

BAB II

TINJAUAN PUSTAKA

2.1 Sistem Informasi

Sistem informasi (*Information system*) adalah sekumpulan komponen informasi yang saling berhubungan saling mengumpulkan atau mendapatkan, memproses, menyimpan, dan mendistribusikan informasi (Nitami et al., 2021).

Komponen Sistem Informasi

Untuk mendukung lancarnya suatu sistem informasi dibutuhkan beberapa komponen yang fungsinya sangat vital di dalam sistem informasi. Komponen-komponen sistem informasi tersebut yaitu :

1. Input

Input adalah semua data yang dimasukkan kedalam sistem informasi. Dalam hal ini yang termasuk dalam *input* yaitu dokumen-dokumen, formulir-formulir, dan file-file.

2. Proses

Proses merupakan kumpulan prosedur yang akan memanipulasi *input* yang kemudian akan disimpan dalam bagian basis data dan seterusnya akan diolah menjadi suatu *output* yang akan digunakan oleh si penerima.

3. Output

Output merupakan semua keluaran atau hasil dari model yang sudah diolah menjadi suatu informasi yang berguna dan dapat dipakai penerima. Komponen ini akan berhubungan langsung dengan pemakai sistem informasi dan merupakan tujuan akhir dari pembuatan sistem informasi.

4. Teknologi

Teknologi merupakan bagian yang berfungsi untuk memasukan *input*, mengolah *input* dan menghasilkan keluaran. Ada tiga bagian dalam teknologi meliputi perangkat keras, perangkat lunak, dan perangkat manusia.

5. Basis data

Basis data merupakan kumpulan data-data yang saling berhubungan satu dengan yang lain dan disimpan dalam perangkat keras komputer dan akan diolah menggunakan perangkat lunak.

6. Kendali

Kendali merupakan semua tindakan yang diambil untuk menjaga sistem informasi tersebut agar bisa berjalan dengan lancar dan tidak mengalami gangguan. Komponen ini sangat penting agar sistem secara keseluruhan memiliki validasi dan integritas yang tinggi.

2.1.1 Pengertian Sistem

Kata sistem berasal dari bahasa yunani yang mengandung arti “*Systema*” yang berarti kesatuan atau kumpulan. Ditinjau dari perkataan kata, Sistem berarti sekumpulan objek yang bekerja bersama-sama untuk menghasilkan kesatuan metode, prosedur atau teknik yang digabungkan dan diatur sedemikian rupa sehingga menjadi satu kesatuan yang berfungsi untuk mencapai tujuan (Nitami et al., 2021).

Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen, batasan sistem, lingkungan luar sistem, penghubung, masukan, keluaran pengolah dan sasaran atau tujuan.

1. Komponen (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem.

2. Batasan sistem (*Boundary*)

Batasan sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan luar sistem (*Environment*)

Lingkungan luar sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan juga merugikan.

4. Penghubung sistem (*Interface*)

Penghubung sistem merupakan media yang menghubungkan antara suatu subsistem dengan yang lain. Melalui penghubung ini kemungkinan sumber-sumber daya mengalir dari suatu subsistem ke subsistem yang lainnya.

5. Masukan sistem (*Input*)

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal *maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat berjalan.

6. Keluaran sistem (*Output*)

Keluaran sistem adalah energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran dapat merupakan masukan untuk subsistem lain.

7. Pengolahan sistem (*Process*)

Pengolahan sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan merubah masukan menjadi keluaran.

8. Sasaran sistem (*Objective*)

Suatu sasaran sistem mempunyai tujuan atau sasaran, kalau sistem tidak mempunyai sasaran maka sistem tidak akan ada. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya. Sasaran sangat berpengaruh pada masukan dan keluaran yang dihasilkan.

2.1.2 Pengertian Informasi

Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan berarti bagi yang menerima. Tanpa suatu informasi, suatu sistem tidak akan berjalan dengan lancar dan akhirnya bisa mati. Dengan kata lain sumber dari informasi adalah data (Nitami et al., 2021).

Kualitas Informasi

Kualitas informasi digunakan untuk mengukur informasi yang dihasilkan dari suatu sistem informasi dengan kualitas yang dapat memberikan nilai bagi pengguna sistem tertentu dengan karakteristik informasi yang sesuai dengan kebutuhan pengguna.

Kualitas informasi tergantung dari 3 hal yang sangat dominan yaitu sebagai berikut :

1. Akurat

Informasi yang dihasilkan harus bebas dari kesalahan-kesalahan dan tidak menyesatkan bagi orang yang menerima informasi tersebut. Ketidakakuratan dapat terjadi karena sumber informasi (data) mengalami gangguan atau kesengajaan sehingga merusak atau merubah data-data asli tersebut.

2. Tepat waktu

Informasi yang diterima harus tepat waktunya, informasi yang dihasilkan atau dibutuhkan tidak boleh terlambat. Informasi yang terlambat tidak mempunyai nilai yang baik, sehingga jika digunakan sebagai dasar dalam pengambilan keputusan akan berakibat fatal atau kesalahan dalam keputusan dan tindakan.

3. Relevan

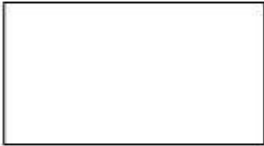
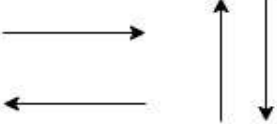
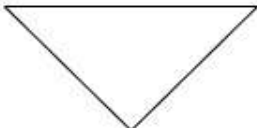
Informasi harus mempunyai manfaat bagi penerimanya, sebab informasi ini akan digunakan untuk pengambilan keputusan dalam pemecahan suatu permasalahan. Relevansi informasi juga berbeda-beda bagi setiap orang.

2.2 Aliran Sistem Informasi (ASI)

Aliran sistem informasi merupakan sebuah bagan untuk menggambarkan sebuah bagan untuk menggambarkan sebuah arus kerja dari awal sampai akhir

pengerjaan program tersebut. Aliran sistem informasi merupakan suatu alat bantu sistem yang digunakan untuk merancang sebuah sistem dari awal sampai akhir perancangan. Dari berbagai pernyataan diatas dapat disimpulkan bahwa aliran sistem informasi sangatlah penting pada suatu proses sistem. Dari proses sistem itulah di dapatkan permasalahan yang sedang di hadapi sehingga dapat di ketahui nilai guna sebuah sistem informasi apakah masih bisa beroperasi dengan baik atau tidak, sistem yang manual atau sistem yang lebih canggih. Jika sebuah sistem informasi sudah tidak layak dipakai lagi maka perlu dilakukan pembaruan (*upgrade system*) pada sistem supaya sistem dapat berjalan dengan lebih baik dan dalam proses pengolahan data dapat lebih akurat (Maydianto & Ridho, Muhammat Rasid, 2021).

Tabel 1. Simbol Aliran Sistem Informasi

NO	SIMBOL	KETERANGAN
1		Proses manual digunakan untuk menggambarkan kegiatan manual atau pekerjaan yang dilakukan tanpa menggunakan komputer
2		Proses komputer, proses yang menggunakan komputer dimana pengolahan data dilakukan secara online
3		Dokumen, merupakan formulir yang digunakan untuk merekam data yang menunjukkan input dan output baik untuk proses manual maupun komputer
4		Garis alir menunjukkan aliran atau arah dalam proses pengolahan data
5		Arsip untuk menggambarkan penyimpanan data baik dalam bentuk arsip atau file komputer. Dapat ditulis F atau A

Sumber : (Maydianto & Ridho, Muhammad Rasid, 2021).

2.3 Pengertian UML (*Unified Modelling Language*)

Unified Modelling Language (UML) adalah sebuah bahasa yang berdasarkan grafik/gambar untuk memvisualisasi, menspesifikasikan, membangun, dan pendokumentasian dari sebuah sistem pengembangan *software* berbasis *Object Oriented*. Penggunaan *UML* tidak terbatas untuk metodologi

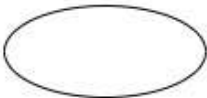
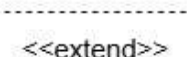
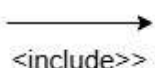
tertentu, walaupun pada kenyataanya *UML* ini paling banyak digunakan pada metodologi berorientasi objek.

UML versi terbaru, yaitu versi 2.5, dikelompokkan menjadi *structure diagram* dan *behavior diagram*. *Structure diagram* menggambarkan data dan hubungan statis dalam suatu sistem informasi yang terdiri *package*, *object*, *component*, *class*, *deployment*, *composite structure*, dan *profile diagram*. *Behavior diagram* menggunakan hubungan dinamis diantara objek yang mewakili sistem informasi bisnis. yang terdiri dari *sequence*, *timing*, *interaction overview*, *activity*, *use case*, *protocol state machine*, *communication* dan *behavior state machine diagram* (Suharni et al., 2023).

2.3.1 Use Case Diagram

Use case diagram merupakan pemodelan perilaku (*behavior*) dari suatu sistem yang dibutuhkan serta diharapkan pengguna. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem dan siapa saja atau aktor-aktor yang berhak menggunakan fungsi-fungsi tersebut (Suharni et al., 2023).

Tabel 2. 1 Simbol *Use Case Diagram*

Simbol	Nama	Deskripsi
	Actor	Digunakan untuk menjelaskan sesuatu atau seseorang yang sedang berinteraksi dengan sistem
	Use Case	Menggambarkan suatu perilaku dari sistem tanpa menggunakan struktur internal dari sistem tersebut
	Assosiation	Jalur komunikasi antar actor dengan use case yang saling berpartisipasi
	Extend	Penambahan perilaku kedalam use case dasar yang tidak tahu tentang hal tersebut
	Use Case generalization	Hubungan antara use case umum dengan use case yang lebih spesifik, yang mewarisi dan menambah fitur terhadapnya
	Include	Penambahan perilaku ke dalam use case dasar yang secara eksplisit menjelaskan penambahannya

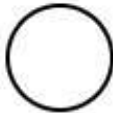
Sumber : (Suharni et al., 2023).

2.3.2 Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak yang perlu diperhatikan disini adalah bahwa diagram

aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor (Suharni et al., 2023).

Tabel 2. 2 Simbol *Activity Diagram*

Simbol	Nama	Deskripsi
	Initial	Menunjukkan dimana aliran kerja dimulai
	Final	Menunjukkan dimana aliran kerja berakhir
	Action	Langkah-langkah dalam sebuah activity
	Decision	Menunjukkan dimana keputusan akan dibuat
	Swimlane	Mengelompokkan activity berdasarkan actor

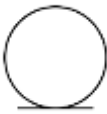
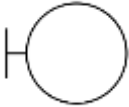
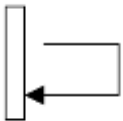
Sumber : (Suharni et al., 2023).

2.3.3 *Sequence Diagram*

Sequce Diagram adalah sebuah diagram yang menggambarkan kelakuan dari objek yang ada pada *use case* dengan cara mendeskripsikan waktu kejadian objek dan pesan yang akan dikirim dan diterima oleh antar objek. *Diagram Sequence* yang digambarkan harus sesuai dengan jumlah *use case* yang telah digambarkan. Sehingga banyaknya jumlah *Diagram Sequence* yang di gambarkan

sesuai dengan jumlah *use case* yang telah ada (Maydianto & Ridho, Muhammad Rasid, 2021).

Tabel 2. 3 Simbol *Sequence Diagram*

No	Simbol	Keterangan
1		<i>Entity class</i> , bagian dari sistem yang berisi kumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data
2		<i>Boundary class</i> , berisi kumpulan kelas yang menjadi <i>interfaces</i> atau interaksi antara satu atau lebih aktor dengan sistem, <i>entry</i> dan <i>form</i> cetak
3		<i>Control class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya kalkulasi dan aturan bisnis yang melibatkan berbagai objek
4		<i>Message</i> , simbol mengirim pesan antar <i>class</i>
5		<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk

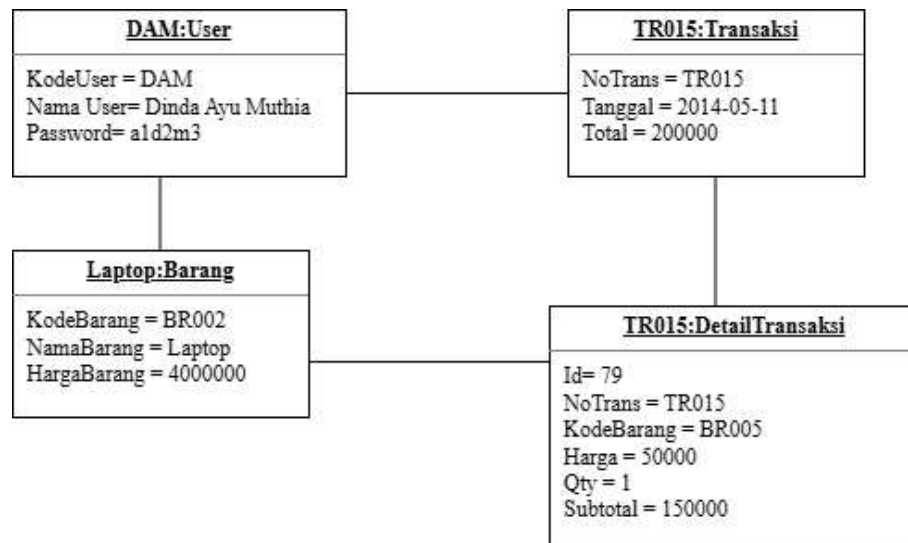
		dirinya sendiri
6		<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi
7		<i>Lifeline</i> , garis titik-titik berhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>

Sumber : (Swastika, 2024).

2.3.4 Object Diagram

Object diagram pada dasarnya merupakan instansiasi dari semua atau sebagian dari *class diagram*. *Object diagram* adalah potret dari objek dalam sistem pada suatu titik waktu. Karena *object diagram* lebih menunjukkan *instance* dari pada *class*, maka *object diagram* sering disebut *instance diagram*.

Instansiasi (*instantiation*) berarti membuat *instance (object)* dari *class* dengan satu set nilai atribut yang sesuai. *Object diagram* menunjukkan sekumpulan objek, hubungannya (*relationship*) dan nilai atribut. *Object diagram* bisa sangat berguna ketika mencoba untuk menemukan detail dari suatu *class*. *Object diagram* digunakan untuk memodelkan tampilan desain statis atau tampilan proses statis dari sistem sama seperti *class diagram*, tetapi dari perspektif contoh nyata atau *prototype* (Swastika, 2024).

Tabel 2. 4 Simbol *Object Diagram*

Sumber : (Swastika, 2024).

2.3.5 Class Diagram

Class Diagram menggambarkan serta deskripsi dari *class*, atribut dan objek serta hubungan satu sama lain. *Class diagram* dapat memberikan pandangan global atas sebuah *system*. Hal tersebut tercermin dari *class* yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa *class diagram*. *Class diagram* sangat membantu dalam visualisasi struktur kelas dari suatu *system*. Diagram ini umum digunakan pada pemodelan *system* berorientasi objek. *Class diagram* berfungsi untuk menjelaskan tipe dari objek sistem dan hubungannya dengan objek yang lain (Suharni et al., 2023).

Tabel 2. 5 Simbol *Class Diagram*

Simbol	Nama	Deskripsi
Classname + field: type + method(type): type	Class	Himpunan objek-objek dari berbagai atribut yang memiliki operasi yang sama
	Association	Relasi antar kelas dengan makna umum dan biasanya disertai multiplicity
	Directed Association	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas lain
	Aggregation	Mengindikasikan keseluruhan bagian relationship disebut sebagai relasi
	Composition	Relasi Composition terhadap class tempat dia bergantung
	Dependency	Menunjukkan operasi pada suatu class yang menggunakan class yang lain

Sumber : (Swastika, 2024).

2.4 UMKM

Pelatihan UMKM dirancang khusus untuk meningkatkan kemampuan dan pengetahuan para pelaku usaha mikro, kecil, dan menengah (UMKM). Tujuan utama dari pelatihan ini adalah untuk membantu para pelaku UMKM agar dapat mengembangkan usahanya dengan lebih baik, sehingga dapat meningkatkan produktivitas, daya saing, dan keuntungan.

Pelatihan UMKM merupakan investasi yang sangat penting untuk meningkatkan kesejahteraan masyarakat. Dengan memberikan pelatihan yang

tepat dan berkelanjutan, pelaku UMKM dapat mengembangkan usahanya, menciptakan lapangan kerja, dan meningkatkan perekonomian.

2.5 Database

Database adalah suatu kumpulan data terhubung yang disimpan secara bersama-sama pada suatu media, tanpa adanya suatu kerangkapan data, sehingga mudah digunakan kembali, dapat digunakan oleh satu atau lebih program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya, data disimpan sedemikian rupa sehingga apabila ada penambahan, pengambilan dan modifikasi data dapat dilakukan dengan mudah dan terkontrol. Basis data dapat dianggap tempat untuk sekumpulan berkas data terkomputerisasi, dengan tujuan utama memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan (Lia Sari & Guna Yanti Kemala Sari Siregar, 2021).

2.6 XAMPP

Xampp adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, dan merupakan kompilasi dari beberapa program. *Xampp* adalah sebuah paket perangkat lunak (*software*) komputer yang sistem penamaanya diambil dari akronim kata *Apache*, *MySQL*, *MariaDB* (Talia et al., 2024).

Untuk sebuah aplikasi berbasis *web* dengan menggunakan bahasa *PHP*, tentu saja diperlukan sebuah *server web* dan interpreter *PHP*. *Server* tidak harus sebuah komputer khusus dengan kinerja tinggi dan berukuran sangat besar, tetapi bisa dibuat PC yang mempunyai fungsi selayaknya sebuah *web server* yaitu, dengan menginstal paket *XAMPP*. *XAMPP* merupakan paket *PHP* dan *MySQL*

berbasis *open source* yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis *PHP*. *XAMPP* mengombinasikan beberapa paket perangkat lunak berbeda kedalam satu paket (Lia Sari & Guna Yanti Kemala Sari Siregar, 2021).

2.7 *MySQL*

MySQL adalah suatu jenis *database server* yang sangat populer. *MySQL* termasuk jenis *RDBMS* (*Relational Database Management System*). *MySQL* mendukung bahasa pemrograman *PH*, bahasa permintaan yang terstruktur, karena pada penggunaannya *SQL* memiliki beberapa aturan yang telah distandarkan oleh asosiasi yang bernama *ANSI*. *MySQL* merupakan *RDBMS* (*Relational Database Management System*) *server*. *RDBMS* adalah program yang memungkinkan pengguna *database* untuk membuat, mengelola, dan menggunakan data pada suatu model relational. Dengan demikian, tabel-tabel yang ada pada *database* memiliki relasi antara satu tabel dengan tabel lainnya (Hermiati et al., 2021).

2.8 *PHP (Hypertext Preprocessor)*

PHP adalah bahasa pelengkap *HTML* yang memungkinkan dibuatnya aplikasi dinamis yang memungkinkan adanya pengolahan data dan pemrosesan data. Semua *syntax* yang diberikan akan sepenuhnya dijalankan pada *server* sedangkan yang dikirimkan ke browser hanya hasilnya saja. Kemudian merupakan bahasa berbentuk *Script* yang ditempatkan dalam *server* dan diproses di *server*. Hasilnya akan dikirimkan ke *client*, tempat pemakai menggunakan *browser*. *PHP* dikenal sebagai sebuah bahasa *scripting*, yang menyatu dengan tag-tag *HTML*, di eksekusi di *server* dan digunakan untuk membuat halaman *web*

yang dinamis seperti halnya *Active Server Pages* (ASP) atau *Java Server Pages* (JSP). *PHP* merupakan sebuah *software Open Source* (Hermiati et al., 2021).

2.9 *HTML (Hyper Text Markup Language)*

HTML (Hyper Text Markup Language) adalah bahasa standar *web* yang dikelola penggunaannya oleh W3C (*World Wide Web Consortium*) berupa tag-tag yang menyusun setiap elemen dari *website*. *HTML* berperan sebagai penyusun struktur halaman *website* yang menempatkan setiap elemen *website layout* yang diinginkan.

HTML biasanya disimpan dalam sebuah *file* berekstensi *HTML* untuk mengetikkan skrip *HTML* dapat menggunakan *text* editor seperti *Notepad* sebagai bentuk paling sederhana atau *text* editor khusus yang dapat mengenali setiap unsur skrip *HTML* dan menampilkannya dengan warna yang berbeda sehingga mudah dibaca, seperti *Notepad++*, *Sublime Text* (Sari & Suhendi, 2020).

2.10 *CSS (Cascading Style Sheet)*

CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen *web* yang berfungsi mengatur elemen *HTML* dengan berbagai *property* yang tersedia sehingga dapat ditampilkan dengan berbagai gaya yang diinginkan.

Cara kerja *CSS* dalam memodifikasi *HTML* dengan memilih elemen *HTML* yang akan diatur kemudian memberikan *property* yang sesuai dengan tampilan yang diinginkan. Dalam memberikan aturan pada elemen *HTML*, skrip *CSS* terdiri dari 3 bagian yaitu *selector* untuk memilih elemen yang akan diberi aturan, *property* yang merupakan aturan yang diberikan dan *value* sebagai nilai dari aturan yang diberikan (Sari & Suhendi, 2020).

2.11 Website

Website dapat diartikan sebagai sekumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang diseluruh dunia. Halaman *website* dibuat dengan menggunakan bahasa standar yaitu *HTML*. Skrip *HTML* ini akan diterjemahkan oleh *web browser* sehingga dapat ditampilkan dalam bentuk informasi yang dapat dibaca oleh semua orang. Secara umum, *website* dibagi menjadi 3 jenis yaitu *website* statis, dinamis, dan interaktif (Sari & Suhendi, 2020).