



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Menurut Saputro (2017:217) menegaskan, “Komputer adalah seperangkat dari teknologi yang dapat digunakan untuk mengolah data teks dan gambar serta untuk mengoperasikan berbagai program maupun untuk control peralatan”.

Menurut Fachri *et.al.* (2020:1), “Komputer adalah sebuah mesin hitung elektronik yang secara cepat menerima informasi masukan digital dan mengolah informasi tersebut menurut seperangkat intruksi yang tersimpan dalam komputer tersebut dan menghasilkan keluaran informasi yang dihasilkan setelah diolah.

Berdasarkan definisi diatas, dapat disimpulkan bahwa komputer merupakan alat elektronik yang dapat menerima, memproses, mencetak, serta menyimpan data yang diolah yang beroperasi dibawah perintah manusia.

2.1.2 Pengertian Perangkat Lunak

Menurut Kadir (2017:2) “Perangkat lunak adalah instruksi-instruksi yang ditujukan kepada komputer agar dapat melaksanakan tugas sesuai kehendak pemakai”.

Menurut Sukanto dan Shalahuddin (2018:2) “Perangkat lunak (*software*) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan (*user manual*). Dari pendapat di atas dapat disimpulkan bahwa Perangkat lunak adalah perintah-perintah yang di arahkan kepada komputer sehingga dapat menjalankan tugas dengan benar sesuai keinginan *user*.



2.1.3 Pengertian Data

Menurut Kristanto (2018:8), “Data merupakan bentuk yang belum dapat memberikan manfaat yang besar bagi penerimanya, sehingga perlu suatu model yang nantinya akan dikelompokkan dan diproses untuk menghasilkan informasi.

Fathansyah (2018:2) menyatakan bahwa “Data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang diwujudkan dalam bentuk angka, huruf, simbol, teks, gambar, bunyi, atau kombinasinya”.

Dari pendapat di atas dapat disimpulkan bahwa data adalah penjelasan atau informasi yang akurat dengan fakta di dunia nyata yang dapat diproses dan dapat menghasilkan suatu informasi.

2.1.4 Pengertian Basis Data (*Database*)

Menurut Fathansyah (2018:2), “Basis Data (*database*) adalah himpunan kelompok data (*arsip*) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.”

Menurut Abdulloh (2018:103) menegaskan bahwa “Basis data atau *database* adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi.”

Dapat disimpulkan bahwa basis data atau *database* adalah sekumpulan data yang saling berhubungan disimpan dalam komputer sehingga dapat diproses untuk memperoleh informasi yang dibutuhkan dalam sebuah organisasi.

2.1.5 Metode Pengembangan Sistem

Menurut Rossa dan Shalahuddin (2018:125) “RUP (*Rational Unified Process*) merupakan pendekatan pengembangan perangkat lunak yang dilakukan berulang-ulang (*iterative*), fokus pada arsitektur (*architecture-centric*), lebih diarahkan berdasarkan berdasarkan penggunaan kasus (*use case driven*)



Adapun tahap-tahap (*fase*) dalam metode pengembangan RUP menurut Sukanto dan Shalahuddin (2018:128-131) adalah sebagai berikut:

1. *Inception* (permulaan)

Tahap ini lebih pada memodelkan proses bisnis yang dibutuhkan (*bussiness modeling*) dan mendefinisikan kebutuhan akan sistem yang akan dibuat (*requirements*).

2. *Elaboration* (perluasan/perencanaan)

Tahap ini lebih difokuskan pada perencanaan arsitektur sistem. Tahap ini juga dapat mendeteksi apakah arsitektur sistem yang diinginkan dapat dibuat atau tidak. Mendeteksi resiko yang mungkin terjadi dari arsitektur yang dibuat Tahap ini lebih pada analisis dan desain sistem serta implementasi sistem yang fokus pada purwarupa sistem (*prototype*).

3. *Construction* (kontruksi)

Tahap ini fokus pada pengembangan komponen dan fitur-fitur sistem. Tahap ini lebih pada implementasi dan pengujian sistem yang fokus pada implementasi perangkat lunak pada kode program. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability Milestone* atau batas/tonggak kemampuan operasional awal.

4. *Transition* (transisi)

Tahap ini lebih pada deployment atau instalasi sistem agar dapat dimengerti oleh *user*. Tahap ini menghasilkan produk perangkat lunak dimana menjadi syarat dari *Initial Operational Capability Milestone* atau batas/tongggak kemampuan operasional awal. Aktifitas pada tahap ini termasuk pada pelatihan *user*, pemeliharaan dan pengujian sistem apakah sudah memenuhi harapan *user*



2.2. Teori Judul

2.2.1. Pengertian Aplikasi

Menurut Jubilee Enterprise (2018:3) “Aplikasi merupakan perangkat lunak (*software*) yang dibuat untuk mempermudah pekerjaan manusia”.

Menurut Chan (2017:4) “Aplikasi adalah koleksi *window* dan objek-objek yang menyediakan fungsi untuk aktivitas *user*, seperti pemasukan data, proses dan pelaporan”.

Dari pendapat di atas dapat disimpulkan bahwa aplikasi adalah koleksi *window* atau program yang berfungsi bagi pengguna untuk keperluan tertentu dan mendukung manusia untuk mengerjakan tugas-tugasnya.

2.2.2. Pengertian Warehouse Management System

Menurut (AK Garside,2017) “WMS adalah subsistem informasi yang membantu dalam pengelolaan produk yang mengalir melalui fasilitas-fasilitas dalam jaringan logistic dan yang tersimpan di fasilitas tersebut”.

Menurut (Prasidi & Lesmini (2019:71) “*Warehouse Management System* merupakan sistem yang digunakan dalam pengelolaan gudang yang mengatur proses penanganan barang sejak penerimaan hingga pengirimnya. Seluruh proses dilakukan dengan menggunakan suatu sistem tertentu dan biasanya dibantu dengan perangkat computer, pallet, forklift dan rak tinggi.

Dari pendapat di atas dapat di simpulkan bahwa WMS adaalah sistem pengelolaan produk dan melalui fasilitas dalam logistic yang tersimpan tersebut.



2.2.3 Pengertian Website

Menurut Abdullah (2018:1) “ *Website* dapat sebagai kumpulan halaman yang berisi informasi satu 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 di seluruh dunia. Halaman *website* dibuat 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”.

Menurut Lestari dan Puspaningrum (2017: 13) “*Website* merupakan adalah sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk digital baik berupa teks, gambar, video,audio, dan animasi lainnya yang disediakan melalui jalur koneksi internet”

Dari pendapat di atas dapat disimpulkan bahwa *website* adalah gabungan dari sekumpulan halaman web di internet yang berguna untuk menyajikan informasi dan membuat sekumpulan halaman yang saling terik

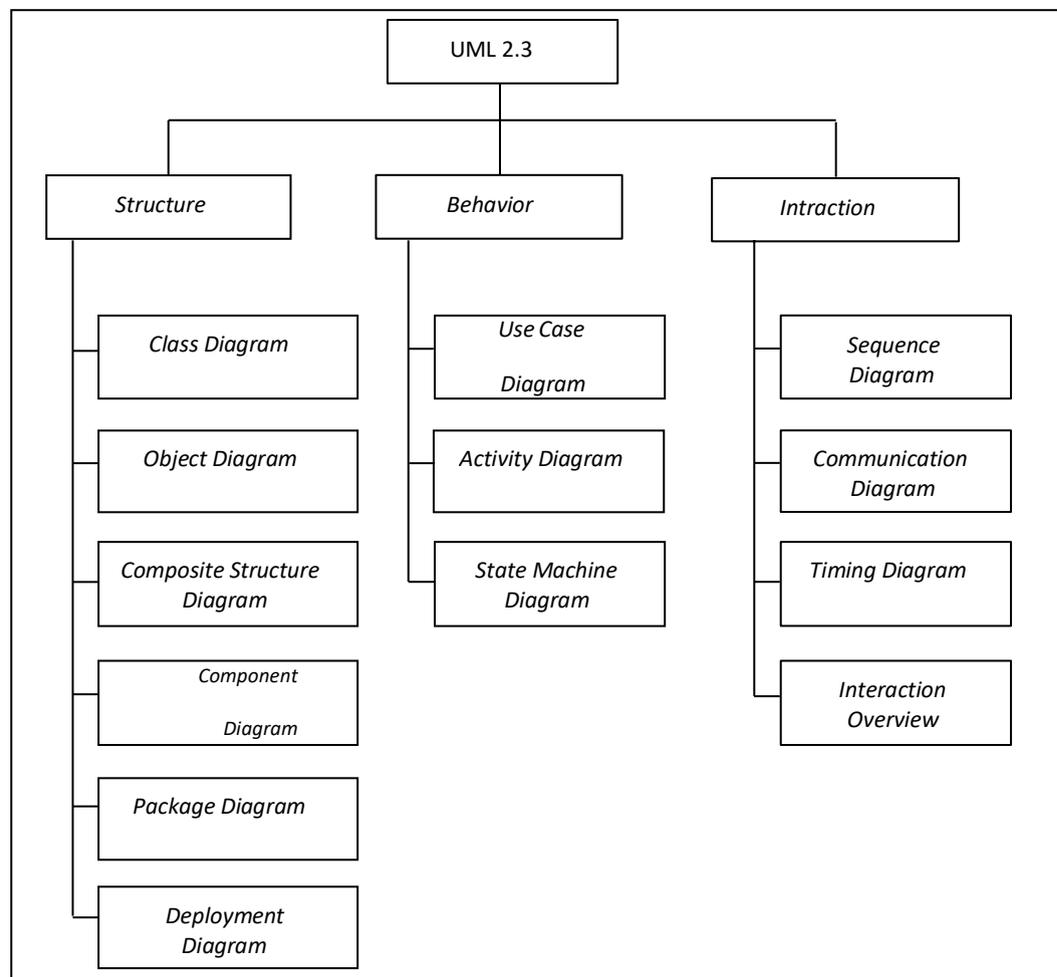


2.3 Teori Khusus

2.3.1 Unified Modeling Language (UML)

Sukanto dan Shalahuddin (2018:137), menjelaskan “Unified Modeling Language (UML) merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks – teks pendukung.”

Pada UML terdiri dari 13 macam diagram yang dikelompokkan dalam 3 kategori. Pembagian kategori dan macam – macam diagram tersebut yaitu :



Gambar 2.1 Diagram UML



Berikut merupakan penjelasan singkat dari pembagian kategori tersebut.

2.3.1.1 *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.

2.3.1.2 *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.

2.3.1.3 *Interactions diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun sistem interaksi antar subsistem pada suatu sistem.

2.3.2 Jenis-Jenis Diagram UML

2.3.2.1 Pengertian *Class Diagram*

Menurut Sukamto dan Shalahuddin (2018:141), menjelaskan tentang *class diagram*, sebagai berikut :

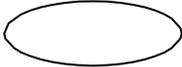
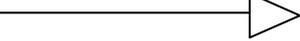
“*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.”

Adapun simbol-simbol yang digunakan dalam *class diagram* adalah sebagai berikut:

Tabel 2.1 Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
1.		Kelas pada struktur sistem

Lanjutan Tabel 2.1 Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
2.	antarmuka / interface  nama_interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
3.	asosiasi / association 	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i>
4.	asosiasi berarah / <i>directed association</i> 	Relasi antarkelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i>
5.	Generalisasi 	Relasi antarkelas dengan makna generalisasi – spesialisasi (umum khusus)
6.	kebergantungan / <i>dependency</i> 	Relasi antarkelas dengan makna kebergantungan antar kelas
7.	agregasi / <i>aggregation</i> 	Relasi antarkelas dengan makna semua-bagian (<i>whole-part</i>)

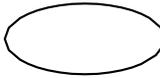
Sumber: Sukamto dan Shalahuddin (2018:146)

2.3.2.2 Pengertian *Use case Diagram*

Sukamto dan Shalahuddin (2018:155), menjelaskan tentang *use case* diagram sebagai berikut : “Use Case merupakan pemodelan untuk kelakuan (behaviour) sistem informasi yang akan dibuat. Use Case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem”



Tabel 2.2. Simbol-simbol *Use case* Diagram

No.	Simbol	Nama	Deskripsi
1.		Actor	Seseorang atau apa saja yang berhubungan dengan sistem
2.		Use case	Menggambarkan bagaimanaseseorang menggunakan Sistem
3.		Relasi asosiasi	Relasi yang dipakai untuk menunjukkan hubungan antara aktor dan use case
4.	<< include >>	Relasi include	Memungkinkan satu use case menggunakan fungsionalitas yang disediakan oleh use case Lainnya
5.	<< extends >>	Relasi extend	Memungkinkan suatu use case secara optional menggunakan fungsionalitas yang disediakan oleh use case lainnya

(Sumber : Rusmawan, 2019:72-73)



Ada dua hal utama pada *use case* yaitu:

2.3.2.1 Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, Jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.

2.3.1.1 *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor



2.3.2.3 Pengertian Activity Diagram

Sukanto dan Shalahuddin (2018:161), menjelaskan tentang *activity diagram* sebagai berikut :

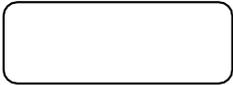
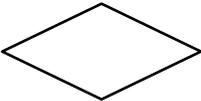
“*Activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.”

Adapun simbol-simbol yang digunakan dalam *activity diagram* adalah sebagai berikut:

Tabel 2.3 Simbol-simbol *Activity Diagram*

No.	Simbol	Nama	Deskripsi
1.		Start state	Titik Awal atau permulaan
2.		End state	Titik akhir atau akhir dari aktivitas
3.		Activity	<i>Activity</i> atau aktivitas yang dilakukan oleh Actor

Lanjutan Tabel 2.3 Simbol-simbol *Activity Diagram*

No.	Simbol	Nama	Deskripsi
3.		Activity	<i>Activity</i> atau aktivitas yang dilakukan oleh Actor
4.		Decision	Pilihan untuk mengambil keputusan
5.		Interaction	Alur

Sumber: Sukamto dan Shalahuddin (2018:162-163)

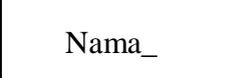
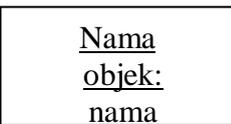
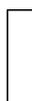
2.3.2.4 Pengertian *Sequence Diagram*

“Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah beserta metode-metode yang dimiliki kelas yang diinstansikan menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada use case” (Sukamto dan Shalahuddin, 2018:165).

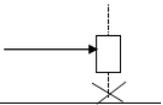
Dapat penulis simpulkan bahwa Sequence diagram adalah penggambaran skenario dari sebuah objek yang ada pada use case yang meliputi rangkaian langkah-langkah aktivitas dari objek berdasarkan waktu hidup objek dan pesan-pesan yang diterima maupun yang dikirimkan objek kepada objek lainnya



Tabel 2.4 Simbol-simbol pada Sequence Diagram

No.	Simbol	Deskripsi
1.	<p>Aktor</p>  <p>Atau</p>  <p>Nama_</p> <p>Akt</p> <p>Tanpa waktu aktif</p>	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tetapi aktor belum tentu berupa orang, biasanya dinyatakan menggunakan kata benda diawal frase nama aktor</p>
2.	<p>Garis hidup/<i>lifeline</i></p> 	<p>Menyatakan kehidupan suatu objek</p>
3.	<p>Objek</p>  <p><u>Nama</u> <u>objek:</u> <u>nama</u> <u>kelas</u></p>	<p>Menyatakan objek yang berinteraksi pesan</p>
4.	<p>Waktu aktif</p> 	<p>Menyatakan objek dalam keadaan aktif dan berinteraksi, semua yang terhubung dengan waktuaktif ini adalah sebuah tahapan yang dilakukan</p> <p>Didalamnya</p>

Lanjutan Tabel 2.4 Simbol-simbol pada *Sequence* Diagram

No.	Simbol	Deskripsi
6.	Pesan tipe <i>call</i> 1 : nama_metode() 	Menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri
7.	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
8.	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 mengarah pada objek yang menerima kembalian
9.	Pesan tipe <i>destroy</i> <<destroy>> 	Menyatakan suatu objek mengakhiri hidup objek yang lain, arah panah mengarah pada objek yang diakhiri, sebaiknya jika ada create maka ada Destroy

Sumber: Sukamto dan Shalahuddin (2018:165)

2.4 Teori Program

2.4.1 Pengertian PHP

Enterprise (2018:1) menegaskan, “PHP merupakan bahasa pemrograman yang digunakan untuk membuat website dinamis dan interaktif. Dinamis artinya, website tersebut bisa berubah tampilan dan



kontennya sesuai kondisi tertentu”

Abdulloh (2018:127) menegaskan, “PHP merupakan bahasa pemrograman web yang dapat disisipkan dalam skrip HTML dan bekerja di sisi server.”

Dapat disimpulkan bahwa PHP adalah bahasa pemrograman universal untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML.

2.4.2 Pengertian CSS

Handayani, dkk (2019: 34) “CSS atau *Cascading Style Sheet* yang merupakan suatu bahasa pemrograman web yang digunakan untuk mengendalikan dan membangun berbagai komponen dalam *web* sehingga tampilan *web* akan lebih rapi, terstruktur dan seragam”.

Mulyani (2020:77) menegaskan, “CSS adalah kumpulan perintah yang diper gunakan untuk menjelaskan sebuah halaman situs web dalam mark-up language.

Dapat disimpulkan bahwa , CSS adalah salah satu kode pemrograman yang bertujuan untuk menghias dan mengatur gaya tampilan/layout halaman *web* supaya lebih elegan dan menarik dan memudahkan *user* untuk mengatur berbagai tampilan yang di dalam *website* dengan tidak mengesampingkan *user experience*

2.4.3 Pengertian MySQL

Enterprise (2018:2) menegaskan, “MySQL merupakan server yang melayani database. Untuk membuat dan mengolah database, kita dapat mempelajari pemograman khusus yang disebut *query* (perintah) SQL”.

Abdulloh (2018:104) menegaskan, “MySQL merupakan bahasa yang digunakan untuk mengakses dan memanipulasi database.”



Jadi, dapat disimpulkan, MySQL adalah sebuah perangkat lunak untuk mengolah database atau basis data .

2.4.4 Pengertian XAMPP

Menurut Iqbal (2019:15), “XAMPP merupakan sebuah *software* web server apache yang didalamnya sudah tersedia database server mysql dan support php programming.”

Poipessy dan Mirna Umasangadji (2018:40) menegaskan, “XAMPP merupakan paket PHP dan MySQL berbasis open source, yang dapat digunakan sebagai tool pembantu pengembangan aplikasi berbasis PHP.” Dapat disimpulkan XAMPP merupakan paket konfigurasi Apache, PHP dan MySQL yang digunakan untuk membantu pengembangan aplikasi.

Berdasarkan definisi di atas bahwa dapat disimpulkan XAMPP merupakan sebuah paket yang di dalamnya sudah tersedia database *server* MYSQL dan support PHP programming.