

BAB II

LANDASAN TEORI

2.1. Konsep Sistem Informasi

Konsep sistem informasi membahas dasar – dasar yang menjadi landasan dalam memahami bagaimana suatu sistem bekerja untuk mengolah data menjadi informasi yang bermanfaat. Pembahasan selanjutnya mengenai pengertian sistem, informasi, dan sistem informasi dapat dipahami secara lebih terarah dan menyeluruh.

2.1.1. Pengertian Sistem

Menurut Kamus Besar Bahasa Indonesia (KBBI), sistem adalah perangkat unsur yang secara teratur saling berkaitan sehingga membentuk suatu totalitas. Secara etimologis, istilah sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*systema*), yang berarti suatu kesatuan yang terdiri dari bagian-bagian atau komponen yang saling berhubungan, saling berinteraksi, dan bekerja sama untuk mencapai suatu tujuan.

Sistem didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan suatu tugas bersama-sama (Irmayani & Munandar, 2020). Secara konseptual Sebuah sistem umumnya terdiri dari input, proses, dan output yang saling berkaitan untuk mencapai tujuan tertentu. Dengan demikian, keberhasilan sistem bergantung pada keterhubungan dan koordinasi antar komponennya, bukan hanya pada elemen yang berdiri sendiri.

2.1.1.1 Klasifikasi Sistem

Sistem dapat diklasifikasikan berdasarkan beberapa sudut pandang. Menurut Ludwig von Bertalanffy dalam bukunya *General System Theory Foundations, Development, Applications*, sistem dibagi menjadi dua jenis, yaitu sistem terbuka dan sistem tertutup. Sistem terbuka adalah sistem yang selalu berinteraksi dengan lingkungan melalui pertukaran informasi, energi, maupun materi, seperti organisasi atau sistem informasi. Sebaliknya, sistem tertutup adalah sistem yang tidak menerima pengaruh dari lingkungan luar dan tidak melakukan pertukaran informasi, sehingga bersifat terisolasi.

2.1.1.2 Karakteristik Sistem

Menurut Tata Sutabri dalam bukunya “Analisis Sistem Informasi” (2017), suatu sistem memiliki beberapa karakteristik utama, yaitu:

1. Komponen Sistem (*Components*)

Suatu sistem yang terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem keseluruhan. suatu sistem dapat mempunyai sistem yang lebih besar yang disebut dengan “supra sistem”.

2. Batasan sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang dapat

dipisah pisahkan.

3. Lingkungan Luar Sistem (*Environtment*)

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga merugikan sistem tersebut. Lingkungan luar yang merugikan harus dikendalikan. Kalau tidak maka akan mengganggu kelangsungan hidup sistem tersebut.

4. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem yang lain disebut dengan penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.

5. Masukan Sistem (*Input*)

Energi yang dimasukkan kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Contoh, didalam suatu unit sistem komputer, “program” adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan “data” adalah *signal input* untuk diolah menjadi informasi.

6. Keluaran Sistem (*output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang

berguna. Keluaran ini merupakan masukan bagi subsistem yang lain. Seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi input bagi subsistem lain.

7. Pengolah Sistem (*Procces*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya adalah sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministic. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.2. Pengertian Informasi

Informasi adalah sekumpulan data atau fakta yang telah diproses dan diolah sedemikian rupa sehingga menghasilkan sesuatu yang bisa dipahami dan memberikan manfaat bagi penerimanya. Menurut (Irmayani & Munandar, 2020) informasi adalah data yang sudah diolah atau dimanipulasikan sesuai dengan keperluan tertentu yang memiliki nilai dan manfaat. Menurut (Yasir, 2020) informasi adalah data yang telah diolah terlebih dahulu dan memberikan manfaat dan memberikan bahan keputusan kepada penerima.

Berdasarkan definisi di atas dapat disimpulkan bahwa informasi adalah data

yang telah diolah atau dimanipulasi sesuai kebutuhan sehingga memiliki nilai, memberikan manfaat, dan dapat digunakan sebagai dasar dalam pengambilan keputusan bagi penerimanya.

2.1.3. Pengertian Sistem Informasi

Sistem informasi adalah serangkaian komponen yang bekerja sama untuk mengumpulkan, mengelola, menyimpan, memproses, dan menyebarkan informasi yang diperlukan untuk mendukung pengambilan keputusan dalam suatu organisasi atau entitas. Salah satu komponen utama dalam sistem informasi adalah pemanfaatan teknologi berbasis perangkat keras (*hardware*) dan perangkat lunak (*software*)(Adham, 2024).

2.2. UML (Unified Modelling Language)

Unified Modelling Language(UML) merupakan sebuah bahasa yang divisualisasikan dalam bentuk gambar atau grafik yang berfungsi untuk memberikan gambaran dan spesifikasi dalam pembangunan dan dokumentasi dari sebuah pengembangan sistem berorientasi objek (*object oriented*)(Dedi Saputra et al., 2023). Menurut (Syafitri, 2016) *Unified Modelling Language (UML)* adalah sebuah bahasa yang berdasarkan gambar untuk divisualisasikan, menspesifikasikan, membangun dan pendokumentasikan dari sebuah sistem pengembangan perangkat lunak berbasis objek.

UML tersusun atas beberapa elemen grafis membentuk 9 diagram, dalam penelitian ini hanya memakai 5 desain diagram yaitu *use case diagram*, *activity diagram*, *sequence diagram*, *class diagram* dan *deployment diagram*.

2.2.1. Use Case Diagram

Use case diagram adalah gambaran yang menerangkan perilaku (*behavior*) dari sebuah sistem yang berada dalam tahap pengembangan. *Use case* dimanfaatkan guna mengidentifikasi fungsi yang berada pada sistem informasi dan menetapkan pihak-pihak yang diberi hak untuk memanfaatkan fungsi-fungsi tersebut (Aryani & Lee, 2025).

Komponen – komponen yang ada pada use case diagram yaitu :

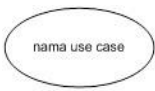
1. Aktor (*actor*)

Aktor merupakan pihak yang berhubungan dengan sistem, dapat manusia, fitur, ataupun sistem lain yang membagikan input ataupun menerima output dari sistem.

2. *Use case*

Use case merupakan guna ataupun layanan yang bisa dicoba oleh sistem buat aktor, yang menggambarkan gimana sistem merespon permintaan ataupun kebutuhan aktor dalam menggapai tujuan tertentu.

Tabel 2.1 Simbol – Simbol *Use Case Diagram*

Simbol	Deskripsi
	Aktor : mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case.
	Use case : abstraksi dan interaksi antara sistem dan aktor.
	Association : abstraksi dari 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.

Sumber : (Suharni et al., 2023)

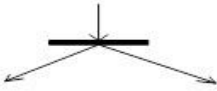
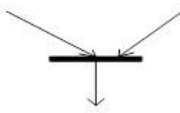
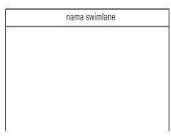
Tabel di atas menjelaskan simbol-simbol tersebut berfungsi untuk menggambarkan interaksi antara aktor dengan sistem, hubungan antar *use case*, serta mekanisme pewarisan dan perluasan fungsi dalam sistem.

2.2.2. Activity Diagram

Activity diagram dalam UML merupakan state diagram khusus, di mana sebagian besar dari state tersebut adalah aksi dan sebagian besar perpindahannya dipicu oleh selesainya state sebelumnya. Oleh karena itu, *activity diagram* tidak menggambarkan kelakuan internal suatu sistem dan interaksi antara satu subsistem dengan subsistem lainnya secara eksak, namun lebih menggambarkan jalur dan proses-proses aktifitas dari level atas secara umum (Ari & Wahid, 2023).

Tabel 2.2 Simbol – Simbol *Activity Diagram*

Simbol	Deskripsi
<i>Initial State</i> 	Status awal aktivitas sistem.
<i>Activity</i> 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
<i>Control Flow</i> 	Urutan perpindahan suatu aktivitas.
<i>Decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.

<i>Final State</i>		Status akhir yang dilakukan sistem.
<i>Fork</i>		Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel.
<i>Join</i>		Digunakan untuk menunjukkan kegiatan yang digabungkan.
<i>Swimlane</i>		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

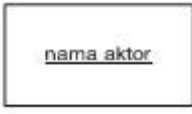
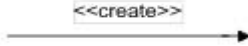
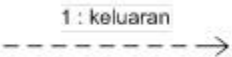
Sumber : (Suharni et al., 2023)

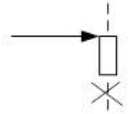
Activity Diagram adalah jenis diagram dalam *Unified Modeling Language* (UML) yang digunakan untuk menggambarkan aliran kerja atau aktivitas dalam suatu sistem atau proses. Diagram ini menyajikan serangkaian kegiatan, tindakan, dan keputusan yang terjadi sepanjang waktu (Rasiban et al., 2024). *Activity Diagram* membantu pengembang dalam memahami alur proses bisnis maupun logika sistem secara visual dan sistematis.

2.2.3. *Sequence Diagram*

Sequence diagram dijelaskan sebagai diagram interaksi antar objek berdasarkan urutan waktu (*over time*). Diagram ini menunjukkan objek mana yang saling mengirim pesan dan bagaimana pesan-pesan tersebut mengalir seiring waktu (Liesnaningsih, 2017). Diagram ini juga membantu dalam memvisualisasikan alur proses sistem secara lebih jelas dan terstruktur dari awal hingga akhir sehingga program dapat berjalan dengan terstruktur.

Tabel 2.3 Simbol – Simbol *Sequence Diagram*

Simbol	Deskripsi
Aktor  nama aktor Atau  nama aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda diawal frase nama aktor.
Garis Hidup/<i>Lifeline</i> 	Menyatakan kehidupan suatu objek.
Objek  nama objek : nama kelas	Menyatakan objek yang berinteraksi pesan.
Waktu aktif 	Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktu aktif ini adalah sebuah tahapan yang dilakukan didalamnya.
Pesan Tipe <i>Create</i>  <<create>>	Menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat.
Pesan Tipe <i>Call</i>  1 : nama_metode()	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
Pesan Tipe <i>Send</i>  1 : masukan	Menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim.
Pesan Tipe <i>Return</i>  1 : keluaran	Menyatakan bahwa suatu objek yang telah menjalankan suatu operasi atau metode menghasilkan suatu kembalian ke objek tertentu, arah panah megarah pada objek yang menerima.

Pesan Tipe <i>Destroy</i> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada <i>create</i> maka ada <i>destroy</i> .
--	---

Sumber : (Rasiban et al., 2024)

Diagram ini memperlihatkan aliran pesan antar objek sehingga pengembang dapat memahami proses komunikasi dan urutan eksekusi dalam sistem secara menyeluruh. Melalui penggunaan *Sequence Diagram*, hubungan antar komponen sistem dapat dijelaskan dengan lebih jelas sebelum tahap implementasi dilakukan (Aurellia et al., 2025).

2.2.4. *Class Diagram*

Dalam jurnal (Cahya et al., 2021) menyebutkan bahwa UML mencakup struktur diagram, di mana *class diagram* adalah bagian untuk menggambarkan struktur statis (kelas, atribut, metode, hubungan) dalam sistem.

Tabel 2.4 Simbol – Simbol *Class Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1.		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2.		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu actor.

5.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7.		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

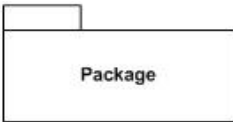
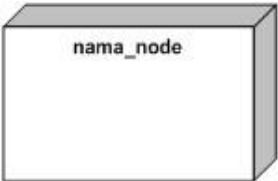
Sumber : (Suharni et al., 2023)

Tabel di atas menjelaskan penggunaan notasi *UML* tersebut bertujuan untuk mempermudah proses perancangan sistem, memperjelas alur interaksi antar komponen, serta membantu pengembang dan pemangku kepentingan dalam memahami desain sistem secara menyeluruh.

2.2.5. *Deployment Diagram*

Deployment diagram adalah diagram yang merepresentasikan model fisik dari *hardware* serta integrasi dan distribusi *software* pada arsitektur *hardware* tersebut. (Hamas & Imaduddin, 2019)

Tabel 2.5 Simbol – Simbol *Deployment Diagram*

Simbol	Deskripsi
<i>Package</i> 	<i>Package</i> merupakan sebuah bungkusan dari satu atau lebih <i>node</i> .
<i>Node</i> 	Biasanya mengacu pada perangkat keras (<i>hardware</i>), perangkat lunak (<i>software</i>) yang tidak dibuat sendiri, jika didalam <i>node</i> disertakan komponen untuk mengkonsistenkan

	rancangan maka komponen yang diikutsertakan harus sesuai dengan komponen yang telah didefinisikan sebelumnya pada diagram komponen.
Kebergantungan/ <i>dependency</i> 	Kebergantungan antara <i>node</i> , arah panah mengarah pada <i>node</i> yang dipakai.
Link 	Relasi antar <i>node</i> .

Sumber : (M.Kom, 2019)

Tabel di atas menjelaskan penggunaan simbol-simbol tersebut bertujuan untuk memvisualisasikan bagaimana komponen sistem ditempatkan dan saling terhubung dalam lingkungan implementasi.

2.3 Perancangan

Menurut Kamus Besar Bahasa Indonesia (KBBI), perancangan adalah proses, cara, atau perbuatan merancang sedangkan kata dasarnya yaitu rancang, berarti mengatur segala sesuatu terlebih dahulu sebelum bertindak, merencanakan, membuat rancangan. Jadi perancangan dapat dimaknai sebagai kegiatan menyusun atau merencanakan sesuatu secara sistematis sebelum pelaksanaan dilakukan, agar tujuan yang diinginkan dapat tercapai dengan baik.

2.4 Pemrograman

Pemrograman adalah proses menulis, menguji, memperbaiki dan memelihara kode yang membangun sebuah program komputer (Naufal, 2018). Dalam sebuah pemrograman dibagi menjadi dua tingkatan, yaitu amatir dan professional, maksud dari pemrogram amatir yaitu sesuatu yang hanya

menghasilkan sebuah program untuk memecahkan masalah-masalah yang kecil, sedangkan pemrogram profesional sendiri ialah memiliki suatu kemampuan untuk menyusun sebuah program yang bertujuan memecahkan masalah-masalah yang besar dan rumit (Riyanti & Yahfizham, 2023).

Selain itu, pemrograman juga dapat dipandang sebagai keterampilan berpikir logis dan sistematis dalam merancang solusi terhadap suatu permasalahan. Melalui pemrograman, seorang pengembang tidak hanya dituntut mampu menulis kode, tetapi juga memahami kebutuhan pengguna, menganalisis masalah, serta menghasilkan solusi yang efektif, efisien, dan dapat dikembangkan sesuai dengan perkembangan teknologi.

2.5 MySQL

MySQL merupakan turunan konsep utama dalam basis data, yaitu *SQL*, *SQL* adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan *input* data, yang memungkinkan pengoperasian data dikerjakan dengan mudah (Irmayani & Munandar, 2020).

2.6 Basis Data

Basis data adalah kumpulan data yang telah disusun secara metodis untuk memudahkan pengelolaan dan akses. Sistem basis data terdiri dari model data, skema, dan bahasa kueri yang memungkinkan interaksi data (Fahzirah & Nasution, 2024). Basis data juga didefinisikan sebagai kumpulan informasi yang terkait secara logis, disimpan bersama, dirancang untuk memenuhi kebutuhan informasi suatu organisasi (Dirgantara et al., 2023).

Jadi, dapat disimpulkan bahwa basis data merupakan Kumpulan data atau informasi yang saling terkait dan disusun secara terstruktur serta metodis, sehingga memudahkan proses penyimpanan, pengelolaan, dan akses data. Basis data dirancang menggunakan model data, skema, dan bahasa kueri agar dapat berinteraksi secara efektif, serta bertujuan untuk memenuhi kebutuhan informasi suatu organisasi secara optimal.

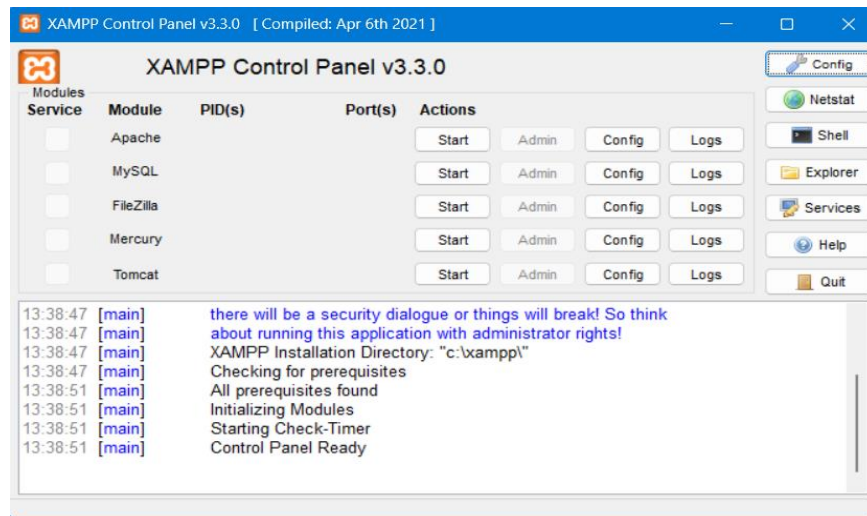
2.7 *PHP*

PHP, atau *Hypertext Preprocessor*, adalah bahasa pemrograman *server-side* yang memungkinkan website untuk berinteraksi dengan database dan menghasilkan konten dinamis (Sinlae et al., 2024). Salah satu keunggulan utama *PHP* adalah kemampuannya untuk terintegrasi secara efisien dengan berbagai jenis basis data, termasuk *MySQL* (Nova, 2025). Hal tersebut membuat *PHP* sebagai bahasa pemrograman yang banyak digunakan serta gampang beradaptasi dalam pengembangan aplikasi web yang terintegrasi dengan basis data.

2.8 *XAMPP*

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program (Irmayani & Munandar, 2020). *XAMPP* merupakan singkatan dari, X yaitu *Cross-platform* (dapat digunakan di Windows, Linux, dan macOS), A yaitu *Apache* (web server), M yaitu *MySQL/MariaDB (database)*, P yaitu *PHP*, dan P yaitu Perl. Secara umum, *XAMPP* digunakan untuk membantu pengembang web dalam mengembangkan dan menguji website atau sistem informasi secara lokal tanpa harus terhubung ke internet.

XAMPP sangat sering digunakan dalam pembuatan aplikasi web berbasis *PHP*.



Gambar 2.1 Tampilan *XAMPP*

2.9 Pengaduan

Menurut Kamus Besar Bahasa Indonesia (KBBI) pengaduan adalah penyampaian keluhan atau laporan kepada pihak yang berwenang mengenai suatu masalah, peristiwa, atau Tindakan yang dianggap merugikan. Pengaduan juga dapat di artikan sebagai penyampaian laporan atau keluhan oleh individu atau masyarakat kepada pihak yang berwenang mengenai suatu permasalahan atau kejadian tertentu dengan tujuan untuk mendapatkan penanganan dan penyelesaian.

2.10 Bencana Alam

Bencana alam adalah peristiwa atau rangkaian peristiwa yang disebabkan oleh faktor alam, seperti gempa bumi, banjir, tanah longsor, letusan gunung berapi, dan putting angin beliung yang mengakibatkan kerusakan lingkungan, kerusakan harta benda, serta dapat menimbulkan korban jiwa.

2.11 BPBD Kabupaten Labuhanbatu Utara

Badan Penanggulangan Bencana Daerah (BPBD) Kabupaten Labuhanbatu Utara merupakan instansi pemerintahan daerah yang berperan dalam menerima, mengelola, dan menindaklanjuti bencana alam serta bertanggung jawab dalam pelaksanaan upaya pencegahan, kesiapsiagaan, tanggap darurat, dan pemulihan pascabencana di wilayah Kabupaten Labuhanbatu Utara.

2.12 Penelitian Terdahulu

Penelitian terdahulu digunakan sebagai bahan acuan dan pembanding dalam pelaksanaan penelitian ini. penelitian terdahulu berfungsi untuk mengidentifikasi perbedaan, kelebihan, dan kekurangan dari penelitian sebelumnya sehingga dapat memperkuat dasar dan arah penelitian yang akan dilakukan.

Tabel 2.6 Penelitian Terdahulu

No.	Judul	Penulis	Tahun	Hasil
1.	Aplikasi Dan Perancangan Sistem Informasi Pemesanan Pada Upi Convention Center Dengan Menggunakan Bahasa Pemrograman <i>PHP</i> Dan <i>MySQL</i>	Syaiful Zuhri Harahap, Muhammad Halmi Dar	2018	Penelitian ini menghasilkan sebuah sistem informasi pemesanan berbasis web yang dibangun menggunakan bahasa pemrograman <i>PHP</i> dan database <i>MySQL</i> . Sistem yang dirancang mampu mempermudah proses pemesanan gedung di UPI <i>Convention Center</i> , mempercepat pengolahan data pemesanan, serta

				membantu pihak pengelola dalam mengelola data customer, produk, dan transaksi secara lebih efektif dan efisien dibandingkan sistem manual.
2.	Analisa Perancangan Sistem Informasi Pada Toko Rafatar Berbasis Web Menggunakan <i>PHP MySQL</i>	Dinda Nurinayah Pane, Masrizal, Syaiful Zuhri Harahap, Ibnu Rasyid Munthe	2025	Penelitian ini menghasilkan sistem informasi penjualan berbasis web yang mampu mengelola data penjualan, stok masuk dan keluar, serta riwayat transaksi secara terkomputerisasi. Penerapan metode FIFO membantu pengelolaan persediaan menjadi lebih akurat, mengurangi kesalahan pencatatan, dan meningkatkan efisiensi operasional Toko Rafatar.
3.	Perancangan Sistem Informasi Akademik SMA Pertiwi 2 Padang Menggunakan Bahasa Pemrograman <i>PHP</i> dan <i>MySQL</i>	Angga Putra Juledi	2021	Penelitian ini menghasilkan sistem informasi akademik berbasis web yang mampu mengelola data siswa, guru, nilai, dan jadwal secara terkomputerisasi, sehingga proses akademik menjadi lebih efektif, efisien, dan dapat diakses secara online dibandingkan sistem manual

				sebelumnya
4.	Perancangan Sistem Informasi Penjualan Pada Kedai Kopi Luwak Bengkulu Menggunakan Bahasa Pemrograman <i>PHP</i> dan Database <i>MYSQL</i>	Desi Mahdalena, Venny Novita Sari, Nofi Qurniati, Prahasti	2023	Penelitian ini menghasilkan sistem informasi penjualan berbasis web yang memudahkan admin dan kasir dalam mengelola transaksi penjualan, data menu, meja, serta laporan penjualan harian, bulanan, dan tahunan secara lebih efektif dan efisien, sehingga membantu pengendalian penjualan dan peningkatan kinerja usaha.
5.	Perancangan Sistem Informasi Akademik Menggunakan Bahasa Pemrograman <i>PHP</i> Dan Database <i>MySQL</i> (Studi Kasus Paud Terpadu Bissmillah Kota Bukittinggi)	Yoga Ananda Putra, Dr. Ir. Sumijan, M.Sc, Mardison, S.Kom, M.Kom	2019	Penelitian ini menghasilkan sistem informasi akademik berbasis web yang mampu mempermudah pengelolaan data akademik, mempercepat pembuatan laporan, serta memudahkan wali murid dalam mengakses informasi secara online dan real time.