

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Sistem

Menurut Okki Ramadhan, Ahmad Kamal, dan Sularno (2025), sistem merupakan sebuah kesatuan yang terdiri dari berbagai elemen yang saling berhubungan dan berinteraksi. Setiap elemen bekerja secara terstruktur untuk melaksanakan suatu proses tertentu dengan tujuan mencapai hasil atau sasaran utama yang telah ditentukan.

Menurut Romney dan Steinbart (2016) dalam Bagus Tri Mahardika (2020), sistem adalah rangkaian dari dua atau lebih komponen yang saling berhubungan dan berinteraksi untuk mencapai suatu tujuan. Sebagian besar sistem terdiri dari subsistem yang lebih kecil yang mendukung sistem yang lebih besar.

2.2 Karakteristik Sistem

Menurut Chr. Jimmy L (2008) dalam (Lela Nurlaela et al., 2020), suatu sistem merupakan hubungan satu unit dengan unit-unit lainnya yang saling berkaitan dan tidak bisa terpisahkan serta menuju suatu kesatuan dalam rangka mencapai tujuan yang telah ditetapkan.

Berikut adalah karakteristik sistem yang bisa membedakan suatu sistem dengan sistem lainnya:

- a. **Komponen Sistem (*Components*)**

Sebuah sistem terdiri dari sejumlah elemen yang saling berinteraksi, yakni saling bekerja sama membentuk satu kesatuan. Komponen-komponen

sistem atau elemen-elemen dapat berupa satu subsistem atau bagian dari sistem.

b. Batas Sistem (*Boundary*)

Batas sistem adalah daerah yang membatasi antara sistem dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dilihat sebagai satu kesatuan. Batas suatu sistem memperlihatkan ruang (*scope*) dari sistem tersebut.

c. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar sistem merupakan semua di luar batas dari suatu sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem bisa bersifat menguntungkan maupun merugikan sistem tersebut.

d. Penghubung (*Interface*)

Penghubung ialah media penghubung di antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini, sumber daya dapat diatur dari satu subsistem ke subsistem lainnya. Keluaran (*output*) dari suatu subsistem akan menjadi masukan (*input*) bagi subsistem lain melalui penghubung. Dengan penghubung, satu subsistem bisa berintegrasi dengan subsistem lainnya membentuk satu kesatuan.

e. Masukan (*Input*)

Masukan merupakan energi yang dimasukkan ke dalam sistem. Masukan sistem bisa dalam bentuk masukan perawatan maupun masukan sinyal. Masukan perawatan merupakan energi yang dimasukkan agar sistem dapat beroperasi guna menghasilkan keluaran.

f. Keluaran (*Output*)

Keluaran adalah hasil energi yang diolah serta diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

g. Pengolahan (*Process*)

Suatu sistem dapat berupa memiliki satu bagian pengolahan atau sistem itu sendiri sebagai pengolahannya. Pengolahan akan mengubah masukan menjadi keluaran.

h. Sasaran (*Objective*) atau Tujuan (*Goal*)

Suatu sistem mesti memiliki tujuan (*goal*) atau sasaran (*objective*). Sasaran dari sistem sangat menentukan hasil masukan yang dibutuhkan sistem serta keluaran yang dihasilkan.

2.3 Pengertian Informasi

Menurut McLeod (2004) dalam Dhika Widiyanto (2022), informasi adalah data yang telah diolah menjadi bentuk yang lebih bermanfaat dan berarti bagi penerimanya.

Informasi juga disebut data yang diproses atau data yang memiliki arti. Informasi merupakan data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakannya.

2.4 Pengertian Sistem Informasi

Menurut Firdinal Juliandre dan Sularno (2024), sistem informasi adalah suatu kesatuan yang terdiri dari komponen-komponen yang saling berhubungan dan bekerja sama untuk mengumpulkan, mengolah, menyimpan, serta

mendistribusikan data menjadi informasi yang berguna bagi pengambilan keputusan dan pengendalian organisasi.

Menurut Yeni Anggraini (2020), sistem adalah suatu jaringan kerja dari prosedur yang saling berhubungan dan berkumpul bersama-sama untuk melakukan sasaran tertentu. Sedangkan informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sehingga dapat disimpulkan bahwa sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, serta kegiatan strategis dari suatu organisasi tertentu dengan laporan-laporan yang dibutuhkan.

2.5 Pengertian Optimalisasi

Menurut Asep Hidayat (2022), optimalisasi merupakan suatu proses untuk mengoptimalkan agar ditemukannya solusi terbaik dari sekumpulan alternatif yang ada. Optimalisasi dilakukan dengan memaksimalkan suatu fungsi objektif tanpa melanggar batasan. Dengan adanya optimalisasi, suatu sistem dapat meningkatkan efektivitasnya, yaitu meningkatkan keuntungan, meminimalisir waktu proses, dan sebagainya.

2.6 Pengertian Penjualan

Menurut Maulana Mustopa (2021), penjualan merupakan pembelian sesuatu (barang atau jasa) dari suatu pihak kepada pihak lain dengan mendapatkan ganti berupa uang dari pihak tersebut. Penjualan juga merupakan sumber pendapatan

perusahaan, semakin besar penjualan maka semakin besar pula pendapatan yang diperoleh perusahaan.

2.7 Pengertian *Website*

Menurut Fadillah, P. N. N., & Gaffar, M. R. (2023) dalam Diva Cisco Pradithya, Dio Prima Mulya dan Sularno (2025), *Website* merupakan sebuah halaman yang terdapat dalam *World Wide Web* (WWW) dan harus memiliki domain serta *subdomain* yang dapat diakses melalui internet. Di dalam sebuah *website* biasanya terdapat teks, gambar, *video*, dan konten lainnya. Untuk dapat mengakses sebuah *website*, pengguna harus membuka format link HTML (*Hyper Text Markup Language*) melalui HTTP yang merupakan alat penyampaian informasi dari *server* kepada pengguna internet.

Menurut Adinda Febriyani (2023), *website* adalah sebuah kumpulan halaman *web* yang terkait dan dapat diakses melalui internet. Halaman *web* tersebut berisi informasi, gambar, *video*, atau elemen lainnya yang dapat diakses melalui *browser web*. *Website* sering digunakan untuk berbagai tujuan, seperti promosi produk atau layanan, menyediakan informasi, menyajikan konten seperti artikel, blog, atau media sosial, serta untuk tujuan hiburan. *Website* dibangun dengan menggunakan berbagai bahasa pemrograman *web* seperti *HTML*, *CSS*, dan *JavaScript*. Halaman *web* diatur dalam struktur hierarki yang disebut "*website architecture*". Sebuah *website* biasanya memiliki berbagai jenis halaman, seperti halaman beranda, halaman tentang, halaman kontak, dan halaman produk atau layanan.

2.8 Pengertian *PHP*

Menurut Prahasti (2022), *PHP (Hypertext Preprocessor)* adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan basis kode program menjadi kode mesin yang dapat dimengerti oleh komputer. *PHP* bersifat *server-side* dan ditambahkan ke dalam *HTML*.

2.9 Pengertian *Xampp*

Menurut Cut Keumala Zuhra (2022), *XAMPP* merupakan perangkat lunak bebas yang mendukung banyak sistem operasi, yaitu kompilasi dari beberapa program. Fungsi *XAMPP* adalah sebagai server yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP Server*, *MySQL Database*, dan penerjemah bahasa yang ditulis dengan menggunakan bahasa pemrograman *PHP* dan *Perl*. Nama *XAMPP* sendiri merupakan singkatan dari X (empat sistem operasi apa pun), *Apache*, *MySQL*, *PHP*, dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan merupakan *web server* yang mudah digunakan serta dapat melayani tampilan halaman *web* yang dinamis.

2.10 Pengertian *Database*

Menurut Deci Irmayani (2020), *database* merupakan kumpulan *file-file* yang saling berkaitan dan berinteraksi, di mana relasi tersebut ditunjukkan dengan kunci dari tiap-tiap *file* yang ada. Satu *database* menunjukkan suatu kumpulan data yang dipakai dalam lingkup perusahaan atau instansi. Pengolahan *database* merupakan suatu cara yang dilakukan terhadap *file-file* pada suatu instansi agar *file* tersebut dapat disusun, diurut, diambil sewaktu-waktu, serta ditampilkan dalam bentuk

laporan sehingga dapat mengolah *file-file* yang berisi informasi tersebut secara rapi.

2.11 Pengertian *MySQL*

Menurut Rulianto Kurniawan (2010) dalam Reza Hermiati (2021), *MySQL* merupakan suatu jenis *database server* yang sangat terkenal. *MySQL* termasuk jenis RDBMS (*Relational Database Management System*). *MySQL* mendukung bahasa pemrograman *PHP* dan bahasa permintaan terstruktur (*SQL*), karena dalam penggunaannya *SQL* memiliki beberapa aturan yang telah distandarkan oleh asosiasi *ANSI*. *MySQL* merupakan RDBMS *server*, yaitu program yang memungkinkan pengguna *database* untuk membuat, mengelola, dan menggunakan data pada suatu model relasional. Dengan demikian, tabel-tabel yang ada pada *database* memiliki relasi antara satu tabel dengan tabel lainnya.

Beberapa keunggulan *MySQL* yaitu:

- a Cepat, handal, dan mudah digunakan

MySQL lebih cepat tiga sampai empat kali daripada *database server* komersial yang beredar saat ini, mudah diatur, dan tidak memerlukan seseorang yang ahli untuk mengatur administrasi pemasangan *MySQL*.

- b Didukung berbagai bahasa

Database server MySQL dapat memberikan pesan error dalam berbagai bahasa seperti Belanda, Portugis, Spanyol, Inggris, Perancis, Jerman, dan Italia.

- c Mampu membuat tabel berukuran sangat besar

Ukuran maksimal dari setiap tabel yang dapat dibuat dengan *MySQL* adalah 4 GB hingga ukuran file yang dapat ditangani oleh sistem operasi yang dipakai.

- d Lebih murah

MySQL bersifat *open source* dan didistribusikan gratis tanpa biaya untuk *UNIX platform*, *OS/2*, dan *Windows platform*.

- e Integrasi kuat dengan *PHP*

Keterikatan antara *PHP* dengan *MySQL* yang sama-sama *open source* sangat kuat, sehingga koneksi yang terjadi lebih cepat dibandingkan menggunakan *database server* lainnya. Modul *MySQL* di *PHP* sudah *built-in* sehingga tidak memerlukan konfigurasi tambahan pada file konfigurasi *PHP*.



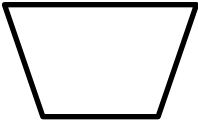
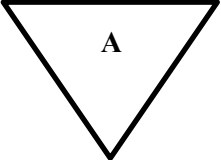
2.12 Pengertian *Visual Studio Code*

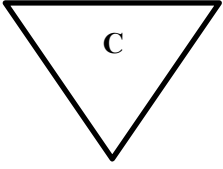
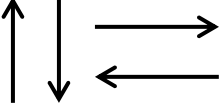


Menurut Januar Firnando (2023), *Visual Studio Code* adalah kode editor sumber yang dikembangkan oleh *Microsoft* untuk *Windows*, *Linux*, dan *macOS*. *Visual Code* memudahkan dalam penulisan kode yang mendukung beberapa jenis bahasa pemrograman dan memberi variasi warna sesuai dengan fungsi dalam rangkaian kode tersebut. Selain itu, fitur lainnya adalah kemampuan untuk menambah ekstensi, di mana para pengembang dapat menambahkan ekstensi untuk menambah fitur yang tidak ada di *Visual Studio Code*.

2.13 Pengertian Aliran Sistem Informasi

Menurut Jogiyanto dalam Alcianno G. Gani (2023), aliran sistem informasi merupakan suatu bagan yang menggambarkan arus kerja data yang akan diproses dari awal sampai akhir program secara menyeluruh. Aliran sistem informasi juga merupakan alat bantu yang digunakan untuk merancang sebuah sistem dari awal sampai akhir perancangan.

Tabel 2.1 Simbol – simbol Aliran Sistem Informasi

Simbol	Nama	Keterangan
	Simbol proses komputerisasi	Menggambarkan proses yang dilakukan secara komputerisasi
	Simbol dokumen	Menggambarkan dokumen masukan dan keluaran system
	Simbol proses manual	Menggambarkan proses manual
	Simbol proses pengarsipan	Menggambarkan <i>file</i> yang diarsipkan menurut alphabet

	<p>Simbol pengarsipan</p>	<p>Menggambarkan <i>file</i> yang diarsipkan menurut kronologi</p>
	<p>Simbol garis alir</p>	<p>Menggambarkan aliran proses dan dokumen</p>
	<p>Penghubung</p>	<p>Digunakan untuk menghubungkan sambungan aliran.</p>
	<p>Basis Data</p>	<p>Untuk media penyimpanan secara terkomputerisasi.</p>

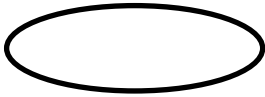
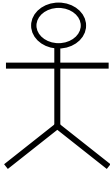

2.14 Pengertian *UML*

Menurut Sukanto dan M. Shalahuddin (2013) dalam Ibnu Rusdi (2020), pengertian *UML* adalah “bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung”. *UML* muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan mendokumentasikan sistem perangkat lunak.

2.14.1 Pengertian *Use Case Diagram*

Menurut David Saputra, Renita Astri, dan Sularno (2025), diagram *Use Case* menunjukkan bagaimana pengguna berinteraksi dengan sistem yang dikembangkan. Diagram ini membantu memahami alur kerja serta fungsi-fungsi utama dari sistem secara keseluruhan.

Tabel 2.2 Simbol – simbol *Use Case Diagram*

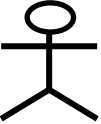
No	Simbol	Keterangan
1.	<i>Use Case</i> 	Fungsional yang disediakan sistem sebagai unit yang saling bertukar pesan antar unit atau <i>actor</i> : biasanya dinyatakan dengan menggunakan kata kerja diawal <i>frase name use case</i> .
2.	<i>Actor</i> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat sendiri. Jadi walaupun simbol <i>actor</i> adalah gambar orang, biasanya dinyatakan menggunakan kata benda diawal <i>frase</i> nama <i>actor</i> .
3.	Asosiasi 	Komunikasi antar <i>actor</i> dan <i>usecase</i> yang berpartisipasi pada <i>usecase</i> atau <i>usecase</i> memiliki interaksi dengan <i>actor</i> .


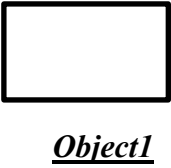

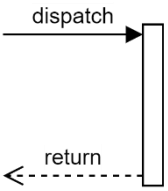
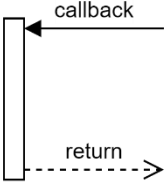
4.	<p style="text-align: center;"><i>Extend</i></p> <p style="text-align: center;"><< extend >></p> <p style="text-align: center;">-----></p>	Perilaku <i>usecase</i> memperluas perilaku <i>usecase</i> yang lain.
5.	<p style="text-align: center;">Generalisasi</p> <p style="text-align: center;">—————></p>	Hubungan generalisasi dan spesialisasi antara dua buah <i>usecase</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya
6.	<p style="text-align: center;"><i>Include</i></p> <p style="text-align: center;"><< include >></p> <p style="text-align: center;">-----></p>	Perilaku <i>usecase</i> merupakan bagian dari <i>usecase</i> yang lain

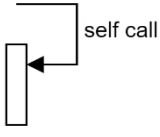
2.14.2 Pengertian *Sequence Diagram*

Menurut Rozaq (2015) dalam Redemptus Siga Dias (2022), *sequence diagram* adalah diagram yang menjabarkan *behavior* sebuah skenario tunggal, seperti menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek-objek tersebut di dalam *use case*.

Tabel 2.3 Simbol – simbol *Sequence Diagram*

No.	Gambar	Nama	Keterangan
1		<i>Actor</i>	Mengambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.

2		<i>Lifeline</i>	Menyatakan kehidupan suatu objek.
3		Objek	Antar muka yang sering berinteraksi
4		Waktu Aktif	Menyatakan objek dalam keadaan aktif dan berintegrasi, semua yang berhubungan dengan waktu aktif ini adalah sebuah tahapan yang akan dilakukan di dalamnya.
5		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi.
6		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi.



7		<i>Self Message</i>	Spesifikasi dari komunikasi untuk menunjukkan kegiatan memuat proses informasi pada aktifitas sendiri.
---	---	---------------------	--

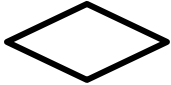


2.14.3 Pengertian *Activity Diagram*

Menurut Wulandari, T., & Nurmiati, S. (2022) dalam Windy Putri Yuza, Zulfahmi, M.Kom., et al., (2025), *Activity diagram* memodelkan *workflow* proses bisnis dan urutan aktivitas dalam sebuah proses. Diagram ini sangat mirip dengan *flowchart* karena memodelkan *workflow* dari suatu aktivitas ke aktivitas lainnya atau dari aktivitas ke status.

Menurut Rio Rafel Limantoro (2021), diagram aktivitas (*activity diagram*) bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem.

Tabel 2.4 Simbol – simbol *Activity Diagram*


No	Gambar	Nama	Keterangan
1		<i>Initial Node</i>	Mengambarkan titik awal aktivitas.
2		<i>Activity</i>	Merupakan gambar aktivitas yang ada pada sistem.

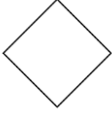
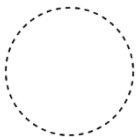
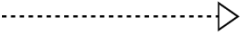
3		<i>Decision</i>	Merupakan pilihan untuk pengambilan suatu keputusan.
4		<i>Join</i>	Penggabungan dimana lebih dari aktifitas.
5		<i>Actifity Final Node</i>	Menggambarkan titik akhir aktifitas.

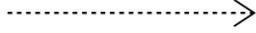

2.14.4 Pengertian *Class Diagram*

Menurut Ibnu Rusdi (2020), *Class diagram* menjelaskan hubungan antar kelas serta memberikan penjelasan secara detail pada setiap kelas dalam sistem. *Class diagram* merupakan gambaran yang menunjukkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat agar dapat berfungsi sesuai dengan kebutuhan sistem, guna mendukung proses pembangunan sistem.

Tabel 2.5 Simbol – simbol *Class Diagram*

No	Gambar	Nama	Keterangan
1		<i>Generalizatio n</i>	Hubungan dimana objek anak (<i>descendene</i>) berbagi perilaku dan struktur data

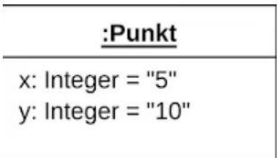
			dari objek yang ada di atasnya objek induk (<i>ancestor</i>).			
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari dua objek.			
3	<table border="1" data-bbox="395 779 683 1115"> <tr> <td>Nama Kelas</td> </tr> <tr> <td>+atribut</td> </tr> <tr> <td>-operasi()</td> </tr> </table>	Nama Kelas	+atribut	-operasi()	<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
Nama Kelas						
+atribut						
-operasi()						
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i> .			
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.			



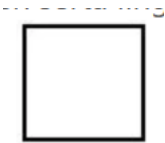

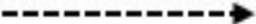
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (independent) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Association</i>	Apa yang menghubungkan objek satu dan lainnya.

2.14.5 Pengertian *Object Diagram*

Menurut Kevin Filbert Susanto (2023), *Object diagram* yaitu salah satu jenis diagram pada UML yang menggambarkan objek-objek pada suatu sistem dan hubungan antarnya.

Tabel 2.6 Simbol – simbol *Object Diagram*

No	Gambar	Nama	Keterangan
1		<i>Instance Specification</i>	Elemen yang mewakili sebuah contoh dalam sistem yang dimodelkan.

2		<i>Link</i>	Hubungan konseptual antar objek dalam sistem.
3		<i>Association</i>	Elemen yang menggambarkan hubungan antar objek yang dimodelkan.
4		<i>Port</i>	Titik interaksi antara komponen dengan lingkungannya.
5		<i>Generalization</i>	Cara untuk berbagi atribut dan perilaku dari suatu kelas atau objek kepada kelas atau objek lain.
6		<i>Dependency</i>	Relasi yang menunjukkan bahwa perubahan pada satu elemen akan mempengaruhi elemen lain yang bergantung padanya.