



## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Teori Umum

##### 2.1.1 Pengertian Perangkat Lunak

Menurut Sukamto dan Shalahuddin (2018:2) “Perangkat lunak (*software*) adalah komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain dan cara penggunaan (*user manual*)”.

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

Berdasarkan pengertian diatas, dapat disimpulkan bahwa perangkat lunak adalah mesin komputer yang di instruksikan sesuai dengan kehendak pemakai berfungsi sebagai alat untuk melaksanakan tugas.

##### 2.1.2 Pengertian Komputer

Menurut kadir (2017:2) “Komputer merupakan peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia”.

Sedangkan menurut Krisbiyanto(2018:1) “Komputer merupakan alat yang dipakai untuk mengolah dan memproses data menurut perintah yang telah dirumuskan”.

Jadi dapat disimpulkan bahwa komputer adalah kumpulan perangkat lunak elektronik yang saling berhubungan dan berinteraksi satu sama lain untuk melakukan proses pengolahan data dengan berbagai perintah program sehingga dapat menghasilkan informasi yang dibutuhkan penggunanya.

##### 2.1.3 Pengertian Basis Data

Menurut Winarno dan Utomo dalam Prayitno dan Safitri (2015:2) “*Database* atau biasa disebut basis data merupakan kumpulan data yang saling berhubungan. Data tersebut biasanya terdapat dalam table-tabel yang saling



berhubungan satu sama lain, dengan menggunakan *field* atau kolom pada tiap tabel yang ada”. Hal serupa dikemukakan lagi oleh Abdullah (2018:103) menyatakan bahwa “*Database* atau basis data adalah kumpulan informasi yang disimpan dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi”.

Berdasarkan pendapat di atas, dapat disimpulkan bahwa Basis Data merupakan suatu kumpulan data atau fakta yang terdapat dalam tabel-tabel dan saling berhubungankemudian disimpan secara bersama sedemikian rupa untuk memperoleh informasi.

## **2.2 Teori Khusus**

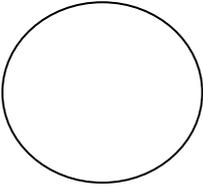
### **2.2.1 Pengertian *Data Flow Diagram (DFD)***

Sukamto dan Shalahuddin (2016:70) berpendapat bahwa “*Data Flow Diagram (DFD)* atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.



Adapun simbol-simbol yang sering digunakan dalam *Data Flow Diagram* (DFD) dapat dilihat pada tabel berikut ini :

**Tabel 2.1** Simbol-Simbol dalam *Data Flow Diagram* (DFD)

No	Notasi	Keterangan
1.		<p>Proses 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</p> <p>Catatan : nama yang diberikan pada sebuah proses biasanya berupa kata kerja</p>
2.		<p><i>File</i> atau basis data atau penyimpanan (<i>storage</i>); 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 (<i>Entity Relationship Diagram</i> (ERD), <i>Conceptual Data Model</i> (CDM), <i>Physical Data Model</i> (PDM))</p> <p>Catatan : nama yang diberikan pada sebuah penyimpanan biasanya kata benda.</p>



**Lanjutan Tabel 2.1** Simbol-Symbol dalam *Data Flow Diagram (DFD)*

No	Notasi	Keterangan
3.		<p>Entitas luar (eksternal <i>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 kata benda, dapat diawali dengan kata data misalnya "data siswa" atau tanpa kata data misalnya "siswa"</p>

Sumber: Sukamto dan Shalahuddin (2016:71-72)

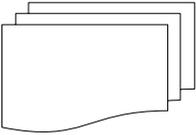
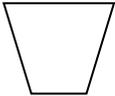
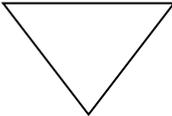
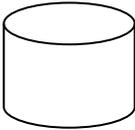
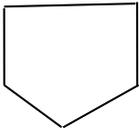


### 2.2.2 Pengertian *Blockchart*

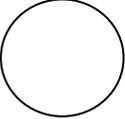
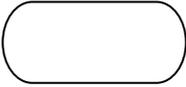
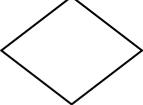
Menurut Kristanto (2018:75) “*Blockchart* berfungsi untuk memodelkan masukan, kelaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

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

**Tabel 2.2** Simbol-Simbol dalam *Blockchart*

No	Simbol	Arti
1.		Multi Dokumen
2.		Proses Manual
3.		Proses yang dilakukan oleh komputer
4.		Menandakan dokumen yang diarsipkan (arsip manual)
5.		Data penyimpanan ( <i>Storage</i> )
6.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik.
7.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.

Lanjutan Tabel 2.2 Simbol-Simbol dalam *Blockchart*

No	Simbol	Arti
8.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
9.		Terminal yang menandakan awal dan akhir dari suatu aliran.
10.		Pengambilan keputusan ( <i>Decision</i> ).
11.		Layar peraga ( <i>monitor</i> ).
12.		Pemasukkan data secara manual.

Sumber : Kristanto (2018:75-77)

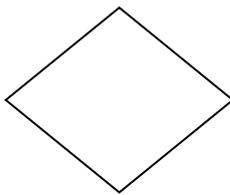
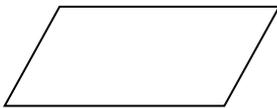


### 2.2.3 Pengertian *Flowchart*

*Flowchart* merupakan gambaran secara grafik dari langkah-langkah dan urutan prosedur suatu program (Indrajani. dikutip Rusmawan, 2019:48).

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

**Tabel 2.3** Simbol-Simbol dalam *Flowchart*

No	Gambar	Simbol Untuk...	Keterangan
1.		Proses/Langkah	Menyatakan Kegiatan yang akan ditampilkan dalam diagram alir
2.		Titik Keputusan	Proses/langkah dimana perlu adanya keputusan atau adanya kondisi tertentu. Dititik ini selalu ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda.
3.		Masukan/Keluaran Data	Digunakan untuk mewakili data masuk, atau data keluar.
4.		Terminasi	Menunjukkan awal atau akhir sebuah proses.
5.		Garis Alir	Menunjukkan arah aliran proses atau algoritma.

Lanjutan Tabel 2.3 Simbol-Simbol dalam *Flowchart*

No	Gambar	Simbol Untuk...	Keterangan
6.		Kontrol/Inspeksi	Menunjukkan proses atau langkah dimana ada inpeksi atau pengontrolan.

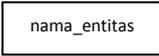
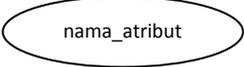
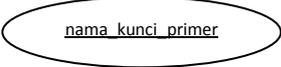
Sumber : Rusmawan (2019:49)

#### 2.2.4 Pengertian *Entity Relationship Diagram (ERD)*

Menurut Sukamto dan Shalahuddin (2018:50) “*Entity Relationship Diagram (ERD)* adalah pemodelan awal basis data yang paling banyak digunakan. ERD digunakan untuk pemodelan basis data relasional”.

Adapun simbol-simbol yang sering digunakan dalam *Entity Relationship Diagram (ERD)* dapat dilihat pada tabel berikut ini:

Tabel 2.4 Simbol-Simbol dalam *Entity Relationship Diagram (ERD)*

No	Sistem	Keterangan
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 computer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama table
2.	Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
3.	Atribut kunci primer 	<i>Field</i> 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)



**Lanjutan Tabel 2.4** Simbol-simbol dalam *Entity Relationship Diagram* (ERD)

No	Sistem	Keterangan
4.	Atribut multinilai / <i>multivalued</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja
6.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian  Kemungkinan jumlah maksimum keterhubungan antara entitas yang lain disebut dengan kardinalitas.

Sumber: sukamto dan Shalahuddin (2013:50)



### 2.2.5 Pengertian Kamus Data

Sukamto dan Shalahuddin (2016:73) menjelaskan, “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan).

Kamus data memiliki beberapa simbol sebagai berikut:

**Tabel 2.5** Simbol-simbol Kamus Data

No	Simbol	Keterangan
1.	=	Disusun atau terdiri dari
2.	+	Dan
3.	[]	Baik...atau...
4.	{ } <sup>n</sup>	N kali diulang/bernilai banyak
5.	()	Data opsional
6.	*..*	Batas komentar

Sumber: sukamto dan Shalahuddin (2016:73)

### 2.2.6 Metode Pengembangan Sistem

Metode pengembangan perangkat lunak yang penulis gunakan adalah Metode Air Terjun (*Waterfall*). Menurut Sukamto dan Shalahuddin (2018:28) Model SDLC air terjun (*Waterfall*) sering juga disebut model sekuensial linier (sequential linier) atau alur hidup klasik (classic life cycle). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian, dan tahap pendukung (support)



Berikut adalah gambaran model air terjun :

a. 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.

1. Penyelidikan Awal

Kristanto (2018:41) Pada tahap penyelidikan awal, analisis belajar dari pemakai mengenai apa yang diharapkan dari sebuah sistem informasi yang baru

Hal-hal yang perlu diperhatikan dalam tahapan ini adalah:

- a. Mencoba memahami dan memperjelas apa yang diharapkan oleh pemakai (sistem informasi bagaimana yang mereka perlukan).
- b. Menentukan ruang lingkup dari studi sistem informasi.
- c. Menentukan kelayakan dari masing-masing alternatif dengan memperkirakan keuntungan/kerugian yang didapat.

2. Studi Kelayakan

Menurut Jogiyanto dalam Syaifullah dan Widiyanto (2014:201) “Suatu studi kelayakan (Feasibility study) adalah suatu studi yang akan digunakan untuk menentukan kemungkinan apakah pengembangan proyek sistem layak diteruskan atau dihentikan. Studi kelayakan disebut juga dengan istilah High point review”

Adapun Hal-hal dalam studi kelayakan adalah sebagai berikut:

a. Kelayakan Teknis

Kelayakan teknis menyoroiti kebutuhan sistem yang telah disusun dari aspek teknologi yang akan digunakan, jika teknologi yang dikehendaki untuk pengembangan sistem merupakan teknologi yang mudah didapat, murah, dan tingkat pemakaiannya mudah, maka secara teknis usulan kebutuhan sistem bisa dinyatakan layak (Al Fatta dalam Syaifullah dan Widiyanto, 2014:201).



b. Kelayakan Operasional

Penilaian terhadap kelayakan operasional digunakan untuk mengukur apakah sistem yang akan dikembangkan nantinya dapat dioperasikan dengan baik atau tidak di dalam organisasi (Jogiyanto dalam Syaifullah dan Widiyanto 2014:202).

c. Kelayakan Ekonomis

Aspek yang paling dominan dari aspek kelayakan yang lain adalah kelayakan ekonomi. Tidak dapat disangkal lagi, motivasi pengembangan sistem informasi pada perusahaan atau organisasi adalah motif keuntungan. Dengan demikian aspek untung rugi jadi pertimbangan utama dalam pengembangan sistem. Kelayakan ekonomi berhubungan dengan return investmen atau berapa lama biaya investasi dapat kembali (Al Al Fatta dalam Syaifullah dan Widiyanto, 2014:201).

b. 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.

c. Implementasi 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.

d. Pengujian

Pengujian fokus kepada perangkat lunak secara logic dan fungsional dan memastikan bahwa semua bagian sudah diuji untuk meminimalisir error dan keluaran harus sesuai. Pemilihan cara pengujian dilakukan dengan menggunakan data-data yang sering digunakan untuk pengolahan data, mulai dari data operasional, data input dan output.



### 1. Pengertian Metode Pengujian

Menurut Sukamto dan Shalahuddin (2016:272) “Pengujian adalah satu set aktifitas yang direncanakan dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan. Aktifitas pengujian terdiri dari satu set atau sekumpulan langkah dimana dapat menempatkan desain kasus uji yang spesifik dan metode pengujian.

Secara umum pola pengujian perangkat lunak adalah sebagai berikut:

- a. Pengujian dimulai dari level komponen hingga integrasi antar komponen menjadi sebuah sistem.
- b. Teknik pengujian berbeda-beda sesuai dengan berbagai isi atau unit uji dalam waktu yang berbeda-beda pula bergantung pada pengujian pada bagian mana yang dibutuhkan.
- c. Pengujian dilakukan oleh pengembang perangkat lunak, dan jika untuk proyek besar, pengujian bisa dilakukan oleh tim uji yang tidak terkait dengan tim pengembang perangkat lunak (*independent test group* (ITG)).
- d. Pengujian dan penirkutuan (*debugging*) merupakan aktivitas yang berbeda tetapi penirkutuan (*debugging*) harus diakomodasikan pada berbagai strategi pengujian.

### 2. Black-Box Testing

Menurut Sukamto dan Shalahuddin (2016:275), “*Black-box testing* (pengujian kotak hitam) yaitu menguji perangkat lunak dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi-fungsi masukan dan keluaran dari perangkat lunak sesuai dengan spesifikasi yang dibutuhkan.”



Kasus uji yang dibuat untuk melakukan pengujian kotak hitam harus dibuat dengan kasus benar dan kasus salah, misalkan untuk kasus proses login maka kasus uji coba yang dilakukan adalah:

1. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang benar.
  2. Jika user memasukkan nama pemakai (*username*) dan kata sandi (*password*) yang salah, misalkan nama pemakai benar tapi kata sandi salah, atau sebaliknya atau keduanya salah.
- e. Pendukung (support) atau pemeliharaan (maintenance).

Dikarenakan adanya perubahan ketika sudah dikirimkan ke user. Perubahan dapat 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 yang baru.

## 2.3 Teori Judul

### 2.3.1 Pengertian Aplikasi

Menurut Juansyah (2015:2) menyatakan bahwa “Aplikasi adalah suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju”.

Sedangkan Menurut Tirtobisono dalam Sallaby, dkk (2015:172) “Aplikasi adalah istilah yang digunakan untuk pengguna komputer bagi pemecahan masalah. Biasanya istilah aplikasi dipasangkan atau digabungkan dengan suatu perangkat lunak”.

Jadi, dari definisi di atas dapat disimpulkan bahwa, aplikasi adalah suatu program yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi yang digunakan sebagai pemecahan masalah.



### 2.3.2 Pengertian Pengolahan Data

Menurut Haerani dan Robiyanto(2019:104) “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Sedangkan Menurut Tata Sutabri dalam Mahyuni, dkk (2014:182) “Pengolahan Data adalah suatu proses menerima data sebagai masukan (*input*), memproses (*processing*) menggunakan program tertentu, dan mengeluarkan hasil proses data tersebut dalam bentuk informasi (*output*).”

Jadi, dari definisi di atas dapat disimpulkan bahwa, Pengolahan Data adalah suatu proses data masukan, proses, dan mengeluarkan hasil proses data dalam bentuk informasi yang memiliki kegunaan.

### 2.3.3 Pengertian Inventaris

Menurut Andani dan Fithri (2016:269) “Inventaris adalah suatu kegiatan yang bertujuan untuk mengelola persediaan barang aset kantor atau perusahaan yang dimiliki”.

Sedangkan Menurut Kamus Besar Bahasa Indonesia (KBBI) Online, Inventaris adalah daftar yang memuat semua barang milik kantor (sekolah, perusahaan, kapal, dan sebagainya) yang dipakai dalam melaksanakan tugas.

Jadi, dari definisi di atas dapat disimpulkan bahwa, Inventaris adalah semua kegiatan yang bertujuan mengelola persediaan barang milik kantor yang akan dipakai untuk melaksanakan tugas.

### 2.3.4 Pengertian Sarana

Menurut Kamus Besar Bahasa Indonesia (KBBI) Online, Sarana adalah segala sesuatu yang dapat dipakai sebagai alat dalam mencapai maksud atau tujuan.

Sedangkan Menurut Jannah dan Sontani (2018:65) “Sarana adalah sebuah perangkatan, peralatan, bahan, perabot yang secara langsung digunakan dalam sebuah kegiatan atau aktivitas”.



Jadi, dari definisi di atas dapat disimpulkan bahwa, Sarana adalah segala sesuatu berupa perangkatan, peralatan dan bahan yang digunakan dalam sebuah aktivitas dan untuk mencapai suatu tujuan.

### **2.3.5 Pengertian Prasarana**

Menurut Kamus Besar Bahasa Indonesia (KBBI) Online, Prasarana adalah segala sesuatu yang merupakan penunjang utama terselenggaranya suatu proses ( usaha, pembangunan, proyek ).

### **2.3.6 Pengertian Aplikasi Pengolahan Data Inventaris Sarana Pada PT Pos Logistic Indonesia PDC Palembang 30900**

Aplikasi Pengolahan Data Inventaris Sarana Pada PT Pos Logistic Indonesia PDC Palembang 30900 adalah suatu aplikasi yang digunakan untuk mengelolah data inventaris sarana kantor PT Pos Logistic Indonesia PDC Palembang 30900.

## **2.4 Teori Program**

### **2.4.1 Pengertian HTML**

Menurut Ardhana dalam Tabrani dan Pudjiarti (2017:31) “HTML atau *Hyper Text Markup Language* merupakan suatu bahasa yang dikenali oleh *webbrowser* untuk menampilkan informasi seperti teks, gambar, animasi bahkan video”.

#### **2.4.1.1 Struktur HTML**

Menurut Tabrani dan Pudjiarti (2017:31-32), “Untuk dapat membuat website dengan baik maka langkah awal yang harus dilakukan yaitu mengenal kode-kode dasar HTML yang sering digunakan oleh programmer web professional. Kode HTML memiliki aturan dan struktur penulisan tersendiri yang disebut tag HTML. Tag adalah kode yang digunakan untuk memoles (*mark-up*) teks menjadi file HTML. Setiap tag diapit dengan tanda kurung runcing. Ada tag pembuka dan tag penutup, yang membedakan tag penutup ditandai dengan tanda



garis miring (*slash*) di depan awal tulisannya. Dokumen html merupakan dokumen yang disajikan dalam web browser dan biasanya hanya untuk menampilkan informasi maupun *interface*. Secara umum, dokumen web dibagi menjadi dua bagian, yaitu *head* dan *body*, sehingga setiap dokumen html harus mempunyai pola dasar”.

## 2.4.2 Sekilas Tentang PHP

### 2.4.2.1 Pengertian PHP

Firman, dkk (2016:30) berpendapat bahwa “PHP atau kependekan dari Hypertext Preprocessor adalah salah satu bahasa pemrograman open source yang sangat cocok atau dikhususkan untuk pengembangan web dan dapat ditanamkan pada sebuah skripsi HTML”.

Senada dengan Firman, dkk, Menurut Anhar dalam Prayitno dan Safitri (2015:2) “PHP adalah (PHP *Hypertext Preprocessor*) adalah bahasa pemrograman web berupa script yang dapat diintegrasikan dengan HTML”.

Berdasarkan pendapat di atas dapat disimpulkan bahwa, PHP merupakan suatu bahasa pemrograman web yang difungsikan untuk membangun suatu website dinamis dan dapat diintegrasikan dengan HTML.

### 2.4.2.2 Script PHP

Menurut Tabrani dan Pudjiarti (2017:32) “berbeda dengan HTML, kode PHP tidak diberikan langsung oleh *server* ketika ada permintaan dari sisi *client* namun dengan cara pemrosesan dari sisi *server*. Kode PHP disisipkan pada kode HTML. Perbedaan dari kode (*script*) HTML dan PHP yaitu setiap kode PHP ditulis selalu diberi *tag* pembuka yaitu `<?` Dan pada akhir kode PHP diberi *tag* penutup yaitu `?>`. PHP sangat berperan besar ketika ingin membuat *website* keren yang dinamis karena dapat melakukan banyak hal, seperti membaca *file*, menulis *file*, menampilkan gambar, animasi atau *movie*, dan yang paling pokok adalah dapat melakukan koneksi terhadap *database*”.



### **2.4.3 Sekilas Tentang XAMPP**

#### **2.4.3.1 Pengertian XAMPP**

Menurut Hanif dan Fitriani (2016:43), “XAMPP adalah sebuah paket kumpulan *software* yang terdiri dari Apache, MySQL, phpmyadmin, PHP, Perl, Freetype2,dll. XAMPP berfungsi untuk memudahkan instalasi lingkungan php, di mana biasa nya lingkungan pengembangan web memerlukan PHP, Apache, MySQL dan phpmyadmin serta *software* yang terkait dengan pengembangan web”.

### **2.4.4 Sekilas Tentang MySQL**

#### **2.4.4.1 Pengertian MySQL**

Menurut Kustiyahningsih dalam Firman, dkk (2016:30), “MySQL adalah sebuah basis data yang mengandung satu atau jumlah tabel.Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel.Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel”.

Sedangkan menurut Anhar dalam Prayitno dan Safitri (2017:2) “MySQL (*My Structure Query Language*) adalah sebuah perangkat lunak sistem manajemen basis data *SQL Database Management System* atau DBMS dari sekian banyak DBMS seperti Oracle, MS SQL, Postagre SQL dan lainnya”.

Dari beberapa pengertian di atas dapat disimpulkan bahwa, MySQL adalah sebuah perangkat lunak yang didistribusikan secara gratis yang digunakan untuk sistem manajemen basis data.



#### 2.4.4.2 Kelebihan MySQL

Tabrani dan Pudjiarti (2017:33) menjelaskan MySQL memiliki kelebihan-kelebihan diantaranya :

1. *Free* (bebas di *download*)
2. Stabil dan tangguh
3. Fleksibel dengan berbagai pemrograman
4. *Security* yang baik
5. Kemudahan manajemen *database*
6. Mendukung transaksi
7. Perkembangan software yang cukup cepat.