

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem Informasi

Dalam merancang dan mengembangkan suatu aplikasi, pemahaman terhadap konsep dasar sistem informasi sangatlah penting. Sistem informasi merupakan kombinasi dari teknologi, manusia, dan prosedur yang dirancang untuk menghasilkan informasi yang berguna dalam pengambilan keputusan. Untuk memahami sistem informasi secara utuh, terlebih dahulu perlu dijelaskan mengenai pengertian sistem, informasi, serta karakteristik dan klasifikasi dari sistem itu sendiri. Konsep-konsep dasar ini menjadi landasan awal dalam membangun sistem informasi yang sesuai dengan kebutuhan pengguna dan tujuan organisasi.

2.1.1 Pengertian Sistem

Sistem adalah suatu kesatuan komponen atau elemen yang saling berinteraksi untuk mencapai tujuan tertentu. Sistem dapat berupa perangkat keras, perangkat lunak, atau kombinasi keduanya yang dirancang untuk melakukan tugas-tugas tertentu [1]. Selain itu juga, sistem bisa didefinisikan sebagai sekumpulan elemen yang saling berhubungan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. Selain itu, sistem dapat didefinisikan sebagai suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi, atau energi untuk mencapai suatu tujuan [2].

2.1.2 Pengertian Informasi

Informasi merupakan data yang telah diolah sedemikian rupa sehingga memiliki makna dan nilai bagi penerimanya, serta dapat digunakan dalam pengambilan keputusan. Menurut Hartono, informasi adalah hasil pengolahan data dalam bentuk yang lebih berarti bagi penerimanya, menggambarkan suatu kejadian nyata yang digunakan untuk pengambilan Keputusan [3]. Dalam konteks sistem informasi, informasi menjadi komponen vital yang menghubungkan berbagai subsistem dalam organisasi. Sistem informasi berfungsi mengumpulkan, menyimpan, memproses, dan mendistribusikan informasi ke seluruh tingkat organisasi, sehingga mendukung operasional dan pengambilan keputusan[4]. Di era digital saat ini, pengolahan data yang efektif menjadi kunci untuk menghasilkan informasi yang relevan dan akurat. Penggunaan teknologi seperti big data dan machine learning dapat meningkatkan kualitas informasi, membantu organisasi dalam menghadapi tantangan disinformasi, dan memastikan bahwa informasi yang digunakan dalam pengambilan keputusan adalah tepat dan dapat diandalkan[5].

2.1.3 Sistem Informasi

Sistem informasi (SI) adalah sebuah sistem yang dirancang untuk mengumpulkan, memproses, menyimpan, menganalisis, dan menyebarkan informasi yang dibutuhkan untuk mencapai tujuan tertentu dalam suatu organisasi. Sistem informasi terdiri dari berbagai komponen yang saling berinteraksi, seperti manusia (*brainware*), perangkat keras (*hardware*), perangkat lunak (*software*), jaringan komunikasi, serta sumber data yang diperlukan untuk mendukung pengambilan keputusan yang tepat [6]. Menurut [7], sistem informasi adalah sebuah sistem yang dirancang oleh manusia yang terdiri dari komponen-komponen dalam

organisasi untuk mencapai tujuan menyajikan informasi. Komponen-komponen ini bekerja sama untuk menghasilkan data yang dapat diolah menjadi informasi yang relevan dan akurat, mendukung pengambilan keputusan dalam organisasi, serta meningkatkan efisiensi dan efektivitas dalam berbagai bidang seperti pendidikan, akuntansi, dan penjualan [8]. Sistem informasi juga memerlukan pengembangan yang tepat, yang dapat dilakukan dengan berbagai metode pengembangan perangkat lunak, seperti *Waterfall*, *Extreme Programming*, dan *Agile*, untuk menghasilkan sistem informasi yang sesuai dengan kebutuhan organisasi. Metode-metode ini memiliki pendekatan yang berbeda dalam merancang dan mengembangkan sistem yang dapat mengoptimalkan fungsionalitas dan kinerja sistem informasi tersebut[9]. Sebagai contoh, sistem informasi dalam manajemen digunakan untuk meningkatkan efisiensi operasional dan membantu pengambilan keputusan yang lebih baik. Dalam konteks ini, sistem informasi memungkinkan pengelolaan data yang lebih efektif dan mempermudah distribusi informasi antar bagian dalam organisasi [10]. Penggunaan sistem informasi yang tepat dapat memberikan manfaat yang besar, baik dalam manajemen inventaris, absensi, maupun keuangan [11]. Secara keseluruhan, sistem informasi adalah komponen kunci dalam setiap organisasi yang ingin mengoptimalkan penggunaan informasi untuk tujuan operasional, analisis, dan pengambilan keputusan.

2.1.4 Karakteristik Sistem

Sistem adalah sekumpulan elemen yang saling berhubungan dan berinteraksi untuk mencapai suatu tujuan tertentu. Menurut [12], setiap sistem memiliki

karakteristik yang membedakannya dari sekedar kumpulan elemen yang tidak terorganisir. Karakteristik utama tersebut meliputi:

1. Komponen

Sistem terdiri dari berbagai elemen atau bagian yang saling berhubungan dan bekerja sama. Komponen ini dapat berupa elemen fisik seperti perangkat keras, elemen logis seperti perangkat lunak, atau bahkan elemen manusia dalam suatu organisasi. Semua komponen ini harus memiliki peran yang jelas agar sistem dapat berfungsi secara optimal.

2. Tujuan

Setiap sistem dirancang untuk mencapai tujuan tertentu. Tujuan inilah yang menjadi alasan keberadaan sistem. Misalnya, sebuah sistem informasi memiliki tujuan untuk mengelola, menyimpan, dan memproses data agar dapat digunakan secara efektif dalam pengambilan keputusan.

3. Batasan

Sistem memiliki batas yang memisahkan elemen internalnya dari elemen eksternal di lingkungannya. Batas ini penting untuk menentukan sejauh mana sistem dapat beroperasi dan berinteraksi dengan elemen di luar dirinya. Dengan kata lain, batasan ini menentukan ruang lingkup dan cakupan sistem.

4. Lingkungan

Lingkungan adalah segala sesuatu di luar batas sistem yang dapat memengaruhi operasionalnya. Lingkungan mencakup faktor eksternal seperti kebijakan, regulasi, teknologi, hingga tren sosial dan ekonomi yang

dapat memengaruhi kinerja sistem secara langsung maupun tidak langsung.

5. Masukan

Setiap sistem memerlukan masukan dari lingkungannya untuk dapat beroperasi. Masukan ini bisa berupa data, informasi, energi, atau sumber daya lain yang diperlukan oleh sistem untuk menjalankan proses internalnya.

6. Proses

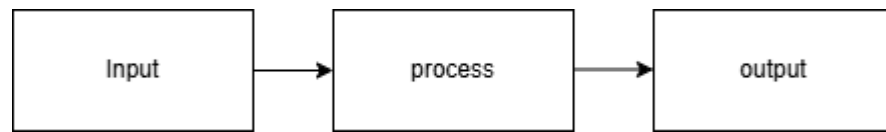
Proses adalah aktivitas atau serangkaian aktivitas yang dilakukan sistem untuk mengolah masukan menjadi keluaran. Proses ini mencerminkan cara kerja sistem untuk memenuhi tujuan yang telah ditentukan.

7. Keluaran

Keluaran adalah hasil akhir dari proses yang dilakukan oleh sistem. Keluaran ini kemudian dapat digunakan oleh pihak luar atau menjadi masukan bagi sistem lain dalam sebuah ekosistem yang lebih besar.

8. Umpan Balik

Sistem memanfaatkan umpan balik sebagai mekanisme evaluasi. Informasi yang dikembalikan ke sistem ini digunakan untuk menilai efektivitas dan efisiensi proses yang berlangsung. Umpan balik juga menjadi dasar untuk melakukan perbaikan dan peningkatan kinerja sistem secara berkelanjutan.



Gambar 2. 1 Karakteristik Sistem

Sumber : (Prehanto,2020) [12].

2.1.5 Klasifikasi Sistem

Klasifikasi sistem adalah pengelompokan sistem berdasarkan karakteristik tertentu yang membedakannya satu sama lain. Menurut literatur dan penelitian terbaru [12], klasifikasi sistem dapat dilakukan berdasarkan berbagai aspek, seperti sifat, tujuan, interaksi, hingga cara kerjanya. Adapun klasifikasi utama sistem adalah sebagai berikut:

1. Sistem Tertutup dan Sistem Terbuka

Sistem tertutup adalah sistem yang tidak berinteraksi dengan lingkungan eksternal. Sistem ini cenderung terisolasi, dan semua proses serta elemennya hanya berada dalam batasan sistem tersebut. Contohnya adalah sistem mekanis seperti jam analog. Sistem terbuka adalah sistem yang berinteraksi dengan lingkungan luar. Sistem ini menerima masukan dari lingkungan dan menghasilkan keluaran yang memengaruhi lingkungan. Contohnya adalah sistem organisasi atau ekosistem alami. Sistem terbuka cenderung lebih dinamis karena harus menyesuaikan diri dengan perubahan lingkungan.

2. Sistem Fisik dan Sistem Non-Fisik

Sistem fisik adalah sistem yang terdiri dari elemen-elemen nyata atau berwujud. Contohnya adalah perangkat keras komputer, kendaraan, dan

jaringan transportasi. Sistem non-fisik adalah sistem yang bersifat konseptual atau tidak berwujud. Contohnya adalah sistem informasi, sistem ekonomi, atau sistem sosial.

3. Sistem Alamiah dan Sistem Buatan

Sistem alamiah adalah sistem yang terbentuk secara alami tanpa campur tangan manusia. Contohnya adalah sistem tata surya, siklus hidrologi, atau ekosistem. Sistem buatan adalah sistem yang dirancang dan dibangun oleh manusia untuk tujuan tertentu. Contohnya adalah sistem komputer, sistem transportasi, dan sistem manajemen.

4. Sistem Sederhana dan Sistem Kompleks

Sistem sederhana memiliki struktur dan proses yang mudah dipahami dan dioperasikan. Contohnya adalah sistem mekanis seperti katrol atau roda gigi. Sistem kompleks memiliki banyak elemen yang saling terkait dan interaksi yang dinamis. Contohnya adalah sistem manajemen organisasi atau sistem ekonomi global.

5. Sistem Terkendali dan Sistem Tak Terkendali

Sistem terkendali adalah sistem yang operasinya dapat diatur dan dipantau oleh manusia atau mekanisme tertentu. Contohnya adalah sistem kontrol lalu lintas udara. Sistem tak terkendali adalah sistem yang berjalan secara alami tanpa adanya kontrol dari pihak tertentu. Contohnya adalah arus sungai atau perubahan cuaca.

6. Sistem Statis dan Sistem Dinamis

Sistem statis tidak mengalami perubahan atau hanya memiliki sedikit perubahan dalam waktu tertentu. Contohnya adalah sistem perpustakaan manual. Sistem dinamis terus mengalami perubahan dan berkembang seiring waktu. Contohnya adalah sistem informasi berbasis teknologi atau sistem pasar saham.

2.2 Pengembangan Sistem Informasi

Pengembangan sistem informasi merupakan proses penting dalam menciptakan sebuah sistem yang mampu memenuhi kebutuhan pengguna secara efektif dan efisien. Proses ini melibatkan serangkaian tahapan yang dikenal sebagai Siklus Hidup Pengembangan Sistem atau System Development Life Cycle (SDLC), yang mencakup analisis kebutuhan, perancangan, implementasi, hingga pemeliharaan sistem. Dalam implementasinya, terdapat berbagai model pengembangan sistem, salah satunya adalah metode Agile yang bersifat iteratif dan adaptif terhadap perubahan. Selain itu, proses pengembangan sistem informasi juga didukung oleh metode penelitian seperti studi pustaka, wawancara, dan observasi untuk mengumpulkan informasi yang relevan. Pemahaman terhadap tahapan dan metode ini sangat penting agar sistem yang dikembangkan dapat berjalan sesuai harapan dan dapat terus disesuaikan dengan kebutuhan pengguna.

2.2.1 Siklus Hidup Perancangan Sistem (*System Deveploment Life Cycle*)

SDLC adalah sebuah kerangka kerja sistematis yang digunakan untuk mengembangkan atau memodifikasi sistem informasi dan perangkat lunak. SDLC mencakup tahapan-tahapan tertentu yang dimulai dari perencanaan hingga

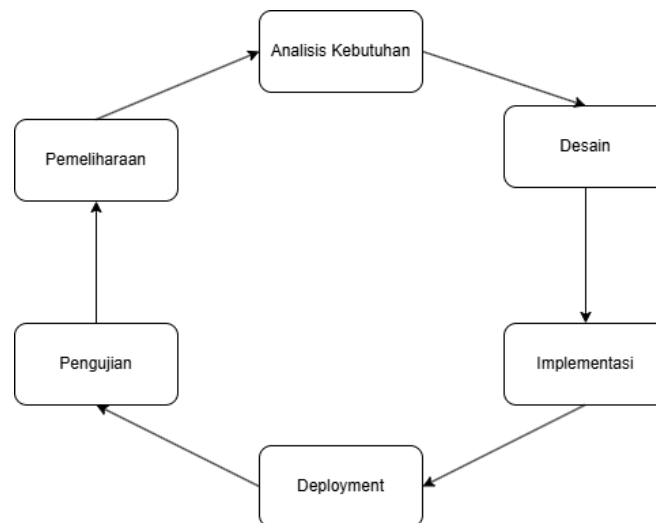
pemeliharaan, memastikan proses pengembangan berjalan secara terstruktur dan efisien.

2.2.2 Definisi dan Konsep SDLC

SDLC dapat didefinisikan sebagai proses sistematis dalam mengembangkan sistem informasi atau perangkat lunak dengan menggunakan pendekatan tertentu. Menurut [13], SDLC adalah metode yang digunakan individu untuk membangun sistem perangkat lunak dengan pendekatan terstruktur, yang bertujuan untuk memenuhi kebutuhan pengguna secara optimal.

1. Analisis Kebutuhan, Tahap ini merupakan langkah awal, di mana tim pengembang mengumpulkan dan menganalisis kebutuhan dari pengguna atau stakeholder. Proses ini melibatkan wawancara, survei, dan studi dokumen yang berkaitan dengan sistem yang akan dikembangkan. Hasil dari tahap ini adalah dokumen spesifikasi kebutuhan perangkat lunak yang akan menjadi acuan dalam tahap selanjutnya.
2. Desain, Setelah kebutuhan sistem ditentukan, tahap desain dilakukan untuk merancang struktur sistem secara lebih detail. Desain ini mencakup arsitektur sistem, desain database, antarmuka pengguna, serta alur proses yang akan diterapkan dalam perangkat lunak. Tujuan dari tahap ini adalah memastikan bahwa sistem dapat dikembangkan secara efisien dengan mempertimbangkan faktor keamanan, skalabilitas, dan performa.
3. Implementasi Pada tahap implementasi, dilakukan pengkodean sesuai dengan desain yang telah disusun. Selain itu, pengujian dan integrasi komponen sistem juga dilakukan untuk memastikan semua bagian berfungsi dengan baik.

4. Deployment Setelah implementasi selesai, sistem yang telah dikembangkan diinstal dan dioperasikan di lingkungan pengguna. Proses ini melibatkan pelatihan pengguna, konfigurasi sistem, dan pengawasan operasional awal.
5. Pengujian Pada tahapan ini dilakukan pengujian perangkat lunak untuk mengetahui perangkat yang dirancang berfungsi dengan baik
6. Pemeliharaan Pemeliharaan adalah tahap akhir dalam SDLC, yang mencakup perbaikan bug, pembaruan sistem, serta penyesuaian terhadap kebutuhan baru. Tahap ini penting untuk memastikan sistem tetap relevan dan dapat digunakan secara optimal.



Gambar 2. 2 Siklus Hidup Perancangan Sistem (SDLC)

Sumber : (Perdana, 2021) [14].

2.2.3 Model-Model SDLC

1. Waterfall Model ini bersifat linear dan berurutan, di mana setiap tahap harus selesai sebelum tahap berikutnya dimulai.
2. Spiral Menggabungkan elemen berulang dengan pendekatan linear untuk mengelola risiko secara lebih efektif.

3. Agile Berfokus pada pengembangan iteratif dan kolaborasi yang erat antara tim pengembang dan pengguna [15].

2.2.4 Keamanan dalam SDLC

Aspek keamanan menjadi perhatian penting dalam SDLC. Dengan mengadopsi *Secure Software Development Life Cycle* (S-SDLC), pengembang dapat mengurangi kerentanan sistem dan meningkatkan keamanannya [16]. Pendekatan ini mencakup integrasi proses keamanan pada setiap tahap pengembangan.

2.2.5 Metode Agile

Metode Agile adalah salah satu metodologi pengembangan sistem informasi yang berfokus pada fleksibilitas, kolaborasi, dan adaptasi terhadap perubahan kebutuhan pengguna. Agile menekankan proses iteratif dan adaptif, memungkinkan pengembangan sistem dilakukan secara bertahap dan responsif terhadap perubahan kebutuhan pengguna [17]. Pendekatan ini berbeda dengan metode tradisional seperti Waterfall, yang cenderung bersifat linear dan tidak fleksibel terhadap perubahan [18]. Menurut [19], metode Agile memberikan kerangka kerja yang fleksibel untuk pengembangan sistem informasi dengan memprioritaskan kolaborasi antara tim pengembang, pengguna, dan pemangku kepentingan lainnya. Selain itu, Agile memungkinkan pengembangan sistem dilakukan dalam siklus pendek (*sprint*), sehingga dapat menghasilkan produk yang dapat digunakan lebih cepat [17].



Gambar 2. 3 Tahapan Metode Agile

Sumber : (Handayani, 2023) [20].

2.2.6 Web

Web, atau *World Wide Web* (WWW), adalah sistem yang memungkinkan akses dan pertukaran informasi melalui internet. Menurut [21], web merupakan bagian penting dalam internet yang memungkinkan pengguna mengakses berbagai informasi dalam bentuk teks, gambar, video, dan lainnya. Sebuah website terdiri dari kumpulan halaman yang saling berhubungan dan dapat diakses melalui internet menggunakan *web browser*. Halaman-halaman ini berisi konten yang disajikan dalam berbagai format, seperti teks, gambar, audio, dan video, yang disimpan di dalam sebuah *server* atau *hosting*. Menurut [21], pengembangan sistem informasi sekolah berbasis web dapat meningkatkan efisiensi dalam penyampaian informasi. *Website* memiliki peran penting dalam berbagai aspek kehidupan, termasuk pendidikan, bisnis, dan komunikasi. Dalam konteks pendidikan, *website* berfungsi sebagai media untuk menampilkan profil sekolah, mempromosikan program atau layanan, dan berinteraksi dengan siswa serta orang tua. Perancangan sistem informasi manajemen ekstrakurikuler berbasis web dapat mempermudah pengelolaan kegiatan ekstrakurikuler di sekolah.

2.2.7 PHP

PHP (*Hypertext Preprocessor*) adalah bahasa pemrograman *server-side* yang dirancang khusus untuk pengembangan web. PHP memungkinkan pengembang membuat halaman web dinamis yang dapat berinteraksi dengan basis data dan pengguna secara *real-time*. Menurut [22], dengan menggabungkan bahasa pemrograman PHP dan MySQL, kita dapat membangun *website* yang menyimpan dan menampilkan data secara terstruktur, seperti daftar produk, profil pengguna, atau artikel berita, serta menyediakan interaksi dengan pengguna, seperti formulir. PHP merupakan bahasa pemrograman yang menyatu dengan HTML, memungkinkan pembuatan halaman web yang dinamis. Salah satu fungsinya adalah untuk menerima, mengolah, dan menampilkan data ke sebuah situs. Data yang diterima akan diolah di sebuah program basis data server, kemudian hasilnya ditampilkan kembali ke layar *browser* sebuah situs. PHP adalah bahasa pemrograman yang dijalankan melalui halaman web dan umumnya digunakan untuk mengolah informasi di internet. PHP merupakan *script* yang menyatu dengan HTML dan berada pada server. Salah satu kelebihan PHP adalah kemudahannya untuk berintegrasi dengan berbagai basis data, seperti MySQL. Hal ini memungkinkan pengembangan aplikasi web yang kompleks dengan lebih efisien. Menurut [23], PHP digunakan dalam perancangan web sistem monitoring stop kontak terkontrol, yang menunjukkan fleksibilitas dan kemampuan PHP dalam pengembangan aplikasi *Internet of Things (IoT)*. Selain itu, PHP didukung oleh berbagai framework yang memudahkan pengembangan aplikasi web, seperti CodeIgniter. CodeIgniter adalah sebuah framework PHP yang dapat membantu mempercepat developer dalam pengembangan aplikasi web berbasis PHP

dibanding jika menulis semua kode program dari awal. Dengan berbagai kelebihan dan dukungan komunitas yang luas, PHP tetap menjadi salah satu bahasa pemrograman yang populer dan andal dalam pengembangan aplikasi web hingga saat ini.

2.2.8 XAMPP

XAMPP adalah perangkat lunak *open-source* yang menyediakan paket lengkap untuk pengembangan web secara lokal. Nama XAMPP merupakan akronim dari "*Cross-Platform* (X), Apache (A), MariaDB (M), PHP (P), dan Perl (P)". Dengan XAMPP, pengembang dapat menjalankan server web Apache, basis data MariaDB, serta mendukung skrip PHP dan Perl pada komputer pribadi tanpa perlu koneksi internet. Hal ini memudahkan proses pengembangan, pengujian, dan debugging aplikasi web sebelum diterapkan pada server sebenarnya. XAMPP kompatibel dengan berbagai sistem operasi, seperti Windows, Linux, dan macOS, sehingga fleksibel digunakan oleh pengembang dengan platform yang berbeda. Selain itu, XAMPP dilengkapi dengan phpMyAdmin, sebuah alat berbasis web yang memudahkan pengelolaan basis data MariaDB atau MySQL. Dengan antarmuka yang ramah pengguna, phpMyAdmin memungkinkan pengembang untuk membuat, mengubah, dan menghapus basis data, tabel, serta menjalankan perintah SQL dengan mudah.

2.2.9 MySQL

MySQL adalah sistem manajemen basis data relasional (RDBMS) yang banyak digunakan dalam pengembangan aplikasi web. Sebagai perangkat lunak *open-source*, MySQL memungkinkan pengguna untuk menyimpan, mengelola, dan

mengambil data secara efisien. Menurut [24], MySQL menawarkan kecepatan tinggi, stabilitas, dan fleksibilitas dalam integrasi dengan berbagai bahasa pemrograman seperti PHP, PERL, C, C++, dan JAVA. Selain itu, MySQL mendukung berbagai sistem operasi dan dapat menangani data dalam jumlah besar dengan performa yang baik. Keunggulan lain dari MySQL adalah kemampuannya dalam mendukung transaksi dan menyediakan fitur keamanan yang baik. Hal ini menjadikannya pilihan populer bagi pengembang yang membutuhkan sistem basis data yang handal dan efisien dalam pengelolaan data. Menurut [24], MySQL juga menyediakan fitur seperti indeks, integritas referensial, dan kemampuan untuk menangani data dalam jumlah besar, yang sangat penting dalam pengembangan aplikasi web yang kompleks.

2.3 Alat Bantu Perancangan Sistem

2.3.1 UML (*Unified Modeling Language*)

UML (*Unified Modeling Language*) adalah bahasa pemodelan standar yang digunakan untuk mendokumentasikan, memvisualisasikan, dan memvalidasi sistem perangkat lunak, khususnya dalam konteks pemrograman berorientasi objek. UML memungkinkan pengembang untuk memodelkan sistem secara grafis sehingga mempermudah pemahaman dan komunikasi antar tim pengembang [28].

UML terdiri dari berbagai diagram yang terbagi dalam tiga kategori utama:

1. Diagram Struktur Digunakan untuk menggambarkan struktur statis dari sistem, seperti *Class Diagram*, *Object Diagram*, *Deployment Diagram* dan *Component Diagram*[25].

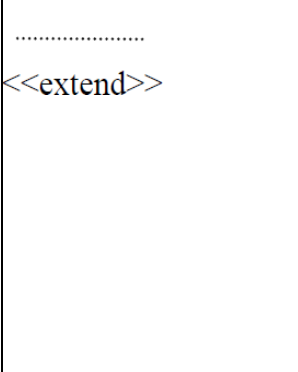
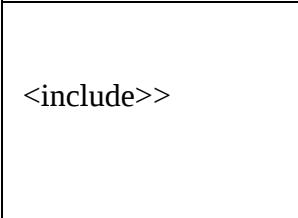
2. Diagram Perilaku Digunakan untuk memodelkan dinamika sistem, seperti *Use Case Diagram*, *Activity Diagram*, dan *State Diagram* [26].
3. Diagram Interaksi Berfokus pada interaksi antar objek dalam sistem, seperti *Sequence Diagram* dan *Communication Diagram* [27].

2.3.2 Use Case Diagram

Use case diagram adalah salah satu diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan interaksi antara pengguna (aktor) dengan sistem yang akan dibuat. Diagram ini memberikan representasi visual dari fungsi-fungsi sistem dan hubungan antara pengguna dengan fungsi tersebut [28]. Dengan *use case diagram*, pengembang dapat memahami kebutuhan sistem dari perspektif pengguna, sehingga sistem yang dikembangkan dapat sesuai dengan kebutuhan tersebut [29].

Table 2. 1 Simbol - simbol Use Case Diagram

Simbol	Nama	Deskripsi
 Actor	<i>Actor</i>	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi akan dibuat di luar sistem informasi yang akan dihasilkan dengan sendirinya, karena walaupun simbol aktor adalah gambaran seseorang, namun pelaku belum tentu adalah orang, biasanya diungkapkan dengan kata benda di awal nama aktor.
	<i>Use Case</i>	Fungsionalitas yang disediakan oleh sistem ketika entitas bertukar pesan antar entitas atau aktor biasanya diungkapkan dengan kata kerja di awal frase nama <i>use case</i> .
	<i>Assosiation</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi di dalam <i>use case</i> memiliki interaksi dengan aktor.

	<i>Extend</i>	Komunikasi antara aktor dan <i>use case</i> yang berpartisipasi dalam <i>use case</i> memiliki interaksi dengan aktor. Hubungan antara <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan bisa berdiri sendiri tanpa <i>use case</i> tambahan: mirip dengan prinsip pewarisan dalam pemrograman berorientasi objek objek: secara umum, <i>use case</i> tambahan memiliki nama yang sama dengan <i>use case</i> yang ditambahkan.
	<i>Use Case generalization</i>	Relasi antara generalisasi dan spesialisasi antara dua <i>use case</i> dimana satu fungsi lebih umum dibandingkan fungsi lainnya.
	<i>Include</i>	Hubungan antara <i>use case</i> tambahan dengan <i>use case</i> dimana <i>use case</i> yang ditambahkan tersebut memerlukan <i>use case</i> untuk memenuhi fungsinya atau sebagai syarat untuk dijalankan <i>use case</i> ini.

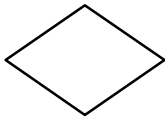
Sumber : (Suharni, dkk, 2023) [30].

2.3.3 Activity Diagram

Activity diagram adalah diagram yang menggambarkan alur aktivitas atau *workflow* dalam suatu sistem. Diagram ini mendeskripsikan bagaimana aktivitas-aktivitas dalam sistem berjalan dan saling berinteraksi [31].

Table 2. 2 Simbol – simbol Activity Diagram

Simbol	Nama	Deskripsi
	Status Awal	Menunjukkan dimana alir sistem kerja dimulai.
	Status Akhir	Menunjukkan dimana aliran sistem kerja berakhir.
	Action	Langkah-langkah suatu aktivitas sistem yang dibuat.

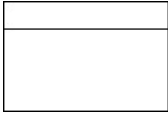
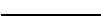
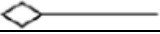
	Decision	Menunjukkan dimana keputusan akan diambil
	Swimlane	Pengelompokan aktivitas berdasarkan aktor.

Sumber : (A. Riyanto Pratama, 2023) [32].

2.3.4 Class Diagram

Class diagram adalah diagram yang menggambarkan struktur dan hubungan antar kelas dalam suatu sistem. Diagram ini mendeskripsikan atribut dan metode yang dimiliki oleh setiap kelas serta relasi antar kelas [28].

Table 2. 3 Simbol - simbol Class Diagram

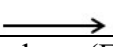
Simbol	Nama	Deskripsi
	<i>Class</i>	Kumpulan objek dengan atribut berbeda yang mempunyai fungsi yang sama.
	<i>Association</i>	Hubungan antar kelas yang mempunyai arti umum dan biasanya bersifat jamak.
	<i>Directed Association</i>	Hubungan antar kelas dengan arti satu kelas digunakan oleh kelas lain.
	<i>Aggregation</i>	Menunjukkan bahwa seluruh bagian dari <i>relationship</i> disebut hubungan.
	<i>Composition</i>	Hubungan <i>Composition</i> dengan class tempat dia bergantung.
	<i>Dependency</i>	Menunjukkan operasi pada suatu kelas yang menggunakan kelas lain.

Sumber : (F. Farhan, dkk., 2023) [33].

2.3.5 Sequence Diagram

Sequence diagram adalah diagram yang menunjukkan atau memperlihatkan interaksi antar objek dalam suatu sistem yang tersusun dalam suatu urutan atau rangkaian waktu. Interaksi antara objek ini termasuk pengguna, tampilan, dan dalam bentuk pesan. Berikut ini adalah simbol – simbol yang ada pada *sequence diagram*:

Table 2. 4 Simbol - simbol Sequence Diagram

No.	Simbol	Nama	Keterangan
1		<i>Actor</i>	Menunjukkan orang-orang yang berinteraksi dengan sistem.
2		<i>Entity Class</i>	Menjelaskan hubungan yang ingin dicapai.
3		<i>Boundary Class</i>	Menjelaskan gambaran dari font.
4		<i>Control Class</i>	Menjelaskan hubungan antara batas dan tabel.
5		<i>A focus of Control & A Life Line</i>	Menjelaskan dimana objek dimulai dan diakhirinya sebuah message
6.		<i>Message</i>	Menjelaskan pengiriman pesan.

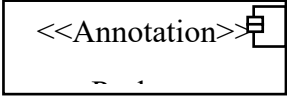
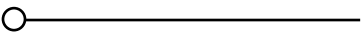
Sumber : (E. S. Harefa, dkk., 2024) [34].

2.3.6 Component Diagram

Component Diagram adalah salah satu jenis diagram dalam *Unified Modeling Language (UML)* yang digunakan untuk menggambarkan struktur sistem perangkat lunak dalam bentuk komponen-komponen independen beserta hubungan di antara mereka. Diagram ini berfungsi untuk memvisualisasikan bagaimana berbagai bagian sistem berinteraksi, termasuk dependensi antar komponen,

antarmuka yang digunakan, serta bagaimana komponen-komponen tersebut saling terhubung untuk membentuk sistem yang utuh. Dengan menggunakan *Component Diagram*, pengembang dapat memahami arsitektur perangkat lunak secara lebih jelas, memudahkan proses pemeliharaan, serta meningkatkan modularitas sistem. Berikut ini adalah notasi-notasi yang digunakan dalam *Component Diagram*.

Table 2. 5 Simbol - simbol Deployment Diagram

No	Simbol	Definisi
1		<i>Package</i> adalah sebuah tempat untuk membungkus satu komponen atau lebih.
2		<i>Component</i> adalah komponen sistem
3		<i>Interface</i> adalah penghubung antar tiap komponen seperti interface yang terdapat dalam konsep pengembangan sistem berbasis objek

Sumber : (Irianto, 2021) [35].

2.4 Ekstrakurikuler

Kegiatan ekstrakurikuler adalah aktivitas pendidikan yang diselenggarakan di luar jam pelajaran formal, bertujuan untuk membantu pengembangan siswa sesuai dengan kebutuhan, potensi, bakat, dan minat mereka. Menurut [36], kegiatan ini dibimbing oleh guru pembina atau tenaga kependidikan yang memiliki kemampuan dan kewenangan di bidangnya, serta diselenggarakan secara berkala dan terprogram.

Pengelolaan kegiatan ekstrakurikuler melibatkan beberapa tahapan penting, yaitu perencanaan, pengorganisasian, pelaksanaan, dan evaluasi. Perencanaan mencakup penentuan jenis kegiatan yang sesuai dengan minat siswa, pengaturan jadwal, serta penyiapan sarana dan prasarana yang dibutuhkan. Pengorganisasian melibatkan pembentukan struktur kepengurusan, pembagian tugas, dan penunjukan pembina untuk setiap kegiatan. Pelaksanaan dilakukan sesuai dengan rencana yang telah ditetapkan, dengan pengawasan dari pembina untuk memastikan kegiatan berjalan lancar. Evaluasi dilakukan secara berkala untuk menilai efektivitas kegiatan dan perkembangan siswa.

Namun, dalam praktiknya, pengelolaan kegiatan ekstrakurikuler sering menghadapi kendala seperti kurangnya koordinasi dan kerja sama antara kepala sekolah dengan pembina ekstrakurikuler. Untuk mengatasi hambatan tersebut, disarankan diadakan rapat evaluasi tiap akhir tahun guna menilai hasil kinerja serta memprogram kegiatan untuk tahun ajaran baru. Kegiatan ini harus diikuti oleh seluruh pemangku kepentingan sekolah.

Selain itu, pengelolaan kegiatan ekstrakurikuler yang efektif tidak hanya mendukung keberhasilan program intrakurikuler, tetapi juga berkontribusi pada pembentukan karakter siswa. Melalui partisipasi dalam berbagai kegiatan, siswa dapat mengembangkan keterampilan sosial, kepemimpinan, dan tanggung jawab [37]. Dengan demikian, pengelolaan kegiatan ekstrakurikuler yang baik di institusi pendidikan diharapkan dapat memberikan manfaat signifikan bagi pengembangan kompetensi dan karakter siswa, serta mendukung pencapaian tujuan pendidikan secara keseluruhan.

2.5 Metode Penelitian

Metode penelitian merupakan serangkaian langkah sistematis yang digunakan untuk mengumpulkan, menganalisis, dan menyajikan data yang relevan guna menjawab permasalahan penelitian. Penelitian ini menggunakan tiga metode utama, yaitu studi pustaka, wawancara, dan observasi, untuk mendapatkan informasi yang mendalam terkait perancangan sistem informasi pengelolaan ekstrakurikuler berbasis metode Agile [38]. Berikut adalah penjelasan dari setiap metode yang digunakan

2.5.1 Studi Pustaka

Studi pustaka dilakukan untuk mengumpulkan berbagai referensi dari buku, jurnal ilmiah, artikel, dan dokumen lain yang berkaitan dengan sistem informasi, metode Agile, dan pengelolaan ekstrakurikuler. Langkah ini bertujuan untuk memperkuat dasar teori dan memahami pendekatan yang sesuai dalam mendesain sistem informasi yang efektif. Referensi yang digunakan dalam studi pustaka berasal dari literatur terkini, khususnya yang relevan dengan teknologi dan manajemen ekstrakurikuler [39].

2.5.2 Wawancara

Wawancara dilakukan dengan pihak-pihak terkait, seperti kepala sekolah, pembina ekstrakurikuler, dan siswa, untuk mengidentifikasi kebutuhan dan permasalahan yang dihadapi dalam pengelolaan kegiatan ekstrakurikuler. Informasi yang diperoleh dari wawancara digunakan sebagai dasar dalam merancang fitur-fitur sistem agar sesuai dengan kebutuhan pengguna [40].

2.5.3 Observasi

Observasi dilakukan dengan mengamati langsung proses pengelolaan ekstrakurikuler di MTs Islamiyah Londut. Metode ini bertujuan untuk memahami alur kerja, kendala, dan kebutuhan yang muncul dalam pengelolaan kegiatan ekstrakurikuler. Data dari observasi membantu dalam memvalidasi temuan dari wawancara dan memberikan gambaran nyata yang diperlukan dalam pengembangan sistem informasi [40].

2.6 Penelitian Terdahulu

Berikut ini tabel yang menyajikan penelitian terdahulu yang relevan :

Table 2. 6 Penelitian Terdahulu

No	Peneliti	Judul	Hasil	Metode
1	Hani Handayani Dkk (2023)	Perancangan sistem informasi inventory barang berbasis web menggunakan metode agile software development	Sistem informasi inventory berbasis Agile dirancang untuk Toko Azura Pekanbaru guna mempermudah pengelolaan barang masuk dan keluar, mengatasi ketidaksesuaian data, dan meningkatkan efisiensi operasional.	Menggunakan metode <i>Agile</i>
2	Fahri Fauzi Hidayat (2021)	Rancang bangun sistem informasi Ekstrakurikuler (SIMAK) berbasis web.	Pengembangan dan implementasi aplikasi Sistem Informasi Manajemen Ekstrakurikuler (SIMAK) berbasis web. Fokus penelitian ini adalah untuk meningkatkan efisiensi dan	Menggunakan metode <i>Agile</i>

			transparansi dalam pengelolaan kegiatan ekstrakurikuler di sekolah.	
3	Muhammad Prayoga Indra Surya, Heri Kurniawan (2024)	Rancang Bangun Sistem Informasi Sekolah Berbasis Web Dengan Menggunakan Metode Agile Pada Sdn 056001 Karang Rejo	pengembangan sistem informasi sekolah berbasis web di SDN 056001 Karang Rejo. Fokus penelitian ini adalah merancang dan membangun sebuah website sekolah menggunakan metode Agile untuk meningkatkan efisiensi dan kecepatan dalam penyampaian informasi.	Menggunakan metode <i>Agile</i>
4	Muhammad Fariz Iqbal, Hanson Prihantoro Putro (2021)	Penerapan Simple Agile Methodology dalam Pengembangan Aplikasi Web	Sistem Informasi Magang diciptakan untuk kemudahan mahasiswa dalam mencari perusahaan yang membuka lowongan magang.	Menggunakan metode <i>Agile</i>