

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Menurut penelitian dari Andri Nugraha Ramdhon dan Fadly Febriya, Penelitian ini membahas mengenai pengembangan sistem presensi berbasis pengenalan wajah untuk meningkatkan efisiensi dan akurasi pencatatan kehadiran mahasiswa. Dalam penelitian tersebut dijelaskan bahwa sistem presensi manual maupun berbasis kartu identitas atau sidik jari masih memiliki berbagai kelemahan, seperti proses yang memakan waktu, risiko terjadinya *human error*, hingga antrian yang panjang karena input data dilakukan secara bergiliran. Penelitian ini memanfaatkan teknologi *face recognition* sebagai solusi yang lebih efisien dan modern. Metode yang digunakan adalah *Local Binary Pattern Histogram (LBPH)*, yang dinilai memiliki kinerja terbaik di antara metode lainnya seperti *Eigenface* dan *PCA*, karena mampu melakukan deteksi dan pengenalan wajah dengan cepat serta menghasilkan akurasi yang tinggi. Sistem dikembangkan menggunakan bahasa pemrograman *Python* dan *library OpenCV*, dengan kamera pengawas (*IP CCTV*) sebagai alat input citra wajah. Sistem ini memungkinkan proses presensi berlangsung secara otomatis dan *real-time*, tanpa perlu kontak fisik atau penggunaan kartu identitas. Hasil dari pengujian sistem menunjukkan bahwa tingkat keberhasilan pengenalan wajah mencapai 86,85%, dengan tingkat kegagalan sebesar 13,15%. Faktor-faktor yang memengaruhi akurasi

pengenalan wajah di antaranya adalah kondisi pencahayaan, jarak kamera ke objek, ekspresi wajah, serta penggunaan atribut tambahan seperti kacamata atau penutup kepala. Selain itu, kualitas data *training* juga menjadi faktor penting, di mana citra wajah yang digunakan sebagai data latih harus diambil dalam kondisi optimal dari berbagai sudut dan ekspresi agar sistem dapat mengenali wajah secara lebih akurat [1].

Menurut penelitian dari Nur Jaini, Ervan Asri, dan Fitri Nova, Penelitian ini mengembangkan sistem presensi digital menggunakan perangkat *ESP32-CAM*. Perangkat ini merupakan modul *mikrokontroler* kecil dengan kamera dan koneksi internet, digunakan untuk mendeteksi dan mengenali wajah mahasiswa secara otomatis. Proses deteksi dilakukan menggunakan algoritma *Haar Cascade*, sementara proses pengenalan wajah dijalankan melalui *library* yang telah ditanam pada *ESP32-CAM*. Sistem dilengkapi sensor jarak *ultrasonik* untuk memicu pemindaian wajah saat mahasiswa mendekat, serta *DF Player* dan *speaker* untuk memberikan notifikasi suara jika presensi berhasil. Seluruh data presensi disimpan secara otomatis dalam *database MySQL*, dan dapat dikelola melalui antarmuka *web*. Sistem ini dirancang agar dapat digunakan oleh banyak kelas secara bergantian hanya dengan satu perangkat. Penelitian juga melibatkan beberapa komponen pendukung seperti *Arduino Nano*, *SIM800L* (untuk koneksi internet via *GPRS*), dan sistem notifikasi berbasis suara. Hasil pengujian menunjukkan bahwa sistem mampu mengenali hingga 200 data wajah, membedakan antara mahasiswa yang hadir dan yang hanya melintas, serta

mencatat data kehadiran ke server hanya dalam waktu sekitar 35 detik. Dengan sistem berbasis *web*, admin dapat mengelola jadwal dan presensi secara digital dan *real-time* [2].

Menurut penelitian dari Sarah Sahira, Hana Nurcahya Sumirat, Ivana Lucia Kharisma, Anggun Fergina, dan Kamdan. Penelitian ini mengembangkan sistem presensi digital menggunakan *Face Recognition* berbasis *web* untuk siswa lintas minat, seperti bahasa Inggris dan Jerman, yang mengharuskan siswa berpindah kelas. Sistem menggunakan metode *Facial Landmark Detection* untuk mendeteksi titik-titik penting pada wajah. Diterapkan dengan pendekatan *Waterfall* dan diuji menggunakan metode *Black Box Testing*, sistem ini mendukung fitur *autentikasi* wajah, validasi lokasi, *dashboard* siswa-guru-admin, serta rekap presensi otomatis dalam format PDF. Hasil pengujian menunjukkan sistem bekerja dengan efisien dan akurat, mampu menolak presensi di luar lokasi atau wajah yang tidak dikenali. Sistem ini membantu mengatasi masalah presensi manual seperti kecurangan, kehilangan form, dan keterlambatan rekap data. Penelitian ini memberikan kontribusi nyata dalam penerapan *face recognition* berbasis *web* di lingkungan sekolah menengah [3].

Menurut penelitian dari Muhammad Hasyim Nasrun, Penelitian ini mengevaluasi penerimaan teknologi *Face ID* dalam sistem presensi siswa dengan pendekatan deskriptif kuantitatif dan model *Technology Acceptance Model* (TAM). Tiga variabel utama yang dianalisis adalah kemudahan penggunaan (X1), kebergunaan (X2), dan niat menggunakan teknologi (X3),

dengan ketepatan waktu kehadiran (Y) sebagai variabel dependen. Dari 97 responden siswa, mayoritas menyatakan sistem *Face ID* mempermudah presensi dan mendorong kehadiran tepat waktu. Hasil analisis menunjukkan semua variabel *independen* memiliki pengaruh signifikan terhadap ketepatan waktu, baik secara parsial maupun simultan. Penelitian ini menegaskan bahwa keberhasilan sistem presensi biometrik juga dipengaruhi oleh penerimaan dan persepsi pengguna, tidak hanya aspek teknis [4].

Menurut penelitian dari Akbar Syarif Pratama Putra dkk. Fokus penelitian ini adalah pengembangan sistem presensi siswa berbasis *web* yang menggabungkan algoritma *Viola-Jones* untuk deteksi wajah dan validasi lokasi GPS guna mencegah kecurangan presensi. Sistem diuji di SMA Negeri 2 Surakarta menggunakan 200 data gambar dari 20 siswa, dan dibangun menggunakan *Laravel 11* pada Windows 11. Hasil pengujian menunjukkan akurasi deteksi wajah mencapai 90%, dengan performa terbaik saat kondisi pencahayaan baik dan wajah berada dalam posisi frontal. Fitur utama sistem meliputi *autentikasi* wajah, validasi lokasi, *dashboard* presensi, dan *feedback* sistem otomatis. Penelitian ini membuktikan bahwa kombinasi teknologi *face recognition* dan pelacakan lokasi dapat menciptakan sistem presensi yang efektif, akurat, dan sulit dimanipulasi, sehingga mendukung kehadiran siswa yang lebih disiplin dan transparan [5].

2.2 Landasan Teori

2.2.1 Proses Sistem Presensi

Sistem presensi berbasis *web* adalah solusi modern yang memanfaatkan teknologi *Face ID* untuk mencatat kehadiran dengan cepat dan akurat. Proses presensi dengan sistem ini dilakukan menggunakan teknologi pengenalan wajah, dimana sistem akan memindai serta mencocokkan wajah pengguna dengan data identitas yang tersimpan dalam *database*. Setelah wajah dikenali, data kehadiran secara otomatis tercatat di sistem berbasis *web*.

Keunggulan utama dari sistem presensi ini adalah efisiensinya. Pengguna tidak perlu lagi membawa kartu atau memasukkan data secara manual, sehingga meminimalkan risiko kehilangan kartu atau kecurangan dalam presensi. Selain itu, sistem ini juga memungkinkan pengelolaan data secara *real-time*, sehingga administrator dapat memantau kehadiran dari mana saja melalui platform berbasis *web*. Dengan adanya integrasi ke dalam sistem *web*, laporan kehadiran dapat diakses dan diunduh sesuai kebutuhan, memberikan kemudahan dalam proses rekapitulasi dan analisis data.

2.2.2 Website

Website merupakan bagian dari teknologi internet, dimana teknologi adalah sistem yang diciptakan oleh manusia untuk maksud dan tujuan tertentu untuk mempermudah manusia dalam meringankan usahanya, meningkatkan hasilnya, dan menghemat tenaga dan sumber

daya yang ada. Saat ini, internet menjadi sumber informasi yang paling banyak digunakan untuk mencari informasi yang dibutuhkan karena pada jaringan internet terdapat sebuah *search engine* atau mesin pencari yang dapat digunakan oleh pengguna untuk mencari halaman-halaman *web* yang dibutuhkannya. Ketika menuliskan informasi yang dibutuhkan pada *search engine* melalui jaringan internet, yang akan tampil adalah berbagai halaman *web* beserta informasi di dalamnya [6].

2.2.3 *Monitoring*

Monitoring berarti pengamatan, pemeriksaan, pengendalian, dan pengoreksian aktivitas yang hendak diketahui. Sistem *monitoring* juga disebut sebagai upaya sistematis untuk menetapkan kinerja standar untuk perencanaan untuk merancang sistem umpan balik informasi, tujuan dari sistem ini adalah untuk membandingkan kinerja actual dengan standar yang telah ditentukan [1].

2.2.4 *MySQL*

MySQL merupakan sistem manajemen *database* yang bersifat *open source*. Sistem *database MySQL* mendukung beberapa fitur seperti *multi-threaded*, *multi-user*, dan *SQL database* manajemen *system*. Sampai saat ini, penggunaan *MySQL* sebagai *database* penyimpanan data adalah yang paling banyak digunakan dibandingkan dengan *database* manajemen *system* yang lain. Keuntungan menggunakan *MySQL* antara lain sangat mudah

mengakses *datasenya*, kinerja *query* cepat, dan mencukupi untuk kebutuhan *database* perusahaan-perusahaan yang berkala kecil sampai berskala besar (Hastomo 2013) [7].



Gambar 2.1 MySQL

2.2.5 Visual Code

Visual Studio Code adalah *Software* yang sangat ringan, namun kuat *editor* kode sumbernya yang berjalan dari *desktop*. *Visual Studio Code* digunakan untuk pembuatan kode-kode program dibutuhkan sebuah aplikasi yang mumpuni (Hartati, 2020). *Visual Studio Code* dapat digunakan untuk berbagai Bahasa pemrograman seperti *JavaScript*, *HTML*, *CSS*, *PHP*, *Python*, *C++*, dan masih banyak lagi. *Visual Studio Code* bekerja pada berbagai sistem operasi seperti *Windows*, *macOS*, dan *Linux*. Selain itu, *Visual Studio Code* menyediakan fitur *Live Share* memungkinkan beberapa pengembang bekerja pada satu proyek yang sama secara bersamaan dari lokasi yang berbeda [8].



Gambar 2.2 *Visual Code*

2.2.6 ***PHP (Hypertext Preprocessor)***

PHP adalah sebuah Bahasa pemrograman yang didesain untuk pengembangan *website*. *PHP* bersifat *open source* sehingga dapat digunakan dan dikembangkan dengan gratis. Kode atau script *PHP* dapat ditambahkan dalam bahasa pemrograman *HTML*. Kebanyakan *programmer web* di seluruh dunia menggunakan *PHP* karena kemudahan penggunaannya (Andre 2014) [7].



Gambar 2.3 *PHP (Hypertext Preprocessor)*

2.2.7 ***Framework Laravel***

Laravel adalah *framework open source PHP* berbasis *web* gratis yang dibuat oleh Taylor Otwell dan ditujukan untuk pengembangan aplikasi *web* mengikuti *model-view-controller* (MVC) atau pola arsitektur. Beberapa fitur dari *Laravel* adalah pengembangan sistem modul-modul yang dapat dimanajemen, mengenalkan cara

yang berbeda untuk mengakses *database* relasional, utilitas yang membantu dalam penyebaran aplikasi dan pemeliharaan yang mudah [9].

Laravel memiliki beberapa kelebihan diantaranya; menggunakan *package manager PHP Composer*, penulisan kode program yang rapi, singkat dan terstruktur, dan mudah dimengerti *developer* [10].



Gambar 2.4 *Framework Laravel*

2.2.8 *OpenCV*

OpenCV (Open Computer Vision Library) merupakan *library* perangkat lunak *open source* yang memiliki lisensi *BSD-licensed product*. *OpenCV* memiliki lebih dari 2500 algoritma yang telah dioptimasi disediakan untuk menangani hal mengenai *computer vision* dan *machine learning*. Algoritma yang ada dapat digunakan untuk mendeteksi wajah, mengenali wajah, mengidentifikasi objek, dan lain-lain. Algoritma *OpenCV* untuk pengenalan wajah sampai pada saat penelitian ditulis, telah tersedia tiga algoritma pengenalan wajah, diantaranya *Eigenface*, *Fisherface*, dan *Local Binary Pattern*

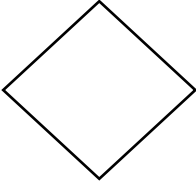
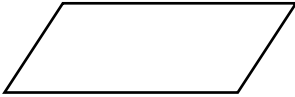
Histogram (LBPH) [1].

2.2.9 *Flowchart*

Flowchart atau sering disebut dengan diagram alir merupakan suatu jenis diagram yang merepresentasikan algoritma atau langkah-langkah instruksi yang berurutan dalam sistem. seorang analis sistem menggunakan *flowchart* sebagai bukti dokumentasi untuk menjelaskan gambaran logis sebuah sistem yang akan dibangun kepada *programmer*. Dengan begitu, *flowchart* dapat membantu untuk memberikan solusi terhadap masalah yang bisa saja terjadi dalam membangun sistem. Pada dasarnya, *flowchart* digambarkan dengan menggunakan simbol-simbol. Setiap simbol mewakili suatu proses tertentu. Sedangkan untuk menghubungkan satu proses ke proses selanjutnya digambarkan dengan menggunakan garis penghubung [11].

Dengan adanya *flowchart*, setiap urutan proses dapat digambarkan menjadi lebih jelas. Selain itu, ketika ada penambahan proses baru dapat dilakukan dengan mudah menggunakan *flowchart* ini. Setelah proses membuat *flowchart* selesai, maka giliran *programmer* yang akan menerjemahkan desain logis tersebut kedalam bentuk program dengan berbagai bahasa pemrograman yang telah disepakati [11].

Tabel 2.1 Tabel Simbol *Flowchart*

No	Gambar	Simbol	Keterangan
1.		Proses / Langkah	Menyatakan kegiatan yang akan ditampilkan dalam diagram alir.
2.		Titik Keputusan	Proses / Langkah di mana perlu adanya keputusan atau adanya kondisi tertentu. Di titik ini selalu ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda.
3.		Masukan / Keluaran Data	Digunakan untuk mewakili data masuk, atau data keluar.
4.		Terminasi	Menunjukkan awal atau akhir sebuah proses.
5.		Garis alir	Menunjukkan arah aliran proses atau algoritma.
6.		Kontrol / Inspeksi	Menunjukkan proses langkah di mana ada inspeksi.

2.2.10 UML (*Unified Modeling Language*)

Unified Modeling Language (UML) adalah sebuah bahasa pemodelan perangkat lunak yang telah distandardisasi sebagai media penulisan cetak biru (*blueprints*) perangkat lunak (*Pressman*). *UML* bisa saja digunakan untuk visualisasi, spesifikasi, konstruksi dan dokumentasi beberapa bagian-bagian dari *system* yang ada dalam perangkat lunak. Dalam kata lain, seperti halnya seorang arsitek dalam membuat dokumen cetak biru yang digunakan oleh perusahaan konstruksi untuk membangun sebuah bangunan, arsitek perangkat lunak membuat diagram-diagram *UML* untuk membantu *programmer developer* membangun perangkat lunak. Untuk selanjutnya, semakin kita mengetahui beberapa kosakata yang digunakan *UML*, kita akan semakin mudah dalam memahami spesifik [12].