

BAB II

LANDASAN TEORI

2.1. Konsep dasar sistem informasi

2.1.1. Konsep Sistem

Sistem dapat dipahami sebagai suatu kesatuan yang terdiri atas berbagai komponen atau elemen yang saling berinteraksi dan bekerja secara terkoordinasi untuk mencapai tujuan tertentu. Setiap sistem memiliki elemen dasar yang meliputi masukan (*input*), proses (*process*), dan keluaran (*output*) yang membentuk suatu alur kerja yang terstruktur. Keterpaduan antar elemen tersebut memungkinkan sistem menjalankan fungsi operasional secara berkelanjutan dan menghasilkan keluaran yang sesuai dengan tujuan yang telah ditetapkan [1].

Selain elemen dasar tersebut, suatu sistem juga memiliki batasan (*boundary*) yang berfungsi untuk membedakan bagian internal sistem dengan lingkungan eksternal yang memengaruhinya. Sistem yang berjalan secara efektif umumnya dilengkapi dengan mekanisme umpan balik (*feedback*) yang berperan sebagai alat pengendalian dan evaluasi kinerja sistem. Melalui umpan balik, sistem dapat melakukan penyesuaian terhadap perubahan lingkungan serta melakukan koreksi apabila hasil yang diperoleh belum sesuai dengan target yang diharapkan [2]. Keberadaan mekanisme ini memungkinkan sistem menjaga kestabilan dan meningkatkan kualitas proses yang dijalankan.

Dalam konteks organisasi publik, konsep sistem memiliki peran yang sangat penting karena keterpaduan antarbagian akan menentukan kelancaran aliran data dan proses kerja. Sistem yang terstruktur dengan baik mampu mendukung aktivitas pencatatan, pengolahan, dan penyajian data secara konsisten dan sistematis. Sebaliknya, sistem yang tidak terorganisasi secara optimal berpotensi menimbulkan permasalahan seperti ketidaksesuaian data, duplikasi pekerjaan, serta penurunan kualitas informasi yang dihasilkan. Oleh karena itu, pemahaman terhadap konsep sistem menjadi landasan utama dalam perancangan sistem informasi yang mampu mendukung kebutuhan operasional dan pengambilan keputusan organisasi secara efektif [3].

2.1.2. Konsep Sistem Informasi

Informasi merupakan hasil dari proses pengolahan data sehingga memiliki makna dan nilai guna bagi penerimanya. Data yang masih bersifat mentah akan diolah melalui suatu proses tertentu agar dapat disajikan dalam bentuk informasi yang relevan untuk mendukung pemahaman terhadap suatu kondisi atau permasalahan [3]. Dengan demikian, informasi berfungsi sebagai sarana untuk mengurangi ketidakpastian dalam pengambilan keputusan.

Agar dapat dimanfaatkan secara optimal, informasi harus memenuhi karakteristik tertentu. Informasi yang baik harus bersifat akurat, relevan, lengkap, dan tepat waktu. Akurasi menunjukkan tingkat kebenaran informasi, relevansi berkaitan dengan kesesuaian informasi terhadap kebutuhan pengguna, kelengkapan mencerminkan cakupan informasi yang disajikan, sedangkan ketepatan waktu berkaitan dengan ketersediaan informasi pada saat dibutuhkan [3].

Kualitas informasi yang rendah dapat menyebabkan kesalahan interpretasi dan berdampak pada keputusan yang diambil. Dalam organisasi, khususnya organisasi publik, informasi memiliki peran strategis karena menjadi dasar dalam perencanaan, pengendalian, dan evaluasi kegiatan. Informasi yang berkualitas memungkinkan pimpinan dan pengelola organisasi memahami kondisi aktual, mengidentifikasi permasalahan, serta menentukan langkah yang tepat secara rasional. Oleh karena itu, pengelolaan informasi yang baik menjadi faktor penting dalam mendukung efektivitas dan akuntabilitas organisasi [1].

2.1.3. Konsep Sistem Informasi

Sistem informasi digunakan sebagai media pendukung dalam menjalankan aktivitas operasional organisasi melalui pengelolaan data yang tersusun secara terintegrasi. Dengan adanya sistem ini, proses pencatatan, pengolahan, hingga penyimpanan data dapat dilakukan dengan lebih terorganisir, sehingga menghasilkan informasi yang lebih tepat dan mudah diakses oleh pengguna. Kondisi tersebut berperan penting dalam menjaga konsistensi data sekaligus meningkatkan efisiensi dalam pelaksanaan pekerjaan [2].

Selain menunjang aktivitas operasional, sistem informasi juga memiliki peran penting dalam fungsi pengendalian serta pengambilan keputusan. Informasi yang dihasilkan dapat dimanfaatkan sebagai bahan evaluasi terhadap kinerja organisasi dan menjadi dasar dalam penyusunan rencana kegiatan berikutnya. Ketersediaan informasi yang relevan dan tepat waktu memungkinkan manajemen untuk menentukan keputusan secara lebih cepat dan akurat sesuai dengan kebutuhan organisasi [1].

Dalam lingkungan instansi pemerintah, penerapan sistem informasi menjadi salah satu faktor penting dalam meningkatkan kualitas tata kelola serta pelayanan kepada masyarakat. Pemanfaatan sistem informasi yang optimal dapat mendorong terciptanya transparansi, akuntabilitas, dan koordinasi yang baik antar unit kerja. Oleh karena itu, sistem informasi tidak hanya berfungsi sebagai alat pengelolaan data, tetapi juga berperan sebagai pendukung utama dalam meningkatkan kinerja serta efektivitas penyelenggaraan pemerintahan [3].

2.2. Alat Bantu Perancangan Sistem

2.2.1. UML (Unified Modeling Language)

Unified Modeling Language (UML) merupakan bahasa pemodelan standar yang digunakan untuk menggambarkan, memvisualisasikan, dan mendokumentasikan struktur serta perilaku suatu sistem informasi. UML menyediakan seperangkat notasi grafis yang membantu pengembang dalam memahami keterkaitan antarkomponen sistem serta memetakan kebutuhan perangkat lunak secara konseptual sebelum proses implementasi dilakukan [4].

UML memiliki fleksibilitas yang tinggi dan dapat diterapkan pada berbagai metodologi pengembangan perangkat lunak, baik model sekuensial seperti Waterfall maupun pendekatan iteratif seperti Agile. UML tidak bergantung pada metode pengembangan tertentu, tetapi berfokus pada penyajian model sistem secara visual agar lebih mudah dipahami dan dianalisis [5]. Melalui berbagai diagram seperti use case, activity, class, sequence, dan component, UML membantu memetakan fungsi sistem, alur proses, struktur data, serta interaksi antarbagian sistem secara terintegrasi.

Dalam pengembangan sistem informasi pada sektor publik maupun organisasi secara umum, UML berperan sebagai alat komunikasi yang efektif untuk menyelaraskan pemahaman antara analis, pengembang, dan pengguna sistem. Representasi visual yang disediakan oleh UML memudahkan proses analisis, validasi kebutuhan, serta dokumentasi sistem, sehingga perancangan yang dihasilkan lebih akurat dan dapat dipertanggungjawabkan [6].

2.2.2. Use Case Diagram

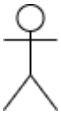
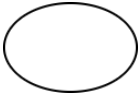


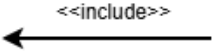
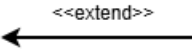
Use Case Diagram merupakan salah satu jenis diagram dalam Unified Modeling Language (UML) yang digunakan untuk menggambarkan hubungan interaksi antara pengguna (aktor) dengan sistem melalui berbagai fungsi yang tersedia. Diagram ini berfungsi untuk menunjukkan layanan yang disediakan oleh sistem serta cara aktor berinteraksi dengan fitur-fitur tersebut [7].

Use Case Diagram memiliki beberapa komponen utama, yaitu aktor, use case, dan system boundary. Aktor menggambarkan pihak eksternal yang berhubungan langsung dengan sistem, sedangkan use case merepresentasikan fungsi atau layanan yang disediakan. Sementara itu, system boundary digunakan untuk menunjukkan batasan ruang lingkup sistem yang sedang dirancang. Dengan struktur tersebut, kebutuhan fungsional sistem dapat diidentifikasi secara lebih jelas tanpa harus membahas aspek teknis secara mendalam [8].

Dalam proses perancangan sistem informasi, Use Case Diagram memiliki peran penting dalam mengidentifikasi kebutuhan pengguna serta menentukan ruang lingkup sistem secara terstruktur. Diagram ini membantu memastikan bahwa seluruh fungsi utama telah terdefinisi dengan baik dan sesuai dengan kebutuhan

pengguna. Selain itu, Use Case Diagram juga berfungsi sebagai sarana komunikasi yang efektif karena mudah dipahami oleh berbagai pihak, baik dari sisi teknis maupun non-teknis [9].

Tabel 2.1 *Use Case Diagram*

Simbol	Keterangan
	Aktor : mewakili peran orang sistem lain atau alat ketika berkomunikasi dengan use case.
	Use case : interaksi antara sistem dan aktor.
	Association : penghubung antara aktor dengan use case.
	Generalisasi : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case.
	Include : Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya.
	Extend : Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi.


2.2.3. Activity Diagram


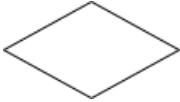



Activity Diagram merupakan salah satu diagram perilaku dalam Unified Modeling Language (UML) yang digunakan untuk merepresentasikan alur aktivitas atau proses bisnis secara sistematis. Diagram ini menunjukkan urutan langkah-langkah aktivitas, percabangan keputusan, kemungkinan proses yang berjalan secara paralel, serta titik awal dan akhir dari suatu proses, sehingga memberikan gambaran yang jelas mengenai alur kerja sistem [10].

Peran Activity Diagram sangat penting dalam tahap analisis kebutuhan karena mampu merepresentasikan proses yang kompleks ke dalam bentuk visual yang sistematis. Melalui diagram ini, analis dapat mengidentifikasi hubungan antaraktivitas, potensi percabangan proses, serta alur kerja yang saling berkaitan. Representasi visual tersebut memudahkan pemahaman proses oleh berbagai pihak yang terlibat dalam perancangan sistem [11].

Activity Diagram juga digunakan untuk mendokumentasikan proses sistem secara jelas dan konsisten. Dokumentasi yang baik mendukung proses pengembangan, pemeliharaan, serta pengembangan lanjutan sistem di masa mendatang. Integrasi Activity Diagram dengan diagram UML lainnya turut berkontribusi dalam meningkatkan kualitas desain sistem secara keseluruhan [12].

Tabel 2.2 *Activity Diagram*

Simbol	Keterangan
	Status awal, penanda awal sistem.
	Aktivitas, aktivitas yang dilakukan sistem.

	
	Percabangan, digunakan bila ada pilihan aktivitas lebih dari satu.
	Penggabungan, apabila ada aktivitas lebih dari satu digabungkan.
	Status akhir, dilakukan bila sistem berakhir.
	Swimlane, memisahkan organisasi yang bertanggung jawab terhadap aktivitas.

2.2.4. Class Diagram


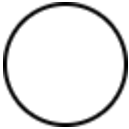


Class Diagram merupakan diagram struktural dalam Unified Modeling Language (UML) yang digunakan untuk menggambarkan struktur statis suatu sistem. Diagram ini merepresentasikan kelas, atribut, operasi, serta hubungan antarkelas yang membentuk kerangka logis sistem berbasis objek [13].


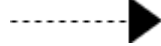
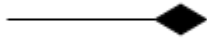
Class Diagram memiliki peran penting dalam mengubah kebutuhan fungsional sistem menjadi rancangan struktur yang tersusun secara sistematis. Hubungan antarkelas, seperti asosiasi, agregasi, komposisi, dan pewarisan, digunakan untuk menunjukkan keterkaitan antarentitas dalam sistem. Pemodelan

ini membantu memastikan bahwa struktur sistem telah dirancang secara konsisten dan mudah dikembangkan [14].

Selain sebagai alat desain, Class Diagram juga berfungsi sebagai media komunikasi visual antara analis sistem dan pengembang. Dengan struktur yang terdokumentasi dengan baik, Class Diagram memudahkan proses pemeliharaan dan pengembangan sistem di masa mendatang serta meningkatkan kualitas arsitektur perangkat lunak secara keseluruhan [15].

Tabel 2.3 *Class Diagram*

Simbol	Keterangan
	Kelas, kelas pada struktur sistem
	Interface, Sama dengan konsep interface dalam pemrograman berorientasi objek.
	Association, Relasi antarclass dengan arti umum, asosiasi biasanya juga disertai dengan Multiplicity.
	Directed association, relasi antar kelas dengan makna kelas yang atau digunakan oleh kelas yang lain

	<p>Generalisasi, Relasi antarkelas dengan makna generalisasi-spesialisasi (umum khusus).</p>
	<p>Dependency, relasi antar kelas dengan makna kebergantungan</p>
	<p>Aggregation, relasi antar kelas dengan makna semua bagian</p>

2.2.5. Sequence Diagram

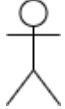
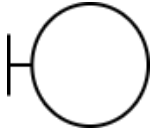

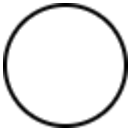
Sequence Diagram merupakan diagram perilaku dalam Unified Modeling Language (UML) yang digunakan untuk menggambarkan interaksi antarobjek dalam sistem berdasarkan urutan waktu. Diagram ini menunjukkan bagaimana pesan dikirimkan dari satu objek ke objek lainnya untuk menjalankan suatu fungsi tertentu [16].

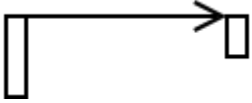
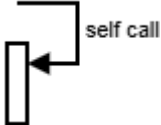


Sequence Diagram membantu memvisualisasikan alur komunikasi dan urutan eksekusi proses secara dinamis. Dengan memodelkan interaksi secara berurutan, diagram ini memudahkan pengembang dalam memahami perilaku sistem serta mengidentifikasi ketergantungan antarobjek. Pemodelan ini juga berguna untuk mengevaluasi konsistensi logika sistem sebelum tahap implementasi dilakukan [15].

Selain mendukung analisis perilaku sistem, Sequence Diagram berfungsi sebagai alat dokumentasi teknis yang memperjelas mekanisme kerja sistem.

Dokumentasi yang baik membantu meningkatkan keandalan sistem serta mempermudah proses pengembangan dan pemeliharaan di masa mendatang [17].

Tabel 2.4 *Sequence Diagram*

Simbol	Keterangan
	<p>Aktor merupakan entitas eksternal, baik berupa pengguna, proses, maupun sistem lain yang berinteraksi dengan sistem informasi yang sedang dikembangkan.</p>
	<p>Boundary object merupakan objek yang berfungsi sebagai penghubung antara sistem dengan pihak luar atau antarmuka pengguna.</p>
	<p>Entity object merupakan objek yang menyimpan data atau informasi terkait aktivitas tertentu yang bersifat tetap dan disimpan di dalam basis data.</p>
	<p>Control object merupakan objek yang berfungsi untuk mengatur dan mengoordinasikan jalannya proses dalam sistem, termasuk mengendalikan alur kerja serta menjalankan fungsi utama sistem.</p>
	<p>Object message merupakan representasi pesan atau komunikasi antar objek yang</p>

	<p>menunjukkan urutan proses atau kejadian yang terjadi dalam sistem.</p>
	<p>Message to self (atau message to life) merupakan pesan yang terjadi pada objek itu sendiri, yang menunjukkan adanya proses atau aktivitas internal dalam objek tersebut.</p>
	<p>Lifeline merupakan garis putus-putus yang merepresentasikan keberadaan suatu objek selama proses interaksi berlangsung, di mana pada sepanjang garis tersebut terdapat activation yang menunjukkan eksekusi aktivitas.</p>
	<p>Activation merupakan elemen yang menunjukkan proses eksekusi suatu operasi pada objek, di mana panjang kotak activation menggambarkan lamanya durasi aktivitas yang berlangsung.</p>

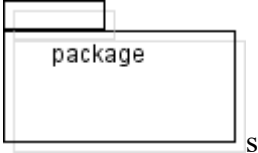
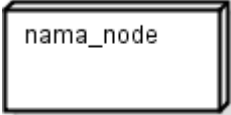
2.2.6. Component Diagram



Component Diagram merupakan diagram struktural dalam Unified Modeling Language (UML) yang digunakan untuk memodelkan arsitektur sistem

pada tingkat komponen. Diagram ini menggambarkan organisasi komponen perangkat lunak serta hubungan dependensi antarbagian sistem [13].

Component Diagram membantu membagi sistem ke dalam komponen-komponen modular yang bersifat mandiri. Pendekatan modular ini meningkatkan fleksibilitas, memudahkan pengelolaan sistem, serta mendukung pengembangan dan pemeliharaan perangkat lunak secara lebih efisien. Dengan struktur komponen yang jelas, kompleksitas sistem dapat dikendalikan dengan lebih baik [14].

Tabel 2.5 Komponen Diagram

Simbol	Deskripsi
<p data-bbox="387 987 501 1016"><i>Package</i></p> 	<p data-bbox="746 987 1366 1167">Package merupakan wadah yang digunakan untuk mengelompokkan satu atau lebih elemen dalam sistem agar tersusun secara terorganisasi.</p>
<p data-bbox="387 1379 448 1408"><i>Node</i></p> 	<p data-bbox="746 1379 1366 1850">Node umumnya mencakup komponen perangkat keras (hardware) maupun perangkat lunak (software) yang bukan hasil pengembangan sendiri. Apabila di dalam node terdapat komponen tambahan, maka komponen tersebut harus disesuaikan dengan rancangan yang telah ditentukan sebelumnya pada diagram komponen.</p>

<p>Kebergantungan/<i>dependency</i></p> 	<p>Dependency menunjukkan hubungan ketergantungan antar node, di mana arah panah mengarah pada node yang digunakan.</p> <p>Penghubungan antar aktor</p>
<p><i>Link</i></p> 	<p>Relasi antar node menunjukkan hubungan atau keterkaitan antara satu node dengan node lainnya dalam sistem.</p>

2.3. Teknologi Web

2.3.1. MySQL

MySQL merupakan salah satu sistem manajemen basis data relasional (Relational Database Management System/RDBMS) yang menggunakan Structured Query Language (SQL) dan banyak dimanfaatkan dalam pengembangan aplikasi berbasis web. MySQL bersifat open source, memiliki kinerja yang stabil, serta mampu mengelola data secara efisien baik dalam skala kecil maupun besar [18]. Sistem ini menerapkan model basis data relasional dengan penyusunan data ke dalam tabel-tabel yang terstruktur, sehingga proses penyimpanan, pengolahan, dan pengambilan data dapat dilakukan secara konsisten dan terorganisir [19].

Dalam implementasi sistem informasi, MySQL berfungsi sebagai media penyimpanan data terpusat yang mendukung integritas serta konsistensi data. MySQL juga menyediakan berbagai fitur seperti pengelolaan transaksi, indeks data,

pengaturan relasi antar tabel, serta fasilitas backup database. Dengan adanya fitur tersebut, sistem informasi dapat berjalan dengan lebih andal serta mampu memenuhi kebutuhan pengolahan data secara berkelanjutan. [19].

Selain itu, MySQL memiliki dokumentasi yang lengkap serta komunitas pengguna yang luas, sehingga memudahkan proses implementasi dan pemeliharaan sistem. Dukungan terhadap berbagai mesin penyimpanan, seperti InnoDB, memungkinkan pengembang menyesuaikan konfigurasi basis data dengan kebutuhan aplikasi. Dengan karakteristik tersebut, MySQL menjadi solusi basis data yang efektif dalam mendukung pengembangan sistem informasi berbasis web yang terstruktur dan terpercaya [20].

2.3.2. XAMPP

XAMPP merupakan paket perangkat lunak berbasis open source yang dirancang untuk menyediakan lingkungan server lokal yang lengkap melalui integrasi beberapa komponen utama, yaitu Apache sebagai web server, MySQL atau MariaDB sebagai sistem manajemen basis data, serta PHP dan Perl sebagai bahasa pemrograman sisi server [21]. Paket ini dikembangkan untuk memudahkan proses instalasi dan konfigurasi server lokal, sehingga pengguna dapat menjalankan aplikasi berbasis web tanpa memerlukan server eksternal.

Sebagai lingkungan pengembangan lokal, XAMPP berfungsi sebagai sarana simulasi sistem sebelum aplikasi diterapkan pada server produksi. Melalui server lokal, pengembang dapat melakukan pengujian terhadap fungsi aplikasi, memeriksa kesesuaian alur program, serta memastikan komunikasi antara aplikasi dan basis data berjalan dengan baik. Lingkungan ini sangat bermanfaat untuk mendeteksi

kesalahan sejak tahap awal pengembangan, sehingga kualitas sistem dapat ditingkatkan sebelum memasuki tahap implementasi yang lebih luas [22].

XAMPP juga dilengkapi dengan berbagai alat bantu pendukung, salah satunya adalah phpMyAdmin yang digunakan untuk mengelola basis data secara visual. Alat ini memudahkan pengguna dalam membuat tabel, mengatur struktur basis data, mengelola relasi antartabel, serta melakukan proses impor dan ekspor data. Keberadaan phpMyAdmin membantu proses perancangan dan pengelolaan basis data menjadi lebih sistematis, terutama pada pengembangan aplikasi yang membutuhkan pengelolaan data secara terstruktur dan konsisten [23].



Gambar 2.1 XAMPP

2.3.3. Visual Studio Code

Visual Studio Code (VS Code) merupakan source code editor modern yang dikembangkan oleh Microsoft dan banyak digunakan dalam pengembangan aplikasi berbasis web maupun perangkat lunak lainnya. Editor ini dirancang bersifat ringan (lightweight), namun memiliki kemampuan yang luas melalui dukungan ekstensi yang beragam. Dengan konsep yang fleksibel, VS Code memungkinkan pengembang menyesuaikan lingkungan kerja sesuai dengan kebutuhan proyek,

sehingga proses pengembangan perangkat lunak dapat berjalan lebih efisien dan terorganisasi [14].

VS Code mendukung berbagai bahasa pemrograman yang umum digunakan dalam pengembangan sistem informasi berbasis web, seperti PHP, JavaScript, HTML, dan CSS. Editor ini dilengkapi dengan fitur IntelliSense yang mampu memberikan saran kode secara otomatis, melakukan pelengkapan sintaks, serta mendeteksi kesalahan penulisan sejak tahap awal. Selain itu, VS Code menyediakan fasilitas debugging terintegrasi yang membantu pengembang dalam menganalisis dan memperbaiki kesalahan logika program secara lebih sistematis. Berbagai penelitian menunjukkan bahwa penggunaan editor dengan fitur bantu cerdas dapat meningkatkan produktivitas serta kualitas kode yang dihasilkan [15].

Keunggulan lain dari Visual Studio Code adalah tersedianya terminal bawaan yang memungkinkan pengembang menjalankan perintah sistem, mengelola package manager, serta mengoperasikan server lokal tanpa harus berpindah ke aplikasi lain. Integrasi berbagai alat pengembangan dalam satu lingkungan kerja mendukung efisiensi siklus pengembangan perangkat lunak, mulai dari penulisan kode, pengujian, hingga pemeliharaan aplikasi. Lingkungan kerja yang terintegrasi juga membantu menjaga konsistensi proses pengembangan serta mengurangi potensi kesalahan teknis. Selain itu, Visual Studio Code mendukung berbagai ekstensi yang dapat disesuaikan dengan kebutuhan pengembang, sehingga fleksibilitas dalam bekerja semakin meningkat. Fitur debugging yang terintegrasi juga memudahkan proses pelacakan dan perbaikan kesalahan secara langsung dalam editor [23].



Gambar 2.2 Aplikasi Visual Studio Code

2.3.4. PHP

PHP merupakan bahasa pemrograman berbasis server (server-side scripting) yang banyak digunakan dalam pengembangan aplikasi web karena kemampuannya dalam mengolah logika program, mengelola data, serta berinteraksi dengan basis data seperti MySQL. PHP dijalankan di sisi server untuk memproses permintaan dari pengguna, kemudian menghasilkan output berupa halaman web dinamis yang ditampilkan melalui browser. Dengan karakteristik tersebut, PHP menjadi salah satu komponen penting dalam pembangunan aplikasi web modern. [24].

Kelebihan PHP terletak pada tingkat fleksibilitas yang tinggi, kemudahan dalam penerapan, serta kemampuannya untuk berjalan pada berbagai jenis web server dan sistem operasi. PHP dapat digunakan pada lingkungan lintas platform (cross-platform) dan mampu terintegrasi dengan berbagai teknologi yang mendukung pengembangan aplikasi web. Selain itu, karena bersifat open source, PHP memiliki dokumentasi yang lengkap serta didukung oleh komunitas pengguna yang luas. Hal ini memberikan kemudahan bagi pengembang dalam proses

pembelajaran, pemeliharaan sistem, maupun pengembangan aplikasi secara berkelanjutan. [25].

Dalam pengembangan sistem informasi, PHP memungkinkan penerapan logika aplikasi secara terstruktur melalui penggunaan fungsi, modul, dan pemisahan kode. Pendekatan ini mendukung pembuatan sistem yang lebih terorganisasi, mudah dipelihara, serta fleksibel terhadap perubahan kebutuhan. PHP juga menyediakan berbagai mekanisme untuk meningkatkan keamanan aplikasi, seperti pengelolaan sesi, validasi data masukan, serta pengaturan hak akses pengguna, sehingga mendukung pengelolaan data yang lebih aman dan terkendali. [26].

2.4. Pengujian Sistem

2.4.1. Black Box Testing pada Sistem Informasi Manajemen

Black Box Testing merupakan metode pengujian perangkat lunak yang berfokus pada evaluasi fungsi sistem berdasarkan input dan output tanpa memperhatikan struktur internal kode program. Pendekatan ini digunakan untuk menilai apakah sistem telah berfungsi sesuai dengan kebutuhan fungsional yang ditetapkan dalam spesifikasi [29]. Penguji hanya berinteraksi melalui antarmuka sistem dan memverifikasi apakah keluaran yang dihasilkan telah sesuai dengan hasil yang diharapkan. Karena tidak menuntut pemahaman teknis terhadap logika pemrograman, metode ini efektif diterapkan pada sistem informasi yang digunakan oleh berbagai kategori pengguna [29].

Metode Black Box Testing relevan digunakan dalam pengujian sistem informasi berbasis web karena mampu mengevaluasi kinerja fungsional setiap modul secara menyeluruh. Melalui penyusunan test case, pengujian dilakukan

terhadap proses input, validasi, penyimpanan, serta penyajian informasi. Pengujian ini bertujuan untuk memastikan bahwa setiap fungsi sistem berjalan sesuai dengan rancangan, antarmuka dapat digunakan dengan baik, serta alur proses tidak menimbulkan kesalahan saat sistem dioperasikan [28].

2.5. Penelitian Terdahulu

Penelitian terdahulu memiliki peran penting sebagai landasan empiris dalam memahami penerapan sistem informasi berbasis web pada berbagai bidang organisasi, terutama di lingkungan pemerintahan. Studi-studi sebelumnya memberikan gambaran mengenai efektivitas pemanfaatan teknologi informasi, berbagai kendala dalam proses implementasi, serta dampaknya terhadap peningkatan kualitas pengelolaan data dan pelayanan. Secara umum, hasil penelitian menunjukkan bahwa penerapan sistem informasi berbasis web mampu meningkatkan kecepatan proses administrasi, memperbaiki akurasi data, serta mendukung terciptanya transparansi dan akuntabilitas dalam penyelenggaraan layanan publik. [30].

Beberapa penelitian juga mengungkapkan bahwa digitalisasi proses kerja melalui sistem informasi berbasis web memberikan peningkatan signifikan terhadap efisiensi operasional organisasi. Penerapan sistem ini memungkinkan otomatisasi proses-proses rutin, pengurangan biaya operasional, serta penghematan waktu dalam pelaksanaan tugas administratif. Selain itu, sistem informasi berbasis web mendukung integrasi antar unit kerja dan pemrosesan data secara real-time, yang pada akhirnya berdampak pada peningkatan produktivitas dan pengambilan keputusan yang lebih cepat serta berbasis data. Hal ini semakin relevan dalam

konteks modernisasi sektor publik, di mana kecepatan dan ketepatan layanan menjadi tuntutan utama dalam memberikan pelayanan kepada masyarakat.

Tabel 2.6 Penelitian Terdahulu

No	Nama Peneliti & Tahun	Judul Penelitian	Metode	Hasil Penelitian
1.	Ritonga, S. M., Ikhwan, A., & Putri, R. A. (2023)	Sistem Informasi Manajemen Pada Dinas Pangan Kabupaten Labuhanbatu	<i>Waterfall</i>	Penelitian ini berhasil merancang dan membangun sistem informasi manajemen berbasis web untuk pengelolaan data pada Dinas Pangan Kabupaten Labuhanbatu dengan menggunakan teknologi PHP dan MySQL.
2.	Ismail, H., Andayani, D. D., & Kaswar, A. B. (2025)	Development of a Web-Based Information System for Food Availability and Production at the Makassar City Food Security Agency	<i>Waterfall</i>	Penelitian ini berhasil mengembangkan sistem informasi yang tidak hanya meningkatkan transparansi terkait harga pangan, tetapi juga membantu masyarakat dalam menentukan keputusan pembelian secara lebih akurat.
3.	Putra, S. A., Suwarti, & Asparizal (2023)	<i>Sistem Informasi Pengelolaan Data Ternak Skala Usaha Mikro Berbasis Web pada Dinas Pertanian dan Pangan Kabupaten Kudus</i>	<i>Waterfall</i>	Penelitian ini menghasilkan sebuah sistem informasi berbasis web yang digunakan untuk mendukung pengelolaan data pertanian pada Dinas Pangan, Tanaman Pangan, dan Hortikultura.

4.	Rosyidah, F., Nugraha, F., & Handayani, P. K. (2022)	Sistem Informasi Pengelolaan Data Ternak Skala Usaha Mikro Berbasis Web pada Dinas Pertanian dan Pangan Kabupaten Kudus	<i>Waterfall</i>	Penelitian ini menghasilkan sistem informasi pengelolaan data ternak mikro yang dirancang untuk Dinas Pertanian dan Pangan Kabupaten Kudus..
5.	Azra, M. (2023)	Sistem Informasi Pendaftaran Izin Pangan Industri Rumah Tangga Berbasis Web Pada Dinas Penanaman Modal Dan Pelayanan Terpadu Satu Pintu	Waterfall	Penelitian ini mengembangkan sistem informasi berbasis web untuk mempermudah proses pendaftaran izin pangan industri rumah tangga di lingkungan instansi pemerintah.
6.	Muhammad Sa'i Ubaidillah (2025)	Perancangan Sistem Informasi Manajemen Pengelolahan Data Berbasis Web Menggunakan Php Dan Mysql Pada Dinas Pangan Kabupaten Labuhanbatu	<i>Waterfall</i>	Dalam tahap penelitian.