

BAB II

KAJIAN LITERATUR

2.1 Sistem Informasi

Sistem didefinisikan sebagai suatu kesatuan terorganisasi atau tatanan yang terdiri dari sejumlah komponen fungsional yang saling berhubungan dan berinteraksi untuk mencapai suatu tujuan atau memenuhi suatu proses pekerjaan tertentu. Dengan demikian, setiap elemen dalam sistem memiliki tugas khusus yang terintegrasi untuk mencapai sasaran akhir yang ditetapkan [11].

Informasi adalah data mentah yang telah diolah dari sumber terpercaya dan disajikan sedemikian rupa sehingga lebih bermakna dan relevan bagi penerimanya. Informasi yang berkualitas harus mengandung pengetahuan yang bermanfaat dan berfungsi sebagai dasar untuk mengambil sebuah keputusan.

Sistem Informasi adalah sebuah sistem yang mengintegrasikan pekerjaan manusia (sebagai sumber/pengguna) dengan penggunaan teknologi (meliputi perangkat keras/*hardware*, perangkat lunak/*software*, data, dan jaringan) untuk mendukung kegiatan manajemen dan operasional organisasi. Sistem ini dirancang secara khusus oleh manusia dengan tujuan utama menghasilkan produk berupa informasi yang akurat dan relevan.

Secara fungsional, Sistem Informasi menjalankan siklus yang mengubah data menjadi informasi yang berguna melalui serangkaian aktivitas utama [12]:

1. Input: Data mentah pertama yang diterima diubah menjadi bentuk yang sesuai untuk diproses.
2. Processing: Data tersebut kemudian dimanipulasi dan diubah menjadi informasi.
3. Storage: Informasi disimpan untuk penggunaan di masa mendatang.
4. Output: Informasi disampaikan kepada pengguna akhir.

2.2 Akademik

Istilah akademik berasal dari bahasa Yunani yaitu *academos*, yang awalnya merujuk pada sebuah taman umum di sebelah barat laut kota Athena. Nama tersebut diambil dari seorang pahlawan yang gugur dalam perang Troya. Di area taman inilah filosof Socrates dahulu berpidato serta membuka ruang perdebatan, yang kemudian juga menjadi tempat bagi Plato untuk berdialog dan mengajarkan pemikiran-pemikiran filosofisnya. Seiring berjalannya waktu, kata *academos* berkembang menjadi istilah akademik yang mengacu

pada lembaga perguruan, di mana para pengikutnya disebut *academist* dan lembaganya dikenal sebagai *academia*. Berdasarkan latar belakang sejarah tersebut, inti dari konsep akademik adalah suatu keadaan di mana individu atau kelompok memiliki kebebasan untuk menyampaikan, menerima, dan menguji gagasan serta ilmu pengetahuan secara jujur, terbuka, dan leluasa [13].

Dalam konteks persekolahan, akademik berkaitan erat dengan pengelolaan kurikulum dan proses pembelajaran untuk meningkatkan pengetahuan siswa [14]. Oleh karena itu, bidang akademik berhubungan langsung dengan seluruh dinamika kegiatan belajar mengajar (KBM) di kelas. Aktivitas ini meliputi tugas-tugas dalam program pembelajaran, diskusi, observasi, serta pengerjaan tugas terstruktur maupun mandiri yang dirancang untuk menunjang capaian belajar siswa. Kegiatan belajar mengajar sendiri merupakan proses hubungan timbal balik (*feedback*) yang komunikatif antara guru dan siswa. Dalam proses ini, guru memiliki peran penting sebagai fasilitator yang membimbing, mengarahkan, dan memotivasi siswa untuk mencapai tujuan pembelajaran tersebut [15].

Dalam ekosistem sekolah, tingkat kehadiran siswa (absensi) serta hasil penilaian merupakan bagian integral yang tidak dapat dipisahkan dari kegiatan akademik. Keduanya saling berkaitan untuk menciptakan proses pengelolaan data pembelajaran yang efektif, terukur, dan berkesinambungan.

2.2.1 Absensi

Absensi atau pencatatan kehadiran merupakan instrumen kendali mutu kedisiplinan sekaligus komponen penting dalam manajemen kesiswaan di lembaga pendidikan [16]. Menurut KBBI, istilah absensi umum diartikan sebagai "tidak hadir" [17]. Namun dalam konteks organisasi sekolah, kata ini merujuk pada rekam jejak kehadiran maupun ketidakhadiran siswa yang disertai dengan pencatatan resmi. Kehadiran siswa di sekolah tidak sekadar mencerminkan keberadaan fisik semata, melainkan menjadi indikator utama dalam memahami tingkat kepatuhan dan kebiasaan disiplin peserta didik terhadap peraturan yang berlaku di lingkungan pendidikan [18].

Secara operasional, data absensi berfungsi mendokumentasikan keikutsertaan siswa dalam proses pembelajaran di kelas. Tingkat kehadiran yang tinggi berkorelasi positif terhadap kontinuitas penerimaan materi pelajaran, sedangkan tingginya angka ketidakhadiran dapat menjadi indikator awal adanya hambatan belajar atau masalah kedisiplinan pada siswa. Oleh karena itu, penggunaan sistem absensi memiliki peran krusial sebagai alat pemantau kinerja siswa secara menyeluruh. Melalui data kehadiran yang tercatat

dengan baik, pihak sekolah dapat melakukan evaluasi perilaku secara objektif serta mengambil langkah-langkah yang diperlukan dalam rangka pembinaan karakter dan penanaman sikap disiplin siswa [19].

2.2.2 Evaluasi dan Penilaian

Evaluasi pembelajaran merupakan serangkaian kegiatan yang dilakukan secara sistematis dan berkesinambungan untuk memperoleh, menganalisis, serta menafsirkan data terkait proses dan hasil belajar siswa. Tujuan utama dari evaluasi ini adalah menghasilkan informasi yang bermakna dalam mendukung pengambilan keputusan instruksional yang tepat [20]. Pelaksanaan evaluasi secara berkala memberikan dampak positif bagi ekosistem sekolah; bagi siswa, proses ini menjadi dorongan untuk meningkatkan semangat belajar, bagi guru berperan dalam meningkatkan kualitas pengajaran, sedangkan bagi pihak sekolah menjadi dasar dalam memperbaiki mutu serta fasilitas pembelajaran secara keseluruhan [21]. Secara lebih luas, evaluasi tidak hanya mengukur hasil akhir, melainkan juga menilai faktor situasi belajar, kemampuan individu, pengetahuan awal, serta perkembangan pencapaian tujuan pendidikan.

Setelah proses pengumpulan data melalui evaluasi selesai, langkah berikutnya adalah penilaian. Penilaian merupakan tindakan memberikan nilai, keputusan, atau interpretasi terhadap hasil pengukuran yang diperoleh selama proses evaluasi, seperti hasil ujian siswa [22]. Di dalam implementasi sistem informasi akademik berbasis *web*, seluruh data evaluasi dan hasil ujian ini direkam secara terpusat. Sistem kemudian melakukan kalkulasi otomatis terhadap seluruh komponen nilai tersebut untuk menghasilkan nilai akhir siswa. Digitalisasi proses ini secara signifikan memangkas waktu rekapitulasi manual yang biasanya dilakukan oleh wali kelas, sekaligus meminimalisir risiko kesalahan hitung.

2.2.3 Sistem Informasi Akademik

Sistem Informasi Akademik merupakan sistem berbasis teknologi informasi yang digunakan untuk mengelola, menyimpan, memproses, dan menyajikan berbagai data akademik dalam suatu institusi pendidikan [23]. Sistem ini berfungsi sebagai sarana pendukung dalam pelaksanaan kegiatan akademik, mulai dari pengelolaan data siswa, data guru, mata pelajaran, jadwal mata pelajaran, dan pengolahan nilai. Dengan adanya Sistem Informasi Akademik, proses administrasi yang sebelumnya dilakukan secara manual dapat diotomatisasi sehingga menjadi lebih efektif dan efisien [24].

Penerapan Sistem Informasi Akademik memberikan berbagai manfaat bagi institusi pendidikan[25]. Sistem ini mampu meningkatkan efisiensi dalam pengelolaan data akademik, mengurangi kesalahan pencatatan data, mempercepat proses pelayanan administrasi, serta memudahkan akses informasi bagi seluruh pengguna. Selain itu, sistem juga mendukung transparansi dan akuntabilitas dalam pengelolaan data akademik karena seluruh informasi tersimpan secara terpusat dan dapat diakses sesuai dengan hak akses masing-masing pengguna [26].

2.3 Metode Waterfall

Metode Waterfall, yang juga dikenal sebagai Model Air Terjun atau *linear sequential life cycle model*, merupakan salah satu model klasik dalam *System Development Life Cycle (SDLC)*. Model ini diperkenalkan pada tahun 1970 dan mendefinisikan pendekatan pengembangan perangkat lunak secara sekuensial. Prinsip dasarnya adalah bahwa setiap tahapan proyek harus diselesaikan secara tuntas sebelum dapat beralih ke tahapan berikutnya, menyerupai aliran air terjun satu arah. Penekanan utama model ini adalah pada perencanaan yang detail dan dokumentasi yang komprehensif di setiap langkah proses [27].

Proses pengembangan menggunakan Metode *Waterfall* terdiri dari lima tahapan utama yang dilakukan secara berurutan [28]:

1. Analisis Kebutuhan: Tahap awal yang fokus pada pengumpulan, pendefinisian, dan pendokumentasian secara rinci semua kebutuhan fungsional dan non-fungsional sistem dari pengguna. Hasil dari tahap ini adalah spesifikasi kebutuhan yang berfungsi sebagai pedoman utama.
2. Desain Sistem: Kebutuhan yang telah didokumentasikan kemudian diterjemahkan menjadi arsitektur perangkat lunak, termasuk perancangan struktur data, antarmuka, dan arsitektur modular. Desain ini harus diverifikasi sebelum implementasi dimulai.
3. Implementasi: Tahap pengkodean, di mana desain sistem diwujudkan menjadi kode program menggunakan bahasa pemrograman terpilih. Selama tahap ini, Pengujian Unit dilakukan untuk memastikan fungsi setiap modul berjalan dengan benar.
4. Pengujian: Seluruh sistem yang telah diintegrasikan diuji secara menyeluruh untuk memastikan sistem berjalan sesuai dengan kebutuhan yang telah didefinisikan pada tahap awal. Pengujian mencakup Pengujian Integrasi, Sistem, dan Penerimaan Pengguna.

5. Pemeliharaan: Fase terpanjang yang dilakukan setelah sistem diluncurkan. Aktivitasnya meliputi perbaikan kesalahan yang ditemukan pasca-rilis, adaptasi terhadap lingkungan operasi yang berubah, dan peningkatan fungsionalitas sistem.

2.4 Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan standar pemodelan fundamental dalam rekayasa perangkat lunak yang dirancang untuk memvisualisasikan, menspesifikasikan, serta mendokumentasikan sistem perangkat lunak [29], khususnya sistem berbasis paradigma berorientasi objek. Perlu ditegaskan bahwa UML bukanlah sebuah metodologi pengembangan perangkat lunak, melainkan bahasa pemodelan visual terhadap proses atau metodologi tertentu. UML berperan sebagai cetak biru (*blueprint*) sistem yang memberikan gambaran arsitektur menyeluruh sebelum tahap implementasi kode. Terdapat beberapa jenis UML, diantaranya adalah Use Case Diagram, Activity Diagram, dan Class Diagram.

2.4.1 Class Diagram

Class Diagram merupakan jenis diagram yang menggambarkan struktur statis sistem dengan menunjukkan kelas-kelas yang terdapat dalam sistem. Diagram ini membantu merancang arsitektur *objek-oriented* [30], dimana setiap kelas merepresentasikan entitas dalam sistem. *Class Diagram* sering menjadi acuan utama dalam pembuatan *Entity Relationship Diagram (ERD)* untuk tabel database, dimana atribut kelas menjadi kolom tabel, dan relasi antar kelas menjadi *foreign key*.

Komponen utama dalam *Class Diagram* [31]:

1. *Class Name* adalah identitas kelas yang merepresentasikan objek nyata dalam sistem.
2. *Attributes* adalah properti atau data yang dimiliki oleh suatu kelas.
3. *Methods* adalah fungsi yang dapat dilakukan oleh suatu kelas.

Dalam penulisan *Class Diagram*, atribut dan metode dilengkapi dengan visibility:

1. *Public* (+), yaitu dapat diakses dari luar kelas.
2. *Private* (-), yaitu hanya dapat diakses di dalam kelas.
3. *Protected* (#), yaitu dapat diakses oleh kelas itu sendiri dan turunannya.

Jenis relasi antar kelas dalam *Class Diagram* meliputi:

1. *Association*, yaitu hubungan umum antar dua kelas.
2. *Aggregation*, yaitu hubungan “bagian dari” yang lemah, dimana objek masih dapat berdiri sendiri.

3. *Composition*, yaitu hubungan “bagian dari” yang kuat, dimana keberadaan objek bagian bergantung pada objek induk.
4. *Generalization*, yaitu hubungan pewarisan atau inheritance.

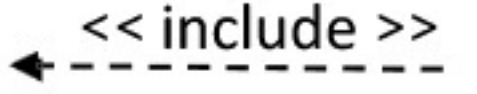
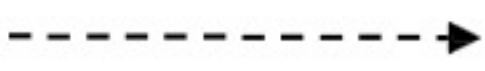
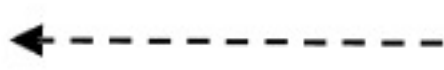
2.4.2 Use Case Diagram

Use Case Diagram merupakan jenis diagram yang menggambarkan interaksi antara aktor dengan fungsionalitas sistem [32]. Diagram ini menggambarkan kebutuhan fungsional sistem dari sudut pandang pengguna. Dengan diagram ini, pengembang dapat memahami layanan apa saja yang harus disediakan agar sesuai dengan kebutuhan pengguna.

Komponen utama dalam *Use Case Diagram* [33]:

1. Aktor adalah entitas eksternal yang berinteraksi langsung dengan sistem, seperti manusia atau sistem komputer lainnya.
2. *Use Case* adalah fungsi yang disediakan oleh sistem untuk memenuhi kebutuhan aktor.

Tabel 2.1 Simbol *Use Case Diagram*

No.	Simbol	Nama	Keterangan
1		<i>Actor</i>	Entitas eksternal yang berinteraksi dengan sistem.
2		<i>Use Case</i>	Aktivitas atau layanan yang digunakan aktor pada sistem.
3		<i>Association</i>	Hubungan komunikasi antara actor dan use case.
4		<i>System</i>	Sistem yang sedang dikembangkan.
5		<i>Include</i>	Use case yang selalu menjadi bagian dari use case lain.
6		<i>Dependency</i>	Ketergantungan antara elemen-elemen dalam diagram.
7		<i>Generalization</i>	Hubungan generalisasi antara aktor atau use case.
8		<i>Realization</i>	Implementasi suatu use case oleh elemen lain.

2.4.3 Activity Diagram

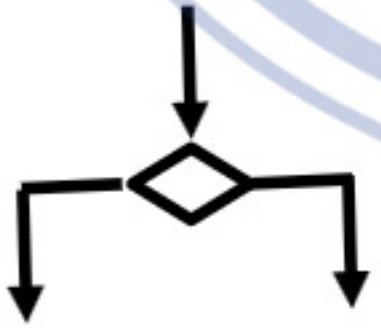
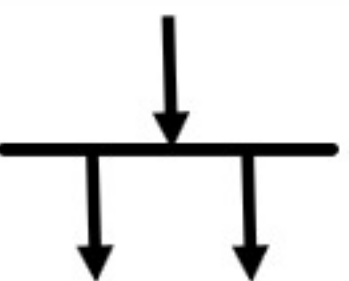
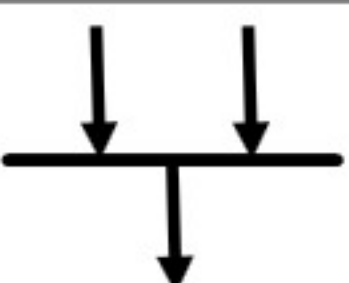
Activity Diagram merupakan jenis diagram yang menggambarkan urutan alur kerja atau rangkaian aktivitas dalam suatu proses sistem [34]. Diagram ini menunjukkan rangkaian aktivitas, pengambilan keputusan, kondisi percabangan, serta kemungkinan proses paralel.

Komponen utama dalam *Activity Diagram*:

1. *Start Node*, yaitu sebagai penanda awal proses.
2. *Activity*, yaitu menunjukkan suatu proses dalam sistem.
3. *Decision Node*, yaitu menunjukkan percabangan keputusan.
4. *Fork & Join Node*, yaitu untuk memecah satu aliran menjadi beberapa aliran paralel atau menggabungkannya kembali.
5. *Final Node*, yaitu sebagai penanda akhir proses.

Simbol-simbol yang ada pada Activity Diagram dapat dilihat pada tabel berikut [35]:

Tabel 2.2 Simbol *Activity Diagram*

No.	Simbol	Nama	Keterangan
1		<i>Start Node</i>	Menandai awal proses dalam sistem.
2		<i>Final Node</i>	Menandai akhir dari seluruh proses.
3		<i>Activity</i>	Menggambarkan proses yang dilakukan dalam sistem.
4		<i>Swimlane</i>	Menunjukkan pembagian proses berdasarkan peran tertentu.
5		<i>Decision Node</i>	Menunjukkan percabangan alur berdasarkan kondisi.
6		<i>Fork Node</i>	Memecah satu alur menjadi beberapa alur yang berjalan secara paralel.
7		<i>Join Node</i>	Menggabungkan kembali beberapa alur paralel menjadi satu alur.

2.5 Landasan Teknologi Website

2.5.1 Definisi dan Karakteristik Website Modern

Website didefinisikan sebagai antarmuka digital yang terdiri dari kumpulan sumber daya yang dihosting pada server dan diakses melalui protokol *HTTP/HTTPS*. *Website* dapat berisi teks, gambar, suara, animasi, dan multimedia lain, serta diidentifikasi dengan nama domain dan URL yang unik [36]. Dalam konteks pengembangan sistem informasi, *website* berfungsi sebagai lapisan presentasi yang menjembatani pengguna dengan logika bisnis dan basis data.

Karakteristik utama website modern yang relevan untuk SIA meliputi:

1. Desain Responsif dan Aksesibilitas

Desain responsif memastikan tampilan *website* menyesuaikan berbagai perangkat (*desktop, tablet, smartphone*), sehingga aksesibilitas bagi siswa dan guru tetap optimal. Fleksibilitas dan aksesibilitas menjadi aspek penting dalam pedagogi digital modern, memungkinkan pembelajaran di mana saja dan kapan saja, serta memperhatikan *universal design for learning*. Pengalaman positif siswa dalam pembelajaran daring juga sangat dipengaruhi oleh kemudahan penggunaan dan literasi perangkat digital [37].

2. Konten Dinamis

Website dinamis memungkinkan konten dihasilkan secara *real-time* dari *database* sesuai permintaan pengguna, seperti menampilkan rencana pembelajaran individu. Hal ini memerlukan integrasi antara *front-end* (presentasi) dan *back-end* (logika bisnis dan basis data), serta penggunaan bahasa pemrograman sisi server. Konten yang dapat disesuaikan dan interaktif meningkatkan fleksibilitas dan relevansi sistem informasi akademik [38].

3. Keamanan Data

Keamanan menjadi aspek krusial, terutama untuk data akademik yang sensitif. Penggunaan protokol *HTTPS* dan penerapan standar keamanan web (misal, *OWASP Top 10*) sangat dianjurkan untuk mencegah kebocoran data dan serangan siber. Selain itu, perlindungan privasi dan kepatuhan pada standar internasional (misal, *WCAG2.1* untuk aksesibilitas) juga penting [39].

2.5.2 Arsitektur Pengembangan Web Tiga Tingkat (*Three-Tier Architecture*)

Arsitektur *Three-Tier* (Tiga Tingkat) sangat direkomendasikan dalam pengembangan Sistem Informasi Akademik (SIA) karena memisahkan fungsionalitas dan

tanggung jawab secara logis, sehingga meningkatkan skalabilitas, keamanan, dan kemudahan pemeliharaan [40].

1. Tingkat Presentasi (*Presentation Tier*): Lapisan presentasi berfungsi sebagai antarmuka pengguna, di mana pengguna seperti siswa dan guru berinteraksi langsung dengan sistem melalui tampilan web yang dibangun menggunakan teknologi seperti HTML5, CSS3, dan framework JavaScript. Lapisan ini berfokus pada kenyamanan dan kemudahan penggunaan (user experience) [41].
2. Tingkat Aplikasi (*Application Tier*): Lapisan aplikasi bertanggung jawab menjalankan logika bisnis, aturan, serta memproses data yang diterima dari pengguna. Lapisan ini menjadi penghubung antara antarmuka pengguna dan basis data, memastikan setiap permintaan diproses sesuai aturan dan kebijakan sistem.
3. Tingkat Data (*Data Tier*): Lapisan data bertugas menyimpan dan mengelola data secara terstruktur menggunakan sistem manajemen basis data relasional (RDBMS) seperti *MySQL* atau *PostgreSQL*. Lapisan ini memastikan data akademik, seperti nilai dan jadwal, tetap aman, konsisten, dan mudah diakses oleh lapisan aplikasi sesuai kebutuhan.

2.5.3 Aspek User Experience (UX) dan Usability pada Website Pendidikan

Keberhasilan adopsi Sistem Informasi Akademik (SIA) berbasis *web* sangat bergantung pada kualitas *User Experience* (UX) dan *Usability*[42].

1. Usability (Kemudahan Penggunaan)

Usability mengacu pada seberapa mudah pengguna dapat mencapai tujuan mereka, seperti mengisi rencana pembelajaran atau input nilai, secara efektif dan efisien. Tingkat kemudahan penggunaan yang tinggi membuat pengguna merasa sistem mudah dipelajari, diingat, dan minim kesalahan. Studi menunjukkan bahwa persepsi kemudahan penggunaan (*perceived ease of use*) sangat memengaruhi sikap dan niat pengguna untuk terus menggunakan sistem *e-learning* atau SIA. Jika usability rendah, proses akademik dapat terhambat dan menurunkan kepuasan pengguna. Kualitas sistem, informasi, dan layanan juga terbukti memperkuat kepuasan pengguna jika sistem dianggap bermanfaat dan mudah digunakan.

2. User Experience (UX):

UX mencakup seluruh emosi, persepsi, dan kepuasan pengguna selama berinteraksi dengan SIA. UX yang baik tidak hanya soal kemudahan, tetapi juga mencakup aspek estetika, konsistensi, kejelasan navigasi, dan responsivitas sistem. Desain UX yang

intuitif dan konsisten terbukti meningkatkan minat, keterlibatan, serta kepuasan pengguna di lingkungan akademik. Pengalaman positif akan mendorong pengguna untuk lebih aktif dan nyaman menggunakan sistem, sehingga mendukung keberhasilan implementasi SIA.

2.6 Sistem Manajemen Basis Data (PostgreSQL)

Basis data merupakan komponen inti dalam sistem informasi yang berfungsi sebagai repositori terpusat untuk menyimpan, memelihara, dan menyediakan data secara terstruktur [43]. Untuk menjamin integritas, konsistensi, dan keamanan data pada Sistem Informasi Akademik (SIA) berbasis *web*, penelitian ini mengimplementasikan *Relational Database Management System* (RDBMS). Model relasional menggunakan struktur tabel yang saling terhubung untuk merepresentasikan data master sekolah, sehingga dapat meminimalisir redundansi dan mencegah terjadinya anomali data melalui proses normalisasi yang tepat [44].

Platform RDBMS yang digunakan dalam pengerjaan tugas akhir ini adalah *PostgreSQL*, sebuah sistem manajemen basis data relasional tingkat lanjut yang bersifat *open-source*. Pemilihan *PostgreSQL* didasarkan pada skalabilitasnya yang tinggi, kinerja yang stabil dalam menangani banyak pengguna secara bersamaan, serta dukungannya yang optimal terhadap integritas data akademik yang kompleks (seperti relasi data guru, siswa, absensi, dan nilai) [45].

2.7 Teknologi Pengembangan Sistem

Dalam melakukan pengembangan Sistem Informasi Akademik pada Yayasan Methodist Indonesia Pangkalan Brandan, pemilihan teknologi yang tepat sangat menentukan performa, efisiensi, dan kemudahan dalam pemeliharaan sistem. Sistem usulan ini dibangun menggunakan dua teknologi utama, yaitu **Svelte** sebagai kerangka kerja di sisi antarmuka pengguna (*frontend*) dan **Spring Boot** sebagai pengolah logika bisnis di sisi server (*backend*).

Berikut adalah penjelasan teoritis dari kedua teknologi pengembangan sistem tersebut:

1. Svelte

Svelte (atau *Svelte.js*) adalah sebuah alat (*tool*) yang digunakan untuk membangun aplikasi web yang cepat. *Svelte* memiliki kemiripan dengan beberapa *framework JavaScript* populer lainnya seperti *React.js* dan *Vue.js*, di mana ketiganya berbagi tujuan

umum dalam memfasilitasi pembuatan antarmuka pengguna (*user interface*) yang mulus, dinamis, serta interaktif [46].

Meskipun memiliki tujuan yang sama, terdapat satu perbedaan kunci yang mendasar antara *Svelte* dengan *framework* modern lainnya. *Svelte* akan mengkompilasi aplikasi menjadi kode *JavaScript* yang optimal pada saat aplikasi dibangun (*build time*), bukan melakukan proses interpretasi kode aplikasi pada saat aplikasi dijalankan (*runtime*) di dalam browser pengguna [47].

2. *Spring Boot*

Spring Boot adalah sebuah *framework* berbasis *Java* yang menyederhanakan pembangunan aplikasi *Spring* melalui konfigurasi otomatis (*auto-configuration*) dan *embedded server*, sehingga aplikasi dapat langsung dijalankan tanpa konfigurasi manual yang kompleks [48]. *Spring Boot* umumnya digunakan untuk membangun layanan *backend* yang menyediakan *REST API*, yaitu antarmuka yang menjadi jembatan pertukaran data antara sisi *frontend* dan basis data [49]. Pada pengembangan sistem informasi akademik ini, aplikasi *backend* disusun menggunakan arsitektur berlapis yang terdiri dari komponen *controller*, *service*, *repository*, *model*, dan *Data Transfer Object* (DTO), sehingga alur data dan logika bisnis aplikasi terorganisir dengan jelas serta mudah dipelihara.

2.8 Analisis PIECES

Analisis PIECES merupakan sebuah metode atau kerangka kerja (*framework*) yang digunakan untuk mengevaluasi operasional sebuah sistem informasi, baik sistem yang sedang berjalan maupun sistem yang akan dikembangkan [49]. Kerangka kerja ini digunakan untuk mengidentifikasi masalah, peluang, hambatan, serta perintah yang terjadi di dalam organisasi. Evaluasi dalam metode ini dibagi ke dalam enam aspek penilaian atau variabel utama, yaitu [50]:

1. **Performance (Keandalan / Kinerja)** Variabel ini digunakan untuk mengetahui kinerja dari sebuah sistem, apakah telah berjalan dengan baik atau tidak. Pengukuran pada aspek ini berfokus pada jumlah data yang dihasilkan serta seberapa cepat suatu data dapat diproses oleh sistem dalam menyelesaikan tugas operasionalnya.
2. **Information and Data (Data dan Informasi)** Variabel ini digunakan untuk menganalisis kualitas data dan informasi yang dikelola. Evaluasi pada aspek ini meninjau seberapa banyak dan seberapa jelas informasi yang dihasilkan oleh sistem

dalam satu kali proses pencarian, sehingga informasi tersebut benar-benar akurat, relevan, dan bermanfaat bagi penggunanya.

3. **Economics (Nilai Ekonomis)** Variabel ini digunakan untuk mengetahui sistem dari segi finansial dan biaya operasional yang dikeluarkan. Evaluasi ini meninjau apakah biaya yang dikeluarkan untuk menjalankan sistem sebanding dengan nilai manfaat yang diperoleh, serta bagaimana sistem mampu mengoptimalkan pemanfaatan sumber daya organisasi secara ekonomis.
4. **Control and Security (Pengendalian dan Pengamanan)** Analisis ini digunakan untuk mengetahui pengawasan, kontrol, dan tingkat keamanan yang dilakukan oleh aplikasi. Pengendalian yang baik berfungsi untuk menjaga integritas data, mengatur pembatasan hak akses pengguna, serta mencegah adanya manipulasi data atau akses ilegal dari pihak yang tidak berwenang.
5. **Efficiency (Efisiensi)** Analisis ini dilakukan untuk mengetahui apakah suatu sistem itu efisien atau tidak dalam memanfaatkan sumber dayanya. Sistem yang dinilai efisien adalah sistem yang mampu meminimalisir pemborosan, di mana dengan input yang sedikit atau usaha yang minimal, sistem tetap bisa menghasilkan sebuah output yang memuaskan dan optimal.
6. **Service (Pelayanan)** Analisis ini digunakan untuk mengetahui bagaimana kualitas pelayanan yang dilakukan oleh sistem kepada para penggunanya. Melalui aspek ini, organisasi dapat mengetahui dan mengevaluasi permasalahan-permasalahan yang ada terkait tentang pelayanan, sehingga sistem yang dikembangkan dapat memberikan kemudahan dan kepuasan yang lebih baik bagi pengguna.

2.9 Black Box Testing

Black Box Testing adalah metode pengujian perangkat lunak yang berfokus pada kesesuaian antara *input* yang diberikan dan *output* yang dihasilkan sistem berdasarkan spesifikasi yang telah ditetapkan, tanpa mempertimbangkan kode program dan struktur internal sehingga memastikan bahwa sistem bekerja dari sudut pandang pengguna [51].

Metode ini memiliki kelebihan utama berupa kemudahan dalam penerapannya karena tidak memerlukan pemahaman teknis mengenai bahasa pemrograman atau arsitektur sistem [52]. Pengujian dilakukan berdasarkan kebutuhan dan fungsi yang telah ditentukan, sehingga efektif untuk menilai apakah sistem memenuhi ekspektasi pengguna akhir. *Black Box Testing* juga mampu mengidentifikasi berbagai kesalahan antarmuka pengguna, validasi data, serta tindakan sistem terhadap berbagai variasi inputan. Manfaat dari metode ini adalah

membantu meningkatkan kualitas perangkat lunak, memastikan keandalan fungsi, serta memverifikasi sistem sebelum digunakan oleh *end user* [53].

Beberapa teknik dalam melakukan black box testing antara lain:

1. *Equivalence Partitioning*

Teknik ini membagi data input menjadi kelompok yang diasumsikan menghasilkan output serupa, untuk mengurangi jumlah kasus uji tanpa mengurangi efektivitas [54].

2. *Boundary Value Analysis*

Teknik ini menguji nilai batas paling awal dan batas paling akhir dari input untuk memastikan aplikasi dapat berfungsi dengan benar saat memasukkan inputan paling kecil, paling besar atau mendekati batas tersebut [55].

3. *Cause-Effect Graphing*

Teknik ini menggambarkan hubungan antara sebab dan akibat ke dalam grafik untuk merancang kasus uji yang efisien, dengan mengidentifikasi kondisi tertentu dan hasil yang diharapkan [56].

4. *State Transition Testing*

Teknik ini menguji perilaku sistem berdasarkan perubahan status dan transisi antar status untuk memastikan setiap kondisi dan peralihannya berjalan sesuai aturan yang telah ditentukan [57].

5. *Fuzzing*

Teknik ini memberikan input acak, tidak valid, atau tidak terduga ke dalam sistem untuk menilai ketahanan aplikasi dan mengidentifikasi potensi kerentanan yang muncul akibat penanganan data yang tidak sesuai [58].

2.10 System Usability Scale

System Usability Scale (SUS) merupakan salah satu instrumen pengujian penerimaan sistem (*acceptance testing*) yang digunakan untuk mengukur persepsi kegunaan (*usability*) suatu aplikasi dari sudut pandang pengguna akhir. SUS merupakan kuesioner global yang andal dan efektif karena dapat diterapkan pada berbagai platform teknologi, termasuk Sistem Informasi Akademik berbasis *web*. Metode ini dipilih untuk mengevaluasi aspek kemudahan penggunaan, efisiensi sistem, dan tingkat kepuasan subjektif pengguna (seperti admin, guru, dan orang tua siswa) setelah berinteraksi dengan antarmuka aplikasi yang telah dibangun [59].

Tabel 2.3 Pernyataan dalam Kuesioner SUS [60]

No.	Pertanyaan	Skor
1	Saya pikir saya ingin menggunakan sistem ini lebih sering.	1-5
2	Saya menemukan sistemnya terlalu rumit.	1-5
3	Saya pikir sistem ini mudah digunakan.	1-5
4	Saya pikir saya memerlukan bantuan dari orang teknis dalam menggunakan sistem ini.	1-5
5	Saya menemukan berbagai fungsi dalam sistem ini terintegrasi dengan baik.	1-5
6	Saya pikir ada terlalu banyak ketidakkonsistenan dalam sistem ini.	1-5
7	Saya membayangkan bahwa kebanyakan orang akan belajar dengan mudah dalam menggunakan sistem ini.	1-5
8	Saya merasa sistem ini sangat tidak praktis.	1-5
9	Saya merasa sangat percaya diri dalam menggunakan sistem ini.	1-5
10	Saya perlu mempelajari banyak hal sebelum saya bisa menjalankan sistem ini.	1-5

Instrumen kuesioner SUS terdiri dari 10 butir pernyataan acak yang menggabungkan pernyataan bernada positif (nomor ganjil) dan negatif (nomor genap). Setiap pernyataan dinilai menggunakan skala Likert 5 poin, mulai dari "Sangat Tidak Setuju" hingga "Sangat Setuju". Hasil pengisian kuesioner oleh responden kemudian dihitung menggunakan rumus konversi khusus untuk menghasilkan skor akhir dengan rentang 0 hingga 100 [61].

Tabel 2.4 Skor *System Usability Scale* [62]

Skor SUS	Grade	Adjektif
> 81	A	<i>Excellent</i>
68 - 81	B	<i>Good</i>
68	C	<i>Fair</i>
51 - 67	D	<i>Poor</i>
< 51	F	<i>Worst</i>

Skor akhir dari pengujian SUS ini memberikan metrik evaluasi yang valid berupa skor SUS, nilai huruf (*grade*), dan kata sifat (adjektif). Melalui penerapan metode SUS pada

tugas akhir ini, peneliti dapat memperoleh data kuantitatif yang objektif untuk membuktikan apakah Sistem Informasi Akademik yang dikembangkan sudah layak dan mudah dioperasikan oleh para pengguna atau masih memerlukan perbaikan pada sisi antarmuka (*user interface*) [63].

2.11 Yayasan Methodist Indonesia Pangkalan Brandan

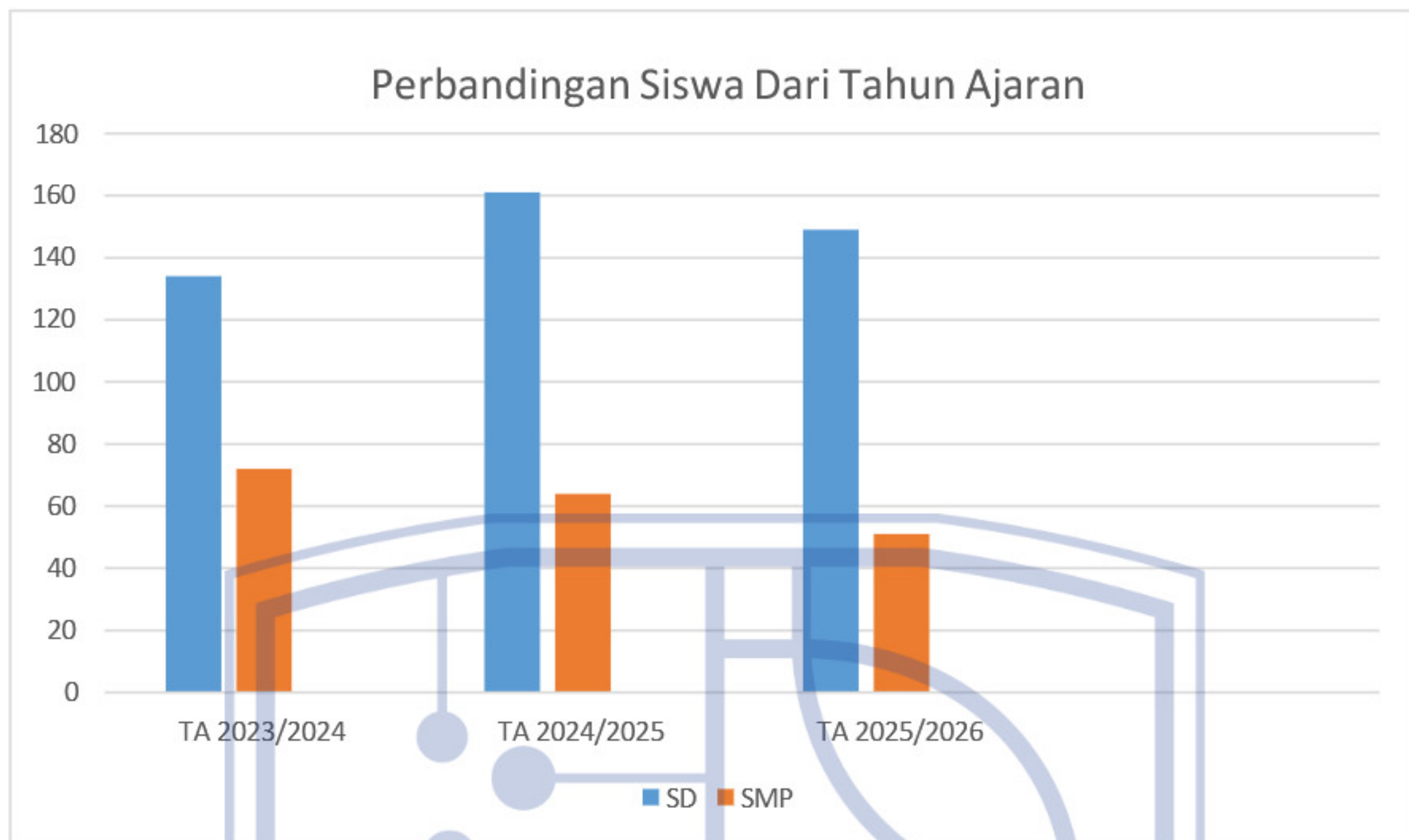


Gambar 2.1 Yayasan Methodist Indonesia Pangkalan Brandan

Yayasan Methodist Indonesia Pangkalan Brandan adalah lembaga pendidikan di bawah naungan Gereja Methodist Indonesia, yang mencakup jenjang SD dan SMP di wilayah Babalan, Kabupaten Langkat, Sumatera Utara, yaitu SD Methodist Babalan (Jalan Dempo) dan SMP Swasta Methodist Babalan. Yayasan Methodist Indonesia Pangkalan Brandan saat ini dipimpin oleh Suka Jaka. Jumlah siswa di Yayasan Methodist Indonesia Pangkalan Brandan pada T.A. 2024/2025 adalah 225 orang, yang terdiri dari 161 siswa jenjang SD dan 64 siswa jenjang SMP.

Adapun Visi dan Misi dari Yayasan Methodist Indonesia Pangkalan Brandan adalah sebagai berikut:

1. Menjadi lembaga pendidikan Kristen yang mencerdaskan kehidupan bangsa, bertakwa kepada Tuhan, penuh kasih sayang, dan berwawasan luas.
2. Menjadi lembaga pendidikan Kristen terbaik yang menghasilkan lulusan unggul dan berdaya saing.



Gambar 2.2 Grafik Perbandingan Siswa

Data menunjukkan bahwa jumlah siswa SD memiliki tren yang cukup dinamis dengan pertumbuhan yang signifikan di pertengahan periode. Pada Tahun Ajaran 2023/2024, jumlah siswa tercatat sebanyak 134 orang, yang kemudian meningkat tajam menjadi 161 siswa pada T.A. 2024/2025. Meskipun pada T.A. 2025/2026 terjadi penurunan menjadi 149 siswa, angka ini tetap berada di level yang lebih tinggi dibandingkan tahun pertama.

Berbeda dengan jenjang SD, jumlah siswa pada jenjang SMP menunjukkan penurunan yang konsisten selama tiga tahun ajaran terakhir secara berturut-turut. Dimulai dengan jumlah 72 siswa pada T.A. 2023/2024, angka tersebut perlahan turun menjadi 64 siswa pada tahun berikutnya, dan kembali menyusut hingga menyisakan 51 siswa pada T.A. 2025/2026.