kemiripan antara dua vektor atau objek, dengan membandingkan sudut antara keduanya dalam ruang vektor. Nilai *cosine similarity* berada pada rentang 0 sampai 1, dimana nilai semakin mendekati 1 menunjukkan kemiripan yang tinggi, sedangkan nilai yang mendekati 0 menunjukkan rendahnya kemiripan (Geralda et al., 2026) .

Sejalan dengan (Mujianto et al., 2024) Menambahkan bahwa *Cosine Similarity* adalah algoritma yang digunakan untuk mengukur

tingkat kesamaan antara dua vektor yang berbeda dengan cara mengukur nilai kosinus sudut di antara kedua vektor tersebut.

$$similarity = \cos(\theta) = \frac{(A \cdot B)}{(|A| \times |B|)} \quad \dots(2.1)$$

$$keputusan = \begin{cases} match(accept) & \text{jika } sim(A, B) \geq \tau \\ nonmatch(reject/UNKNOWN) & \text{jika } sim(A, B) < \tau \end{cases} \quad \dots(2.2)$$

Dari definisi tersebut, dapat disimpulkan bahwa *Cosine Similarity* adalah matriks pembandingan dua vektor *embedding* wajah berdasarkan arah, bukan jarak, sehingga lebih stabil terhadap perbedaan skala data dan cocok digunakan pada data berdimensi tinggi seperti *embedding* wajah hasil ekstraksi *ArcFace*.

d. Matrix evaluasi pengenalan biometrik

Evaluasi performa pengenalan biometrik wajah, secara umum menggunakan *confusion matrix* yang mengklasifikasi hasil kedalam empat yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN) (Anisa et al., 2026; Hidayat et al., 2025).

$$Akurasi = \frac{Jumlah\ Prediksi\ Benar}{Total\ Data} \quad \dots(2.3)$$

$$Precision = \frac{TP}{TP + FN} \quad \dots(2.4)$$

$$Recal = \frac{TP}{TP + FN} \quad \dots(2.5)$$

$$F1 - score = 2x \frac{Precision \cdot Recall}{Precision + Recall} \quad \dots(2.6)$$

Dapat disimpulkan bahwa penilaian keberhasilan sistem pengenalan wajah dilakukan dengan memetakan hasil deteksi ke dalam *confusion matrix*. Melalui kombinasi nilai TP, TN, FP, dan FN, sistem

tidak hanya diuji dari aspek ketepatan pengenalan, tetapi juga dari kemampuannya dalam menangkal celah keamanan.

1) False Accept Rate

FAR (*False Acceptance Rate*) adalah matrik yang digunakan untuk mengukur berapa sering sistem salah menerima individu yang tidak terdaftar atau tidak terdaftar sebagai individu yang sah, sehingga sistem memberikan akses kepada pihak yang seharusnya ditolak (Anisa et al., 2026 ; Tanujaya et al., 2026)

FAR dihitung menggunakan rumus sebagai berikut :

$$FAR = + \frac{FP}{FP + TN} \times 100\% \quad \dots(2.7)$$

Dimana :

FP (*False Positive*) : Seharusnya ditolak, namun diterima sistem

TN (*True Negatif*) : Ditolak sistem sesuai kondisi nyata

Dapat disimpulkan FAR merupakan indikator kunci untuk melihat tingkat kerentanan pada sistem keamanan biometrik. Lewat perbandingan antara nilai *False Positive* (FP) dan total kondisi negatif (FP + TN), metrik ini bisa nunjukkin data konkret seberapa sering sistem kecolongan dalam memberikan hak akses kepada pihak yang seharusnya ditolak.

2) False Reject Rate

False Reject Rate (FRR) adalah matriks yang digunakan untuk menghitung berapa sering sistem salah menolak identitas yang

sebenarnya terdaftar, sehingga identitas pengguna yang sah tidak dikenali meskipun valid (Anisa et al., 2026; Tanujaya et al., 2026)

FRR dihitung menggunakan rumus berikut :

$$FRR = + \frac{FN}{FN + TP} \quad \dots(2.8)$$

Dimana :

FN (*False Negative*) : jumlah yang seharusnya diterima, tapi ditolak

TP (*True Positive*) : jumlah wajah yang seharusnya diterima dan benar

Dapat disimpulkan FRR adalah metrik kunci untuk melihat tingkat kegagalan sistem keamanan biometrik dalam mengenali identitas yang valid. Parameter ini berfungsi mencatat frekuensi kesalahan sistem ketika keliru menolak pengguna sah yang seharusnya diberikan hak akses.

e. MediaPipe

MediaPipe menyediakan model *pre-trained* yang digunakan sebagai implementasi cepat untuk face landmarking, serta memiliki fungsi landmarking yang memungkinkan penangkapan wajah dan titik-titik landmark wajah (Zhi et al., 2023).

MediaPipe merupakan salah satu *framework* pengembangan *machine learning* yang mampu berjalan di berbagai platform dan mendukung berbagai bahasa pemrograman (Sugeng & Nizar, 2023).

Sehingga disimpulkan bahwa *MediaPipe* adalah *framework open source* buatan Google yang fleksibel dan mendukung berbagai

platform serta bahasa pemrograman, dengan kemampuan memproses berbagai jenis data mulai dari gambar, video, hingga audio.

f. OpenCV

OpenCV adalah *library open source* yang tersedia secara gratis untuk keperluan *computer vision* dan *machine learning*, dengan dukungan berbagai bahasa pemrograman seperti C/C++, Python, Java, dan MATLAB (Yahya et al., 2025).

Menurut (Irfan et al., 2023) turut menegaskan bahwa *OpenCV* merupakan *library open source* yang secara khusus digunakan untuk menampilkan dan memproses gambar maupun video.

Dari beberapa definisi tersebut, dapat disimpulkan bahwa *OpenCV* adalah *library open source* yang dikembangkan untuk mempermudah pemrosesan gambar dan video, dilengkapi berbagai fitur *computer vision* dan *machine learning*, serta mendukung berbagai bahasa pemrograman sehingga fleksibel digunakan di berbagai kebutuhan pengembangan aplikasi.

5. WhatsApp Gateway

WhatsApp Gateway adalah aplikasi yang berfungsi untuk mengirim dan menerima pesan melalui koneksi *Application Programming Interface* (API), yang berperan sebagai perantara dalam menerima permintaan pengguna, memberi tahu sistem terkait tindakan yang perlu dilakukan, serta menanggapi permintaan tersebut (Ismawati et al., 2023).

(Uddin et al., 2025) menjelaskan bahwa *WhatsApp Gateway* merupakan layanan perantara yang memungkinkan sistem mengirimkan notifikasi pesan, baik secara personal maupun *broadcast*, melalui antarmuka yang terprogram. Hal ini sejalan dengan (Utama et al., 2024) yang menyebutkan bahwa *WhatsApp Gateway* adalah teknologi yang mengintegrasikan aplikasi atau sistem informasi dengan platform WhatsApp guna mengirim dan menerima pesan secara otomatis melalui API.

Berdasarkan beberapa definisi tersebut, dapat disimpulkan bahwa *WhatsApp Gateway* adalah sebuah layanan berbasis API yang berperan sebagai jembatan antara sistem informasi dengan platform *WhatsApp*, sehingga memungkinkan pengiriman dan penerimaan pesan secara otomatis, baik personal maupun *broadcast*.

6. Geolokasi

Geolokasi atau *Global Positioning System* (GPS) adalah metode yang digunakan untuk menentukan posisi suatu objek di permukaan bumi dalam berbagai kondisi cuaca (Muthmainnah et al., 2024). *Geolocation* merupakan sebuah fitur yang memiliki kemampuan untuk melacak lokasi suatu perangkat seluler berdasarkan GPS, *IP address*, RFID, data seluler, maupun koneksi Wi-Fi. Fitur *geolocation* ini dapat diakses secara fleksibel melalui berbagai perangkat pengguna seperti *smartphone*, tablet, laptop, hingga PC (Fadila & Amalia, 2025).

Dari pendapat diatas maka dapat disimpulkan bahwa GPS atau geolokasi adalah sistem navigasi berbasis satelit yang digunakan untuk menentukan posisi geografis suatu objek secara akurat di permukaan bumi dalam berbagai kondisi.

7. Website

Menurut (Yuniarti et al., 2022) dan (Ahmad et al., 2024) *Website* adalah kumpulan halaman yang memuat berbagai informasi digital seperti teks, gambar, audio, video, maupun animasi, yang dapat diakses melalui jaringan internet menggunakan aplikasi peramban (*browser*) seperti *Google Chrome*, *Mozilla Firefox*, atau *Microsoft Edge*.

Berdasarkan beberapa definisi tersebut, dapat disimpulkan bahwa website adalah media informasi berbasis internet yang menampilkan konten digital secara interaktif dan multimedia, dapat diakses dari berbagai lokasi melalui peramban, serta dirancang untuk memenuhi berbagai kebutuhan informasi pengguna secara efisien.

8. Teknologi Pengembangan Sistem

a. PHP

Menurut (Kansha et al., 2023). PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman *open source* yang dapat disisipkan langsung ke dalam kode HTML dan berjalan di sisi server (*server-side scripting*), artinya semua proses pengolahan data dilakukan di server sebelum hasilnya dikirim ke browser pengguna.

PHP merupakan sebuah bahasa skrip yang digunakan untuk pengembangan web, sehingga memungkinkan pembuatan konten yang dinamis pada sebuah halaman situs web (Junaedi 2024:9)

Dari pendapat diatas maka dapat disimpulkan bahwa PHP adalah bahasa pemrograman *server-side* berbasis *open source* yang disisipkan ke dalam HTML, digunakan untuk membangun website dinamis dan interaktif dengan cara memproses data di sisi server sebelum ditampilkan ke pengguna

b. TypeScript

TypeScript adalah bahasa pemrograman turunan dari JavaScript yang dilengkapi fitur pengetikan statis, pemrograman berorientasi objek, dan modularisasi kode, dengan tujuan meningkatkan keamanan, keterbacaan, serta efisiensi dalam pengembangan aplikasi web (Prayetno, 2023).

(Fatah et al., 2023) menambahkan bahwa *TypeScript* merupakan *superset* dari *JavaScript*, artinya sintaks yang digunakan pada dasarnya sama dengan *JavaScript* namun dengan kemampuan yang lebih luas.

Berdasarkan beberapa definisi tersebut, dapat disimpulkan bahwa *TypeScript* adalah bahasa pemrograman berbasis *JavaScript* yang dikembangkan dengan menambahkan fitur pengetikan statis dan pemrograman berorientasi objek, sehingga menghasilkan kode yang lebih terstruktur, aman, dan mudah dipelihara.

c. Python

Python dikenal sebagai bahasa pemrograman yang menggunakan interpreter untuk mengeksekusi kode program secara langsung. Fitur ini memberikan fleksibilitas dan memungkinkannya berjalan di berbagai sistem operasi, termasuk Windows dan Linux (Rahman et al., 2023:1).

(Mahawardana et al., 2022) menambahkan bahwa *Python* termasuk bahasa pemrograman tingkat tinggi yang cukup populer di kalangan *programmer*, dengan sintaks yang sederhana dan tidak terlalu rumit sehingga mudah dipelajari maupun digunakan dalam pengembangan berbagai jenis aplikasi.

Maka dapat disimpulkan bahwa *Python* adalah bahasa pemrograman tingkat tinggi yang bersifat *dynamically-typed*, memiliki sintaks yang sederhana, dan mudah dipelajari, menjadikannya salah satu bahasa yang banyak digunakan dalam pengembangan berbagai aplikasi.

d. FastAPI

Menurut (Hanifah et al., 2025) FastAPI merupakan *framework* untuk pengembang *website* modern yang memiliki kecepatan tinggi untuk membangun API (*Application Programming Interface*) dengan bahasa pemrograman *Python*. Sejalan dengan (Zhi et al., 2023) FastAPI adalah *framework* web berbasis *Python* yang ringan, dirancang untuk membuat API berkinerja tinggi menggunakan *type hint* standar *Python*.

Dapat disimpulkan bahwa Dalam pengembangan *backend*, FastAPI hadir sebagai *framework Python* yang menawarkan performa tinggi dalam menyusun *RESTful API* dengan kode yang lebih ringkas. Efektivitas *framework* ini didukung oleh kemampuan eksekusi asinkron, fitur validasi mandiri, dan pembuatan dokumentasi API secara langsung tanpa memerlukan modul tambahan.

e. Laravel

Menurut (Fauzi & Darmawan, 2023) *Laravel* merupakan *framework PHP* yang mengimplementasikan konsep arsitektur MVC (*Model, View, Controller*) untuk memisahkan komponen tampilan UI secara terstruktur dari logika pemrograman dan pengelolaan data.

Laravel dirancang sebagai *framework PHP* untuk mempermudah pengembangan aplikasi web modern melalui struktur yang kuat dan sintaks yang bersih. Dikembangkan pertama kali oleh Taylor Otwell pada tahun 2011 untuk mengatasi keterbatasan *framework* tradisional seperti CodeIgniter, *Laravel* kini telah tumbuh menjadi salah satu *framework PHP* paling populer secara global (Devega et al., 2026:1).

Berdasarkan beberapa definisi tersebut, dapat disimpulkan bahwa *Laravel* adalah *framework PHP* berbasis arsitektur MVC yang mempermudah dan mempercepat proses pengembangan aplikasi web, dengan berbagai fitur bawaan seperti *artisan* yang membuat pengelolaan aplikasi menjadi lebih terstruktur dan efisien.

f. React JS

Menurut (Sulistyorini et al., 2022) React JS adalah *library* JavaScript yang bersifat *open source* dan banyak digunakan untuk membangun *User Interface* (UI), khususnya pada aplikasi berbasis satu halaman (*single page application*). Sependat (Fatah et al., 2023) menambahkan bahwa *React JS* merupakan pustaka *frontend* yang dikembangkan oleh *Facebook*, yang difungsikan untuk menangani lapisan tampilan pada aplikasi web maupun aplikasi seluler

Maka dapat disimpulkan bahwa *React JS* adalah *library* JavaScript *open source* buatan *Facebook* yang digunakan untuk membangun tampilan antarmuka pengguna secara efisien, baik pada aplikasi web maupun seluler.

g. InsightFace

InsightFace adalah pustaka *open source* berbasis *deep learning* yang mendukung berbagai tugas analisis wajah seperti deteksi wajah, penyelarasan wajah (*face alignment*), dan ekstraksi fitur wajah (*face embedding*), baik dalam format 2D maupun 3D (Setiawan et al., 2025).

(Kamil et al., 2025) menjelaskan bahwa *InsightFace* dirancang untuk mempelajari fitur diskriminatif yang mampu membedakan wajah secara akurat menggunakan dataset wajah berskala besar.

(Rasyid et al., 2025) menambahkan bahwa *InsightFace* menyediakan berbagai model *state-of-the-art* untuk tugas analisis wajah, termasuk deteksi, pengenalan, dan segmentasi wajah, di mana

dalam penerapannya model ini juga mampu melakukan *cropping*, *resize*, dan penyimpanan hasil deteksi ke dalam folder tertentu.

Dari beberapa pendapat diatas maka dapat disimpulkan bahwa *InsightFace* adalah *framework deep learning open source* yang menyediakan berbagai fitur analisis wajah secara lengkap, mulai dari deteksi, penyelarasan, hingga ekstraksi fitur wajah, dengan kemampuan membedakan identitas wajah secara akurat menggunakan model *state-of-the-art*.

h. Basisdata

(Mutoffar et al.,2024:1) Basis data merupakan sekumpulan data yang diorganisasikan dan disimpan secara sistematis demi kemudahan pencarian, pengambilan, serta pengelolaan data yang efisien. Dalam penerapannya, terdapat beberapa jenis basis data yang sering digunakan, seperti basis data relasional *SQL* dan *NoSQL*. (Siregar et al., 2024) menambahkan bahwa basis data disusun secara sistematis agar data yang tersimpan dapat diakses, dimanipulasi, dan dikelola dengan mudah oleh sistem maupun aplikasi perangkat lunak.

Hal ini sejalan dengan (Pane et al., 2022) yang menyebutkan bahwa basis data merupakan kumpulan informasi yang disimpan di komputer secara sistematis sehingga dapat diperiksa dan diambil kembali menggunakan program komputer.

Dari beberapa pendapat diatas dapat disimpulkan bahwa basis data adalah kumpulan data yang disusun secara terstruktur dan

sistematis di dalam komputer, sehingga memudahkan proses penyimpanan, pengelolaan, dan pengaksesan informasi oleh sistem atau aplikasi yang membutuhkannya

1) Laragon

Laragon adalah perangkat lunak *open source* yang berfungsi sebagai server virtual *localhost* dan dapat digunakan di berbagai sistem operasi, memungkinkan pengembang menjalankan lingkungan pengembangan web secara lokal dengan mudah dan efisien (Andarsyah et al., 2022).

(Raharjo et al., 2026) menambahkan bahwa Laragon merupakan aplikasi server lokal yang ringan dan gratis, dengan kompatibilitas yang luas mencakup *Windows*, *MacOS*, dan *Linux*.

Berdasarkan beberapa definisi tersebut, dapat disimpulkan bahwa Laragon adalah aplikasi *open source* yang menyediakan lingkungan pengembangan web secara lokal, ringan, gratis, dan kompatibel dengan berbagai sistem operasi, sehingga mempermudah proses pengembangan aplikasi berbasis *website*.

2) MySQL

Menurut (Setiawan et al., 2022) menjelaskan bahwa *MySQL* merupakan DBMS (*Database Management System*) yang sudah sangat populer digunakan, dengan *SQL* sebagai bahasa perintah utamanya. *MySQL* tersedia dalam dua jenis lisensi, yakni *Free Software* dan *Shareware*, sehingga penggunaannya cukup

fleksibel tergantung kebutuhan pengembangan aplikasi. Hal ini sejalan dengan apa yang disampaikan (Pane et al., 2022) bahwa *MySQL* sebagai DBMS *open source* memang hadir dalam dua bentuk lisensi *Free Software* yang bebas digunakan, dan *Shareware* yang penggunaannya lebih terbatas karena masih ada hak kepemilikannya.

Dari beberapa pendapat di atas, dapat disimpulkan bahwa *MySQL* adalah sistem manajemen basis data (DBMS) berbasis *open source* yang menggunakan SQL sebagai bahasa utamanya.

9. Metode Pengembangan Sistem

a. Extreme programming (XP)

Extreme Programming (XP) adalah salah satu metodologi pengembangan perangkat lunak yang memfokuskan pada proses pengembangan yang bersifat adaptif, kolaboratif (Sari & Cahyani, 2022).

Sepakat dengan (Rudianto, 2023) metodologi XP (*Extreme Programming*) merupakan pengembangan perangkat lunak yang berfokus pada fleksibilitas yang adaptif dan fleksibel.

Dari beberapa pendapat di atas, dapat disimpulkan bahwa XP merupakan metodologi *agile* yang cocok digunakan ketika kebutuhan proyek sering berubah.

10. Perancangan Sistem

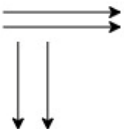
a. Flowchart

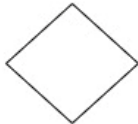
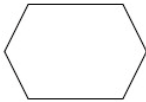
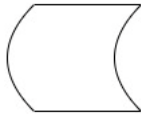
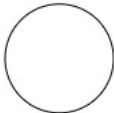
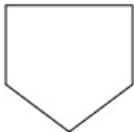
Menurut (Riyadi & Astutik, 2022) *flowchart* adalah instrumen grafis berbentuk diagram dua dimensi yang digunakan untuk menjelaskan urutan proses dalam sistem komputer secara logis. Melalui simbol-simbol khusus yang mewakili fungsi operasional dan garis kendali.

(Rasiban et al., 2024) menyebutkan bahwa *flowchart* atau bagan alur merupakan diagram yang memperlihatkan langkah-langkah beserta keputusan yang ada dalam sebuah proses program.

Dari kedua pendapat tersebut, dapat disimpulkan bahwa *flowchart* adalah alat bantu visual berupa diagram yang digunakan untuk menggambarkan alur dan tahapan suatu proses secara sistematis, sehingga memudahkan dalam memahami maupun merancang alur kerja sebuah sistem atau program.

Tabel 2. 1 Simbol *Flowchart*

Simbol	Nama	Fungsi
	Flowline	Menunjukkan arah aliran proses
	Terminator	Menandai titik awal (<i>start</i>) dan akhir (<i>stop</i>) alur proses
	Process	Menunjukkan proses pengolahan data atau tindakan operasional

	Input Output	Simbol yang menyatakan keluar masuk data
	Decision	Menunjukkan kondisi untuk pengambilan keputusan (ya/tidak)
	Preparation	Menyiapkan dan memberikan nilai awal suatu variabel/memor
	Document	Menampilkan output data dalam bentuk dokumen fisik/cetak
	Data (Input/Output)	Menunjukkan proses masukan (<i>input</i>) atau keluaran (<i>output</i>) data
	On-Page Connector	Menghubungkan alur proses dalam halaman yang sama
	Off-Page Connector	Menghubungkan alur proses pada halaman yang berbeda

(Sumber : (Rasiban et al., 2024))

b. UML

UML (*Unified Modeling Language*) digunakan untuk menggambarkan spesifikasi, konstruksi, serta dokumentasi bagian-bagian dari sistem perangkat lunak yang bertujuan mempermudah pengembang dalam membangun sistem. Selain itu, pemodelan UML

membuat komunikasi antara pengguna dan pengembang menjadi lebih mudah dan efektif (Anardani et al., 2023).

Sedangkan (Raharjo et al., 2026) berpendapat UML adalah bahasa yang digunakan untuk menggambarkan, menentukan, membangun dan mencatat komponen- komponen pada perangkat lunak. Artefak- artefak ini mencakup seperti model-model, penjelasan, atau perangkat dalam sistem perangkat lunak, seperti bisnis dan yang lainnya

Dari beberapa pendapat diatas, dapat disimpulkan bahwa UML adalah alat pemodelan standar yang banyak digunakan dalam pengembangan sistem berbasis *object oriented*, karena mampu membantu perancangan sistem secara lebih terstruktur dan mudah dipahami.

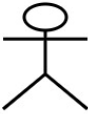
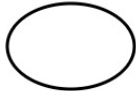
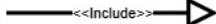
1) Use Case Diagram

Use case diagram merupakan sebuah diagram yang menekankan pada apa yang akan diperbuat oleh sistem serta siapa saja aktor yang akan menggunakan sistem tersebut. Diagram ini berfungsi untuk menggambarkan seluruh kebutuhan sistem (Endraswari & Putri Mentari, 2022:18).

Dari ke tiga pendapat di atas, dapat disimpulkan bahwa *use case diagram* merupakan salah satu diagram dalam UML yang berfungsi untuk menggambarkan interaksi antara pengguna (*aktor*)

dengan sistem, sehingga memudahkan dalam memahami kebutuhan dan alur penggunaan sebuah aplikasi

Tabel 2. 2 *Use Case Diagram*

Simbol	Nama	Deskripsi
	Actor	Menjelaskan pengguna yang berinteraksi dengan sistem.
	Use Case	Menggambarkan fungsi sistem tanpa menunjukkan struktur internal.
	Assosiation	Menunjukkan hubungan komunikasi antara actor dan use case.
	Extend	Menambahkan perilaku pada use case utama secara tidak langsung
	Use case Generalization	Hubungan antara use case umum dan use case khusus yang mewarisi fitur.
	Include	Menambahkan perilaku pada use case dasar secara eksplisit.

(Sumber : Suharni et al., 2023)

2) Sequence Diagram

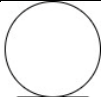
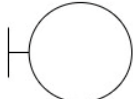
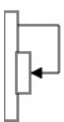
Menurut (Rasiban et al., 2024; Wayahdi & Ruziq, 2023)

sequence diagram merupakan diagram UML yang

memvisualisasikan interaksi dan pertukaran pesan antar-objek dari waktu ke waktu.

Dapat disimpulkan bahwa *sequence diagram* merupakan diagram UML yang digunakan untuk menggambarkan alur interaksi antar objek secara berurutan berdasarkan waktu, sehingga memudahkan pemahaman terhadap proses dan komunikasi yang terjadi di dalam sebuah sistem.

Tabel 2. 3 Simbol *Sequence Diagram*

Gambar	Nama	Keterangan
	Entity Class	Sistem sebagai landasan dalam menyusun basisdata
	Boundary Class	Menangani komunikasi antar lingkungan sistem
	Controll Class	Bertanggung jawab terhadap kelas objek yang berisi
	Recursive	Pesan untuk dirinya
	Activation	Mewakili proses durasi aktivitas sebuah operasi
	Life Line	Komponen yang digambarkan garis putus putus terhubung dengan objek

(Sumber: Rasiban at al., 2024)

3) Activity Diagram

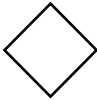
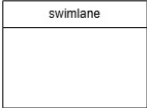
Activity diagram merupakan diagram yang berfungsi untuk menggambarkan dan memvisualisasikan diagram alir dari sebuah sistem atau organisasi. Diagram ini memuat serangkaian aktivitas, tindakan (*action*), percabangan, perulangan, hingga *concurrency* (proses paralel) untuk menjelaskan alur aktivitas yang terjadi secara sistematis (Endraswari & Putri Mentari, 2022:20).

Activity diagram merupakan cara untuk memodelkan aktivitas-aktivitas yang ada dalam suatu *use case*, sekaligus memperlihatkan bagaimana alur kendali berpindah dari satu aktivitas ke aktivitas berikutnya (Suharni et al., 2023).

Dapat disimpulkan bahwa *activity diagram* adalah diagram yang berfungsi untuk memodelkan dan menggambarkan alur aktivitas dalam sebuah sistem secara berurutan, sehingga memudahkan dalam memahami proses.

Tabel 2. 4 Simbol Activity Diagram

Simbol	Nama	Deskripsi
	Initial	Menunjukkan dimana aliran kerja dimulai.
	Final	Menggambarkan dimana aliran kerja berakhir.
	Action	Langkah langkah dalam sebuah activity.

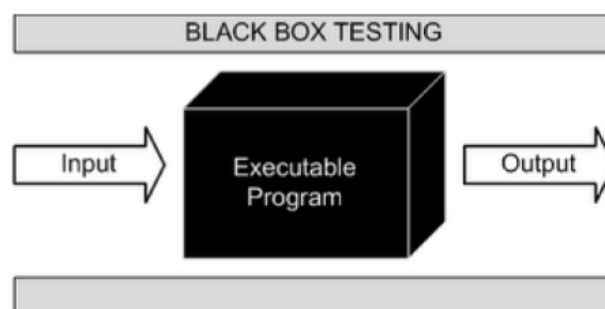
	Decision	Menunjukkan dimana keputusan akan dibuat
	Swimlane	Mengelompokkan activity berdasarkan actor.

(Sumber : Suharni at al., 2023)

11. Blackbox Testing

Menurut (Jibril et al.,2024 .; Mintarsih, 2023) *Black box* adalah metode yang befokus pada fungsi dan tampilan aplikasi. Dalam proses ini dilakukan dengan mengamati hasil eksekusi dari input data tertentu tanpa memeriksa *sourcecode*.

Dengan demikian, dapat disimpulkan bahwa metode *Black Box* berfungsi sebagai instrumen validasi untuk memastikan bahwa sistem beroperasi dan berpenampilan sesuai dengan spesifikasi kebutuhan pengguna (*user requirement*) dari sudut pandang pengguna akhir secara objektif.



Gambar 2. 1 Blackbox Testing

(Sumber : Abdillah et al., 2024)

12. Pengujian System Usability Scale (SUS)

SUS (*System Usability Scale*) merupakan metode penilaian yang menerapkan *quick and dirty* berdasarkan kusioner untuk mengukur tingkat kegunaan atau *usability* sistem informasi, yang diukur dari aspek efisiensi, efektifitas, dan kepuasan pengguna saat berinteraksi dengan *user interface* (Ridwan et al., 2024)

Menurut (Hakim et al., 2025) SUS menggunakan 10 pertanyaan dengan ukuran 5 poin yang terbagi menjadi dua arah, yaitu pernyataan positif (*favorable items*) dan pernyataan negatif (*unfavorable*).

(Yani et al., 2025) Skor SUS berfungsi sebagai Instrumen evaluasi untuk melihat potensi dan keterbatasan sistem, sejak diperkenalkan pada tahun 1986 yang diakui sebagai sarana pengujian perangkat lunak yang sederhana dan valid secara lintas domain.

Dapat disimpulkan bahwa SUS merupakan kusioner dengan 10 pertanyaan baik positif maupun negatif yang dapat dipercaya untuk menilai efisiensi, efektifitas, dan kepuasan pengguna. Selain untuk menentukan kelayakan sistem, hasil skor dari uji SUS juga menjadi analisis bagi pengembang untuk memperbaiki UI/UX (*user interface* dan *user experince*).

13. Metodologi Research and Development

Metode *Research and Development* (R&D) adalah metode penelitian yang berfokus pada perancangan dan pembuatan suatu produk

tertentu sekaligus melakukan pengujian terhadap tingkat efektivitas produk yang dihasilkan tersebut (Fayrus 2022:1; Sati et al., 2023).

Dapat disimpulkan bahwa metode *Research and Development* memiliki dua inti utama, yaitu pembuatan (*development*) dan pengujian (*research*).

B. Kajian Empiris

Dalam menyusun penelitian ini, peneliti menggunakan beberapa jurnal yang masih berkaitan dengan topik yang diangkat, yaitu implementasi *face recognition* dan *liveness detection* pada lingkungan Pemerintahan Desa Sirapan. Jurnal jurnal digunakan sebagai referensi yang diharapkan dapat membantu mematangkan dan memperkuat teori dalam penyusunan penelitian ini.

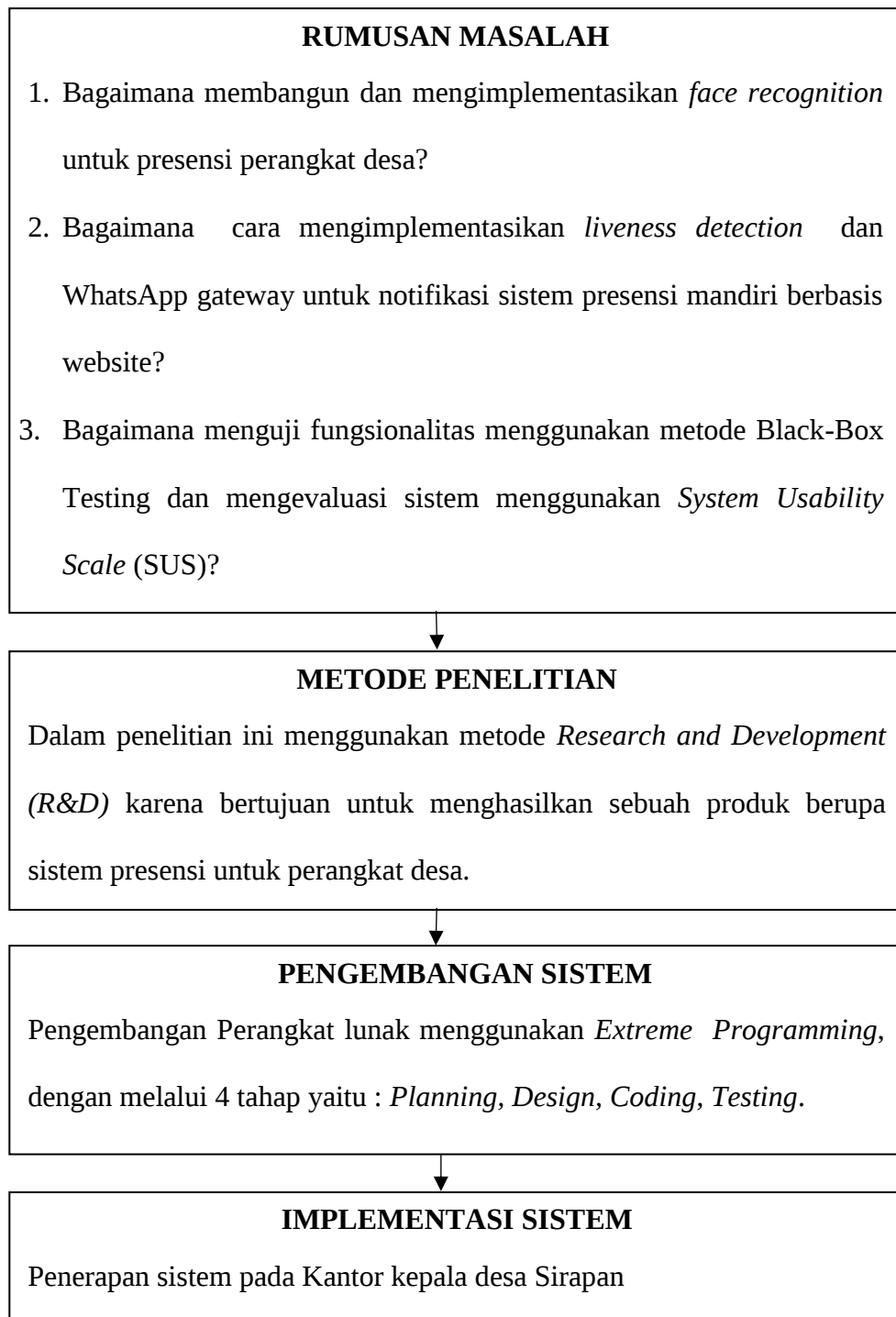
Pada beberapa penelitian terdahulu, sudah membahas *Face Recognition*, *liveness detection*, dan notifikasi *WhatsApp Gateway* secara terpisah. Penelitian yang dilakukan oleh (Nurlita et al., 2024) dan (Kamil et al., 2025) membuktikan bahwa *Insighface (ArcFace)* masih akurat dalam mengenali penggunanya meskipun variasi cahayanya beragam. Didukung oleh (Tanujaya et al., 2026) yang membuktikan algoritma YOLO bisa mendeteksi wajah dan *cropping* otomatis dengan *bounding box*. Pada penelitiannya (Dewi et al., 2025) dan (Sugeng & Nizar, 2023) memanfaatkan algoritma YAW dan *eye blink*, *head movement* untuk mencegah objek statis melalui *active liveness*. Penelitian yang dilakukan oleh (Rudianto, 2023) dan (Sari & Cahyani, 2022) memanfaatkan metode pengembangan sistem yaitu XP (*Extreme Programming*) yang mengedepankan ke efisiensinya.

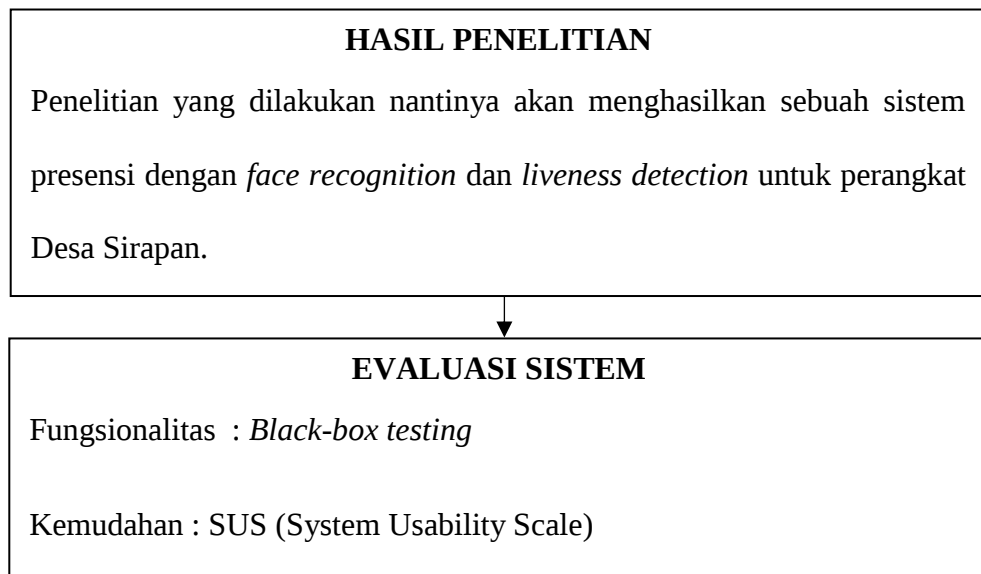
Sejalan dengan (Hanifah et al., 2025) menyatakan bahwa *fastAPI* memiliki kecepatan dan asinkronus untuk pengembangan backend modern, oleh karena itu *FastAPI* cocok digunakan sebagai *face recognition service* dalam penelitian ini.

Akan tetapi masih menyisakan celah yang masih berkaitan dan diselesaikan dalam penelitian ini. Pada aspek *Face Recognition* (Achmad & Supatman, 2024) masih memakai manual *cropping* berukuran 64 x 64 pixel yang berakibat menurunkan otomatisasi sistem, ditambah dengan penelitiannya (Husni & Munir, 2025) menemukan model FaceNet Menurun pada saat pencahayaan gelap. (Geralda et al., 2026) merekomendasikan penambahan keamanan berbentuk *liveness detection*.

Berdasarkan studi literatur yang dilakukan oleh peneliti, celah yang didapat berupa sistem presensi pemerintahan desa yang mengambil *Multi Authentication* yaitu *Active Liveness random challege*, *Face Recognition*, Verifikasi *Global Positioning System* (GPS), dalam penelitian ini, peneliti memilih algoritma YOLO yang akan di gunakan untuk mendeteksi objek wajah pengguna dan *auto cropping*, penggunaan *MediaPipe* dan *OpenCV* sebagai perhitungan rumus YAW untuk mendeteksi objek hidup, selanjutnya model *Insighface (ArcFace)* Buffalo_1 di pilih karena kuat untuk mengenali identitas meskipun cahaya redup. Dukungan *WhatsApp gateway* sebagai konfirmasi keberhasilan presensi dan peneliti memilih pendekatan XP (*Extreme Programming*) karena sangat cocok untuk proses *development* yang membutuhkan skalabilitas dan fleksibilitas dalam pembangunan sistem presensi berbasis *face recognition*.

C. Kerangka Berpikir





Gambar 2. 2 Rencana Penelitian