
BAB II

TINJUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Pengertian Aplikasi

“Aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju. Menurut kamus komputer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang diharapkan” (Budiyanto, 2023).

Secara singkat aplikasi adalah program yang dirancang untuk melaksanakan fungsi tertentu bagi penggunaannya, dengan tujuan memecahkan masalah melalui pemrosesan data sesuai kebutuhan atau harapan pengguna.

2.1.2 Pengertian Pengenalan Wajah (*Face Recognition*)

“Pengenalan Wajah adalah teknologi yang digunakan untuk mengidentifikasi dan memverifikasi individu berdasarkan fitur wajah mereka” (Walangitan et al., 2024).

Secara singkat Pengenalan Wajah adalah teknologi yang digunakan untuk mengidentifikasi dan memverifikasi individu dengan menganalisis fitur wajah mereka.

2.1.3 Pengertian Absensi

“Absensi merupakan sebuah daftar hadir sekelompok orang yang tergabung dalam suatu instansi secara resmi yang mempunyai peraturan, ketentuan, dan pembatasannya, serta mereka yang terlibat, terikat oleh peraturan itu” (Ayunita Pertiwi et al., 2023).

Secara singkat, Absensi adalah daftar hadir resmi yang mencatat kehadiran anggota suatu instansi sesuai dengan aturan dan ketentuan yang berlaku.

2.1.3 Pengertian Metode *Linear Discriminant Analysis (LDA)*

“*Linear Discriminant Analysis (LDA)* merupakan salah satu metode yang dipakai untuk pengenalan pola pada perhitungan statistika dengan cara menemukan proyeksi linear dari data yang akan memaksimalkan jarak antar kelas dan meminimalkan jarak data yang memiliki kesamaan” (Budiman et al., 2021).

Secara singkat *Linear Discriminant Analysis (LDA)* adalah metode pengenalan pola yang menggunakan proyeksi linier untuk memaksimalkan jarak antar kelas dan meminimalkan jarak dalam kelas yang sama.

2.1.4 Pengertian Metode *Euclidean Distance*

“*Euclidean Distance* adalah perhitungan jarak dari 2 buah titik dalam Euclidean Space. Euclidean Space diperkenalkan oleh Euclid, seorang matematikawan dari Yunani sekitar tahun 300 B.C.E. untuk mempelajari hubungan antara sudut dan jarak. Euclidean ini berkaitan dengan Teorema Pythagoras dan biasanya diterapkan pada 1, 2 dan 3 dimensi. Tapi juga sederhana jika diterapkan pada dimensi yang lebih tinggi” (Agustina et al., 2022).

2.1.5 Pengertian Aplikasi *E-Visitor* dengan *Face Recognition* untuk Absensi Tamu Menggunakan Metode *Linear Discriminant Analysis* di Dinas Kearsipan Provinsi Sumatera Selatan

Judul ini menggambarkan sebuah sistem aplikasi berbasis *website* yang dirancang khusus untuk mencatat kehadiran atau absensi tamu secara otomatis di lingkungan Dinas Kearsipan Provinsi Sumatera Selatan. Aplikasi ini dinamakan *E-Visitor*, yang merujuk pada sistem kunjungan elektronik.

Untuk memastikan kehadiran yang valid dan efisien, aplikasi ini dilengkapi dengan fitur *Face Recognition* atau pengenalan wajah, yang mampu mengidentifikasi wajah tamu yang berkunjung. Teknologi ini menggantikan metode manual seperti tanda tangan di buku tamu, sehingga lebih modern, cepat, dan minim penyalahgunaan.

Dalam melakukan proses pengenalan wajah, aplikasi ini menggunakan metode *Linear Discriminant Analysis (LDA)*, yaitu suatu teknik dalam bidang pembelajaran mesin (*machine learning*) yang bertujuan untuk mengklasifikasikan data dengan mengurangi dimensi dan memaksimalkan pemisahan antar kelas. *LDA* membantu sistem untuk membedakan wajah satu tamu dengan tamu lainnya secara akurat.

2.1.6 Pengertian Diagram Konteks

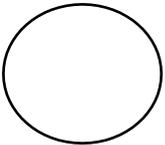
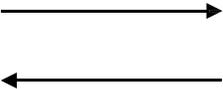
“Diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram Konteks merupakan level 1 tertinggi dari DFD yang menggambarkan seluruh *Input* ke sistem atau *Output* dari sistem” (Safwandi, 2021).

“Diagram konteks merupakan suatu bentuk diagram yang menggambarkan

proses serta ruang lingkup kerja suatu sistem” (Fadilla & Wulandari, 2023).

Kedua pendapat tersebut menyimpulkan bahwa diagram konteks adalah representasi visual dari proses utama dan ruang lingkup suatu sistem, yang menunjukkan seluruh aliran data berupa input dan output dari atau ke entitas luar. Diagram ini berada pada level tertinggi dalam *Data Flow Diagram* (DFD) dan digunakan untuk menggambarkan batasan sistem serta interaksi eksternal yang terjadi tanpa memperlihatkan detail proses internal.

Tabel 2. 1 Simbol-Simbol Diagram Konteks

No	Notasi	Keterangan
1.		Proses (<i>Process</i>) atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.
2.		File atau basis data atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.
3.		Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.
4.		Aliran data; merupakan data yang dikirim antar-proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Catatan : Nama yang digunakan pada aliran data

Sumber : Safwandi (2021)

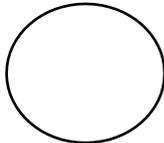
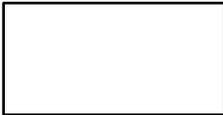
2.1.7 Pengertian *Data Flow Diagram* (DFD)

“*Data Flow Diagram* (DFD) merupakan sebuah teknik yang digunakan untuk menggambarkan komponen-komponen utama dari suatu sistem, serta menunjukkan

aliran-aliran data yang terjadi di antara komponen tersebut, termasuk asal data, tujuan akhir, dan tempat penyimpanannya dalam sistem secara keseluruhan.” (Safwandi, 2021).

Dengan demikian, teknik ini bertujuan untuk memberikan pemahaman menyeluruh tentang bagaimana data mengalir melalui berbagai komponen dalam sistem, mulai dari sumber data, proses yang dilalui, hingga penyimpanan dan output akhirnya.

Tabel 2. 2 Simbol-Simbol pada *Data Flow Diagram* (DFD)

No	Notasi	Keterangan
1.		Proses (Process) atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.
2.		File atau basis data atau penyimpanan (storage); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.
3.		Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.
4.		Aliran data; merupakan data yang dikirim antar-proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Catatan : Nama yang digunakan pada aliran data

Sumber : Safwandi (2021)

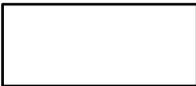
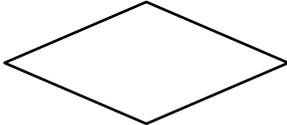
2.1.8 Pengertian *Entity Relationship Diagram* (ERD)

“*Entity Relationship Diagram* (ERD) adalah sebuah diagram yang digunakan untuk merancang hubungan antar tabel-tabel dalam basis data. ERD menggambarkan entitas yang ada dalam sistem dan relasi antar entitas tersebut. Diagram ini membantu dalam pemodelan data yang jelas dan mempermudah proses pengembangan sistem

informasi. ERD juga digunakan untuk mengidentifikasi atribut-atribut yang diperlukan dalam setiap entitas. (Nurfitriana et al., 2021).

Dari pengertian diatas, dapat disimpulkan bahwa *Entity Relationship Diagram (ERD)* adalah diagram yang menggambarkan hubungan antar tabel, entitas, relasi, dan atribut dalam basis data, sehingga mempermudah pemodelan data dan pengembangan sistem.

Tabel 2. 3 Simbol-Simbol pada *Entity Relationship Diagram (ERD)*

No.	Simbol	Deskripsi
1.	 Entitas/ <i>entity</i>	Entitas adalah data inti yang akan disimpan sebagai tabel dalam basis data, mewakili objek yang memiliki data dan perlu diakses.
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut kunci primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan.
4.	Atribut multinilai/ <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas ; biasanya diawali dengan kata kerja

Sumber : Nurfitriana et al. (2021)

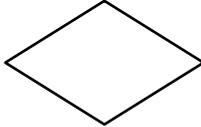
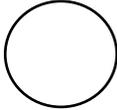
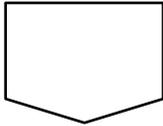
2.1.9 Pengertian *Flowchart*

“*Flowchart* merupakan gambar atau bagan yang memperlihatkan urutan atau langkah-langkah dari suatu program dan hubungan antar proses beserta pernyataannya” (Fauzi, 2020).

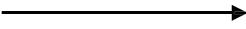
Dari pengertian diatas, dapat disimpulkan bahwa *flowchart* adalah diagram

yang menunjukkan urutan langkah dan hubungan antar proses dalam suatu program.

Tabel 2. 4 Simbol-simbol pada *Flowchart*

No.	Simbol	Nama	Keterangan
1.		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program
2.		<i>Input/Output</i>	Menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
3.		<i>Proses</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
4.		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban ya/tidak
5.		<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
6.		<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
7.		<i>Predefined Process</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda

Lanjutan Tabel 2. 5 Simbol-simbol pada *Flowchart*

No.	Simbol	Nama	Keterangan
8.		<i>Punched Card</i>	Menyatakan <i>input</i> berasal dari kartu atau <i>output</i> ditulis ke kartu
9.		<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui printer)
10.		<i>Flow</i>	Menyatakan jalannya arus suatu proses

Sumber : Fauzi (2020)

2.1.10 Pengertian *Hypertext Preprocessor (PHP)*



Gambar 2. 1 Logo *Hypertext Preprocessor (PHP)*

(Sumber : Wikipedia)

“*PHP* atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan *web* dan dapat ditanamkan pada sebuah skripsi HTML. Bahasa *PHP* dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti C, Java, dan Perl serta mudah untuk dipelajari. *PHP* merupakan bahasa *scripting server – side*, dimana pemrosesan datanya dilakukan pada sisi server” (Noviantoro et al., 2022).

Singkatnya, *PHP* adalah bahasa pemrograman *open source* untuk

pengembangan *web* yang mudah dipelajari dan berjalan di sisi server, serta dapat disisipkan dalam HTML.

2.1.11 Pengertian *Cascading Style Sheet (CSS)*



Gambar 2. 2 Logo *Cascading Style Sheets (CSS)*

(Sumber : Wikipedia)

“*Cascading Style Sheet (CSS)* merupakan salah satu bahasa pemrograman *web* yang bertujuan untuk membuat *website* agar lebih menarik dan terstruktur. Dari beberapa pendapat di atas, dapat disimpulkan bahwa CSS adalah suatu bahasa pemrograman *web* yang berfungsi mengatur tampilan teks dan gambar dari suatu *website* agar terlihat lebih menarik dan terstruktur. Cara kerja CSS dalam memodifikasi HTML dengan memilih elemen HTML yang akan diatur kemudian memberikan property yang sesuai dengan tampilan yang diinginkan” (Noviantoro et al., 2022).

Singkatnya, CSS adalah bahasa pemrograman *web* yang digunakan untuk mengatur tampilan elemen HTML agar *website* terlihat lebih menarik dan terstruktur.

2.1.12 Pengertian *Javascript (JS)*



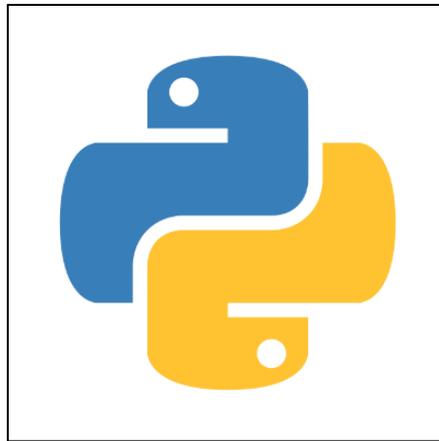
Gambar 2. 3 Logo *Javascript (JS)*

(Sumber : Wikipedia)

“*JavaScript* (JS) adalah bahasa pemrograman tingkat tinggi yang pertama kali dibuat untuk membuat *Website* lebih "hidup". *JavaScript*, bersama dengan HTML dan CSS, adalah bahasa pemrograman yang paling umum digunakan untuk membuat aplikasi berbasis *website*. Bahasa ini dapat membuat *website* lebih logis -- membuatnya lebih Interaktif” (Arisantoso et al., 2023).

JavaScript adalah bahasa pemrograman tingkat tinggi yang digunakan untuk menambahkan interaktivitas dan logika pada *website*. Bersama HTML dan CSS, *JavaScript* membentuk fondasi utama dalam pengembangan aplikasi berbasis web, menjadikan tampilan web lebih dinamis, responsif, dan "hidup" bagi pengguna.

2.1.13 Pengertian *Python*



Gambar 2. 4 Logo *Python*

(Sumber : Wikipedia)

“*Python* merupakan salah satu bahasa pemrograman yang banyak digunakan oleh perusahaan besar maupun para developer untuk mengembangkan berbagai macam aplikasi berbasis desktop, web dan mobile. *Python* diciptakan oleh Guido van Rossum di Belanda pada tahun 1990 dan namanya diambil dari acara televisi kesukaan Guido Monty *Python*'s Flying Circus. Van Rossum mengembangkan *Python* sebagai hobi, kemudian *Python* menjadi bahasa pemrograman yang dipakai secara luas dalam industri dan pendidikan karena sederhana, ringkas, sintak intuitif dan memiliki pustaka yang luas” (Muhammad Romzi & Kurniawan, 2020).

“*Python* adalah bahasa pemrograman interpretatif yang dianggap mudah dipelajari serta berfokus pada keterbacaan kode. *Python* secara umum berbentuk pemrograman berorientasi objek, pemrograman imperatif dan pemrograman

fungsional” (Kurniawan & Romzi, 2022).

Python adalah bahasa pemrograman yang populer dan mudah dipelajari, dengan sintaks sederhana dan pustaka luas. Diciptakan oleh Guido van Rossum pada 1990, *Python* mendukung berbagai paradigma pemrograman dan banyak digunakan dalam pengembangan aplikasi di berbagai bidang.

2.1.14 Pengertian *Bootstrap*



Gambar 2. 5 Logo *Bootstrap*

(Sumber : Wikipedia)

“*Bootstrap* adalah kerangka kerja (framework) front-end open-source yang digunakan untuk membangun tampilan situs *web* dan aplikasi *web* responsif (responsive *web* design) yang dapat diakses melalui perangkat apapun, seperti *desktop*, *tablet*, atau *smartphone*. *Bootstrap* dikembangkan oleh Twitter dan dirilis pada tahun 2011 sebagai alat internal untuk mengembangkan proyek mereka sendiri, kemudian dilepaskan sebagai proyek open-source” (Noviantoro et al., 2022)

Singkatnya, *Bootstrap* adalah framework front-end open-source yang dikembangkan oleh *Twitter* untuk membuat situs *web* responsif yang dapat diakses di berbagai perangkat, dan dirilis sebagai proyek *open-source* pada 2011.

2.1.15 Pengertian *MY Structure Query Language (MySQL)*



Gambar 2. 6 Logo *MY Structure Query Language (MySQL)*

(Sumber : Wikipedia)

“*MySQL* (*MY Structure Query Language*) adalah salah satu dari sekian banyak DBMS seperti Oracle, MS SQL, Postagre SQL, dan lainnya. *MySQL* adalah sistem manajemen basis data yang menggunakan SQL untuk mengelola data. *MySQL* adalah database *open source*, yang artinya Anda dapat menggunakannya secara gratis. Pemrograman *PHP* juga sangat mendukung atau support dengan Basis Data *MySQL*” (Noviantoro et al., 2022).

Singkatnya, *MySQL* adalah DBMS *open source* yang menggunakan SQL untuk mengelola data, banyak digunakan karena gratis dan sangat kompatibel dengan pemrograman *PHP*.

2.1.16 Pengertian *Quick Response Code (QR Code)*



Gambar 2. 7 Logo *Quick Response Code (QR Code)*

(Sumber : Wikipedia)

“*QR Code* adalah teknik yang mengubah data tertulis menjadi kode-kode dua dimensi yang tercetak ke dalam suatu media yang lebih ringkas. *QR Code* merupakan dua dimensi yang diperkenalkan pertama kali oleh perusahaan Jepang *Denso-Wave* pada tahun 1994. *Barcode* pertama kali digunakan untuk pendataan inventaris produksi suku cadang kendaraan dan sekarang sudah digunakan dalam berbagai bidang” (Mubarok & Chotijah, 2021).

Singkatnya, *QR Code* adalah kode dua dimensi buatan *Denso-Wave* yang berfungsi mengubah data menjadi bentuk ringkas dan mudah dipindai, awalnya digunakan untuk suku cadang, kini digunakan di berbagai bidang.

2.2 State Of The Art

Tabel 2. 6 Tabel State Of The Art

No	Judul	Hasil Penelitian
1.	Penggunaan Metode <i>Linear Discriminant Analysis</i> Untuk Pengenalan Wajah Dengan Membandingkan Banyaknya Data Latih (Kosasih, 2021)	Akurasi tertinggi 97,5% dicapai dengan 9 citra per orang dan $K=1$, menunjukkan kombinasi terbaik antara jumlah data latih dan parameter <i>Euclidean Distance</i> .
2.	Optimalisasi Pengenalan Wajah Berbasis <i>Linear Discriminant Analysis</i> Dan <i>K-Nearest Neighbor</i> Menggunakan <i>Particle Swarm Optimization</i> (Ramdani <i>et al.</i> , 2022)	Kombinasi <i>LDA</i> , <i>Euclidean Distance</i> , dan <i>PSO</i> meningkatkan akurasi dari 70% ke 71,67% dan mengurangi waktu komputasi sebesar 0,1009 detik.
3.	Face Detection Using <i>Linear Discriminant Analysis (LDA)</i> And Support Vector Machine (SVM) Methods (Hariadi dan Enda, 2022)	<i>LDA</i> dan SVM mencapai akurasi 84,2% pada dataset CCTV, dipengaruhi oleh kualitas gambar dan proses cropping yang konsisten.
4.	A Method of Partial <i>Face Recognition</i> Using <i>LDA</i> and Back Propagation Neural Network (Bairagi <i>et al.</i> , 2020)	<i>LDA</i> + MLP untuk wajah parsial capai akurasi 98,3%; <i>LDA</i> untuk reduksi dimensi, MLP atasi pose tak lengkap.
5.	Penerapan Metode <i>Linear Discriminant Analysis</i> Dalam Mendeteksi Kematangan Buah Tomat (Setiawan, 2025)	CNN + Liveness Detector capai akurasi 100% (wajah terdaftar) & 73,33% (wajah acak), serta mendukung absensi tanpa kontak fisik.
6.	Automated Attendance System using Facial Recognition : A Survey (Baheti, 2024)	HOG + Raspberry Pi dukung absensi real-time, update ke Excel & kirim notifikasi email.

Lanjutan Tabel 2. 5 *State Of The Art*

No	Judul	Hasil Penelitian
7.	Perancangan Sistem Absensi Facial Recognition Menggunakan CNN Dan Liveness Detector Pada BPR Central Dana Mandiri (Khatina Sari <i>et al.</i> , 2022)	<i>LDA</i> klasifikasikan kematangan tomat berbasis HSV dengan akurasi 85%, potensial untuk riset lanjut.
8.	Sistem Absensi Pengenalan Wajah Dengan Menggunakan Pustaka Dlib Dan Metoda K-NN Pada Jaringan LAN (Sugeng dan Mulyana, 2022)	Dlib + K-NN di LAN raih akurasi 98,3%, mendukung absensi cepat & protokol kesehatan.
9.	Pengembangan Sistem Absensi Berbasis <i>Face Recognition</i> Dengan Metode <i>LDA</i> (Tan <i>et al.</i> , 2022)	<i>LDA</i> lebih akurat (95%) dibanding PCA (89,5%) dalam absensi wajah.
10.	Aplikasi <i>E-Visitor</i> dengan <i>Face Recognition</i> untuk Absensi Tamu Menggunakan Metode <i>Linear Discriminant Analysis</i> di Dinas Arsip Provinsi Sumatera Selatan (Alfarizi, 2025)	Aplikasi ini berfungsi untuk absensi tamu otomatis menggunakan <i>Face Recognition</i> , mempercepat pencatatan, mengurangi kontak fisik, serta meningkatkan keamanan dan efisiensi di Dinas Arsip. Berhasil menghasilkan akurasi sebesar 83.3% dari 120 data uji.

2.3 Keterbaruan Penelitian

Berdasarkan beberapa penelitian sebelumnya, metode *Linear Discriminant Analysis (LDA)* telah banyak digunakan untuk sistem pengenalan wajah, baik dengan algoritma pembandingan seperti K-NN, SVM, maupun MLP. Namun, belum ada penelitian yang secara spesifik menerapkan *LDA* dalam sistem absensi tamu berbasis web menggunakan *QR Code* dan kamera internal perangkat secara langsung tanpa instalasi tambahan, seperti yang dilakukan pada Aplikasi *E-Visitor* di Dinas Kearsipan. Keterbaruan dari penelitian ini terletak pada integrasi sistem absensi *web-based* dengan *Face Recognition* berbasis *LDA* yang mudah diakses, praktis, dan relevan untuk instansi pemerintah dalam mendukung digitalisasi layanan publik.

