



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Puspitosari (2013:1), “Komputer adalah sebuah alat yang digunakan untuk mengolah data menurut perintah yang telah dirumuskan”.

Kadir (2013:2), “Komputer merupakan peralatan elektronis yang biasa dipakai orang untuk membantu pelaksanaan pekerjaan. Sebagai contoh, mahasiswa menggunakan program statistik seperti SPSS untuk mengolah hasil penelitian dan sekretaris menggunakan pengolah kata seperti Microsoft Word atau OpenOffice Writer untuk membuat surat dan dokumen”.

Dari pernyataan di atas dapat disimpulkan, bahwa komputer adalah alat yang digunakan untuk mengolah data yang dipakai orang untuk membantu pelaksanaan pekerjaan.

2.1.2 Pengertian Perangkat Keras

Puspitosari (2013:7), “*Hardware* merupakan salah satu elemen dari sistem komputer suatu alat yang bisa dilihat dan diraba oleh manusia secara langsung yang mendukung proses komputerisasi”.

Kadir (2013:2), “Perangkat keras berupa peranti-peranti yang terlihat secara fisik. Termasuk dalam kelompok ini adalah monitor, keyboard, mouse dan printer”.

Jadi dapat disimpulkan dari pernyataan di atas, bahwa perangkat keras adalah alat yang bisa dilihat dan diraba oleh manusia secara fisik.

2.1.3 Pengertian Perangkat Lunak

Puspitosari (2013:7), “*Software* merupakan kumpulan beberapa perintah yang dieksekusi oleh mesin komputer dalam menjalankan tugasnya”.



Kadir (2013:2), “Perangkat lunak berupa intruksi-intruksi yang ditujukan kepada komputer agar komputer dapat melaksanakan tugas sesuai dengan kehendak pemakai”.

Dari pernyataan di atas dapat disimpulkan, bahwa perangkat lunak adalah kumpulan perintah atau intruksi yang dieksekusi komputer dalam menjalankan tugasnya.

2.1.4 Pengertian Data

Asropudin (:22), “kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti. Data dapat diolah sehingga menghasilkan informasi”.

Setyaningrum (2013:1), “Data adalah catatan atas sekumpulan fakta yang belum mempunyai arti bagi penerimanya dan masih memerlukan suatu pengolahan”.

Dari pernyataan di atas dapat disimpulkan bahwa, Data adalah kejadian yang berbentuk masih mentah dan belum mempunyai arti bagi penerima yang masih perlu diolah.

2.1.5 Pengertian Sistem Informasi

Hutahaean (2014:13), “Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan”.

Sarosa (2017:2), “Sistem informasi didefinisikan sebagai organisasi yang menyediakan proses dan informasi yang berguna bagi anggota dan pemangku kepentingannya”.

2.1.6 Metode Pengembangan Sistem

Rosa dan Shalahuddin (2013:2) menjelaskan bahwa ada beberapa model SDLC (Software Development Life Cycle) yang dapat digunakan, salah satunya adalah model air terjun (*waterfall*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis,



desain, pengodean, pengujian, dan tahap pendukung. Berikut adalah tahapan dari model air terjun :

1. Analisis kebutuhan perangkat lunak

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

2. Desain

Desain perangkat lunak adalah proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean. Tahap ini mentranslasi kebutuhan perangkat lunak dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

3. Pembuatan kode program

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

4. 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.

5. Pendukung (*support*) atau pemeliharaan (*maintenance*)

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pendukung atau pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis



spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

2.2 Teori Khusus

2.2.1 Pengertian *Data Flow Diagram* (DFD)

Saputra (2013:118) menyatakan *Data Flow Diagram* atau yang disingkat DFD merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas. DFD juga dapat diartikan sebagai teknis grafis yang menggambarkan alir data atau masukan menuju keluaran atau *output*.

Rosa dan Shalahudin (2013:69) Menyatakan *Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD :

1. Membuat DFD Level 0 atau sering disebut juga Context Diagram DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas luar.
2. Membuat DFD Level 1
DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem yang akan dikembangkan. DFD Level 1 merupakan hasil breakdown DFD Level 0 yang sebelumnya sudah dibuat.
3. Membuat DFD Level 2
Modul-modul pada DFD Level 1 dapat di-breakdown menjadi DFD Level 2. Modul mana saja yang harus di-breakdown lebih detail tergantung pada kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu di-breakdown lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level 1 yang di-breakdown.

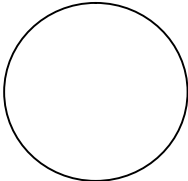



4. Membuat DFD Level 3 dan seterusnya

DFD Level 3, 4, 5, dan seterusnya merupakan breakdown dari modul pada DFD Level di-atasnya. Breakdown pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2

Dari pernyataan di atas dapat disimpulkan bahwa, *Data Flow Diagram* adalah proses yang menggambarkan aliran informasi dari masukan dan keluaran yang menyimpan data, proses yang menghasilkan data tersebut.

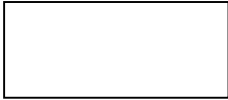

Tabel 2.1 Notasi-notasi pada DFD (Edward Yourdan dan Tom Demarco)

No	Proses	Keterangan
1		Prosedur atau fungsi atau prosedur, pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program. Catatan: Nama yang diberikan pada sebuah proses biasanya kata kerja.
2.		File atau basis data atau penyimpanan (storage); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (Entity Relationship Diagram (ERD), Conceptual Data Model (CDM), Physical Data Model (PDM)). Catatan: Nama yang diberikan pada sebuah penyimpanan biasanya berupa kata benda.



--	--	--

Lanjutan Tabel 2.1 Notasi-notasi pada DFD (Edward Yourdan dan Tom Demarco)

3.		<p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakai/berinteraksi dengan perangkat lunak yang dimodelkan atau sistem lain yang terkait dengan aliran data dari sistem yang dimodelkan.</p> <p>Catatan: Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda</p>
4		<p>Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>).</p> <p>Catatan: Nama yang digunakan pada aliran data biasanya berupa data benda, dapat diawali dengan kata data misalnya “data siswa” atau tanpa kata data misalnya “siswa”</p>

Sumber: Rosa dan shalahuddin (2013:71-72)

2.2.2 Pengertian *Block Chart*



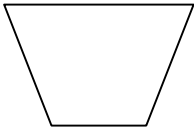

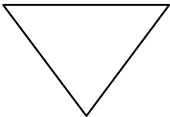
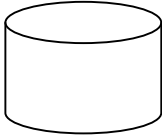
Kristanto (2008:75), “Block Chart berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu.



Pembuatan *Block Chart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

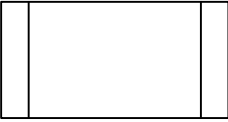
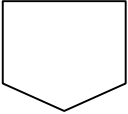
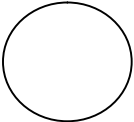
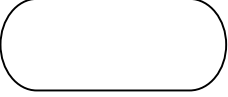
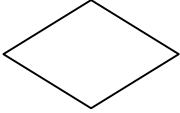
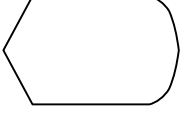

Adapun simbol-simbol yang sering digunakan dalam *Block Chart* dapat dilihat pada tabel berikut ini:

Tabel 2.2 Simbol-simbol dalam *Block Chart*

No	Simbol	Arti
1.		Menandakan dokumen bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.
2.		Multi dokumen
3.		Proses manual
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (data storage)



Lanjutan Tabel 2.2 Simbol-simbol dalam *Block Chart*

7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain
9.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran
11.		Pengambilan keputusan (decision)
12.		Layar peraga (monitor)
13.		Pemasukan data secara manual

Sumber : Kristanto (2008:75)



2.2.3 Pengertian *Flowchart*



Saputra (2013:120), “*flowchart* merupakan suatu diagram yang menggambarkan alur kerja suatu sistem”.

Jogiyanto (2006:662), “Bagan alir program (program *flow chart*) adalah suatu bagan yang menggambarkan arus logika dari data yang akan diproses dalam suatu program dari awal sampai akhir”.


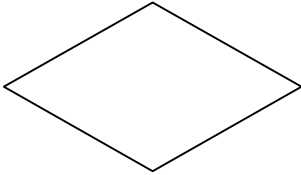

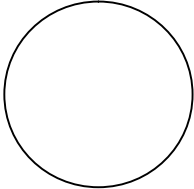
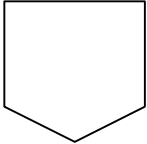
Dari pernyataan di atas dapat disimpulkan, bahwa *flowchart* adalah diagram yang menggambarkan alur kerja suatu sistem yang akan diproses dalam suatu program dari awal sampai akhir”.

Berikut ini adalah simbol-simbol program *flow chart* menurut ANSI (American National Standard Institute):


Tabel 2.3 Simbol-simbol pada *flowchart*

No	Simbol	Deskripsi
1.	Simbol Terminal (<i>terminal symbol</i>) 	Digunakan untuk menunjukkan awal dan akhir program
2.	Simbol Persiapan (<i>preparation symbol</i>) 	Digunakan untuk memberikan nilai awal dari suatu variable atau counter

Lanjutan Tabel 2.3 Simbol-simbol pada *flowchart*

3.	<p>Simbol Pengolahan (processing symbol)</p> 	Digunakan untuk pengolahan arthmatika dan pemindahan data
4.	<p>Simbol Keputusan (decision symbol)</p> 	Digunakan untuk mewakili operasi perbandingan logika
5.	<p>Simbol proses terdefinisi (<i>predefined process symbol</i>)</p> 	Digunakan untuk proses yang detilnya dijelaskan terpisah, misalnya dalam bentuk <i>subroutine</i>
6.	<p>Simbol penghubung (<i>connector symbol</i>)</p> 	Digunakan untuk menunjukkan hubungan arus proses yang terputus masih dalam halaman yang sama
7.	<p>Simbol penghubung halaman lain (<i>off page connector symbol</i>)</p> 	Digunakan untuk menunjukkan hubungan arus proses yang terputus masih dalam halaman yang sama

Lanjutan Tabel 2.3 Simbol-simbol pada *flowchart*

7.	Simbol penjelasan (<i>annotation flag symbol</i>) 	Digunakan untuk memberikan keterangan-keterangan guna memperjelas simbol-simbol yang lain
----	--	---

Sumber: Jogiyanto (2006:662)

2.2.4 Pengertian *Entity Relationship Diagram*

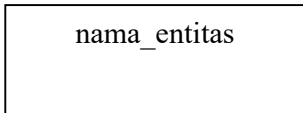
Setyaningrum (2013:17), “*Entity Relationship Diagram* (ERD) merupakan model data berupa notasi grafis dalam pemodelan data konseptual yang menggambarkan hubungan antar penyimpanan”.

Rosa dan Shalahuddin (2013:50), “Pemodelan awal basis data yang paling banyak digunakan adalah menggunakan *Entity Relationship Diagram* (ERD).

ERD digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow’s Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen.

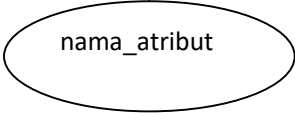
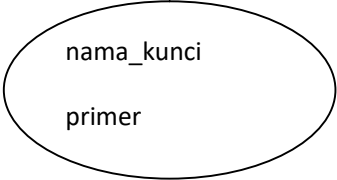
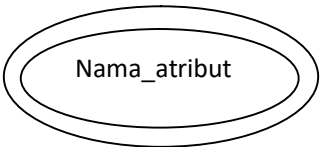
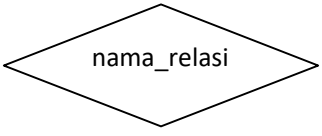
Berikut adalah simbol-simbol yang digunakan pada *ERD* dengan notasi Chen:

Tabel 2.4 Simbol-simbol pada ERD dengan Notasi Chen

No	Simbol	Deskripsi
1.	Entitas / <i>Entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel



Lanjutan Tabel 2.4 Simbol-simbol pada ERD dengan Notasi Chen

2.	<p>Atribut</p> 	<p>Field atau kolom data yang butuh disimpan dalam suatu entitas</p>
3.	<p>Atribut Kunci Primer</p> 	<p>Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)</p>
4.	<p>Atribut multi nilai / <i>multi value</i></p> 	<p><i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu</p>
5.	<p>Relasi</p> 	<p>Relasi yang menghubungkan antar entitas, biasanya diawali dengan kata kerja</p>

**Tabel 2.5** Simbol-simbol pada Kamus Data

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
[]	Baik . . . atau . . .
{ } ⁿ	n kali diulang/ bernilai banyak
()	Data opsional
* . . . *	Batas komentar

Sumber :Rosa dan Shalahuddin(2013:74)

2.3 Teori Judul

2.3.1 Pengertian Aplikasi

Menurut Sujatmiko (2012:23), “Aplikasi adalah program komputer yang dibuat suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu”.

Menurut Asropudin (2013:6) “Aplikasi adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu misalnya MS-Word, MS-Excel.

Dari beberapa pengertian aplikasi diatas dapat ditarik kesimpulan bahwa aplikasi adalah suatu program yang memiliki fungsi tertentu sesuai dengan kebutuhan pengguna komputer.

2.3.2 Pengertian Layanan

Menurut Atmadji (2018:1), “Layanan adalah suatu kegiatan atau urutan kegiatan yang terjadi dalam interaksi langsung antara seseorang dengan orang lain atau mesin secara fisik dan menyediakan kepuasan pelanggan”.

Menurut Windasuri et al (2016:92), “Layanan adalah perilaku atau sikap yang berlandaskan ketulusan dan empati, untuk membantu pelanggan”.

Dari kedua pengertian tersebut dapat diketahui bahwa layanan adalah kegiatan yang berlandaskan ketulusan dan empati dalam interaksi langsung seseorang dengan orang lain.



2.3.2.1 Kualitas Layanan

Konsep kualitas dianggap sebagai ukuran relative kebaikan suatu produk atau jasa yang terdiri dari desain kualitas dari desain kualitas dan kualitas kesesuaian. Pengertian kualitas paling mendasar adalah bebas dari cacat. Kebanyakan perusahaan bertumpu pada pelanggan mendefinisikan kualitas sebagai kepuasan pelanggan.

Kotler et al dalam Paweningtyas (2016:41) mengemukakan bahwa kualitas (*quality*) adalah totalitas fitur dan karakteristik produk atau jasa yang bergantung pada kemampuannya untuk memuaskan kebutuhan konsumen atau pengguna jasa yang dinyatakan. Ada lima dimensi atau lima faktor utama pelayanan yang digunakan konsumen untuk menilai atau menenukan kualitas pelayanan. Kelima dimensi tersebut adalah sebagai berikut:

- a. Keandalan (*Reability*) yaitu kemampuan orang memberikan layanan yang dijanjikan dengan segera, akurat dan memuaskan
- b. Daya tanggap (*Responsiveness*) yaitu keinginan para staf untuk membantu para pelanggan dan memberikan layanan dengan tanggap
- c. Jaminan (*Asurance*) mencakup pengetahuan, kompetensi, kesopanan dan sifat dapat dipercaya yang dimiliki oleh para staf, bebas dari bahaya, risiko atau keraguan.
- d. Empati (*Emphaty*) meliputi kemudahan dalam menjalin relasi, komunikasi yang baik, perhatian pribadi dan pemahaman atas kebutuhan individual para pelanggan.
- e. Bukti fisik (*Tangible*) meliputi fasilitas fisik, perlengkapan, pegawai dan sarana komunikasi.

Layanan dapat dibagi menjadi 4 kuadran layanan:

1. *The Freezer*

Pemberi layanan dalam kelompok ini biasanya dilabeli pelanggan sebagai petugas yang “menyebalkan”. Dimata pelanggan, pemberi layanan memiliki kesan buruk, baik personal maupun prosedural. Pelanggan dilayani berdasarkan kepentingan dirinya, tidak berorientasi pada layanan.



2. *The Friendly Zoo*

Pemberi layanan dalam *Friendly Zoo* tidak sepenuhnya buruk. Ketika berhadapan dengan pegawai, pelanggan mendapat kesan bahwa pegawai sudah berusaha memahami, sudah berusaha keras mencari solusi untuk permintaan dan permasalahan pelanggan.

3. *The Factory*

The Factory justru tidak memperlihatkan dimensi personal yang menyenangkan, walaupun dari sisi procedural, layanan yang diberikan cukup memuaskan. Pelanggan yang datang dilayani untuk memecahkan keluhannya atau dipenuhi kebutuhannya tanpa sentuhan personal.

4. *Quality Customer Service*

Kelompok layanan ini adalah tipe ideal yang diharapkan dimiliki setiap perusahaan dan ditemui oleh setiap pelanggan sebab pemberi layanan pelanggan yang berkualitas mampu memadukan keunggulan dimensi prosedural dengan kualitas personal yang profesional dan berorientasi pada layanan. Pelanggan yang datang mendapat layanan yang ramah dan memperoleh kesan bahwa mereka berhadapan dengan seseorang yang sungguh peduli.

2.3.3 Pengertian Jasa

Menurut Kotler dalam Pawestriningtyas (2016:41), “Jasa merupakan setiap tindakan atau perbuatan yang dapat ditawarkan oleh suatu pihak kepada pihak lain dalam hal ini misalnya penjual dengan pembeli dan pada dasarnya bersifat tidak berwujud fisik (*intangible*) dan tidak menghasilkan kepemilikan sesuatu sehingga jasa tersebut tidak dapat dilihat”.

Menurut Zeithaml dalam Pawestriningtyas (2016:41), “Jasa adalah suatu kegiatan ekonomi yang *outputnya* bukan produk dikonsumsi yang penggunaannya bersamaan dengan waktu produksi dan memberikan nilai tambah kepada konsumen (seperti kenikmatan, hiburan, santai, sehat) serta biasanya bersifat tidak berwujud”.



Dari kedua pengertian tersebut dapat diketahui bahwa jasa tidak dapat dilihat atau diraba dikarenakan tidak memiliki wujud namun dapat dirasakan oleh pengguna jasa tersebut.

Kotler dalam Pawestriningtyas (2016:41) menyatakan bahwa jasa memiliki beberapa karakteristik yang secara umum diklasifikasikan dalam 4 karakteristik, yaitu:

a. Tidak berwujud (*Intangibility*)

Jasa berbeda dengan barang. Bila barang dapat dimiliki. Jasa bersifat *Intangibility* / tidak dapat dilihat, dirasa, dicium, didengar atau diraba sebelum dibeli dan dikonsumsi. Dengan demikian, orang tidak dapat menilai kualitas jasa tersebut sebelum ia merasakan/ mengkonsumsinya sendiri.

b. Tidak dapat dipisahkan (*Inseparability*)

Jasa bersifat *Inseparability* artinya bahwa dalam memasarkan jasa interaksi antara penyedia jasa dan pelanggan merupakan ciri khusus dalam pemasaran jasa, keduanya mempengaruhi hasil jasa tersebut.

c. Variabilitas (*Perishability*)

Jasa bersifat variabel karena merupakan *non-standardized output*, artinya banyak variasi bentuk, kualitas dan jenis, tergantung pada siapa, kapan dan dimana jasa tersebut dihasilkan.

d. Tidak tahan lama (*Perishability*)

Jasa merupakan komoditas tidak tahan lama dan tidak dapat disimpan. Dengan demikian bila jasa tidak digunakan, maka jasa tersebut akan berlalu begitu saja.

Menurut Bery dalam Pawestriningtyas(2016:41) menjelaskan 3 karakteristik jasa, yaitu:

1. Lebih bersifat tidak berwujud daripada berwujud (*More intangible than tangible*).
2. Produksi dan konsumsi bersamaan waktu (*Simultaneous production and consumption*).
3. Kurang memiliki standar dan kesamaan (*less standardized and uniform*).



2.3.4 Pengertian Aplikasi Layanan Jasa di Austin Car Wash Kabupaten Muara Enim

Kegiatan pencucian mobil yang berlandaskan ketulusan dan empati dalam interaksi langsung seseorang dengan orang lain yang tidak dapat dilihat atau diraba dikarenakan tidak memiliki wujud namun dapat dirasakan dari pihak Austin Car Wash kepada pelanggan yang datang dengan menggunakan suatu program sesuai dengan kebutuhan di Kabupaten Muara Enim.

2.4 Teori Program

2.4.1 Pengertian Basis Data

Ramakrishnan dan Gehrke (2003), “Basis data adalah kumpulan data yang terdiri atas *entity* dan *relationship*, umumnya mendeskripsikan aktivitas dari satu organisasi atau lebih yang berhubungan”.

Stephens dan Plew (2000) menyatakan, basis data sebagai mekanisme yang digunakan untuk menyimpan informasi atau data. Sedangkan menurut Connolly dan Carolyn (2005), basis data adalah sebuah koleksi data yang dipakai bersama dan terhubung secara logis.

Jadi, dapat disimpulkan bahwa basis data merupakan kumpulan data yang terdiri dari *atribut*, *entity* dan *relationship* dari informasi suatu instansi atau perusahaan yang dirancang untuk memenuhi kebutuhan informasi dari sebuah organisasi.

2.4.2 Pengertian MySQL (*My Structure Query Language*)

Nugroho (2004:29), “MySQL (*My Structure Query Language*) atau yang biasa dibaca “mai-se-kuel” adalah sebuah program pembuat database yang bersifat *open source*, artinya siapa saja boleh menggunakannya dan tidak dicekal”.

Bunafit (2013:1), “MySQL adalah Relational Database Management System (RDMS), yaitu database relasi yang memiliki perintah standar adalah SQL (Structured Query Language). MySQL termasuk Database Server, karna mendukung perintah SQL secara penuh dan dapat diakses dalam jaringan (bisa sebagai Server dan Client)”.



Dari pernyataan di atas dapat disimpulkan bahwa, MySQL adalah perangkat lunak pembuat database yang memiliki perintah.

2.4.3 Pengertian XAMPP

Riyanto (2014:11), “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 ke dalam satu paket”.

Nugroho (2013:1), “XAMPP adalah paket program berbasis web, di dalamnya berisi; Software Apache, PHP dan database MySQL”.

Dari pernyataan di atas dapat disimpulkan bahwa, XAMPP adalah program berbasis web yang merupakan paket PHP dan MySQL dan software Apache.

2.4.4 Pengertian HTML (*Hypertext Markup Language*)

Sulhan (2006:23), “*Hypertext Markup Language* (HTML) adalah suatu sistem untuk menambahkan dokumen dengan tabel yang menandakan bagaimana teks di dokumen harus disajikan dan bagaimana dokumen dihubungkan bersama”.

Pramono (2015:1), “HTML (*Hypertext Markup Language*) adalah bahasa pengkodean untuk menghasilkan dokumen-dokumen hypertext untuk digunakan di *World Wide Web*. Akan tetapi, HTML bukanlah sebuah bahasa pemrograman. HTML hanya berisi perintah-perintah yang telah terstruktur berupa tag-tag penyusun”.

Jadi dari pernyataan di atas dapat disimpulkan bahwa, HTML adalah bahasa pengkodean untuk menambahkan dokumen dengan tabel berisi perintah-perintah yang telah terstruktur tag-tag penyusun.

2.4.5 Pengertian PHP

Winarno dkk (2014:49), “PHP atau *Hypertext Preprocessor* adalah sebuah pemrograman web berbasis server (*server-side*) yang mampu parsing kode PHP



dari kode web dengan ekstensi .php, sehingga menghasilkan tampilan website yang dinamis di sisi client (browser)”.

Saputra (2013:4), “PHP merupakan bahasa *Server Side Scripting*, dimana PHP selalu membutuhkan web *server* dalam menjalankan aksinya”.

Secara prinsip, server akan bekerja apabila ada permintaan dari client, yaitu kode-kode php. Client tersebut akan dikirimkan ke server, kemudian server akan mengembalikan pada halaman sesuai intruksi yang diminta.

Jadi dapat disimpulkan dari pernyataan di atas bahwa, PHP adalah script pemrograman yang membutuhkan web *server*.

2.4.6 Pengertian CSS

Menurut Jayan (2012:2) mengemukakan bahwa “CSS merupakan singkatan dari *Cascading Style Sheet* yang digunakan untuk mengatur tampilan dokumen HTML. Contohnya seperti pengaturan jarak antar baris, teks, warna dan format border bahkan penampilan file gambar.

Wahana Komputer (2014:2), “CSS adalah bahasa yang digunakan untuk mengatur tampilan halaman web. CSS merupakan singkatan dari *Cascading Style Sheet*”.

Dari pernyataan di atas dapat disimpulkan bahwa, CSS adalah *Cascading Style Sheet* yang digunakan untuk mengatur tampilan halaman web seperti pengaturan jarak antar baris, teks, warna dan format border bahkan penampilan file gambar.