BAB II

TINJAUAN PUSTAKA

2.1. Teori Dasar

2.1.1 Konsep Sistem Informasi

Kata "sistem" berasal dari bahasa Yunani yaitu "*system*", yang berarti suatu kesatuan atau kumpulan yang merujuk pada sekelompok komponen yang saling berhubungan untuk mencapai sasaran tertentu (Ali Ikhwan et al., 2024). Berkaitan dengan hal tersebut, data mentah yang telah diproses dan diubah sehingga memiliki arti serta nilai bagi penggunanya didefinisikan sebagai informasi, di mana hal ini berbeda dengan data yang masih berupa fakta lapangan tanpa makna khusus (Rasid Ridho, 2021).

Sistem Informasi (SI) sendiri merupakan kombinasi terintegrasi antara manusia, perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, dan data yang saling bekerja sama (Prayogo et al., 2021). Fungsi utama dari sistem ini adalah untuk mengumpulkan, mengolah, menyimpan, dan menyebarkan informasi untuk mendukung aktivitas operasional harian, membantu kebutuhan administrasi, hingga menyediakan laporan penting untuk mendukung pimpinan dalam mengambil keputusan (Rasid Ridho, 2021).

2.1.2 Pendidikan dan Kedisiplinan

Penjadwalan secara umum didefinisikan sebagai suatu rencana pengaturan urutan kerja sekaligus proses pengalokasian sumber daya, baik berupa waktu maupun fasilitas, untuk menyelesaikan setiap operasi yang telah ditentukan (Simarangkir, 2021). Selain itu, kegiatan penjadwalan (*scheduling*) ini juga

mencakup proses pembagian fasilitas, peralatan, hingga tenaga kerja untuk suatu aktivitas operasi, serta menjadi penentu urutan pelaksanaannya di mana dalam hierarki pengambilan keputusan, proses ini merupakan langkah terakhir sebelum operasi dimulai (Simarangkir, 2021). Melalui penyusunan jadwal yang dirancang secara efektif, risiko terjadinya bentrok jadwal dapat dihindari sekaligus dapat memaksimalkan efisiensi waktu selama Kegiatan Belajar Mengajar (KBM) berlangsung (Sri Novida Sari et al., 2022). Sebagai inti utama dari jalannya proses pendidikan, KBM sendiri diartikan sebagai bentuk interaksi timbal balik yang terjadi antara guru dan siswa. Dalam menjalankan struktur ini, Wali Kelas memegang tanggung jawab yang sangat penting sebagai jembatan komunikasi antara pihak sekolah dengan orang tua murid, terutama dalam memastikan validitas atau kebenaran dari surat izin serta keterangan absen siswa (Sardiman, 2024).

Di sisi lain, perkembangan diri siswa serta penanganan masalah tata tertib di sekolah menjadi tugas utama dari bagian Bimbingan dan Konseling. Tenaga pendidik seperti Bimbingan Konseling memiliki peran yang sangat besar untuk mengarahkan sekaligus memaksimalkan seluruh potensi yang ada pada diri siswa (Harita et al., 2022). Agar proses penanganan siswa tersebut dapat berjalan secara optimal, pihak Bimbingan Konseling memerlukan data perilaku yang cepat dan akurat supaya langkah pencegahan bisa segera diambil.

2.1.3 *Monitoring*

Dalam lingkungan sebuah organisasi, *monitoring* atau pemantauan diartikan sebagai salah satu landasan mendasar yang bersifat krusial dalam pelaksanaan setiap kegiatan. Proses ini digunakan sebagai sebuah tolok ukur utama untuk menilai sejauh mana program kerja telah terlaksana serta memastikan apakah target

dan tujuan organisasi tersebut sudah berhasil dicapai. Secara struktur, kegiatan monitoring ini melibatkan berbagai unsur, komponen, atau variabel terorganisasi yang saling terikat satu sama lain, di mana seluruh komponen tersebut bekerja bersama dengan tujuan untuk mengumpulkan data dari beraneka ragam sumber daya yang ada (Trisnawati & Setiawan, 2022).

2.1.4 Analisis Efisiensi Waktu

Menurut metode studi waktu (*time study*), tingkat keberhasilan peningkatan efisiensi sebuah proses kerja dihitung berdasarkan persentase penurunan durasi waktu siklus sebelum dan sesudah dilakukannya perbaikan metode terhadap waktu awal (Putra et al., 2026). Pada rumpun perancangan sistem informasi, perhitungan selisih kuantitatif tersebut diadopsi untuk mengukur performa perangkat lunak yang diusulkan. Komponen waktu awal diposisikan sebagai Durasi Konvensional, sedangkan waktu sesudah direpresentasikan sebagai Durasi Sistem. Dengan demikian, formula untuk menghitung persentase efisiensi akhir dijabarkan pada Rumus 2.1.

$$\text{Presentase Efisiensi} = \frac{\text{Durasi Konvensional} - \text{Durasi Sistem}}{\text{Durasi Konvensional}} \text{ x } 100\% \quad (2.1)$$

2.1.5 REST API

Representational State Transfer atau REST merupakan sebuah gaya arsitektur dalam merancang layanan *website* (*web service*) yang berpusat pada objek atau sumber daya (*resource*), di mana setiap sumber daya tersebut dapat diakses secara spesifik melalui alamat *URL HTTP* yang unik. Sebagai suatu arsitektur perangkat lunak, sistem ini memuat serangkaian aturan serta batasan teknis terstruktur yang wajib dipenuhi dalam proses pembuatan layanan *website*.

Batasan-batasan di dalam rancangan REST API ini memiliki peran yang sangat krusial untuk mengatur, mengontrol, sekaligus mengamankan proses pertukaran data serta interaksi yang terjadi antara aplikasi klien dengan basis data pusat di dalam server (Lesmana & Razaq, 2023).

2.1.6 Framework

Framework merupakan kerangka kerja dimana berupa *library*, fungsi atau *method* yang bersifat dapat digunakan kembali (*re-use*) kapan saja oleh pengembang perangkat lunak. Keberadaan komponen ini bertujuan agar programmer tidak perlu menyusun kode program yang sama berulang kali untuk menyelesaikan tugas atau fungsi yang serupa. Sebagai contoh, saat ingin menampilkan data pada halaman *website* dengan sistem penomoran halaman (*pagination*), fungsi tersebut umumnya telah disediakan langsung oleh framework. Dengan demikian, pengembang cukup memanggil fungsi yang sudah ada tersebut saat proses *coding* berlangsung dengan tetap mengikuti aturan tata cara penulisan yang telah ditetapkan oleh masing-masing framework (Purnama Sari & Wijanarko, 2020).

2.1.7 Laravel

Laravel adalah sebuah framework berbasis PHP yang bersifat *open-source* dan dapat digunakan secara gratis untuk mengembangkan aplikasi *website* dengan menerapkan pola desain *Model-View-Controller* (MVC) dikenal sebagai Laravel. Struktur pola MVC pada Laravel memiliki karakteristik yang sedikit berbeda jika dibandingkan dengan arsitektur MVC pada umumnya. Perbedaan tersebut terletak pada keberadaan fitur *routing* yang berfungsi sebagai jembatan untuk mengatur setiap *request* dari pengguna menuju ke bagian *controller*, sehingga *controller* tidak

menerima permintaan tersebut secara langsung (Purnama Sari & Wijanarko, 2020).

2.1.8 Flutter

Flutter ialah sebuah framework *open-source* dari Google yang digunakan untuk membangun antarmuka (*user interface*) pada sistem operasi *Mobile* dan iOS. Keunggulan utama dari teknologi ini adalah sifatnya yang *cross-platform* sehingga pengembang dapat menghemat waktu tanpa perlu mempelajari bahasa *native* dari masing-masing platform. Selain itu, proses pengembangan aplikasi dapat berjalan lebih cepat berkat adanya fitur *hot reload* untuk memperbarui kode program dalam hitungan detik, serta kemudahan kustomisasi tampilan UI yang menarik melalui penggunaan sistem *widget* yang fleksibel (Rachmat Setiawan et al., 2025).

2.1.9 *Database*

Database atau basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematik sehingga dapat diperiksa menggunakan suatu program komputer untuk memper*oleh* informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil *query* basis data disebut sistem manajemen basis data (*database management system*, DBMS) (Rezeki, 2025).

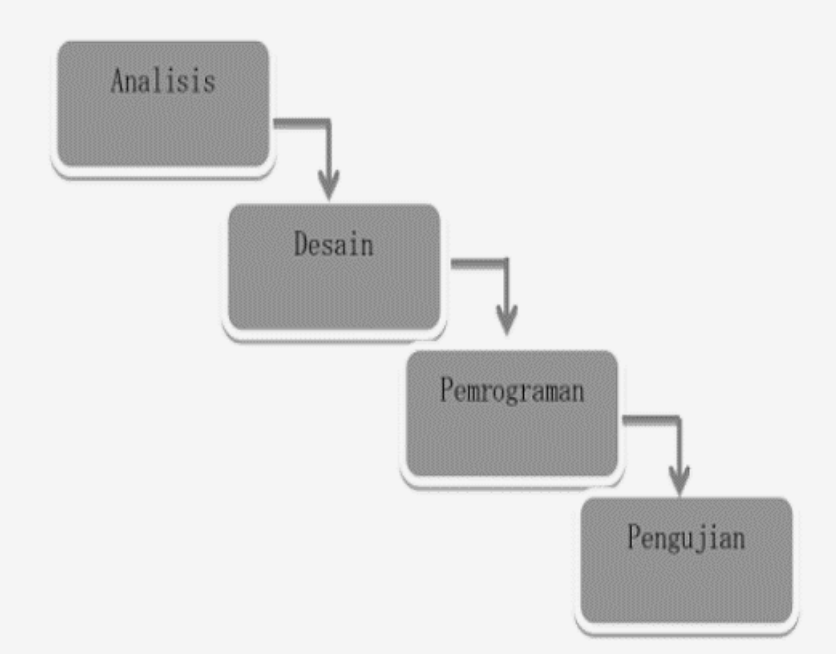
2.1.10 *QR Code*

QR Code adalah kode matriks berbentuk dua dimensi yang dirancang untuk menyimpan informasi dalam berbagai format, termasuk teks dan URL, serta dapat diakses dengan cepat melalui *scanner* atau kamera *smartphone*. Kode ini dikenal karena kapasitas penyimpanannya yang tinggi, kemampuan membaca yang cepat, dan teknologi *error correction* yang membuatnya tahan terhadap kerusakan sebagian. Teknologi *QR Code* dirancang tidak hanya untuk menyimpan data

berkapasitas besar, tetapi juga mendukung penyandian karakter kompleks seperti Kanji dan Kana (Chotijah et al., 2025).

2.1.11 Metode *Waterfall*

Model *Waterfall* atau sering disebut sebagai *classic life cycle* merupakan salah satu metode pengembangan rekayasa perangkat lunak tertua yang menerapkan pendekatan secara terurut dan sistematis. Berdasarkan alur pada Gambar 2.1, proses pengerjaan metode ini dilakukan secara sekuensial linear yang wajib dimulai dari tahap analisis kebutuhan sistem, dilanjutkan ke perancangan desain, proses pemrograman atau pengodean, hingga diakhiri dengan tahap pengujian perangkat lunak (Irwanto, 2021).



Gambar 2. 1 Metode Waterfall
(Sumber: Irwanto, 2021)

2.1.12 Diagram UML

Sebagai sebuah standar bahasa pemodelan, UML digunakan secara luas untuk menganalisis, merancang, sekaligus menggambarkan arsitektur program

pada pengembangan perangkat lunak berbasis objek (*object-oriented*) (Kurniawan et al., 2021). Dalam penerapannya, setiap komponen diagram tersebut memiliki sekumpulan simbol grafis standar dengan fungsi spesifik, di mana seluruh aturan visual dan teknis penggunaan simbol dalam perancangan sistem ini murni mengadopsi standar baku yang merujuk pada penelitian terdahulu oleh Harlina (2025).

1. *Activity* Diagram

   Activity diagram digunakan untuk memodelkan *workflow* atau alur kerja dari suatu proses bisnis dan rangkaian urutan aktivitas di dalam sistem. Sifat diagram ini sangat menyerupai *flowchart* karena menggambarkan perpindahan dari satu aktivitas ke aktivitas lainnya secara berurutan. Di dalam pengembangan sistem, diagram ini berfungsi krusial untuk memberikan penjelasan mengenai jalannya aktivitas program secara logis, tanpa perlu melihat visual tampilan antarmuka maupun detail baris kode pemrograman (*coding*) yang ada (Kurniawan et al., 2021).

2. *Use Case* Diagram

   Use case diagram didefinisikan sebagai struktur deskriptif yang menggambarkan urutan kelompok aktivitas saling terkait dan membentuk sistem secara teratur untuk dilakukan atau diawasi oleh seorang aktor. Secara visual, diagram ini menyajikan gambaran grafis mengenai aktor, fungsi use case, serta interaksi dasar yang memperkenalkan cara kerja suatu sistem. Pemodelan use case tidak ditujukan untuk menjelaskan detail teknis secara mendalam, melainkan hanya memberikan gambaran ringkas mengenai hubungan antar-komponen agar fungsi-fungsi di dalam sistem

yang dirancang dapat diketahui dengan jelas (Kurniawan et al., 2021).

3. *Sequence* Diagram

   Sequence diagram bertugas untuk menggambarkan jalannya interaksi dinamis dan kolaborasi antara sejumlah objek berdasarkan urutan waktu tertentu. Sebagai salah satu bagian integral dari rumpun UML, diagram ini memiliki kegunaan utama untuk menunjukkan kronologis rangkaian pesan (*message*) yang saling dikirim serta diterima antarobjek. Melalui pemodelan ini, pengembang dapat melacak setiap peristiwa atau interaksi data yang terjadi pada titik tertentu ketika sistem sedang dieksekusi (Kurniawan et al., 2021).

4. *Class* Diagram

   Class diagram berfungsi untuk menggambarkan struktur statis, penjelasan komponen, serta hubungan keterkaitan antara *class, package,* dan *object* di dalam sistem. Selain menjadi media komunikasi antar-pengembang perangkat lunak, pemodelan ini juga digunakan untuk memetakan kemampuan sistem dalam mengelola data sekaligus mengeksekusi tindakan tertentu pada tahap perancangan (*design phase*) (Wayahdi & Ruziq, 2023).

2.1.13 *Entity Relationship Diagram*

Entity Relationship Diagram (ERD) didefinisikan sebagai sebuah diagram terstruktur yang digunakan untuk merancang hubungan keterkaitan antartabel di dalam suatu basis data. Selain itu, ERD juga menjadi sebuah teknik penting yang digunakan oleh sistem analis pada tahap analisis persyaratan proyek untuk memodelkan seluruh kebutuhan data dari suatu organisasi (Kurniawan et al., 2021).

2.1.14 Pengujian *Blackbox*

Black box testing atau pengujian kotak hitam merupakan metode evaluasi perangkat lunak yang berfokus murni pada spesifikasi fungsional sistem tanpa melibatkan pemeriksaan struktur kode program maupun desain internalnya. Pengujian ini diterapkan untuk memastikan bahwa seluruh fungsi masukan (*input*) dan keluaran (output) pada aplikasi telah beroperasi sesuai dengan kebutuhan pengguna yang ditetapkan (Yulianti & Prastowo, 2021).

2.2. Penelitian Terdahulu

Tinjauan terhadap penelitian terdahulu dilakukan untuk mengetahui perkembangan teknologi terkini yang relevan dengan topik penelitian, serta untuk mengidentifikasi kesenjangan (gap) yang membedakan penelitian ini dengan sistem yang sudah ada sebelumnya. Berikut adalah matriks perbandingan dari tiga penelitian yang relevan dengan pengembangan sistem monitoring dan presensi:

Tabel 2. 1 Penelitian Terdahulu

No	Peneliti (Tahun)	Judul Penelitian	Metode & Teknologi	Hasil & Perbedaan dengan Penelitian Ini
1	Nuraiman, dkk. (2024)	Rancang Bangun Sistem Absensi Cepat untuk Guru Bimbingan Konseling Berbasis Android	Waterfall, Android, ISO 9126	Hasil: Sistem Presensi internal untuk mencatat kehadiran Bimbingan Konseling sebagai pegawai. Perbedaan: Penelitian ini fokus pada pemantauan kedisiplinan dan presensi siswa harian.
2	Hidaya & Prasetyo (2024)	*Dashboard* Monitoring Sistem Berbasis *Website* Santri pada Pondok Sekolah Multazam Bogor	Waterfall, UML, *Website*, CodeIgniter, PHP, MySQL, Black Box	Hasil: *Dashboard website* untuk mendokumentasikan pelanggaran santri dan mempermudah pelaporan ke orang tua. Perbedaan: Penelitian tersebut fokus mencatat pelanggaran santri via *website*. Penelitian ini fokus pada presensi KBM dan sinkronisasi izin menggunakan integrasi *Mobile* dan *Website*.
3	Sufandi, dkk. (2023)	Pengembangan Aplikasi Presensi Berbasis Android dan Web di Polnep	Android, *Website*, GPS Geolocation, Waterfall	Hasil: Aplikasi presensi pegawai berbasis lokasi (GPS) untuk WFH/WFO dan lembur. Perbedaan: Penelitian ini berfokus pada presensi kehadiran siswa per jam pelajaran di dalam kelas.

Tabel 2. 1 Penelitian Terdahulu (Lanjutan)

No	Peneliti (Tahun)	Judul Penelitian	Metode & Teknologi	Hasil & Perbedaan dengan Penelitian Ini
4	Lahagu & Siregar (2023)	Penerapan Aplikasi Presensi Pegawai Bagian Produksi Menggunakan *QR Code* Berbasis Web Dan Android	*QR Code*, Geofencing, Web, Android, UML	Hasil: Sistem pencatatan presensi pegawai produksi menggunakan *QR Code* yang dilengkapi validasi batas area (geofencing) untuk mencegah kecurangan. Perbedaan: Penelitian ini berfokus pada siswa, terintegrasi dengan validasi izin Wali Kelas dan pemantauan Bimbingan Konseling.
5	Saputra, dkk. (2022)	Sistem Informasi Monitoring Kinerja Kepala Sekolah dan Guru Pada Koordinator Wilayah Kecamatan Muara Kuang Berbasis Web	Prototype, *Website* (PHP, MySQL), DFD, ERD	Hasil: *Website* untuk penilaian supervisi kinerja kepala sekolah dan guru. Perbedaan: Penelitian tersebut fokus pada penilaian kinerja pegawai. Penelitian ini fokus pada presensi KBM dan izin siswa terintegrasi *Mobile* dan *Website*.