

BAB II

LANDASAN TEORI

2.1 Kajian Teori

2.1.1 Sistem Informasi

Sistem informasi adalah suatu sistem yang digunakan untuk memproses data menjadi informasi yang bermanfaat bagi sebuah organisasi. Sistem ini terdiri dari berbagai bagian yang saling bekerja sama, seperti manusia, alat fisik, program, jaringan komunikasi, dan data, untuk mengumpulkan, memproses, dan menyebarkan informasi (Zufria et al., 2022).

Sistem informasi berperan penting dalam membantu berbagai kegiatan seperti operasional, manajemen, dan pengambilan keputusan dengan memberikan informasi yang diperlukan oleh pengguna. Sistem informasi melakukan beberapa proses seperti mengumpulkan, mengolah, menyimpan, dan mendistribusikan informasi agar data yang dihasilkan bisa digunakan secara efektif untuk mendukung berbagai kegiatan organisasi. Sistem informasi berperan penting dalam menyediakan informasi yang akurat dan bermanfaat bagi penggunanya (Sarosa, 2021).

Perkembangan teknologi informasi membuat sistem informasi yang dulunya dilakukan secara manual berubah menjadi sistem yang menggunakan komputer, jaringan, dan internet. Perubahan tersebut memberi banyak manfaat, seperti meningkatkan kecepatan kerja, mengurangi kesalahan yang terjadi karena manusia, serta memudahkan dalam pengelolaan dan mengakses data. Selain itu, sistem informasi modern memungkinkan penggabungan data yang lebih luas dan terjadi secara *real-time* berkat teknologi berbasis *web* (Permana et al., 2023).

Dalam bidang pendidikan, sistem informasi berperan penting dalam membantu mengelola data pelajaran serta urusan administrasi sekolah. Sistem informasi akan berbagai data, seperti data siswa, guru, mata pelajaran, jadwal pelajaran, serta informasi akademik lainnya secara terintegrasi. Penggunaan sistem informasi di lingkungan sekolah dapat meningkatkan efisiensi dalam mengelola data,

mempercepat akses terhadap informasi, meningkatkan ketepatan data, serta memudahkan proses pengambilan keputusan bagi pihak yang terlibat (Rifai, 2024).

Dari penjelasan di atas, bisa disimpulkan bahwa sistem informasi berperan penting dalam membantu mengelola data dengan cara yang lebih efektif dan efisien. Oleh karena itu, penelitian ini membuat sistem absensi sidik jari berbasis *Internet of Things* (IoT) yang terhubung dengan sistem informasi manajemen sekolah berbasis *web*, agar data dapat dikelola secara terpusat dan meningkatkan kualitas pelayanan administrasi di MI Panyocokan I.

2.1.2 Sistem Informasi Manajemen Sekolah

Sistem Informasi Manajemen Sekolah adalah sistem yang menggunakan teknologi informasi untuk membantu mengelola data akademik, non-akademik, administrasi, dan kegiatan operasional sekolah secara terpadu. Sistem informasi manajemen sekolah digunakan untuk meningkatkan kemudahan dalam mengelola data dan menyampaikan informasi, sehingga proses administrasi di sekolah bisa berjalan lebih cepat, tepat, dan hemat dibandingkan menggunakan sistem manual (Zufria et al., 2022).

Sistem informasi manajemen sekolah secara umum digunakan untuk menggabungkan berbagai kegiatan belajar mengajar dan urusan administrasi sekolah ke dalam satu sistem yang saling terkait. Sistem ini bisa dipakai untuk mengelola berbagai data seperti data siswa, data guru, proses belajar mengajar, penilaian, kehadiran siswa, serta memberi informasi kepada orang tua. Dengan adanya sistem ini, pengelolaan sekolah dapat menjadi lebih baik dan lebih cepat. Sistem informasi manajemen sekolah juga membantu menyediakan informasi dengan lebih cepat dan memudahkan berbagai pihak dalam mendapatkan informasi yang dibutuhkan (Rahmadi et al., 2021).

Sistem Informasi Manajemen Sekolah membantu meningkatkan kualitas pelayanan bagi semua pihak yang terlibat, seperti kepala sekolah, guru, staf administrasi, siswa, dan orang tua. Kepala sekolah bisa menggunakan sistem ini sebagai bantuan dalam mengambil keputusan, guru lebih mudah dalam mengurus data pelajaran, sedangkan staf administrasi lebih terbantu dalam mengelola data dan

membuat laporan. Selain itu, siswa dan orang tua bisa mendapatkan informasi tentang pelajaran serta urusan administrasi dengan lebih cepat, lebih mudah, dan lebih jelas. Dengan demikian, penggunaan Sistem Informasi Manajemen Sekolah dapat meningkatkan efektivitas layanan administrasi dan mendukung proses penyelenggaraan pendidikan yang lebih efisien (Malik et al., 2024).

Dengan berkembangnya teknologi informasi, sistem informasi manajemen sekolah saat ini semakin banyak dikembangkan berbasis *web*, sehingga bisa diakses dengan lebih fleksibel melalui internet. Pengembangan berbasis *web* merekan kepala sekolah, guru, tenaga administrasi, siswa, dan orang tua mereaksi oleh lokasi dan jenis perangkat yang digunakan. Dengan demikian, sistem informasi manajemen sekolah berbasis *web* dapat meningkatkan kinerja pelayanan administrasi, memperluas akses terhadap informasi, serta membantu pengelolaan sekolah menjadi lebih efisien (Mindani et al., 2023).

Dalam penelitian ini, sistem yang dibuat bertujuan membangun sistem presensi menggunakan sidik jari berbasis *Internet of Things* (IoT) yang terhubung dengan Sistem Informasi Manajemen Sekolah berbasis *web*. Sistem itu meliputi pengelolaan data siswa, guru, kelas, dan proses absensi yang terhubung secara otomatis ke dalam sistem informasi. Dengan adanya integrasi tersebut, diharapkan proses pengelolaan data kehadiran menjadi lebih efektif, akurat, dan dapat dilakukan secara *real-time*, serta mampu mendukung peningkatan kualitas layanan administrasi dan penyampaian informasi mengenai kehadiran siswa kepada pihak sekolah maupun orang tua di MI Panyocokan I.

2.1.3 Website

Website merupakan sekumpulan halaman *web* yang saling terhubung dalam sebuah *domain* atau *subdomain* dan dapat diakses melalui jaringan internet menggunakan *web browser*. Keberadaan *website* memungkinkan informasi disajikan secara digital sehingga pengguna dapat memperoleh maupun mengelola informasi dengan lebih cepat dan efisien. Seiring berkembangnya teknologi informasi, fungsi *website* tidak lagi terbatas sebagai media publikasi informasi, tetapi juga dimanfaatkan sebagai platform yang mendukung berbagai aktivitas

interaktif, termasuk pengelolaan data dan layanan pada institusi pendidikan (Miftahuljannah, 2023).

Berdasarkan karakteristiknya, *website* dapat dikelompokkan menjadi dua jenis, yaitu *website* statis dan *website* dinamis. *Website* statis memiliki konten yang bersifat tetap sehingga setiap perubahan informasi harus dilakukan secara langsung pada kode program. Berbeda dengan *website* statis, *website* dinamis mampu menampilkan informasi yang berubah secara otomatis sesuai dengan data yang tersimpan di dalam basis data maupun kebutuhan pengguna. Kemampuan tersebut menjadikan *website* dinamis lebih banyak diterapkan dalam pengembangan sistem informasi karena lebih fleksibel dalam mengelola dan menyajikan data secara interaktif (Maulana et al., 2023).

Sistem berbasis *web* merupakan sistem informasi yang memanfaatkan *website* sebagai media utama untuk menjalankan berbagai fungsi aplikasi dan dapat diakses melalui *web browser* tanpa memerlukan proses instalasi pada perangkat pengguna. Pendekatan ini memberikan kemudahan dalam mengakses informasi dari berbagai lokasi, mempercepat proses pengelolaan data, serta mempermudah penyampaian informasi kepada pengguna. Di lingkungan pendidikan, sistem berbasis *web* banyak dimanfaatkan untuk mengelola data siswa, guru, jadwal pelajaran, nilai, hingga administrasi sekolah sehingga berbagai aktivitas akademik dan administrasi dapat dilakukan secara lebih efektif dan terorganisir (Setiawan et al., 2023).

Perkembangan teknologi *web* juga melahirkan konsep *Responsive Web Design* (RWD), yaitu pendekatan pengembangan antarmuka yang memungkinkan tampilan *website* menyesuaikan secara otomatis dengan ukuran layar perangkat yang digunakan. Melalui konsep ini, pengguna tetap memperoleh pengalaman yang nyaman ketika mengakses *website* menggunakan komputer, tablet, maupun *smartphone* tanpa kehilangan fungsi ataupun kualitas tampilan. Oleh karena itu, penerapan *Responsive Web Design* menjadi salah satu aspek penting dalam pengembangan *website* modern karena mampu meningkatkan aksesibilitas, kenyamanan penggunaan, serta kualitas penyajian informasi pada berbagai perangkat (Susanto & Andriana, 2020).

Berdasarkan uraian tersebut, dapat dipahami bahwa sistem berbasis *web* menjadi salah satu teknologi yang mampu mendukung pengembangan sistem informasi secara efektif dan terintegrasi. Dalam penelitian ini, teknologi *website* dipilih sebagai media utama untuk mengelola data sekolah secara terpusat, mempermudah akses informasi, serta mendukung integrasi dengan sistem presensi *fingerprint* berbasis *Internet of Things (IoT)*. Dengan demikian, data kehadiran dapat diproses, disimpan, dan ditampilkan secara *real-time* melalui Sistem Informasi Manajemen Sekolah.

2.1.4 Teknologi Website

Teknologi *website* merupakan kumpulan bahasa pemrograman dan teknologi pendukung yang digunakan dalam proses pembangunan serta pengembangan sebuah *website*. Pemilihan teknologi yang sesuai menjadi salah satu faktor penting agar *website* mampu berfungsi dengan baik, memiliki tampilan yang menarik, serta memberikan pengalaman penggunaan yang optimal. Dalam pengembangan sistem informasi berbasis *web*, beberapa teknologi yang umum digunakan antara lain HTML, CSS, JavaScript, dan PHP. Masing-masing teknologi memiliki fungsi yang berbeda, namun saling mendukung dalam membentuk struktur halaman, mengatur tampilan, menciptakan interaksi, serta menjalankan proses pengolahan data pada sistem berbasis *web* (Setiawan et al., 2023).

2.1.4.1 HTML (HyperText Markup Language)

HTML (*HyperText Markup Language*) merupakan bahasa *markup* yang digunakan sebagai dasar dalam membangun struktur sebuah halaman *website*. Melalui HTML, berbagai elemen seperti judul, paragraf, gambar, tabel, tautan, maupun formulir dapat disusun sehingga dapat dikenali dan ditampilkan oleh *web browser*. Dalam proses pengembangan *website*, HTML menjadi fondasi utama yang kemudian dipadukan dengan CSS untuk memperindah tampilan serta *JavaScript* untuk menambahkan berbagai fitur interaktif pada halaman *web* (Sholikhan, 2022).

2.1.4.2 CSS (*Cascading Style Sheets*)

CSS (*Cascading Style Sheets*) merupakan bahasa *stylesheet* yang berfungsi mengatur tampilan visual dari halaman *website* yang dibangun menggunakan HTML. Penggunaan CSS memungkinkan pengembang mengelola berbagai aspek antarmuka, seperti warna, jenis huruf, ukuran teks, jarak antar elemen, hingga tata letak halaman. Dengan memisahkan pengaturan tampilan dari struktur halaman, proses pengembangan maupun pemeliharaan *website* menjadi lebih mudah, sekaligus menghasilkan antarmuka yang lebih rapi, konsisten, dan nyaman digunakan (Sholikhan, 2022).

2.1.4.3 JavaScript

JavaScript merupakan bahasa pemrograman yang berperan dalam menciptakan interaksi pada halaman *website*. Melalui JavaScript, halaman *web* dapat memberikan respons secara langsung terhadap tindakan pengguna, misalnya melakukan validasi data pada formulir, menampilkan notifikasi, mengelola berbagai *event*, maupun memperbarui isi halaman tanpa perlu memuat ulang keseluruhan halaman. Integrasi JavaScript dengan HTML dan CSS menghasilkan *website* yang lebih dinamis serta mampu meningkatkan kenyamanan pengguna ketika berinteraksi dengan sistem (Sholikhan, 2022).

2.1.4.4 PHP (*Hypertext Preprocessor*)

PHP (*Hypertext Preprocessor*) merupakan bahasa pemrograman yang berjalan pada sisi server (*server-side scripting*) dan banyak digunakan dalam pengembangan *website* dinamis. Seluruh proses pengolahan data dilakukan di server, kemudian hasilnya dikirimkan ke *web browser* dalam bentuk halaman *web* yang siap ditampilkan kepada pengguna. Selain dapat diintegrasikan dengan HTML, PHP juga mendukung koneksi dengan berbagai sistem manajemen basis data, termasuk MySQL, sehingga banyak dimanfaatkan dalam pembangunan sistem informasi berbasis *web*. Pada penelitian ini, PHP digunakan sebagai bahasa pemrograman utama untuk mengembangkan Sistem Informasi Manajemen Sekolah, termasuk

dalam mengelola proses bisnis aplikasi, pengolahan data, serta komunikasi dengan basis data (Endra et al., 2021).

Berdasarkan uraian tersebut, dapat dipahami bahwa HTML, CSS, JavaScript, dan PHP merupakan teknologi yang saling melengkapi dalam pengembangan *website*. HTML berfungsi sebagai penyusun struktur halaman, CSS mengatur tampilan antarmuka, JavaScript memberikan kemampuan interaktif, sedangkan PHP menangani proses pengolahan data pada sisi server. Penerapan keempat teknologi tersebut secara terpadu pada penelitian ini mendukung pembangunan Sistem Informasi Manajemen Sekolah berbasis *web* yang terintegrasi dengan sistem presensi *fingerprint* berbasis *Internet of Things* (IoT), sehingga proses pengelolaan data serta penyampaian informasi dapat dilakukan secara *real-time*.

2.1.5 Database dan MySQL

Database menjadi salah satu komponen utama dalam sistem informasi karena berfungsi sebagai tempat penyimpanan data yang disusun secara terstruktur sehingga mudah dikelola dan diakses kembali ketika diperlukan. Seluruh data yang tersimpan di dalam *database* memiliki hubungan satu sama lain sehingga proses pencarian, pembaruan, maupun pengolahan informasi dapat dilakukan secara lebih cepat dan akurat. Selain mendukung efisiensi pengelolaan data, penggunaan *database* juga membantu menjaga konsistensi informasi serta mengurangi kemungkinan terjadinya kehilangan maupun duplikasi data (Zufria et al., 2022).

Pada lingkungan sekolah, *database* dimanfaatkan untuk mengelola berbagai informasi akademik dan administrasi, seperti data siswa, data guru, data kelas, jadwal pelajaran, nilai, hingga data presensi. Penyimpanan seluruh informasi dalam satu basis data terpusat memberikan kemudahan dalam melakukan pencarian, pembaruan, maupun penyusunan laporan secara sistematis. Oleh karena itu, keberadaan *database* menjadi bagian yang sangat penting dalam mendukung operasional Sistem Informasi Manajemen Sekolah berbasis *web* agar proses pengelolaan data dapat berlangsung secara efektif dan efisien (Zulfa, 2022).

2.1.5.1 MySQL

MySQL merupakan salah satu *Relational Database Management System* (RDBMS) yang digunakan untuk mengelola basis data dengan memanfaatkan *Structured Query Language* (SQL) sebagai bahasa utama dalam pengolahan data. Melalui SQL, pengguna dapat melakukan berbagai operasi, seperti menambahkan, memperbarui, menghapus, maupun menampilkan data sesuai kebutuhan sistem. Sebagai perangkat lunak yang bersifat *open source*, MySQL banyak dipilih dalam pengembangan aplikasi berbasis web karena memiliki performa yang baik, mudah digunakan, serta dapat diintegrasikan dengan berbagai bahasa pemrograman, salah satunya PHP (Siregar, 2021).

2.1.5.2 Kelebihan MySQL

Popularitas MySQL dalam pengembangan sistem informasi tidak terlepas dari berbagai keunggulan yang dimilikinya. Selain tersedia secara *open source*, MySQL mampu menangani pengelolaan data secara efisien serta mendukung akses oleh banyak pengguna (*multi-user*) dalam waktu yang bersamaan. Penggunaan SQL sebagai bahasa standar juga mempermudah proses pengelolaan basis data. Di samping itu, kompatibilitas MySQL dengan PHP menjadikannya sebagai salah satu kombinasi teknologi yang banyak digunakan dalam pembangunan aplikasi berbasis *web* karena mampu mendukung proses pengembangan yang lebih fleksibel dan mudah dipelihara (Siregar, 2021).

2.1.5.3 Kekurangan MySQL

Di samping berbagai kelebihannya, MySQL juga memiliki beberapa keterbatasan yang perlu diperhatikan dalam proses pengembangan sistem. Pada implementasi yang melibatkan basis data berukuran sangat besar dengan tingkat kompleksitas yang tinggi, diperlukan proses optimasi agar performa sistem tetap terjaga. Selain itu, beberapa fitur lanjutan yang tersedia pada sistem manajemen basis data lain belum sepenuhnya dimiliki oleh MySQL. Oleh sebab itu, pemilihan MySQL sebaiknya mempertimbangkan kebutuhan, karakteristik, serta skala sistem

yang akan dibangun agar dapat memberikan kinerja yang optimal (Endra et al., 2021).

Dari pembahasan yang telah dipaparkan, dapat dipahami bahwa *database* dan MySQL memiliki peran yang saling berkaitan dalam mendukung pembangunan sistem informasi berbasis *web*. *Database* berfungsi sebagai media penyimpanan dan pengelolaan data, sedangkan MySQL berperan sebagai sistem manajemen basis data yang mengatur proses penyimpanan, pengolahan, dan pengambilan informasi. Pada penelitian ini, MySQL dipilih karena mampu terintegrasi dengan baik menggunakan PHP serta mendukung kebutuhan pengelolaan data pada Sistem Informasi Manajemen Sekolah berbasis *web* yang terintegrasi dengan sistem presensi *fingerprint* berbasis *Internet of Things* (IoT).

2.1.6 Sistem Presensi

Presensi merupakan salah satu aktivitas administrasi yang bertujuan untuk mencatat kehadiran seseorang dalam suatu organisasi atau institusi. Pada lingkungan pendidikan, sistem presensi digunakan untuk mendokumentasikan kehadiran siswa, guru, maupun tenaga kependidikan sebagai bagian dari pengelolaan administrasi sekolah. Informasi kehadiran yang tersimpan tidak hanya berfungsi sebagai arsip, tetapi juga menjadi dasar dalam melakukan pemantauan kedisiplinan, evaluasi tingkat kehadiran, serta penyusunan laporan secara berkala. Dengan adanya sistem presensi, proses pengelolaan data kehadiran dapat dilakukan secara lebih teratur dan sistematis (Fadhilah, 2024).

Perkembangan teknologi telah mengubah cara pencatatan kehadiran yang sebelumnya dilakukan secara manual menggunakan buku absensi menjadi sistem yang memanfaatkan teknologi digital. Metode manual sering kali memerlukan waktu yang lebih lama, rentan terhadap kesalahan pencatatan, serta memungkinkan terjadinya manipulasi data kehadiran. Untuk mengatasi berbagai keterbatasan tersebut, banyak institusi mulai menerapkan sistem presensi digital dengan memanfaatkan teknologi seperti RFID, barcode, QR code, *fingerprint*, maupun aplikasi berbasis *mobile*. Pemanfaatan teknologi tersebut mampu meningkatkan akurasi pencatatan, mempercepat proses presensi, serta memudahkan pemantauan

data kehadiran secara *real-time*, sehingga pengelolaan presensi menjadi lebih efektif dan efisien (Kurniawan et al., 2022).

2.1.6.1 Tujuan Sistem Presensi

Penerapan sistem presensi memiliki tujuan utama untuk menghasilkan data kehadiran yang akurat, terdokumentasi dengan baik, dan mudah dikelola sebagai bagian dari proses administrasi. Selain berfungsi sebagai media pencatatan, sistem presensi juga mendukung peningkatan kedisiplinan pengguna, mempermudah proses pemantauan kehadiran baik secara berkala maupun *real-time*, serta membantu penyusunan laporan sebagai bahan pertimbangan dalam pengambilan keputusan administratif. Dengan memanfaatkan sistem presensi digital, proses pencatatan kehadiran dapat dilakukan secara lebih cepat, efisien, dan mampu mengurangi kesalahan yang sering dijumpai pada metode presensi manual (Fitriani et al., 2021).

2.1.6.2 Kelebihan Sistem Presensi Digital

Penerapan sistem presensi digital memberikan berbagai keuntungan dibandingkan dengan metode presensi manual. Proses pencatatan kehadiran dapat dilakukan secara otomatis sehingga waktu yang dibutuhkan menjadi lebih singkat dan tingkat akurasi data meningkat. Seluruh informasi kehadiran juga tersimpan dalam basis data sehingga lebih mudah dikelola, dicari kembali, dan memiliki tingkat keamanan yang lebih baik. Selain itu, sistem presensi digital mendukung pembuatan laporan secara otomatis serta memungkinkan pihak sekolah melakukan pemantauan kehadiran secara *real-time*, sehingga informasi yang diperoleh menjadi lebih cepat, tepat, dan mendukung pengelolaan administrasi secara efisien (Permana et al., 2023).

2.1.6.3 Kekurangan Sistem Presensi Digital

Di balik berbagai keunggulan yang dimiliki, sistem presensi digital tetap mempunyai beberapa keterbatasan dalam penerapannya. Kinerja sistem sangat dipengaruhi oleh ketersediaan perangkat keras dan kualitas jaringan yang digunakan, sehingga gangguan teknis dapat menghambat proses pencatatan

kehadiran. Selain itu, implementasi sistem ini memerlukan biaya awal untuk penyediaan perangkat, perangkat lunak, serta infrastruktur pendukung. Pengguna juga perlu memahami cara pengoperasian sistem agar dapat dimanfaatkan secara optimal. Aspek keamanan data menjadi hal yang tidak kalah penting karena informasi kehadiran yang tersimpan secara digital berpotensi menghadapi risiko akses tanpa izin maupun kebocoran data apabila tidak didukung mekanisme pengamanan yang memadai (Permana et al., 2023).

Uraian yang telah dipaparkan menunjukkan bahwa sistem presensi memiliki peranan penting dalam mendukung pengelolaan data kehadiran secara lebih akurat, efektif, dan efisien. Perkembangan menuju sistem presensi digital memberikan kemudahan dalam proses pencatatan, pemantauan kehadiran, serta penyusunan laporan secara otomatis. Pada penelitian ini, sistem presensi yang dikembangkan memanfaatkan teknologi *fingerprint* yang terintegrasi dengan *Internet of Things* (IoT) dan Sistem Informasi Manajemen Sekolah berbasis *web*, sehingga data kehadiran dapat diproses, disimpan, dan disajikan secara *real-time*.

2.1.7 Fingerprint

Fingerprint merupakan teknologi biometrik yang memanfaatkan karakteristik sidik jari sebagai media untuk mengenali dan memverifikasi identitas seseorang. Setiap individu memiliki pola sidik jari yang berbeda serta bersifat permanen, sehingga teknologi ini banyak dimanfaatkan sebagai metode autentikasi yang memiliki tingkat akurasi tinggi. Dalam proses identifikasi, sensor *fingerprint* akan membaca pola sidik jari pengguna, kemudian membandingkannya dengan data referensi yang telah tersimpan di dalam basis data untuk menentukan kesesuaian identitas pengguna (Putriana et al., 2025).

Dibandingkan dengan metode presensi manual maupun penggunaan kartu identitas, teknologi *fingerprint* memberikan tingkat keamanan yang lebih baik karena sidik jari tidak dapat dipindahtanggankan kepada orang lain. Kondisi tersebut menjadikan proses autentikasi lebih terpercaya serta mampu mengurangi kemungkinan terjadinya penyalahgunaan, seperti praktik titip absen. Selain itu,

proses verifikasi dapat dilakukan dalam waktu yang relatif singkat sehingga pencatatan kehadiran menjadi lebih cepat, akurat, dan efisien (Sari et al., 2025).

Pemanfaatan *fingerprint* pada lingkungan pendidikan terus berkembang seiring dengan penerapan sistem informasi berbasis *web*. Perangkat *fingerprint* kini banyak digunakan sebagai media pencatatan kehadiran siswa maupun guru yang terhubung secara langsung dengan sistem informasi sekolah. Integrasi tersebut memungkinkan data kehadiran tersimpan secara otomatis ke dalam basis data, mudah dipantau oleh pihak sekolah, serta mendukung proses administrasi secara lebih efektif dan terorganisir (Andrean et al., 2025).

2.1.7.1 Cara Kerja Fingerprint

Proses kerja sistem *fingerprint* diawali dengan tahap *enrollment*, yaitu proses perekaman sidik jari pengguna sebagai data referensi yang akan disimpan di dalam sistem. Setelah proses pendaftaran selesai, setiap kali pengguna melakukan presensi, sensor akan memindai pola sidik jari untuk memperoleh karakteristik biometrik yang selanjutnya dibandingkan dengan data yang telah tersimpan. Apabila hasil pencocokan menunjukkan kesesuaian, sistem akan memverifikasi identitas pengguna dan secara otomatis mencatat data kehadiran. Informasi tersebut kemudian disimpan ke dalam basis data serta dapat ditampilkan melalui sistem informasi berbasis *web* sehingga proses pengelolaan maupun pemantauan kehadiran dapat dilakukan secara *real-time* (Sumijan, 2021).

2.1.7.2 Kelebihan Fingerprint

Teknologi *fingerprint* memiliki sejumlah keunggulan yang menjadikannya banyak diterapkan pada sistem presensi modern. Pola sidik jari yang unik pada setiap individu memberikan tingkat keamanan yang tinggi karena sulit dipalsukan maupun digunakan oleh orang lain. Kondisi tersebut mampu meminimalkan terjadinya penyalahgunaan dalam proses presensi, termasuk praktik titip absen. Selain itu, proses identifikasi berlangsung secara cepat sehingga waktu pencatatan kehadiran menjadi lebih efisien. Data yang dihasilkan juga memiliki tingkat akurasi yang tinggi karena seluruh proses dilakukan secara otomatis oleh sistem.

Kemampuan *fingerprint* untuk diintegrasikan dengan basis data dan sistem informasi berbasis *web* semakin mempermudah penyimpanan, pengelolaan, serta penyajian informasi kehadiran secara terpusat (Sari et al., 2025).

2.1.7.3 Kekurangan *Fingerprint*

Walaupun menawarkan berbagai keunggulan, penerapan teknologi *fingerprint* juga memiliki beberapa keterbatasan. Proses pembacaan sidik jari dapat terganggu apabila kondisi jari pengguna sedang basah, kotor, atau mengalami luka sehingga sensor kesulitan mengenali pola sidik jari secara optimal. Di samping itu, penggunaan *fingerprint* memerlukan perangkat keras khusus yang membutuhkan biaya pengadaan dan perawatan secara berkala agar tetap berfungsi dengan baik. Sistem ini juga bergantung pada pasokan listrik dan koneksi jaringan yang stabil, sehingga apabila terjadi gangguan pada salah satu komponen tersebut, proses pencatatan maupun pengiriman data kehadiran dapat mengalami kendala (Sari et al., 2025).

2.1.7.4 Sensor *Fingerprint* AS608

Sensor *Fingerprint* AS608 merupakan modul biometrik optik yang digunakan untuk melakukan proses perekaman, penyimpanan, dan verifikasi sidik jari sebagai media autentikasi pengguna. Sensor ini bekerja dengan menangkap citra sidik jari, kemudian mengekstraksi karakteristik unik menjadi *template* digital yang disimpan di memori internal untuk digunakan sebagai data pembandingan pada proses identifikasi berikutnya. AS608 mendukung komunikasi serial (UART), sehingga mudah diintegrasikan dengan berbagai mikrokontroler, seperti *Arduino* maupun ESP32, dalam pengembangan sistem berbasis *Internet of Things* (IoT). Selain memiliki waktu identifikasi yang relatif cepat dan tingkat akurasi yang baik, sensor ini juga mampu menyimpan sejumlah *template* sidik jari sehingga banyak dimanfaatkan pada sistem keamanan dan sistem presensi berbasis biometrik. Pada penelitian ini, sensor *Fingerprint* AS608 digunakan sebagai perangkat utama untuk melakukan autentikasi sidik jari siswa dan guru sebelum data kehadiran dikirimkan ke Sistem Informasi Manajemen Sekolah berbasis *web* melalui ESP32 (Zuhair, A et al., 2025).

Dari pembahasan yang telah dijelaskan, dapat dipahami bahwa *fingerprint* merupakan teknologi autentikasi biometrik yang mampu memberikan tingkat keamanan dan akurasi yang tinggi dalam proses identifikasi pengguna. Penerapan teknologi ini tidak hanya meningkatkan efektivitas proses presensi, tetapi juga membantu mengurangi potensi penyalahgunaan serta menghasilkan data kehadiran yang lebih akurat dibandingkan metode konvensional. Pada penelitian ini, *fingerprint* dimanfaatkan sebagai media autentikasi yang diintegrasikan dengan teknologi *Internet of Things* (IoT) dan Sistem Informasi Manajemen Sekolah berbasis web, sehingga data kehadiran dapat dikirim, disimpan, serta dipantau secara *real-time*.

2.1.8 *Internet of Things* (IoT)

Internet of Things (IoT) merupakan konsep yang memungkinkan berbagai perangkat fisik saling terhubung melalui jaringan internet sehingga mampu berkomunikasi, bertukar informasi, dan menjalankan fungsi tertentu secara otomatis. Pada implementasinya, perangkat IoT umumnya dilengkapi dengan sensor, mikrokontroler, serta media komunikasi yang berperan dalam mengumpulkan, mengolah, dan mengirimkan data secara *real-time*. Perkembangan teknologi ini telah mendorong pemanfaatannya di berbagai bidang, seperti industri, kesehatan, pertanian, smart home, hingga pendidikan karena mampu meningkatkan efisiensi, efektivitas, serta kemudahan dalam proses pemantauan maupun pengendalian sistem (Syahfitri, 2025).

Mekanisme kerja *Internet of Things* diawali dengan proses pengambilan data oleh sensor yang terpasang pada perangkat. Data tersebut selanjutnya diproses oleh mikrokontroler sebelum dikirim melalui media komunikasi, seperti Wi-Fi, *Bluetooth*, atau jaringan seluler menuju server maupun sistem informasi. Setelah diterima, data disimpan ke dalam basis data dan dapat ditampilkan kepada pengguna melalui aplikasi atau *website*. Proses tersebut memungkinkan pengguna memperoleh informasi secara *real-time* serta melakukan pemantauan maupun pengendalian sistem dari berbagai lokasi yang terhubung dengan jaringan internet (Pratama et al., 2022).

2.1.8.1 Kelebihan Internet of Things

Pemanfaatan teknologi *Internet of Things* memberikan berbagai manfaat dalam mendukung pengelolaan sistem yang lebih modern. Salah satu keunggulannya adalah kemampuan melakukan pemantauan secara *real-time*, sehingga kondisi perangkat maupun informasi yang dihasilkan dapat diketahui dengan cepat dan akurat. Selain itu, proses pengumpulan serta pengolahan data berlangsung secara otomatis sehingga mampu meningkatkan efisiensi kerja sekaligus mengurangi ketergantungan pada proses manual. Otomatisasi tersebut juga membantu meminimalkan kesalahan yang disebabkan oleh faktor manusia dalam proses pencatatan maupun pengolahan data. Keunggulan lainnya adalah pengguna dapat mengakses informasi serta mengendalikan perangkat dari lokasi yang berbeda melalui jaringan internet, sehingga pengelolaan sistem menjadi lebih fleksibel (Pratama et al., 2022).

2.1.8.2 Kekurangan Internet of Things

Di samping berbagai kelebihannya, penerapan teknologi *Internet of Things* juga memiliki sejumlah tantangan yang perlu diperhatikan. Salah satunya adalah ketergantungan terhadap koneksi internet yang stabil, karena gangguan jaringan dapat memengaruhi proses komunikasi antarperangkat maupun pengiriman data. Selain itu, implementasi IoT memerlukan perangkat keras serta infrastruktur pendukung yang dapat meningkatkan biaya pembangunan sistem. Dari aspek keamanan, pertukaran data melalui jaringan internet juga berpotensi menimbulkan risiko terhadap privasi maupun akses oleh pihak yang tidak berwenang, sehingga diperlukan mekanisme pengamanan yang memadai. Agar sistem tetap berjalan dengan optimal, perangkat IoT juga membutuhkan pemeliharaan secara berkala, baik pada sisi perangkat keras maupun perangkat lunaknya (Pratama et al., 2022).

Berdasarkan pembahasan di atas, dapat dipahami bahwa *Internet of Things* merupakan teknologi yang memungkinkan berbagai perangkat saling terhubung untuk melakukan pertukaran data dan menjalankan proses secara otomatis melalui jaringan internet. Kemampuan tersebut menjadikan IoT sebagai salah satu teknologi yang efektif dalam mendukung sistem pemantauan dan pengelolaan

informasi secara *real-time*. Pada penelitian ini, IoT berperan sebagai media komunikasi antara perangkat ESP32 dan sensor *fingerprint* dengan Sistem Informasi Manajemen Sekolah berbasis *web*, sehingga data hasil presensi dapat dikirim, disimpan, dan ditampilkan secara otomatis dalam basis data.

2.1.9 Mikrokontroler ESP32

ESP32 merupakan mikrokontroler berbasis *System on Chip* (SoC) yang dikembangkan oleh *Espressif Systems* dan dirancang untuk memenuhi kebutuhan pengembangan perangkat berbasis *Internet of Things* (IoT). Mikrokontroler ini mengintegrasikan prosesor berperforma tinggi, konektivitas Wi-Fi dan *Bluetooth*, serta berbagai pin *General Purpose Input Output* (GPIO) dalam satu perangkat. Kombinasi tersebut memungkinkan ESP32 berkomunikasi dengan berbagai sensor maupun aktuator tanpa memerlukan modul komunikasi tambahan. Selain mampu mengolah data dari perangkat masukan, ESP32 juga dapat mengirimkan informasi ke server melalui jaringan internet sehingga data dapat disimpan, diproses, dan ditampilkan secara *real-time*. Kemampuan tersebut menjadikan ESP32 banyak diterapkan pada sistem pemantauan, otomasi, serta pengendalian berbasis IoT karena menawarkan komunikasi data yang cepat, efisien, dan andal (Nizam et al., 2022).

Dalam implementasi sistem IoT, ESP32 berfungsi sebagai pusat pengendali yang menghubungkan perangkat keras dengan sistem informasi berbasis *web*. Data yang diperoleh dari sensor terlebih dahulu diproses oleh ESP32 sebelum dikirimkan melalui jaringan internet menuju server atau basis data. Selanjutnya, data tersebut dapat diakses dan ditampilkan melalui aplikasi berbasis *web* sehingga pengguna dapat melakukan pemantauan secara *real-time*. Pada penelitian ini, ESP32 dimanfaatkan sebagai mikrokontroler utama yang mengintegrasikan sensor *fingerprint* AS608 dengan Sistem Informasi Manajemen Sekolah berbasis *web*. Melalui integrasi tersebut, hasil autentikasi sidik jari dapat dikirimkan secara otomatis ke server untuk mendukung proses pencatatan dan pengelolaan data presensi (Nizam et al., 2022).

2.1.9.1 Kelebihan ESP32

ESP32 memiliki berbagai keunggulan yang menjadikannya salah satu mikrokontroler paling banyak digunakan dalam pengembangan perangkat IoT. Konektivitas Wi-Fi dan *Bluetooth* yang telah terintegrasi memungkinkan komunikasi data dilakukan tanpa memerlukan perangkat tambahan. Selain itu, ESP32 dibekali prosesor *dual-core* yang mampu menangani berbagai proses dengan kinerja yang baik, serta menyediakan banyak pin GPIO yang mendukung integrasi dengan beragam sensor dan perangkat elektronik lainnya. Mikrokontroler ini juga dilengkapi fitur *deep sleep* yang membantu mengurangi konsumsi daya ketika perangkat tidak sedang melakukan proses aktif. Kombinasi antara performa, fleksibilitas, serta biaya implementasi yang relatif terjangkau menjadikan ESP32 sebagai pilihan yang sesuai untuk kebutuhan prototipe maupun pengembangan sistem IoT (Nizam et al., 2022).

2.1.9.2 Kekurangan ESP32

Di samping berbagai keunggulannya, penggunaan ESP32 juga memiliki beberapa keterbatasan yang perlu diperhatikan. Proses pengembangan menggunakan ESP32 membutuhkan pemahaman mengenai pemrograman mikrokontroler, konfigurasi jaringan, serta pengelolaan berbagai *library*, sehingga implementasinya relatif lebih kompleks dibandingkan perangkat yang lebih sederhana. Selain itu, ESP32 memerlukan catu daya yang stabil karena perubahan tegangan dapat menyebabkan perangkat mengalami *restart* maupun gangguan koneksi. Dari sisi perangkat lunak, kesalahan dalam penulisan program atau pengelolaan memori dapat memengaruhi performa sistem. Kinerja komunikasi ESP32 juga sangat dipengaruhi oleh kualitas jaringan Wi-Fi yang digunakan, sehingga koneksi internet yang tidak stabil dapat menghambat proses pertukaran data dengan server (Nizam et al., 2022).

Berdasarkan pembahasan tersebut, ESP32 dapat dipahami sebagai mikrokontroler yang memiliki kemampuan pemrosesan dan komunikasi yang sangat mendukung pengembangan sistem berbasis *Internet of Things* (IoT). Berbagai fitur yang dimilikinya memungkinkan ESP32 mengintegrasikan sensor,

jaringan internet, serta sistem informasi dalam satu kesatuan yang saling terhubung. Pada penelitian ini, ESP32 berperan sebagai pengendali utama yang menghubungkan sensor *fingerprint* AS608 dengan Sistem Informasi Manajemen Sekolah berbasis *web*, sehingga proses pengiriman, penyimpanan, dan penyajian data presensi dapat dilakukan secara otomatis dan *real-time*.

2.1.10 Email (*Electronic Mail*)

Email (*Electronic Mail*) merupakan layanan komunikasi elektronik yang memungkinkan proses pengiriman dan penerimaan pesan melalui jaringan internet. Dalam perkembangan sistem informasi, *email* tidak hanya digunakan sebagai sarana komunikasi antar pengguna, tetapi juga dimanfaatkan sebagai media penyampaian informasi secara otomatis yang dihasilkan oleh sistem. Pemanfaatan *email* memberikan berbagai keuntungan, seperti proses pengiriman informasi yang cepat, dokumentasi pesan yang tersimpan dengan baik, serta kemudahan bagi pengguna untuk menerima pemberitahuan tanpa harus selalu membuka aplikasi. Oleh karena itu, email banyak diterapkan pada berbagai sistem informasi, termasuk di bidang pendidikan, sebagai media penyampaian informasi yang efektif dan efisien (Syahputra & Putri, 2025).

Pada sistem informasi berbasis *web*, *email* umumnya diintegrasikan sebagai fitur notifikasi otomatis untuk menyampaikan informasi kepada pengguna ketika terjadi suatu aktivitas tertentu di dalam sistem. Dengan mekanisme tersebut, sistem dapat mengirimkan pemberitahuan secara langsung tanpa memerlukan tindakan tambahan dari pengguna. Penerapan notifikasi *email* membantu mempercepat penyampaian informasi, meningkatkan efektivitas komunikasi antara sistem dengan pengguna, serta mendukung proses pemantauan berbagai aktivitas secara lebih efisien. Dalam penelitian ini, email dimanfaatkan sebagai media penyampaian informasi kehadiran siswa kepada orang tua setelah proses presensi *fingerprint* berhasil dilakukan sehingga informasi dapat diterima secara *real-time* (Syahputra & Putri, 2025).

2.1.10.1 SMTP (Simple Mail Transfer Protocol)

Simple Mail Transfer Protocol (SMTP) merupakan protokol standar yang digunakan untuk mengirimkan pesan *email* melalui jaringan internet. Protokol ini berfungsi sebagai perantara antara aplikasi dengan server *email* sehingga proses pengiriman pesan dapat dilakukan secara otomatis menuju alamat tujuan. Dalam pengembangan sistem informasi berbasis *web*, SMTP banyak dimanfaatkan untuk mendukung layanan notifikasi otomatis karena mampu mengirimkan informasi secara cepat, terstruktur, dan andal. Pada penelitian ini, SMTP digunakan sebagai media komunikasi antara Sistem Informasi Manajemen Sekolah dengan layanan *Gmail* sehingga notifikasi kehadiran dapat dikirimkan secara otomatis kepada orang tua setelah proses presensi berhasil dilakukan (Harbani, 2022).

2.1.10.2 PHPMailer

PHPMailer merupakan *library* yang dikembangkan untuk bahasa pemrograman PHP dengan tujuan mempermudah implementasi pengiriman *email* pada aplikasi berbasis *web*. *Library* ini mendukung penggunaan protokol *Simple Mail Transfer Protocol* (SMTP) serta menyediakan berbagai fitur, seperti autentikasi pengguna, enkripsi koneksi, pengiriman berkas lampiran (*attachment*), dan penyusunan isi pesan dalam format teks maupun HTML. Kehadiran fitur-fitur tersebut menjadikan proses pengiriman *email* lebih praktis, aman, dan stabil dibandingkan penggunaan fungsi *email* bawaan PHP. Oleh karena itu, PHPMailer banyak dimanfaatkan dalam pengembangan berbagai sistem informasi yang memerlukan pengiriman notifikasi secara otomatis kepada pengguna. Pada penelitian ini, PHPMailer berfungsi sebagai media integrasi antara Sistem Informasi Manajemen Sekolah dengan layanan *Gmail* SMTP sehingga informasi kehadiran siswa dapat dikirimkan secara otomatis kepada orang tua setelah proses presensi *fingerprint* berhasil dilakukan (Pahrizal et al., 2022).

2.1.10.3 Notifikasi Email

Notifikasi *email* merupakan fitur yang memungkinkan sistem mengirimkan informasi secara otomatis kepada pengguna melalui layanan surat elektronik ketika

terjadi suatu peristiwa tertentu. Mekanisme ini membantu pengguna memperoleh informasi secara cepat tanpa harus masuk atau mengakses sistem terlebih dahulu. Pada penelitian ini, fitur notifikasi *email* digunakan untuk mengirimkan informasi kehadiran siswa kepada orang tua setelah data presensi berhasil diverifikasi menggunakan *fingerprint* dan tersimpan di dalam basis data. Selanjutnya, sistem akan memproses data tersebut dan mengirimkannya melalui layanan *Gmail* menggunakan protokol SMTP, sehingga informasi kehadiran dapat diterima secara cepat, terdokumentasi dengan baik, serta mendukung proses monitoring kehadiran secara *real-time* (Syahputra & Putri, 2025).

Dari pembahasan tersebut dapat dipahami bahwa *email* merupakan salah satu media komunikasi yang efektif untuk mendukung penyampaian informasi secara otomatis dalam sistem informasi berbasis *web*. Dukungan protokol SMTP memungkinkan proses pengiriman pesan dilakukan secara cepat, terstruktur, dan andal. Pada penelitian ini, teknologi email dimanfaatkan sebagai media notifikasi kehadiran yang terintegrasi dengan Sistem Informasi Manajemen Sekolah berbasis *web*, sehingga orang tua dapat memperoleh informasi presensi siswa secara otomatis setelah proses autentikasi *fingerprint* berhasil dilakukan.

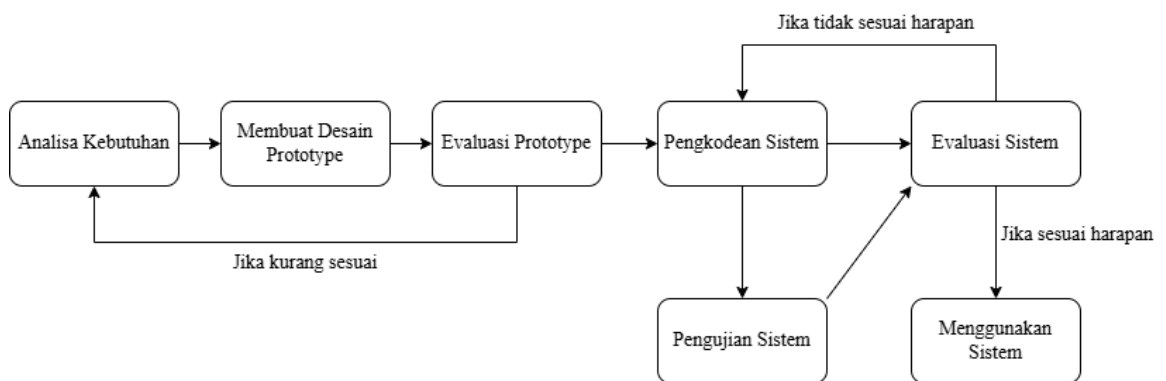
2.1.11 Prototype

Metode *Prototype* merupakan salah satu pendekatan dalam pengembangan perangkat lunak yang dilakukan dengan membangun sebuah model awal sebagai gambaran dari sistem yang akan dikembangkan. Model tersebut berfungsi untuk memperlihatkan rancangan antarmuka, alur proses, serta fungsi utama sistem sehingga pengguna dapat memahami bentuk awal sistem sebelum proses pengembangan dilakukan secara menyeluruh. Melalui *prototype* yang telah dibuat, pengguna dapat memberikan masukan maupun saran perbaikan sehingga pengembang lebih mudah mengidentifikasi kebutuhan, menemukan kekurangan, dan melakukan penyempurnaan secara bertahap hingga sistem yang dihasilkan sesuai dengan kebutuhan pengguna (Agustana, 2025).

Metode *Prototype* umumnya diterapkan pada pengembangan sistem yang kebutuhan penggunanya belum dapat didefinisikan secara lengkap pada tahap awal.

Dengan adanya model awal sistem, pengguna dapat melakukan evaluasi secara langsung terhadap tampilan maupun fungsi yang telah dirancang. Hasil evaluasi tersebut kemudian digunakan sebagai dasar untuk melakukan perbaikan secara berulang (*iterative*) hingga diperoleh sistem yang sesuai dengan kebutuhan serta harapan pengguna. Pendekatan ini memungkinkan komunikasi antara pengembang dan pengguna berlangsung lebih efektif sehingga risiko kesalahan dalam pengembangan sistem dapat diminimalkan (Agustana, 2025).

2.1.11.1 Tahapan Metode *Prototype*



Gambar 2. 1 Tahapan Metode *Prototype*
Sumber : Kustanto et al. (2024)

Secara umum, metode *Prototype* memiliki beberapa tahapan sebagai berikut :

1. Pengumpulan Kebutuhan
Pada tahap ini dilakukan identifikasi kebutuhan pengguna melalui observasi, wawancara, maupun studi dokumentasi. Informasi yang dikumpulkan meliputi kebutuhan fitur, data, serta permasalahan yang terjadi pada sistem berjalan.
2. Membangun *Prototype* Awal
Berdasarkan hasil identifikasi kebutuhan, pengembang menyusun *prototype* awal yang menggambarkan antarmuka, alur proses, dan fungsi utama sistem sebagai media evaluasi bersama pengguna.
3. Evaluasi *Prototype* oleh Pengguna
Prototype yang telah dibuat kemudian ditinjau bersama pengguna untuk memperoleh masukan mengenai tampilan, fungsi, maupun alur sistem. Hasil

evaluasi tersebut menjadi dasar bagi pengembang untuk melakukan penyempurnaan sebelum sistem dikembangkan lebih lanjut.

4. Mengkodekan Sistem

Apabila *prototype* telah disetujui oleh pengguna, tahap berikutnya adalah mengimplementasikan rancangan tersebut ke dalam bentuk program melalui proses coding menggunakan bahasa pemrograman yang telah ditentukan.

5. Menguji Sistem

Sistem yang telah dikembangkan kemudian diuji untuk memastikan bahwa seluruh fungsi berjalan dengan baik. Pengujian dilakukan untuk menemukan kesalahan (*bug*) pada sistem sebelum digunakan oleh pengguna.

6. Evaluasi Sistem

Sistem yang telah diuji kemudian dievaluasi kembali bersama pengguna untuk memastikan seluruh kebutuhan telah terpenuhi. Apabila masih ditemukan kekurangan atau fitur yang belum sesuai, maka pengembang akan melakukan perbaikan sebelum sistem diterapkan.

7. Penggunaan Sistem

Tahap terakhir adalah implementasi atau penggunaan sistem oleh pengguna. Sistem yang telah melalui tahap pengujian dan evaluasi siap digunakan dalam lingkungan nyata.

Apabila pada tahap evaluasi *prototype* maupun evaluasi sistem masih ditemukan kekurangan, maka proses pengembangan akan kembali ke tahapan sebelumnya hingga sistem memenuhi kebutuhan pengguna sesuai dengan konsep metode *Prototype*.

2.1.12 Unified Modelling Language (UML)

Unified Modeling Language (UML) merupakan bahasa pemodelan visual yang digunakan untuk menggambarkan, merancang, mendokumentasikan, serta memvisualisasikan sistem perangkat lunak, khususnya yang dikembangkan dengan pendekatan *object-oriented*. UML menyediakan berbagai notasi dan simbol standar yang membantu pengembang dalam merepresentasikan struktur maupun perilaku sistem secara lebih jelas. Sebagai standar pemodelan yang digunakan secara luas

dalam rekayasa perangkat lunak, UML mampu mempermudah proses perancangan sistem sekaligus meningkatkan pemahaman terhadap kebutuhan dan fungsi perangkat lunak yang akan dikembangkan (Wayahdi & Ruziq, 2023).

Selain berfungsi sebagai media visualisasi, UML juga mendukung proses pengembangan perangkat lunak yang dilakukan secara bertahap dan berulang (*iterative*). Melalui model yang disusun menggunakan UML, pengembang dapat membangun komunikasi yang lebih baik dengan pengguna maupun anggota tim pengembang karena rancangan sistem dapat dipahami secara lebih mudah sebelum proses implementasi dilakukan. Penggunaan UML juga membantu mengurangi kompleksitas dalam pengembangan perangkat lunak, mempermudah proses pemeliharaan, serta mendukung analisis kebutuhan sistem tanpa bergantung pada bahasa pemrograman atau teknologi tertentu (Wayahdi & Ruziq, 2023).

UML menyediakan berbagai jenis diagram yang dapat digunakan sesuai dengan kebutuhan analisis dan perancangan sistem. Setiap diagram memiliki fungsi yang berbeda dalam merepresentasikan aspek tertentu dari sistem, seperti kebutuhan fungsional, alur aktivitas, interaksi antarobjek, maupun struktur kelas. Pada penelitian ini, diagram yang digunakan meliputi *Use Case Diagram*, *Activity Diagram*, *Sequence Diagram*, *Class Diagram*, *Component Diagram*, dan *Deployment Diagram*. Keenam diagram tersebut digunakan untuk menggambarkan kebutuhan pengguna, alur proses, interaksi antarobjek, struktur sistem, hubungan antarkomponen, serta arsitektur implementasi sistem sehingga proses analisis, perancangan, dan dokumentasi perangkat lunak dapat dilakukan secara lebih sistematis dan mudah dipahami (Wayahdi & Ruziq, 2023).

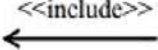
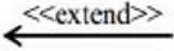
2.1.12.1 Use Case Diagram

Use Case Diagram merupakan salah satu jenis diagram pada *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan kebutuhan fungsional suatu sistem berdasarkan interaksi antara aktor dengan sistem. Diagram ini menunjukkan berbagai layanan atau fungsi yang dapat dijalankan oleh sistem beserta pihak-pihak yang berhak mengaksesnya. Dengan adanya *Use Case Diagram*, pengembang maupun pengguna dapat memahami ruang lingkup sistem

serta kebutuhan fungsional yang harus dipenuhi sejak tahap analisis, sehingga proses pengembangan perangkat lunak menjadi lebih terarah dan sesuai dengan kebutuhan pengguna (Wayahdi & Ruziq, 2023).

Tabel 2. 1 Komponen *Use Case Diagram*

Sumber : Ramdany et al. (2024)

Simbol	Nama	Keterangan
	Aktor	Mewakili peran pengguna, sistem lain, atau perangkat yang berinteraksi dengan sistem.
	<i>Use Case</i>	Menggambarkan fungsi atau layanan yang disediakan sistem kepada aktor.
	<i>Association</i>	Menunjukkan hubungan komunikasi antara aktor dengan <i>use case</i> .
	<i>Generalization</i>	Menggambarkan hubungan pewarisan (<i>inheritance</i>) antara aktor maupun <i>use case</i> .
	<i>Include</i>	Menunjukkan bahwa suatu <i>use case</i> selalu menggunakan fungsi dari <i>use case</i> lainnya.
	<i>Extend</i>	Menunjukkan hubungan tambahan yang dijalankan apabila kondisi tertentu terpenuhi.

Dalam *Use Case Diagram*, aktor merepresentasikan pengguna atau entitas lain yang berinteraksi dengan sistem, sedangkan *use case* menggambarkan layanan atau fungsi yang disediakan oleh sistem kepada aktor tersebut. Hubungan antara aktor dan *use case* dibatasi oleh *system boundary*, yaitu batas yang membedakan proses yang dikerjakan oleh sistem dengan aktivitas yang berasal dari lingkungan di luar sistem. Oleh karena itu, aktor selalu ditempatkan di luar *system boundary*, sedangkan seluruh *use case* berada di dalam batas sistem sebagai representasi dari fungsi-fungsi yang tersedia (Wayahdi & Ruziq, 2023).

Selain hubungan antara aktor dan *use case*, pemodelan ini juga mengenal beberapa jenis relasi, yaitu *include*, *extend*, dan *generalization*. Relasi *include* digunakan ketika suatu *use case* harus selalu menjalankan fungsi dari *use case* lain sebagai bagian dari prosesnya. Sebaliknya, relasi *extend* menggambarkan fungsi tambahan yang hanya dijalankan apabila kondisi tertentu terpenuhi. Adapun relasi *generalization* digunakan untuk menunjukkan hubungan pewarisan perilaku atau karakteristik, baik pada aktor maupun *use case*. Penggunaan ketiga relasi tersebut membantu menghasilkan model sistem yang lebih lengkap, terstruktur, serta mampu merepresentasikan interaksi antarkomponen secara lebih jelas (Ramdany et al., 2024).

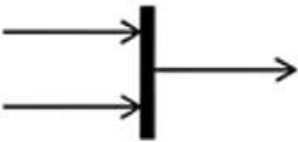
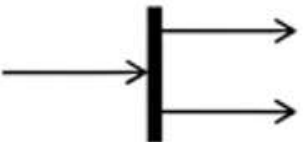
2.1.12.2 Activity Diagram

Activity Diagram merupakan salah satu jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan alur aktivitas atau *workflow* yang terjadi di dalam suatu sistem. Diagram ini memperlihatkan urutan proses mulai dari aktivitas awal hingga aktivitas akhir, termasuk setiap langkah, percabangan keputusan, maupun perpindahan proses yang terjadi selama sistem berjalan. Dengan visualisasi tersebut, *Activity Diagram* membantu pengembang dan pengguna memahami logika proses bisnis maupun alur kerja sistem secara lebih jelas sehingga mempermudah proses analisis dan perancangan perangkat lunak (Neptunus ARTEII, 2024).

Fokus utama *Activity Diagram* adalah menggambarkan *control flow* atau aliran kendali yang menunjukkan perpindahan dari satu aktivitas ke aktivitas lainnya. Selain menampilkan urutan proses secara bertahap, diagram ini juga mampu merepresentasikan percabangan keputusan, aktivitas yang berjalan secara bersamaan (*parallel process*), serta proses sinkronisasi sebelum alur dilanjutkan ke tahap berikutnya. Kemampuan tersebut menjadikan *Activity Diagram* sebagai salah satu media pemodelan yang efektif untuk menggambarkan proses sistem secara sistematis dan mudah dipahami oleh seluruh pihak yang terlibat dalam pengembangan perangkat lunak (Wayahdi & Ruziq, 2023).

Tabel 2. 2 Komponen *Activity Diagram*

Sumber : Ramdany et al. (2024)

Simbol	Nama	Keterangan
	<i>Activity</i>	Merepresentasikan suatu aktivitas atau proses yang dilakukan dalam alur kerja sistem.
	<i>Control Flow</i>	Menunjukkan urutan perpindahan dari satu aktivitas ke aktivitas berikutnya.
	<i>Object Flow</i>	Menggambarkan perpindahan objek atau data antaraktivitas dalam sistem.
	<i>Start Point</i>	Menunjukkan titik awal dimulainya suatu proses atau aktivitas.
	<i>End Point</i>	Menunjukkan titik akhir dari suatu proses atau aktivitas.
	<i>Join / Penggabungan</i>	Digunakan untuk menggabungkan kembali beberapa aktivitas yang berjalan secara paralel.
	<i>Fork</i>	Digunakan untuk membagi satu alur menjadi beberapa aktivitas yang berjalan secara bersamaan.
	<i>Decision</i>	Menunjukkan percabangan proses berdasarkan kondisi atau keputusan tertentu.

Setiap komponen pada *Activity Diagram* memiliki peran yang berbeda dalam menggambarkan alur kerja sistem. *Start Point* digunakan sebagai penanda

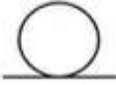
dimulainya proses, sedangkan *Activity* merepresentasikan setiap aktivitas yang dilakukan selama sistem berjalan. Apabila terdapat kondisi tertentu yang memerlukan percabangan proses, maka digunakan simbol *Decision*. Sementara itu, simbol *Fork* dan *Join* digunakan untuk menggambarkan aktivitas yang berlangsung secara paralel beserta proses sinkronisasinya sebelum alur kembali menjadi satu. Seluruh rangkaian aktivitas tersebut diakhiri dengan *End Point* yang menandakan bahwa proses telah selesai. Dengan memahami fungsi setiap komponen tersebut, *Activity Diagram* dapat digunakan untuk merepresentasikan alur sistem secara jelas, terstruktur, dan mudah dipahami oleh pengembang maupun pengguna (Wayahdi & Ruziq, 2023).

2.1.12.3 Sequence Diagram

Sequence Diagram merupakan salah satu diagram pada *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan urutan interaksi antarobjek dalam suatu sistem berdasarkan kronologi waktu. Diagram ini memperlihatkan bagaimana setiap objek saling mengirim dan menerima pesan selama proses berlangsung hingga suatu fungsi atau layanan sistem dapat dijalankan. Dengan menampilkan alur komunikasi secara berurutan, *Sequence Diagram* membantu pengembang memahami mekanisme kerja sistem secara lebih rinci sehingga proses analisis, perancangan, maupun implementasi perangkat lunak dapat dilakukan dengan lebih sistematis (Wayahdi & Ruziq, 2023).

Tabel 2. 3 Komponen *Sequence Diagram*

Sumber : Ramdany et al. (2024)

Simbol	Nama	Keterangan
	<i>Actor</i>	Merepresentasikan pengguna atau entitas yang berinteraksi dengan sistem.
	<i>Entity Class</i>	Merepresentasikan objek yang berhubungan dengan pengelolaan dan penyimpanan data dalam sistem.

Simbol	Nama	Keterangan
	<i>Boundary Class</i>	Berfungsi sebagai antarmuka yang menghubungkan aktor dengan sistem.
	<i>Control Class</i>	Mengendalikan alur proses serta logika bisnis yang berlangsung dalam sistem.
	<i>Message</i>	Menunjukkan pertukaran pesan atau komunikasi antarobjek.
	<i>Recursive</i>	Menggambarkan objek yang mengirimkan pesan kepada dirinya sendiri.
	<i>Lifeline</i>	Menunjukkan keberadaan suatu objek selama proses interaksi berlangsung.
	<i>Activation</i>	Menunjukkan periode ketika suatu objek sedang menjalankan suatu aktivitas atau operasi.

Dalam *Sequence Diagram*, setiap simbol memiliki fungsi yang saling berkaitan untuk menggambarkan urutan interaksi di dalam sistem. *Actor* berperan sebagai pihak yang memulai proses interaksi dengan sistem, sedangkan *Boundary Class* berfungsi sebagai media komunikasi antara aktor dan sistem. Selanjutnya, *Control Class* mengatur alur proses serta logika bisnis yang dijalankan, sementara *Entity Class* bertanggung jawab terhadap pengelolaan maupun penyimpanan data. Komunikasi antarobjek ditunjukkan melalui *Message*, sedangkan *Recursive Message* digunakan ketika suatu objek menjalankan proses terhadap dirinya sendiri. Selain itu, *Lifeline* menggambarkan keberadaan objek selama proses berlangsung, dan *Activation* menunjukkan rentang waktu ketika objek sedang menjalankan suatu operasi. Kombinasi seluruh komponen tersebut memungkinkan *Sequence Diagram* merepresentasikan alur komunikasi antarobjek secara kronologis, sehingga proses

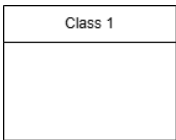
analisis dan perancangan sistem menjadi lebih jelas, terstruktur, dan mudah dipahami (Ramdany et al., 2024).

2.1.12.4 Class Diagram

Class Diagram merupakan diagram pada *Unified Modeling Language* (UML) yang digunakan untuk memodelkan struktur sebuah sistem melalui kumpulan kelas (*class*) beserta hubungan di antaranya. Setiap kelas mendefinisikan karakteristik yang dimiliki objek dalam bentuk atribut (*attribute*) dan operasi (*method*), sehingga diagram ini mampu memberikan gambaran mengenai bagaimana komponen-komponen penyusun sistem saling berhubungan. Oleh karena itu, *Class Diagram* menjadi salah satu acuan penting dalam proses perancangan karena dapat menggambarkan struktur sistem sebelum diimplementasikan ke dalam kode program (Wayahdi & Ruziq, 2023).

Tabel 2.4 Komponen *Class Diagram*

Sumber : Ramdany et al. (2024)

Simbol	Nama	Keterangan
	<i>Class</i>	Merepresentasikan kelas yang berisi nama kelas, atribut, dan operasi (<i>method</i>) sebagai dasar pembentukan objek.
	<i>Association</i>	Hubungan antar kelas yang menunjukkan adanya interaksi atau keterkaitan antar objek.
	<i>Aggregation</i>	Hubungan keseluruhan dan bagian (<i>whole-part relationship</i>) dimana bagian masih dapat berdiri sendiri tanpa objek induknya.
	<i>Composition</i>	Hubungan keseluruhan dan bagian yang kuat (<i>strong aggregation</i>), dimana bagian tidak dapat berdiri sendiri tanpa objek induknya.

Simbol	Nama	Keterangan
	<i>Generalization</i>	Menggambarkan hubungan pewarisan (<i>inheritance</i>) antara kelas induk dan kelas turunan.
	<i>Dependency</i>	Menunjukkan hubungan ketergantungan ketika suatu kelas memanfaatkan kelas lain untuk menjalankan fungsinya.

Pemodelan menggunakan *Class Diagram* tidak hanya menampilkan daftar kelas yang terdapat pada sistem, tetapi juga menjelaskan bagaimana setiap kelas saling berhubungan dalam membentuk suatu struktur yang utuh. Hubungan tersebut dapat berupa *Association* sebagai keterkaitan umum antar kelas, *Aggregation* yang menunjukkan hubungan antara objek induk dan objek bagian tanpa ketergantungan penuh, maupun *Composition* yang menggambarkan hubungan kepemilikan yang lebih kuat karena keberadaan objek bagian bergantung sepenuhnya pada objek induknya.

Selain hubungan tersebut, *Class Diagram* juga memanfaatkan *Generalization* untuk menunjukkan konsep pewarisan sifat atau fungsi dari kelas induk kepada kelas turunan. Sementara itu, *Dependency* digunakan ketika suatu kelas membutuhkan layanan atau fungsi dari kelas lain selama proses tertentu berlangsung. Dengan memadukan atribut, metode, serta berbagai jenis relasi tersebut, *Class Diagram* mampu memberikan representasi struktur sistem secara menyeluruh. Informasi ini menjadi dasar bagi pengembang dalam merancang arsitektur perangkat lunak yang terorganisasi, mudah dipelihara, serta lebih mudah dikembangkan pada tahap implementasi berikutnya (Ramdany et al., 2024).

2.1.13 *Entity Relationship Diagram (ERD)*

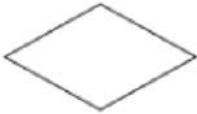
Entity Relationship Diagram (ERD) merupakan model konseptual yang digunakan untuk menggambarkan struktur data beserta hubungan antarentitas dalam suatu basis data. Diagram ini memberikan representasi visual mengenai objek-objek yang dikelola oleh sistem, atribut yang dimiliki setiap objek, serta

hubungan yang terbentuk di antaranya. Melalui pemodelan tersebut, pengembang dapat menyusun rancangan basis data secara lebih terstruktur sebelum diimplementasikan ke dalam *Database Management System* (DBMS). Selain itu, penggunaan ERD juga membantu mengurangi terjadinya redundansi data dan menjaga konsistensi hubungan antar data yang tersimpan di dalam basis data (Azzahra, 2022).

Pada proses pengembangan sistem informasi, ERD berperan sebagai dasar dalam penyusunan skema basis data relasional. Setiap entitas yang diidentifikasi akan dikembangkan menjadi tabel, sedangkan atribut dan relasi yang dimiliki akan menjadi acuan dalam menentukan struktur tabel beserta hubungan antar tabel tersebut. Proses ini melibatkan penentuan *primary key* sebagai identitas unik setiap data serta *foreign key* sebagai penghubung antar tabel yang saling berelasi. Dengan rancangan yang baik, struktur basis data menjadi lebih terorganisasi, mendukung integritas data, serta mempermudah pengelolaan maupun integrasi antar modul dalam sistem informasi (Putra et al., 2022; Aulia, 2025).

Tabel 2. 5 Komponen *Entity Relationship Diagram* (ERD)

Sumber : Triaulia, E. (2021)

Simbol	Nama	Keterangan
	Entitas	Merepresentasikan objek atau konsep yang datanya disimpan di dalam basis data.
	Relasi	Menunjukkan hubungan yang terjadi antar entitas dalam sistem.
	Atribut	Menjelaskan karakteristik yang dimiliki oleh suatu entitas, termasuk atribut yang berfungsi sebagai key.
	Garis	Menghubungkan entitas dengan relasi maupun atribut yang berkaitan.

Setiap simbol dalam ERD memiliki fungsi tertentu untuk menggambarkan struktur basis data secara menyeluruh. *Entity* digunakan sebagai representasi objek yang akan disimpan di dalam basis data, misalnya data siswa, guru, kelas, maupun presensi. Karakteristik dari setiap entitas dijelaskan melalui *Attribute*, sedangkan keterkaitan antarentitas digambarkan menggunakan *Relationship*. Seluruh komponen tersebut dihubungkan oleh *Line* sehingga hubungan antar data dapat divisualisasikan secara jelas dan mudah dipahami.

Selain menampilkan entitas dan atribut, ERD juga menggambarkan tingkat hubungan atau kardinalitas antar entitas. Kardinalitas menunjukkan banyaknya hubungan yang dapat terjadi, seperti *one to one* (1:1), *one to many* (1:N), maupun *many to many* (M:N). Konsep ini membantu pengembang menentukan bagaimana data saling berinteraksi serta menjadi dasar dalam pembentukan relasi antar tabel pada basis data relasional. Dengan demikian, rancangan basis data yang dihasilkan akan lebih konsisten, efisien, dan sesuai dengan kebutuhan sistem (Azzahra, 2022).

Berdasarkan uraian tersebut, dapat dipahami bahwa *Entity Relationship Diagram* (ERD) merupakan alat pemodelan yang berperan penting dalam proses perancangan basis data. ERD memberikan gambaran mengenai entitas, atribut, relasi, serta kardinalitas sehingga hubungan antar data dapat dirancang secara sistematis sebelum proses implementasi dilakukan. Dalam penelitian ini, ERD digunakan sebagai landasan untuk merancang struktur basis data pada Sistem Informasi Manajemen Sekolah berbasis *web* yang terintegrasi dengan sistem presensi *fingerprint*. Melalui perancangan tersebut, setiap data yang berkaitan dengan pengguna, guru, siswa, presensi, maupun modul lainnya dapat saling terhubung dan dikelola secara efektif sesuai kebutuhan sistem.

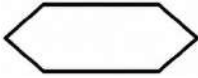
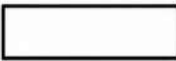
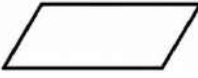
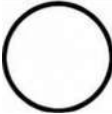
2.1.14 Flowchart

Flowchart merupakan representasi visual yang digunakan untuk menggambarkan urutan langkah atau alur proses dalam suatu sistem menggunakan simbol-simbol yang telah distandarkan. Diagram ini membantu menunjukkan hubungan antaraktivitas secara logis, mulai dari proses awal hingga proses berakhir, sehingga alur kerja sistem dapat dipahami dengan lebih mudah. Dalam

pengembangan perangkat lunak, *flowchart* sering dimanfaatkan sebagai media untuk menganalisis, merancang, serta mendokumentasikan proses bisnis sebelum sistem diimplementasikan ke dalam bentuk program. Dengan demikian, komunikasi antara pengembang dan pengguna mengenai mekanisme kerja sistem dapat berlangsung secara lebih efektif (Darmawan et al., 2026).

Tabel 2. 6 Komponen *Flowchart*

Sumber : Anggreani dkk. (2024)

Simbol	Nama	Keterangan
	<i>Terminator</i>	Menunjukkan titik awal atau akhir suatu proses maupun program.
	Garis Alir (<i>Flow Line</i>)	Menunjukkan arah aliran proses dari satu langkah ke langkah berikutnya.
	<i>Preparation</i>	Digunakan untuk proses inisialisasi atau pemberian nilai awal.
	<i>Process</i>	Merepresentasikan aktivitas atau proses pengolahan data yang dilakukan sistem.
	<i>Input/Output data</i>	Menunjukkan proses pemasukan maupun keluaran data atau informasi.
	<i>Decision</i>	Digunakan untuk menggambarkan percabangan berdasarkan suatu kondisi atau keputusan.
	<i>On Page Connector</i>	Menghubungkan alur yang terpisah tetapi masih berada pada halaman yang sama.

Simbol	Nama	Keterangan
	<i>Off Page Connector</i>	Menghubungkan alur proses yang berlanjut ke halaman lain.

Setiap simbol dalam *flowchart* memiliki fungsi yang berbeda sesuai dengan perannya dalam menggambarkan alur proses. *Terminator* digunakan sebagai penanda dimulainya maupun berakhirnya suatu proses, sedangkan *Process* menunjukkan aktivitas yang dikerjakan oleh sistem. Proses penerimaan atau penyajian data direpresentasikan menggunakan simbol *Input/Output*, sementara *Decision* digunakan ketika terdapat kondisi yang menghasilkan lebih dari satu kemungkinan alur. Hubungan antaraktivitas selanjutnya ditunjukkan melalui *Flow Line*, sehingga urutan proses dapat diikuti secara sistematis dari awal hingga akhir.

Selain simbol utama tersebut, *flowchart* juga memanfaatkan *On Page Connector* dan *Off Page Connector* untuk menghubungkan alur yang terpisah karena keterbatasan ruang pada diagram. Penggunaan kedua simbol ini membantu menjaga keterbacaan diagram tanpa mengurangi kejelasan urutan proses yang digambarkan. Dengan menerapkan simbol-simbol sesuai standar, *flowchart* mampu menjadi media dokumentasi yang efektif dalam menjelaskan logika sistem, mendukung proses analisis dan perancangan, serta mempermudah pengembangan maupun pemeliharaan perangkat lunak di masa mendatang (Nadiyah, 2023).

2.2 Penelitian Terdahulu

Penelitian terdahulu mengenai sistem informasi manajemen sekolah menunjukkan bahwa penerapan sistem berbasis teknologi informasi dapat membantu meningkatkan efektivitas pengelolaan data dan administrasi sekolah. Zufria et al. (2022) mengembangkan sistem informasi manajemen sekolah berbasis *web* untuk mendukung pengelolaan data akademik dan nonakademik. Hasil penelitian tersebut menunjukkan bahwa sistem informasi dapat membantu meningkatkan efisiensi pengelolaan administrasi sekolah, tetapi belum terintegrasi dengan perangkat presensi. Sementara itu, Rahmadi et al. (2021) mengembangkan

manajemen sekolah terintegrasi berbasis sistem informasi yang mampu mengelola data sekolah secara terintegrasi, namun belum dilengkapi dengan sistem presensi otomatis.

Penelitian mengenai presensi berbasis *Internet of Things* juga telah banyak dilakukan. Permana et al. (2023) mengembangkan sistem absensi siswa menggunakan teknologi *Internet of Things* yang memungkinkan proses monitoring kehadiran secara *real-time*, tetapi belum mencakup pengelolaan data sekolah secara keseluruhan. Andrean et al. (2025) juga mengembangkan sistem monitoring absensi siswa berbasis *Internet of Things* yang memudahkan pemantauan kehadiran secara cepat, tetapi masih terbatas pada fungsi monitoring dan belum terintegrasi dengan sistem informasi sekolah. Hasil penelitian tersebut menunjukkan bahwa penerapan IoT dapat meningkatkan efisiensi proses presensi dan pemantauan kehadiran, tetapi masih terdapat kebutuhan untuk mengintegrasikan data presensi dengan pengelolaan informasi sekolah.

Penelitian lain memanfaatkan teknologi identifikasi otomatis seperti RFID dan *fingerprint* untuk meningkatkan keandalan proses presensi. Kurniawan et al. (2022) mengembangkan sistem informasi presensi berbasis RFID yang terintegrasi dengan *database* berbasis *web* dan menggunakan *webhook* untuk pertukaran data serta pengujian *black box*. Sistem tersebut dapat meningkatkan efisiensi pencatatan presensi, tetapi belum menggunakan *fingerprint* sebagai metode autentikasi. Adejumobi et al. (2021) mengembangkan sistem presensi berbasis *fingerprint* yang dilengkapi dengan notifikasi menggunakan *Simple Mail Transfer Protocol* (SMTP). Penelitian tersebut memiliki kesamaan dalam penggunaan *fingerprint* dan mekanisme notifikasi, tetapi masih berfokus pada sistem presensi dan belum mencakup integrasi dengan sistem informasi manajemen sekolah.

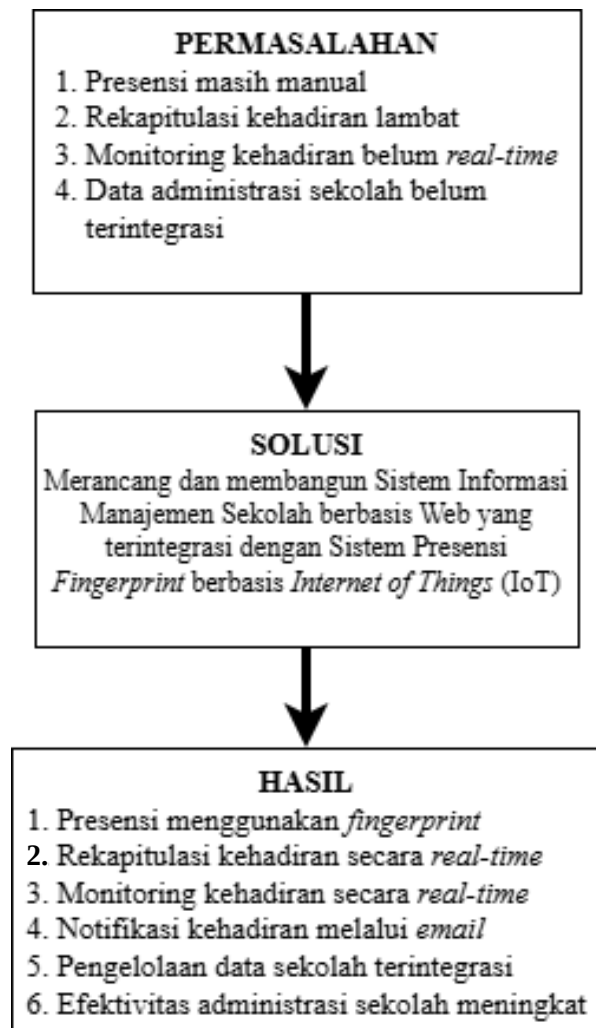
Penelitian yang mengintegrasikan teknologi IoT dan biometrik juga menunjukkan bahwa data kehadiran dapat diproses dan dikirimkan secara terhubung ke *server*. Nguyen et al. (2022) mengembangkan sistem absensi cerdas berbasis IoT dan biometrik yang terhubung dengan server serta dievaluasi melalui survei. Penelitian Aposika (2026) juga menunjukkan bahwa teknologi biometrik

dapat mendukung peningkatan efektivitas pemantauan kehadiran mahasiswa. Meskipun demikian, kedua penelitian tersebut belum secara khusus berfokus pada integrasi sistem presensi biometrik dengan sistem informasi manajemen sekolah yang mengelola data administrasi sekolah secara terpadu.

Berdasarkan beberapa penelitian terdahulu tersebut, dapat diketahui bahwa penelitian mengenai sistem informasi sekolah, presensi berbasis IoT, RFID, dan *fingerprint* telah banyak dikembangkan secara terpisah maupun dalam lingkup tertentu. Namun, masih terdapat ruang pengembangan pada integrasi antara sistem informasi manajemen sekolah dengan sistem presensi *fingerprint* berbasis IoT secara *real-time*. Oleh karena itu, penelitian ini mengembangkan sistem presensi *fingerprint* berbasis IoT yang terintegrasi dengan Sistem Informasi Manajemen Sekolah berbasis *web* di MI Panyocokan I. Sistem yang dikembangkan tidak hanya melakukan pencatatan kehadiran, tetapi juga mengintegrasikan data presensi dengan pengelolaan data sekolah serta menyediakan informasi kehadiran secara *real-time* dan notifikasi melalui *email* kepada orang tua.

2.3 Kerangka Pemikiran

Kerangka berpikir merupakan gambaran alur pemecahan masalah yang menjadi dasar dalam pelaksanaan penelitian ini



Gambar 2. 2 Kerangka Pemikiran

Kerangka berpikir penelitian ini menggambarkan alur penyelesaian permasalahan yang terjadi di MI Panyocokan I. Penelitian diawali dengan identifikasi permasalahan berupa proses presensi yang masih dilakukan secara manual, rekapitulasi kehadiran yang lambat, monitoring kehadiran yang belum *real-time*, serta pengelolaan data administrasi sekolah yang belum terintegrasi. Berdasarkan permasalahan tersebut, dirancang solusi berupa Sistem Informasi Manajemen Sekolah berbasis *web* yang terintegrasi dengan sistem presensi *fingerprint* berbasis *Internet of Things* (IoT), sehingga diharapkan mampu menghasilkan proses presensi yang lebih efektif, monitoring kehadiran secara *real-time*, notifikasi kehadiran melalui *email*, serta pengelolaan data sekolah yang terintegrasi.