

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Dalam beberapa tahun terakhir, muncul sejumlah penelitian yang secara langsung atau tidak langsung terkait dengan pengembangan sistem presensi berbasis pengenalan wajah dan *platform web*. Berikut beberapa penelitian penting yang relevan.

Absensi *Face Recognition* Berbasis *Web* di SDN 07 Sindang Danau, Penelitian ini dilakukan oleh Effendy, Pratama, dan Syakti pada tahun 2023 dengan mengembangkan sistem informasi absensi dewan guru yang memanfaatkan *face recognition* berbasis *website* untuk mencatat kehadiran guru secara *real-time*. Sistem ini berhasil mengurangi kesalahan dan kehilangan data, sekaligus meningkatkan efisiensi dalam pencarian dan rekapitulasi absensi di SDN 07 Sindang Danau [1].

Penelitian yang dilakukan oleh Rifaldi dan Indriani dengan judul Rancang Bangun Aplikasi Presensi Mobile dengan *Face Recognition*. Penelitian ini dilakukan pada tahun 2024 dengan membangun aplikasi presensi berbasis mobile yang dilengkapi dengan *face recognition*. Dikembangkan menggunakan *framework Flutter* dan *backend Golang*, aplikasi ini memiliki tingkat keberhasilan presensi hingga 90% berdasarkan pengujian *black box* dan uji terima pengguna (UAT) [2].

Penelitian dengan judul *Presensi Face Recognition* di BiMBA AIUEO Cibinong ditahun 2024 yang dilakukan oleh Keyzha dan Fathurahman, penelitian ini merancang sistem presensi gabungan antara *face recognition* dan *fingerprint* dengan dukungan *IoT* (menggunakan ESP32Cam) yang terintegrasi melalui *platform website*. Teknologi *deep learning FOMO (Faster Objects, More Objects)* digunakan untuk deteksi wajah dengan akurasi sekitar 80% [3]. Penelitian yang berjudul *Face Recognition dengan Metode SVM* di tahun 2023 merupakan skripsi yang ditulis oleh Mahendra dari Universitas Teknologi

Digital Indonesia. Penelitian ini mengembangkan sistem presensi yang mengandalkan algoritma face recognition dengan pendekatan *Support Vector Machine* (SVM). Meskipun detail evaluasi performa rinci tidak dipublikasikan dalam abstrak, penelitian ini tetap relevan sebagai contoh penerapan algoritma klasik dalam sistem presensi berbasis wajah [4].

Penelitian yang dilakukan oleh Muliyadi B, Siska Anraeni dan Herman Herman dengan judul Rancang Bangun Sistem Absensi Online berbasis *Face Recognition* menggunakan *Platform Android* pada tahun 2022. Dalam penelitian ini, mereka membuat sebuah aplikasi, yang mana nantinya, dosen tidak repot lagi melakukan absensi secara manual, dan mahasiswa dapat dengan mudah absensi dengan cara scan wajah atau *Face Recognition* menggunakan metode pengenalan wajah Eigenface. Dari hasil penelitian ini diperoleh tingkat total akurasi keseluruhan adalah 75% [5].

Keseluruhan penelitian di atas menunjukkan bahwa integrasi face recognition dengan platform berbasis website semakin berkembang dan diaplikasikan dalam berbagai konteks mulai dari mobile, embedded system, hingga sistem berbasis *IoT*. Akurasi berkisar antara 75 hingga 95% dan beberapa sistem telah teruji secara real time.

Semua penelitian tersebut memberi pelajaran penting bagi pengembangan sistem presensi guru di SD Negeri Mejasem Timur 02, yaitu pentingnya efisiensi dan akurasi dalam pengenalan wajah serta kemudahan dalam penggunaan platform berbasis web untuk akses *real time* dan manajerial.

Dengan demikian, penelitian Tugas Akhir ini dapat memperkuat pendekatan yang telah dilakukan sebelumnya menggabungkan *face recognition* cepat dan akurat dengan *interface website* yang mudah digunakan dan diakses oleh pihak sekolah.

2.2. Landasan Teoretis

2.2.1. Sistem Informasi

Sistem informasi merupakan suatu sistem yang dirancang untuk mengolah data menjadi informasi yang berguna dalam mendukung kegiatan operasional, pengelolaan organisasi, serta proses pengambilan keputusan. Sistem ini terdiri atas beberapa komponen yang saling terintegrasi, yaitu perangkat keras, perangkat lunak, data, prosedur, dan sumber daya manusia. Kerja sama antarkomponen tersebut menghasilkan informasi yang akurat, relevan, dan tersedia pada saat diperlukan[6].

2.2.2. Sistem Informasi Pemesanan

Sistem informasi pemesanan adalah sistem berbasis komputer yang dimanfaatkan untuk mengelola seluruh proses pemesanan produk maupun layanan secara lebih efektif dan efisien. Melalui sistem ini, data pemesanan dapat diproses secara otomatis sehingga mempercepat pelayanan, memudahkan pengelolaan informasi, serta mengurangi risiko kesalahan dalam proses pencatatan.

2.2.3. *Pengenalan Wajah*

Laragon adalah sebuah *local development environment* yang dirancang untuk memudahkan developer dalam membuat dan menjalankan aplikasi *web* secara lokal tanpa perlu melakukan konfigurasi rumit. Sebagai software yang ringan dan portabel, *Laragon* dapat dijalankan dengan cepat dan tidak membebani sistem, bahkan bisa dipindahkan ke perangkat lain hanya dengan menyalin *folder* instalasinya. Selain itu, *Laragon* menyediakan fitur seperti *auto-virtual*

host yang memungkinkan setiap proyek otomatis mendapatkan domain lokal sendiri, *hashing password*, dan validasi input. Selain itu, terdapat *Artisan CLI*, yaitu alat baris perintah untuk mempercepat pembuatan file penting. *Laragon* mendukung penggunaan berbagai versi *PHP* yang dapat diganti dengan mudah, sehingga sangat membantu ketika mengerjakan proyek yang membutuhkan versi berbeda. Dengan integrasi komponen seperti *Apache/Nginx*, *MySQL/MariaDB*, *PHP*, dan *Node.js* dalam satu paket, *Laragon* menawarkan pengalaman pengembangan yang efisien, terorganisir, dan nyaman, menjadikannya pilihan ideal untuk pemula maupun *developer* berpengalaman yang ingin bekerja lebih cepat tanpa konfigurasi yang kompleks [13].

Pengenalan wajah (face recognition) adalah teknologi biometrik yang mengidentifikasi atau memverifikasi identitas seseorang berdasarkan pola unik pada wajahnya. Teknologi ini telah berkembang pesat dalam satu dekade terakhir dan banyak diterapkan dalam berbagai bidang, mulai dari keamanan nasional, perbankan, hingga manajemen kepegawaian.

Secara umum, sistem face recognition terdiri dari beberapa tahap utama, yaitu:

1. Akuisisi Gambar

Pengambilan gambar wajah menggunakan webcam/ IP camera / kamera terpasang (kiosk). Untuk aplikasi web sering digunakan WebRTC pada browser atau modul capture pada perangkat edge (Raspberry Pi/ESP32Cam).

2. Deteksi Wajah (Face Detection)

Menemukan lokasi/koordinat wajah pada citra (bounding box). Algoritma populer: Haar Cascade, HOG + linear SVM, MTCNN, YOLO/SSD. Deteksi harus cepat dan tahan terhadap variasi latar/pencahayaannya.

3. Preprocessing & Alignment

Menormalkan citra (resizing, histogram equalization), mendeteksi

titik referensi wajah (landmarks seperti mata, hidung, bibir) dan melakukan normalisasi pose (affine transform). Alignment meningkatkan konsistensi ekstraksi fitur.

4. Ekstraksi Fitur (Feature Extraction / Embedding)

Mengubah wajah menjadi representasi numerik (vektor/embedding) yang bersifat diskriminatif. Pendekatan klasik: LBPH, Eigenface (PCA), Fisherface (LDA). Pendekatan modern: CNN-based embeddings (mis. FaceNet, ArcFace, MobileFaceNet) yang menghasilkan vektor berdimensi tetap (contoh umum: 128/512 dimensi).

5. Pencocokan (*Matching*)

Membandingkan vektor embedding yang diperoleh dengan database (enrolled templates) menggunakan jarak (Euclidean) atau kesamaan kosinus. Keputusan hadir/identitas didasarkan pada threshold yang ditentukan (*trade-off antara False Accept Rate dan False Reject Rate*).

6. Manajemen Data & Pelaporan

Menyimpan hasil presensi ke database (mis. MySQL), manajemen akun guru, serta menghasilkan laporan kehadiran otomatis.

Keunggulan utama *face recognition* dibanding metode biometrik lainnya adalah kemudahan penggunaan dan non-invasif. Guru hanya perlu berdiri di depan kamera selama beberapa detik tanpa perlu menyentuh perangkat apapun. Hal ini sangat menguntungkan di lingkungan sekolah, terutama di masa pasca-pandemi, di mana kebersihan dan jarak fisik masih menjadi pertimbangan.

Selain itu, face recognition dapat diintegrasikan dengan perangkat yang relatif murah, seperti kamera web atau kamera IP, sehingga biaya implementasi tidak terlalu tinggi. Meskipun demikian, sistem ini memiliki tantangan, seperti variasi pencahayaan, sudut pandang wajah, atau perubahan penampilan (misalnya memakai masker, kacamata, atau janggut). Oleh karena

itu, penting untuk merancang sistem yang mampu beradaptasi dengan kondisi lingkungan sekolah, seperti ruang guru yang memiliki pencahayaan cukup dan posisi kamera yang tetap.

2.2.4. Website

Website merupakan kumpulan halaman *website* yang saling terhubung dan dapat diakses melalui jaringan internet menggunakan aplikasi peramban (*browser*). *Website* berfungsi sebagai media untuk menyampaikan informasi, menyediakan layanan digital, serta memfasilitasi interaksi antara pengguna dengan sistem melalui berbagai konten, seperti teks, gambar, audio, video, maupun fitur interaktif lainnya.

2.2.5. PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman yang dijalankan pada sisi server (*server-side scripting*) dan banyak digunakan dalam pengembangan aplikasi berbasis web yang bersifat dinamis. Bahasa pemrograman ini mampu memproses data yang diterima dari pengguna, mengelola logika aplikasi, serta berkomunikasi dengan berbagai sistem basis data sehingga menjadi salah satu teknologi yang banyak diterapkan dalam pengembangan *website*[8].



Gambar 2. 1 PHP (*Hypertext Preprocessor*)

2.2.6. Laravel

Laravel merupakan *framework* berbasis PHP yang mengadopsi konsep *Model-View-Controller (MVC)* untuk mempermudah proses pengembangan aplikasi web. Dengan struktur yang terorganisasi, Laravel membantu pengembang dalam membangun aplikasi yang lebih rapi, mudah dipelihara, dan efisien. Selain itu, *framework* ini menyediakan berbagai fitur bawaan, seperti *routing*, autentikasi pengguna, pengelolaan *database*, hingga sistem keamanan yang mendukung proses pengembangan aplikasi[9].



Gambar 2. 2 Laravel

2.2.7. MySQL

MySQL merupakan sistem manajemen basis data relasional (*Relational Database Management System* atau RDBMS) yang digunakan untuk menyimpan, mengelola, dan memproses data menggunakan bahasa *SQL (Structured Query Language)*. Sebagai perangkat lunak yang bersifat *open source*, MySQL banyak dimanfaatkan dalam pengembangan aplikasi web karena memiliki kinerja yang baik, mudah digunakan, serta kompatibel dengan berbagai bahasa pemrograman maupun *framework*, termasuk Laravel. Pada

penelitian ini, MySQL digunakan sebagai media penyimpanan seluruh data yang berkaitan dengan sistem informasi pelayanan dan pemesanan umroh berbasis *website*[10].



Gambar 2. 3 MySQL

2.2.8. Visual Studio Code

Visual Studio Code (VS Code) merupakan aplikasi penyunting kode (*source code Editor*) yang dikembangkan oleh *Microsoft. Editor* ini mendukung berbagai bahasa pemrograman, seperti PHP, HTML, CSS, dan JavaScript. Selain memiliki tampilan yang ringan, VS Code juga dilengkapi dengan berbagai fitur pendukung, seperti ekstensi, debugger, *IntelliSense*, serta integrasi Git yang membantu mempercepat proses pengembangan perangkat lunak[11].



Gambar 2. 4 Visual Studio Code

2.2.9. Laragon

Laragon merupakan perangkat lunak yang berfungsi sebagai server lokal (*localhost*) untuk menjalankan dan menguji aplikasi berbasis web sebelum dipublikasikan ke server *online*. Dengan menggunakan Laragon, proses pengembangan aplikasi menjadi lebih mudah karena pengembang dapat melakukan pengujian sistem secara langsung pada lingkungan lokal tanpa memerlukan koneksi internet[12].



Gambar 2. 5 Laragon

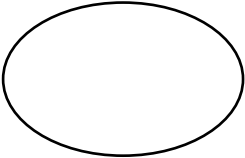
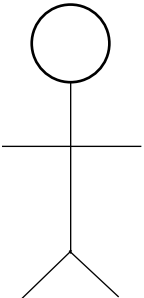
2.2.10. *Unified Modeling Language* (UML)

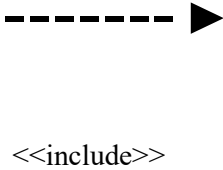
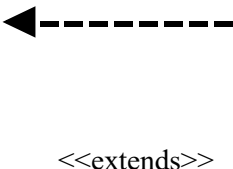
Unified Modeling Language (UML) merupakan bahasa pemodelan standar yang digunakan untuk menganalisis, merancang, memvisualisasikan, dan mendokumentasikan sistem perangkat lunak. UML membantu pengembang dalam menggambarkan struktur maupun alur kerja sistem sehingga proses pengembangan menjadi lebih terarah dan mudah dipahami. Pada penelitian ini, diagram UML yang digunakan meliputi *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, dan *Class Diagram*.

1. *Use Case Diagram*

Use Case Diagram digunakan untuk menggambarkan interaksi antara aktor dan sistem serta mengidentifikasi fungsi-fungsi yang tersedia sesuai kebutuhan pengguna. Simbol yang digunakan disajikan pada Tabel 2.1.

Tabel 2. 1 Simbol *Use Case Diagram*

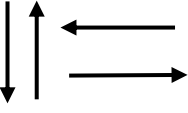
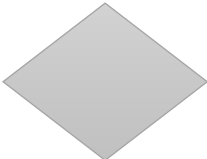
No	Simbol	Keterangan
1		<i>Use Case</i> merupakan penjelasan rinci merupakan penjelasan rinci dari sebuah <i>Use Case</i> yang menggambarkan fungsionalitas yang disediakan oleh sistem.
2		<i>Actor</i> yang berinteraksi dengan sistem untuk menjalankan fungsi tertentu.
3		<i>Association</i> garis yang menghubungkan aktor dengan <i>Use Case</i> untuk menunjukkan adanya interaksi antara keduanya.

4		<i>Include</i> , merupakan gambaran jika <i>Use Case</i> dipanggil usecase lain.
5		<i>Extends</i> , merupakan Gambaran jika memperluas <i>usecase</i> target.
6		<i>Dependency</i> merupakan garis panah yang menunjukkan jika <i>actor</i> berinteraksi secara pasif.

2. Activity Diagram

Activity Diagram digunakan untuk menggambarkan alur aktivitas atau proses bisnis dalam suatu sistem. Diagram ini menunjukkan urutan aktivitas dari awal hingga akhir, termasuk percabangan (*decision*) dan proses paralel yang mungkin terjadi selama sistem berjalan. Simbol-simbol yang digunakan dalam *Activity Diagram* ditunjukkan pada tabel berikut.

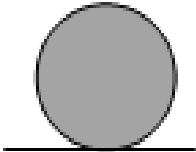
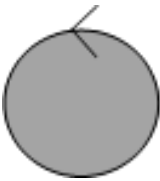
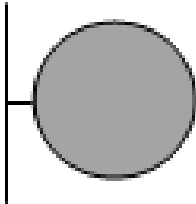
Tabel 2. 2 Simbol *Activity Diagram*

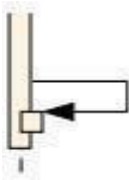
No	Simbol	Keterangan
1		<i>End Point</i> atau <i>Final Node</i> , merupakan Gambaran akhir dari suatu aktivitas.
2		<i>Start Point</i> , merupakan awal dari suatu aktivitas yang peletakkannya pada pojok kiri atas.
3		<i>Line Connector</i> , digunakan untuk menghubungkan satu symbol dengan symbol lainnya.
4		<i>Activity</i> , merupakan Gambaran dari suatu proses.
5		<i>Action</i> , State dari sistem yang mencerminkan eksekusi dari suatu aksi.
6		<i>Decision</i> , merupakan pilihan pengambilan suatu Keputusan <i>false or true</i> .

3. *Sequence Diagram*

Sequence Diagram digunakan untuk menggambarkan interaksi antar objek dalam sistem berdasarkan urutan waktu. Diagram ini menunjukkan bagaimana pesan (*message*) dikirim dan diterima antar objek untuk menjalankan suatu proses tertentu secara berurutan. Simbol-simbol yang digunakan dalam *Sequence Diagram* ditunjukkan pada tabel berikut.

Tabel 2. 3 Simbol *Sequence Diagram*

No	Simbol	Keterangan
1		<i>Entity Class</i> merupakan bagian sistem yang membentuk deskripsi awal sistem dan berisi Kumpulan kelas dalam bentuk.
2		<i>Control Class</i> merupakan Gambaran penghubung antara <i>Boundary class</i> dengan suatu <i>table</i> .
3		<i>Boundary Class</i> merupakan Gambaran dari penggambaran from.

4		Pesan atau <i>message</i> , menunjukkan pengiriman pesan antar <i>class</i>
5		<i>SelfMessage</i> , menunjukkan pengiriman suatu pesan yang akan dikirim ke objek itu sendiri
6		Activation menggambarkan suatu objek yang melakukan sebuah aksi/eksekusi operasi.
7		<i>Lifeline</i> garis titik yang terhubung ke objek disepanjang garis <i>lifeline</i> memiliki aktivitas.

4. *Class Diagram*

Class Diagram digunakan untuk menggambarkan struktur sistem dalam bentuk kelas, atribut, metode, serta hubungan antar kelas. Diagram ini menunjukkan bagaimana data dan fungsi dalam sistem saling berinteraksi. Selain itu, *Class Diagram* juga menggambarkan hubungan seperti asosiasi, generalisasi, dan agregasi antar kelas dalam sistem.

Elemen utama dalam *Class Diagram* meliputi kelas, atribut, metode (operasi), serta visibilitas akses.

Tabel 2. 4 Simbol *Class Diagram*

No	Simbol	Pengertian	Keterangan
1.	1	Hanya Satu	Satu dan hanya satu
2.	0...*	Nol atau Lebih	Boleh tidak ada atau 1 atau lebih
3.	1...*	Satu atau Lebih	Setidaknya satu
4.	0...1	Nol atau satu	Bisa jadi ada satu atau tidak ada sama sekali
5.	n...n	Batasan tertentu	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 maksimal 4