



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Sudjiman (2018:61) komputer adalah mesin penghitung elektronik yang cepat dan dapat menerima informasi input digital, kemudian memprosesnya sesuai dengan program yang tersimpan di memorinya, dan menghasilkan output berupa informasi.

Wahyudin, S.T., M.T (2018:1), “Komputer adalah suatu alat elektronik yang mampu melakukan beberapa tugas, yaitu menerima *input*, memproses *input* sesuai dengan instruksi yang diberikan, menyimpan perintah-perintah dan hasil pengolahannya, serta menyediakan *input* dalam bentuk informasi”.

Dari pendapat di atas dapat disimpulkan bahwa komputer adalah merupakan alat elektronik yang dapat menerima, memproses, mencetak, serta menyimpan data yang diolah yang beroperasi dibawah perintah manusia.

2.1.2 Pengertian Web

(Sudarto, Kristiadi, & Hidayat, 2021), Web merupakan kumpulan laman yang berisi informasi yang dapat diambil melalui media internet berupa tulisan, gambar suara, video yang memiliki alamat pada browser.

Menurut Abdulloh dalam (Sa'ad, 2020), website atau web adalah kumpulan beberapa halaman yang terdiri dari informasi berbentuk data digital, baik berupa teks, gambar, video, suara, dan animasi lainnya yang disediakan melalui jalur koneksi internet. Lebih jelasnya, website merupakan halaman-halaman yang berisi informasi yang dapat diakses oleh browser dan mampu memberikan informasi yang berguna bagi para pengaksesnya.

Berdasarkan penjelasan di atas bahwa website adalah laman berisi informasi berupa teks, gambar, animasi, suara, dan video yang diakses melalui koneksi internet.



2.1.3 Pengertian Data

Indrajani (2018:2) menegaskan, “Data adalah fakta atau observasi mentah yang biasanya mengenai fenomena fisik atau transaksi bisnis”.

Fathansyah (2018:2) menegaskan, “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 sehingga digunakan sebagai dasar analisis.

2.1.4 Metode Pengembangan Sistem

Pengertian Waterfall merupakan salah satu metode dalam SDLC yang mempunyai ciri khas pengerjaan yaitu setiap fase dalam waterfall harus diselesaikan terlebih dahulu sebelum melanjutkan ke fase selanjutnya. (Tabrani & Aghniya, 2019)

Model air terjun menyediakan pendekatan alur hidup terurut mulai dari analisis, desain, pengkodean, pengujian dan pemeliharaan.

a. Analisis Kebutuhan

Perangkat Lunak Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh user. Spesifikasi kebutuhan perangkat lunak pada tahap ini perlu untuk didokumentasikan.

b. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program sistem termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka dan prosedur pengkodean.



c. Pembuatan Kode Barang

Pada tahap pengkodean, desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

d. Pengujian

Pengujian fokus pada perangkat lunak secara dari segi logik dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (error) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan. Dalam penelitian ini pengujian sistem akan menggunakan pengujian Black-Box. Pengujian Black-box berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian metode ini memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk suatu program.

e. Pendukung (*support*) dan pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah mengirimkan ke user. Perubahan bisa terjadi adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru

2.2 Teori Khusus

2.2.1 Kamus Data

(Santi, 2020), menyatakan bahwa, Kamus data adalah kumpulan daftar elemen yang berisi data atau katalog fakta tentang data dari suatu sistem informasi. Kamus data biasanya dibuat berdasarkan arus data yang mengalir pada DAD (fisik dan logika) dimana di dalamnya terdapat struktur dari arus data secara lengkap.

Tabel 2.1 Simbol-simbol pada Kamus Data

No	Simbol	Keterangan
1.	=	Terbentuk dari atau terdiri dari
2.	+	Dan



Lanjutan Tabel 2.1. Simbol-simbol pada Kamus Data

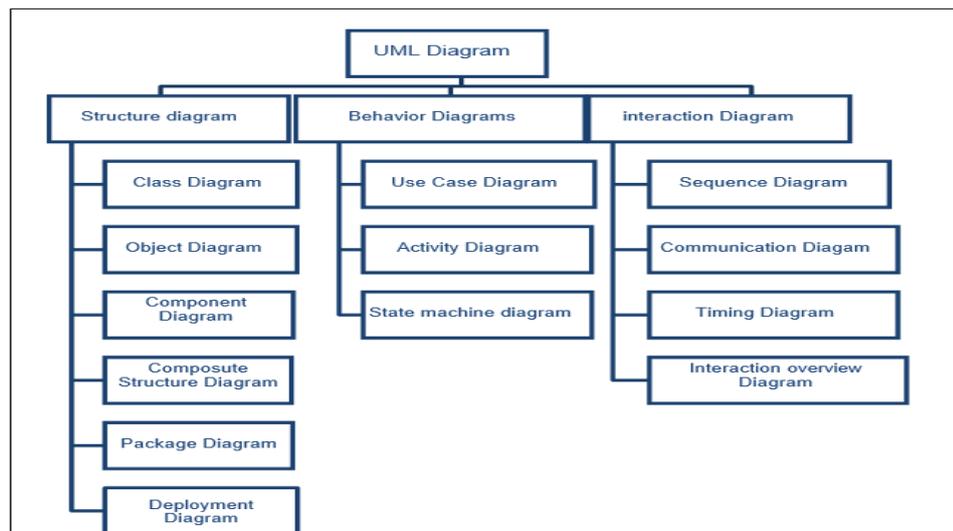
No	Simbol	Keterangan
3.	[]	Memilih salah satu elemen data
4.	N{M}	N kali/ bernilai banyak berulang
5.	()	Data opsional
6.	*...*	Batas komentar
7.		Pemisah sejumlah alternatif
8.	@	Identifikasi atribut kunci

Sumber : (Maniah & Hamidin, 2017)

2.2.2 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.



1. *Structure diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan kelakuan sistem atau rangkaian perubahan yang terjadi pada sebuah sistem.
3. *Interactions diagrams* yaitu kumpulan diagram yang digunakan untuk menggambarkan interaksi sistem dengan sistem lain maupun sistem interaksi antar subsistem pada suatu sistem.

2.2.2 Jenis-Jenis Diagram UML

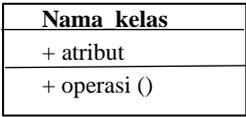
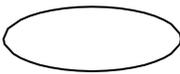
2.2.2.1 Pengertian *Class Diagram*

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

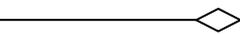
“*Class Diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Diagram kelas dibuat agar pembuat program atau *programmer* membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.”

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

Tabel 2.2 Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
1.	 Nama kelas + atribut + operasi ()	Kelas pada struktur sistem
2	 nama_interface	Sama dengan konsep interface dalam pemrograman berorientasi objek
3.	 →	Relasi antarkelas dengan makna umum, asosiasi biasanya juga disertai <i>multiplicity</i>

Lanjutan Tabel 2.2. Simbol-simbol *Class Diagram*

No	Simbol	Deskripsi
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>)

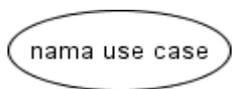
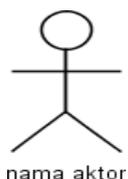
Sumber: Sukanto dan Shalahuddin (2018:146)

2.2.2.2 Pengertian *Use case Diagram*

Sukanto dan Shalahuddin (2018:155), menjelaskan tentang *use case* diagram sebagai berikut :

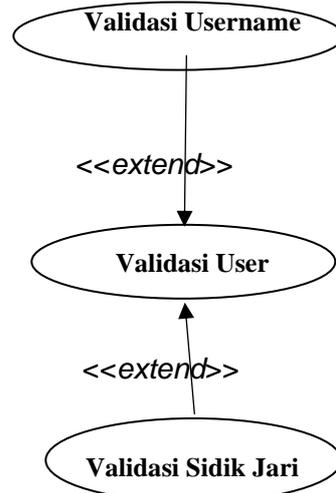
“*Use case* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem.” Adapun simbol-simbol yang digunakan dalam *use case* adalah sebagai berikut:

Tabel 2.3. Simbol-simbol *Use case Diagram*

No	Simbol	Deskripsi
1.	<i>Use case</i> 	fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal-awal frase nama <i>use case</i> .
2.	aktor / <i>actor</i> 	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 di awal frase nama aktor.

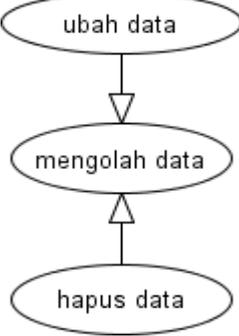


Lanjutan tabel 2.3. Simbol-simbol *Use case* Diagram

No	Simbol	Deskripsi
3.	asosiasi / <i>association</i> 	komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan actor.
4.	ekstensi / <i>extend</i> 	relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang di tambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misalnya  <pre>graph TD; A([Validasi Username]) -- <<extend>> B([Validasi User]); B -- <<extend>> C([Validasi Sidik Jari]);</pre> arah panah mengarah pada <i>use case</i> yang ditambahkan; biasanya <i>use case</i> yang menjadi <i>extend</i> -nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya

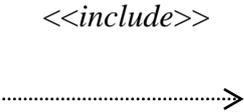
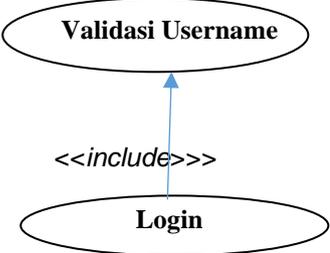
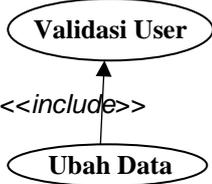


Lanjutan tabel 2.3. Simbol-simbol *Use case* Diagram

No	Simbol	Deskripsi
5.	Generalisasi <i>/</i> <i>Generalization</i> 	hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya,  misalnya: arah panah mengarah pada <i>use case</i> yang menjadi generalisasinya (umum).



Lanjutan tabel 2.3. Simbol-simbol Use case Diagram

No	Simbol	Deskripsi
6.	menggunakan / include / uses  	<p>relasi tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini</p> <p>ada dua sudut pandang yang cukup besar mengenai include di <i>use case</i>:</p> <ul style="list-style-type: none">• <i>Include</i> berarti <i>use case</i> yang ditambahkan akan selalu di panggil saat <i>use case</i> tambahan dijalankan, misalnya pada kasus berikut:  <ul style="list-style-type: none">• <i>Include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang di tambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan, misal pada kasus berikut:  <p>kedua interpretasi di atas dapat dianut salah satu atau keduanya tergantung pada pertimbangan dan interpretasi yang dibutuhkan.</p>

Sumber: Sukamto dan Shalahuddin (2018:156)

Ada dua hal utama pada *use case* yaitu:

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

2.2.2.3 Pengertian *Activity Diagram*

Sukamto 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.”

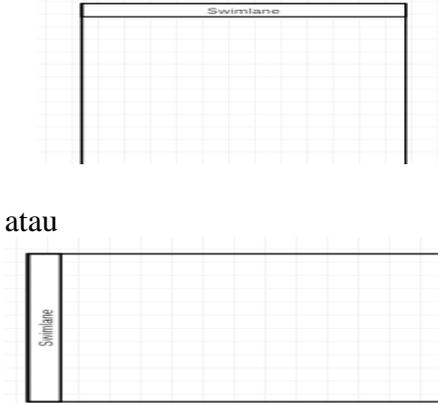
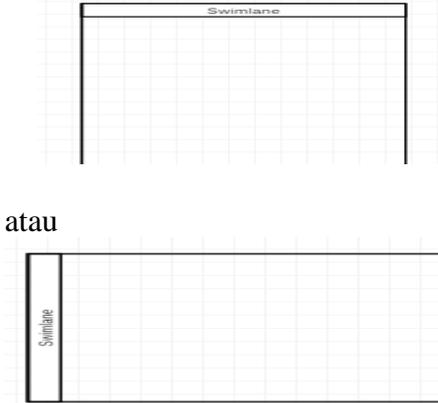
Adapun simbol-simbol yang digunakan dalam *activity diagram* adalah sebagai berikut:

Tabel 2.4 Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
1.	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
2.	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja
3.	Percabangan / <i>decision</i> 	Asosiasi percabangan di mana jika ada pilihan aktivitas lebih dari satu
4.	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
5.	Status akhir 	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir



Lanjutan Tabel 2.4 Simbol-simbol *Activity Diagram*

No	Simbol	Deskripsi
6.	Swimlane  atau 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

Sumber: Sukamto dan Shalahuddin (2018:162)

2.2.2.4 Pengertian *Sequence Diagram*

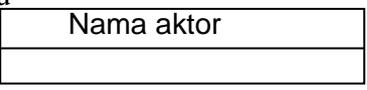
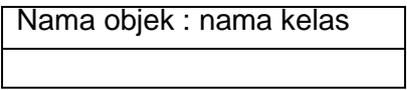
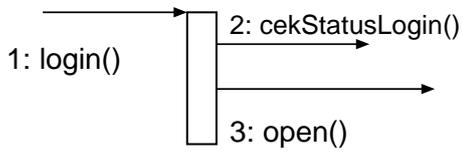
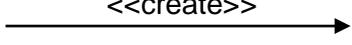
“Diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah beserta metode-metode yang dimiliki 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.

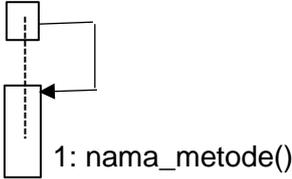
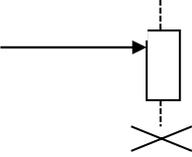


Berikut simbol-simbol pada Sequence Diagram :

Tabel 2.5 Simbol-simbol pada Sequence Diagram

No	Simbol	Deskripsi
1.	<p>Actor</p>  <p>nama aktor</p> <p>atau</p>  <p>tanpa waktu aktif</p>	<p>orang, proses, atau sistem lain yang berinteraksi dengan 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 di awal frase nama aktor</p>
2.	<p>Garis hidup / <i>lifeline</i></p> 	<p>menyatakan kehidupan suatu objek</p>
3.	<p>Objek</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 waktu aktif ini adalah sebuah tahapan yang dilakukan di dalamnya, misalnya</p>  <p>maka cekStatusLogin () dan open() dilakukan di dalam metode login() aktor tidak memiliki waktu aktif</p>
5.	<p>Pesan tipe create</p> 	<p>menyatakan suatu objek membuat objek yang lain, arah panah mengarah pada objek yang dibuat</p>

Lanjutan Tabel 2.5 Simbol-simbol pada Sequence Diagram

No	Simbol	Deskripsi
6.	Pesan tipe call 1: nama_metode() 	menyatakan suatu objek memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri,  arah panah mengarah pada objek yang memiliki operasi/metode, karena ini memanggil operasi/metode maka operasi/metode yang dipanggil harus ada pada diagram kelas sesuai dengan kelas objek yang berinteraksi
7.	Pesan tipe send 1: masukan 	menyatakan bahwa suatu objek mengirimkan data/masukan/informasi ke objek lainnya, arah panah mengarah pada objek yang dikirim
8.	Pesan tipe return 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 destroy <<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 (2018:165-167)



2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Indrajani (2018:3) menegaskan, “Aplikasi adalah program yang menentukan aktivitas pemrosesan informasi yang dibutuhkan untuk penyelesaian tugas-tugas khusus dari pemakai komputer”.

Pengertian Aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya, aplikasi merupakan suatu perangkat komputer yang siap pakai bagi *user*. (Melani, dkk, 2018).

Dari pendapat di atas dapat disimpulkan bahwa aplikasi adalah koleksi *window* atau program yang berfungsi bagi pengguna untuk keperluan tertentu.

2.3.2 Pengertian Pengolahan Data

Pengolahan Data adalah suatu proses menerima data sebagai masukan (*input*) memproses (*processing*) menggunakan proses tertentu, dan mengeluarkan hasil proses data tersebut dalam bentuk informasi (*output*). (Marhamelda, 2018).

2.3.3 Pengertian Pembayaran

Pengertian pembayaran adalah pelunasan utang oleh debitur kepada kreditur. Pembayaran seperti ini dilakukan dalam bentuk uang atau barang. Namun, pengertian pembayaran dalam arti yuridis tidak hanya dalam bentuk uang atau barang, tetapi juga dalam bentuk jasa, seperti jasa dokter dan lain-lain. (Evy, 2019).

2.3.4 Pengertian Pemesanan

Pemesanan adalah pelunasan utang oleh debitur kepada kreditur. Pembayaran seperti ini dilakukan dalam bentuk uang atau barang. Namun, pengertian pembayaran dalam arti yuridis tidak hanya dalam bentuk uang atau barang, tetapi juga dalam bentuk jasa, seperti jasa dokter dan lain-lain. (Satri & Seabtian, 2019).



2.4. Teori Program

2.4.1 Pengertian Sublime Text

Putratama (2018:14) Sublime Text merupakan perangkat lunak *text editor* yang digunakan untuk membuat atau meng-*edit* suatu aplikasi.

Syifani dan Ardiansyah (2018:25) menegaskan, “*Sublime Text* adalah aplikasi editor untuk kode dan teks yang dapat berjalan diberbagai platform *operating system* dengan menggunakan teknologi *Phyton API*”.

2.4.2 Pengertian HTML

Menurut Indrajani (2018:55) merupakan bahasa standar yang digunakan untuk mendesain seluruh halaman web, di mana kita dapat mengontrol tampilan web page dan *content*-nya, mempublikan dokumen secara *online*, membuat *form online* untuk pendaftaran atau transaksi, dan menambahkan objek-objek, seperti *image*, audio, video, dan *java applet* ke dalam dokumen HTML.

Menurut Hidayatullah dan Jauhari (2020:15), “HTML adalah Bahasa *standard* yang digunakan untuk menampilkan *web*”.

2.4.3 Pengertian CSS

Abdullah (2018:45) menegaskan, “CSS adalah singkatan dari *Cascading Style Sheet* yaitu dokumen web yang berfungsi mengatur elemen HTML dengan berbagai property yang tersedia sehingga dapat tampil dengan berbagai gaya yang diinginkan.”

Sulistiono (2018:2) menegaskan, “CSS (*cascading style sheet*) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah web sehingga akan lebih terstruktur dan seragam”.

2.4.4 Pengertian Bootstrap

Menurut Alatas dalam Sa’ad (2020:31), “Bootstrap adalah *framework* atau tools CSS yang memudahkan pengembangan untuk membangun website menarik dan responsif. Bootstrap memberikan solusi rapi dan seragam terhadap solusi umum, tugas *interface* yang setiap pengembang hadapi”.



Menurut Abdulloh (2018:261), “Bootstrap merupakan *framework* CSS paling populer dari sekian banyak *framework* CSS yang ada. Bootstrap memungkinkan desain sebuah web menjadi responsive sehingga dapat dilihat dari berbagai macam ukuran device dengan tampilan tetap menarik”.

2.4.5 Pengertian PHP

Menurut Hidayatullah dan Jauhari (2020:223), “PHP adalah suatu Bahasa *scripting* khususnya digunakan untuk *web development*”.

Putratama (2018:3) PHP adalah suatu bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat *server-side* yang dapat ditambahkan ke dalam HTML.

2.4.6 Pengertian Code Igniter

Code Igniter adalah sebuah aplikasi open source yang berupa kerangka kerja atau framework untuk membangun website menggunakan bahasa pemrograman PHP. Tujuannya memungkinkan pengembangan proyek yang lebih cepat daripada penulisan kode dasar atau kode terstruktur, dengan menyediakan banyak Library yang biasanya digunakan dalam pengerjaan. (Rachman & Marjito 2020).

2.4.7 Pengertian MySQL

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

Dwiartara (2020:6) MySQL adalah database, database sendiri merupakan suatu jalan untuk dapat menyimpan berbagai informasi dengan membaginya berdasarkan kategori-kategori tertentu. Dimana informasi-informasi tersebut saling berkaitan, satu dengan yang lainnya.



Jadi, dapat disimpulkan, MySQL adalah sebuah perangkat lunak basis data yang memiliki fungsi sebagai *relational database manajemen system* RDBMS yang multithread dan multi-user.

2.4.8 Pengertian XAMPP

Idcloud (2019) menegaskan, “XAMPP adalah perangkat lunak gratis yang mendukung untuk banyak sistem operasi, yang merupakan kompilasi dari beberapa program. Fungsi XAMPP sendiri adalah sebagai server yang berdiri sendiri (localhost), yang terdiri beberapa program antara lain Apache HTTP Server, *MySQL* database, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.”

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.

