



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1 Pengertian Komputer

Kadir (2017:2), “Komputer adalah peralatan elektronik yang bermanfaat untuk melaksanakan berbagai pekerjaan yang dilakukan oleh manusia.”

Paramytha (2016:4), “Komputer berasal dari bahasa latin yaitu *Computer* yang berarti menghitung (*to compute* atau *to reckon*). Kata komputer itu sendiri pada awalnya dipergunakan untuk menggambarkan pekerjaan orang yang melakukan perhitungan aritmatika dengan atau tanpa alat bantu.”

Jadi dapat penulis simpulkan pengertian komputer adalah peralatan elektronik yang dipergunakan untuk menggambarkan pekerjaan yang dilakukan oleh manusia.

2.1.2 Pengertian Aplikasi

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

Paramytha (2016:56), “Aplikasi (*application software*), merupakan perangkat lunak yang dikembangkan untuk menyelesaikan suatu aplikasi tertentu.”

Maka dari dua pengertian diatas dapat penulis simpulkan pengertian aplikasi adalah program yang menentukan aktivitas pemrosesan informasi untuk menyelesaikan tugas-tugas khusus dari pemakai komputer.

2.1.3 Pengertian Perangkat Lunak

Kadir (2017:2), “Perangkat Lunak adalah instruksi-instruksi yang ditujukan kepada komputer agar dapat melaksanakan tugas sesuai kehendak pemakai. Sistem operasi seperti *Windows*, *Mac OS*, dan *Linux*, dan aplikasi seperti *Microsoft Word* dan *Microsoft Excel* adalah contoh perangkat lunak.”



Rosa 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*).”

Berdasarkan pengertian diatas dapat penulis simpulkan pengertian perangkat Lunak adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak agar dapat melaksanakan tugas sesuai kehendak pemakai.

2.1.4 Pengertian Website

Abdulloh (2018:1), “*Website* dapat diartikan sebagai kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya yang disediakan melalui jalur koneksi internet sehingga dapat diakses dan dilihat oleh semua orang di seluruh dunia.”

Sidik (2017:1), “Situs web (*Web site*) awalnya merupakan suatu layanan sajian informasi yang menggunakan konsep hyperlink, yang memudahkan surfer (sebutan bagi pemakai komputer yang melakukan penelusuran informasi di Internet) untuk mendapatkan informasi, dengan cukup mengklik suatu link berupa teks atau gambar, maka Informasi dari teks atau gambar akan ditampilkan secara lebih rinci (*detail*).”

Jadi dapat penulis simpulkan pengertian *website* adalah kumpulan halaman yang berisi informasi data digital baik berupa teks, gambar, animasi, suara dan video atau gabungan dari semuanya untuk mendapatkan informasi, dengan cukup mengklik suatu link berupa teks atau gambar.

2.1.5 Metode Pengembangan Sistem

Rosa dan Shalahudin (2018:28) menjelaskan tentang metode pengembangan sistem yaitu *waterfall*. Metode air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequential linear*) atau alur hidup klasik (*classic life cycle*). 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.



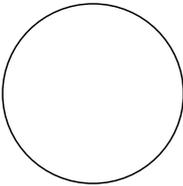
2.2 Teori Khusus

2.2.1 Pengertian DFD (*Data Flow Diagram*)

Rosa dan Shalahuddin (2018:71) menjelaskan, “*Data Flow Diagram* (DFD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*ouput*).

Adapun notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut :

Tabel 2.1 Simbol-simbol *Data Flow Diagram*

No	Notasi	Keterangan
1.		Proses atau fungsi atau prosedur, pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program Catatan : Nama yang diberikan pada sebuah proses biasanya berupa kata kerja
2.		File basisdata atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harusnya 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)) Catatan : Nama yang diberikan pada sebuah penyimpanan biasanya kata benda



Lanjutan Tabel 2.1 Simbol-simbol *Data Flow Diagram*

3.		<p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang memakain/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 masukan (<i>input</i>) atau keluaran (<i>output</i>)</p> <p>Catatan:</p> <p>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 : Rosa dan Shalahuddin, 2018:71-72)

Rosa dan Shalahuddin (2018:72-73) menjelaskan, Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 (nol) atau sering disebut juga Context Diagram

DFD Level 0 (nol) menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem lain. DFD Level 0 (nol) 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 (nol) 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 lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-breakdown lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul ada DFD Level 1(satu) yang di-breakdown.

4. Membuat DFD Level 3 dan seterusnya

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

2.2.2 Pengertian *Blockchart*

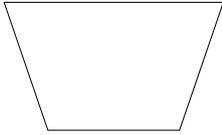
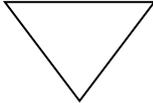
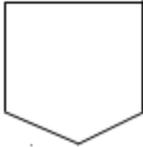
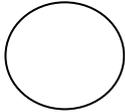
Kristanto (2008:68) menjelaskan, “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu”. Pembuatan blockhart harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

Adapun simbol-simbol yang sering digunakan dalam block chart dapat dilihat pada tabel berikut ini:

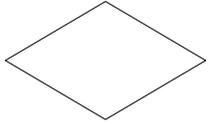
Tabel 2.2 Simbol-simbol *Blockchart*

No	Simbol	Arti
1.		Multi Dokumen
2.		Proses Manual

Lanjutan Tabel 2.2 Simbol-simbol *Blockchart*

3.		Proses yang dilakukan oleh komputer
4.		Menandakan dokumen yang diarsipkan (arsip manual)
5.		Data penyimpanan (<i>data 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.
8.		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
9.		Terminasi yang menandakan awal dan akhir dari suatu aliran

Lanjutan Tabel 2.2 Simbol-simbol *Blockchart*

10.		Pengambilan keputusan (<i>decision</i>)
11.		Layar peraga (<i>monitor</i>).
12.		Pemasukkan data secara manual.
13.		

(Sumber : Kristanto, 2008:68-70)

2.2.3 Pengertian *Flowchart*

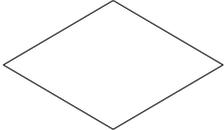
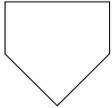
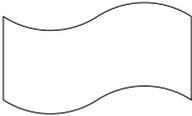
Lamhot (2015:14) menjelaskan, “*Flowchart* adalah langkah-langkah menyelesaikan masalah yang dituliskan dalam simbol-simbol tertentu.”

Menurut Lamhot (2015:14-16), Gambaran simbol-simbol *Flowchart* sebagai berikut:

Tabel 2.3 Simbol-simbol *Flowchart*

No.	Simbol	Nama	Fungsi
1		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program.
2		<i>Input/output</i>	Menyatakan proses input atay output tanpa tergantung jenis peralatannya.

Lanjutan Tabel 2.3 Simbol-simbol *Flowchart*

3		<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
4		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya atau tidak.
5		<i>Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama.
6		<i>Offline Connector</i>	Menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda.
7		<i>Predefined process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
8		<i>Fanned Card</i>	Menyatakan input berasal dari kartu atau output data ke kartu.
9		<i>Punch Tape</i>	Menyatakan input atau output yang menggunakan pita kertas berlubang.
10		<i>Document</i>	Mencetak keluaran dalam bentuk dokumen (melalui Printer)

Lanjutan Tabel 2.3 Simbol-simbol *Flowchart*

11		<i>Flow</i>	Menyatakan jalannya arus suatu proses
----	--	-------------	---------------------------------------

(Sumber: Lamhot, 2015:14-15)

2.2.4 Pengertian Kamus Data

Rosa dan Shalahuddin (2018:73) menjelaskan, “Kamus data adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (input) dan keluaran (ouput) dapat dipahami secara umum (memiliki standar cara penulisan).”

Kamus data memiliki beberapa simbol sebagai berikut :

Tabel 2.4 Simbol-simbol Kamus Data

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



2.3 Teori Judul

2.3.1 Pengertian Aplikasi

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

Paramytha (2016:56), “Aplikasi (*application software*), merupakan perangkat lunak yang dikembangkan untuk menyelesaikan suatu aplikasi tertentu.”

Jadi dapat penulis simpulkan pengertian aplikasi adalah program yang menentukan aktivitas pemrosesan informasi untuk menyelesaikan tugas-tugas khusus dari pemakai komputer.

2.3.2 Pengertian Pengolahan

Menurut Kamus Besar Bahasa Indonesia, “Pengelolaan adalah proses yang memberikan pengawasan pada semua hal yang terlibat dalam pelaksanaan kebijaksanaan dan pencapaian tujuan.”

Balderton (dalam Adisasmita, 2011:21), “istilah pengelolaan sama dengan manajemen yaitu menggerakkan, mengorganisasikan, dan mengarahkan usaha manusia untuk memanfaatkan secara efektif material dan fasilitas untuk mencapai suatu tujuan.”

Kesimpulannya, pengelolaan adalah suatu rangkaian kegiatan yang meliputi merencanakan, mengorganisasikan dan mengarahkan, dan mengawasi kegiatan manusia dengan memanfaatkan material dan fasilitas yang ada untuk mencapai tujuan yang telah ditetapkan secara efektif dan efisien.

2.3.3 Pengertian Data

Fathansyah (2012:2) , “Data adalah representasi fakta dunia nyata yang mewakii 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 kombinasi.”



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

Jadi dapat penulis simpulkan bahwa pengertian data adalah fakta yang mewakili suatu objek yang berwujud dalam angka, huruf, simbol, teks, gambar, bunyi atau kombinasi dan juga biasanya mengenai fenomena fisik dengan kata lain data adalah informasi yang nyata.

2.3.4 Pengertian Pengaduan

Fanar (2009:29), ”Pengaduan merupakan suatu sumber informasi yang sangat penting bagi upaya-upaya pihak penyelenggara pelayanan untuk memperbaiki kesalahan yang mungkin terjadi, sekaligus secara konsisten menjaga dan meningkatkan pelayanan yang dihasilkan agar selalu sesuai dengan standar yang telah ditetapkan.”

Tarigan et al. (2014:494), “pengaduan didefinisikan sebagai laporan kepada pihak yang berwenang mengenai tindakan atau perilaku seorang profesional, badan hukum swasta atau lembaga negara yang melanggar peraturan perundang-undangan yang berlaku serta hak atau kepentingan individu atau masyarakat, agar pihak yang berwenang tersebut dapat mengambil tindakan sesuai kewenangannya untuk mencegah atau mengembalikan kerugian yang diderita individu atau masyarakat tersebut, dan/atau membuat profesional, pejabat publik, badan hukum swasta atau lembaga negara tersebut menerima konsekuensi atas tindakan yang telah dilakukan.”

Kesimpulannya, Pengaduan adalah suatu sumber informasi yang sangat penting mengenai tindakan atau perilaku seorang profesional untuk memperbaiki kesalahan yang mungkin terjadi untuk memperbaiki kesalahan yang mungkin terjadi.

2.3.5 Pengertian Serikat Kerja

Berdasarkan ketentuan umum pasal 1 Undang-undang Tenaga Kerja tahun 2003 No 17, “Serikat buruh/serikat kerja adalah organisasi yang dibentuk dari,



oleh, dan untuk pekerja baik di perusahaan maupun diluar perusahaan, yang bersifat bebas, terbuka, mandiri, demokratis, dan bertanggung jawab guna memperjuangkan, membela serta melindungi hak dan kepentingan pekerja serta meningkatkan kesejahteraan pekerja dan keluarganya.”

Budiarti (2008:2), “Serikat pekerja adalah organisasi demokratis yang berkesinambungan dan permanen dibentuk secara sukarela dari, oleh dan untuk pekerja.”

Kesimpulannya, Serikat kerja adalah organisasi yang dibentuk dari, oleh, dan untuk pekerja yang berkesinambungan dan permanen dibentuk secara sukarela guna memperjuangkan, membela serta melindungi hak dan kepentingan pekerja serta meningkatkan kesejahteraan pekerja dan keluarganya.

2.3.6 Pengertian Aplikasi Pengolahan Data Anggota Serikat Pekerja Seluruh Indonesia pada PT Kencana Sawit Indonesia Berbasis *Website*

Aplikasi Pengolahan Data Anggota Serikat Pekerja Seluruh Indonesia pada PT Kencana Sawit Indonesia Berbasis *Website* adalah suatu aplikasi yang berisi form dan juga informasi mengenai kegiatan pengaduan anggota tentang kebijakan perusahaan dan juga laporan dari kegiatan pengaduan keluh kesah tersebut.

2.4 Teori Program

2.4.1 Pengertian Basis Data

Rosa dan Shalahuddin (2018:43), “Basis data adalah sistem komputerisasi yang tujuan utamanya adalah memelihara data yang sudah ada yang diolah atau informasi dan membuat informasi tersedia saat dibutuhkan.”

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

Maka dari dua pengertian diatas dapat penulis simpulkan pengertian basis data adalah sistem komputerisasi yang saling berhubungan yang diorganisasi sedemikian ruoa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.



2.4.1.1 Istilah-istilah dalam Basis Data

Kadir (2017:103), menjelaskan beberapa istilah yang sering digunakan pada pembahasan *database* (basis data), yaitu sebagai berikut :

1. **DBMS**, yaitu sistem perangkat lunak yang menyediakan layanan bagi user untuk membuat, mengontrol dan mengakses database.
2. **Table**, merupakan kumpulan data yang diorganisasikan dalam baris (record) dan kolom (field). Dalam satu database biasanya terdiri dari beberapa tabel.
3. **Field**, merupakan kolom pada tabel yang memiliki ukuran dan tipe data tertentu.
4. **Record**, merupakan baris data yang memiliki nilai untuk masing-masing field pada tabel. Satu tabel biasanya terdiri dari banyak record.

2.4.1.2 Atribut (*attribute*)

Atribut merupakan semua informasi yang berkaitan dengan entitas. Yanto (2016:34-38), menjelaskan macam-macam bentuk atribut :

1. **Atribut Sederhana (*Simple Attribute*)**, adalah atribut yang nilainya tidak dapat dibagi lagi menjadi banyak atribut yang lebih kecil.
2. **Atribut Komposit (*Composite Attribute*)**, adalah atribut gabungan yang nilainya dapat dipecah menjadi bagian yang lebih kecil. Atau sering disebut atribut yang terdiri dari beberapa atribut kecil di dalamnya.
3. **Atribut Bernilai tunggal (*single Value Attribute*)**, adalah jenis atribut yang nilainya hanya satu dari suatu entitas.
4. **Atribut Bernilai Banyak (*multivalues Attribute*)**, adalah jenis atribut yang nilainya lebih dari satu dalam suatu entitas tertentu.
5. **Atribut Turunan (*Derived Attribute*)**, adalah jenis atribut yang nilainya diperoleh dari atribut lain.
6. **Atribut Identitas (*Key Attribute*)**, adalah atribut yang dijadikan sebagai kunci pada suatu tabel. Sifat atribut identitas ini unik, tidak ada yang menyamai, atribut identitas terdiri dari beberapa jenis yaitu :



- a. **Super Key**, adalah satu atribut atau kumpulan atribut yang secara unik mengidentifikasi sebuah baris di dalam relasi atau himpunan dari satu atau lebih entitas yang dapat digunakan untuk mengidentifikasi secara unik sebuah entitas dalam set entitas.
- b. **Candidate Key**, adalah atribut yang menjadi determinan yang dapat dijadikan identitas baru pada sebuah relasi.
- c. **Primary Key**, adalah kandidat key yang dipilih untuk mengidentifikasi baris data secara unik dalam relasi.
- d. **Alternative Key**, adalah candidate key yang tidak terpilih sebagai primary key atau atribut untuk menggantikan kunci utama.
- e. **Foreign Key**, adalah atribut dengan domain yang sama yang menjadi kunci utama sebuah relasi, tetapi pada relasi lain atribut tersebut sebagai atribut biasa.
- f. **Composite Key**, adalah kunci yang terdiri dari dua atribut atau lebih.

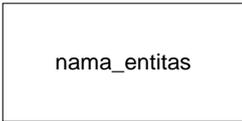
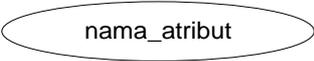
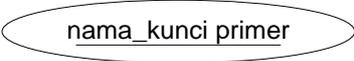
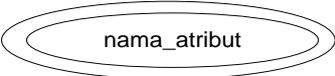
2.4.2 Pengertian ERD (*Entity Relational Diagram*)

Rosa dan Shalahuddin (2018:50) menjelaskan, “*Entity Relational Diagram* (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.

Adapun simbol-simbol yang digunakan pada ERD dengan notasi Chen adalah sebagai berikut:

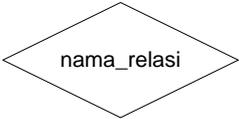
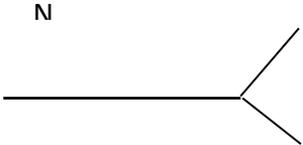


Tabel 2.5 Simbol-simbol *Entity Relational Diagram*

No	Simbol	Deskripsi
1.	Entitas / <i>entity</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
2.	Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record 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)
3.	Atribut kunci primer 	Field atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
4.	Atribut multivalai / multivalue 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja



Lanjutan Tabel 2.5 Simbol-simbol *Entity Relational Diagram*

5.	Relasi 	Penghubung antar relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> menghubungkan entitas A dan entitas B maka
6.	Asosiasi / <i>association</i> 	

(Sumber: Rosa dan Shalahuddin , 2018:50-51)

2.4.3 Pengertian XAMPP

Aryanto (2016:4), “*Xampp* merupakan sebuah aplikasi perangkat lunak pemrograman dan *database* yang di dalamnya terdapat berbagai macam aplikasi pemrograman seperti : *Apache*, *HTTP*, *MySQL*, *database*, bahasa pemrograman *PHP* dan *Perl*.”

Madcoms (2014:186), “*Xampp* adalah sebuah paket kumpulan software yang terdiri dari *Apache*, *MySQL*, *PhpMyAdmin*, *Perl*, *Filezilla* dan lain-lain.”

Maka dari dua pengertian diatas dapat penulis simpulkan pengertian *XAMPP* adalah sebuah aplikasi perangkat lunak pemrograman dan *database* yang di dalamnya terdapat berbagai macam aplikasi pemrograman yang terdiri dari *Apache*, *MySQL*, *PhpMyAdmin*, *Perl*, *Filezilla* dan lain-lain.



2.4.4 Pengertian MySQL

Rosa dan Shalahuddin (2018:46), “SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus.”

Raharjo (2015:16), “MySQL adalah *software* RDMS (atau *server database*) yang dapat mengelola *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user* (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau bebarengan (*multi-threaded*).”

Berdasarkan pengertian diatas dapat penulis simpulkan pengertian MySQL adalah bahasa yang digunakan untuk mengelola data SQL (*database management system*) atau DBMS yang *multithreaded, multi-user*.

2.4.5 Pengertian PHP

Murya (2014:65), “PHP *Hypertext Preprocessor* atau sering disebut PHP merupakan bahasa pemrograman berbasis *server-side* yang dapat melakukan parsing script php menjadi script web sehingga dari sisi client menghasilkan suatu tampilan yang menarik.”

Madcoms (2016:2), “PHP (*Hypertext preprocessor*) adalah bahasa script yang dapat ditanamkan atau disisipkan ke dalam HTML.”

Jadi dapat penulis simpulkan pengertian *PHP* adalah bahasa pemograman berbasis *server-side* yang bisa kita gunakan untuk membuat aplikasi web yang disisipkan pada HTML.



Gambar 2.1 Tampilan Logo PHP



2.4.5.1 Sintaks Dasar PHP

Kode (Script) PHP yang sering disebut dengan istilah embedded script yaitu script PHP yang disisipkan di antara script HTML. Jadi dapat dikatakan script PHP hanya ditulis atau disisipkan ketika dibutuhkan saja, seperti menampilkan data dari database meng-upload file, delete data, edit data dan lain sebagainya. Contoh script :

```
<HTML>
  <HEAD>
    <TITLE>Contoh Awal</TITLE>
  </HEAD>
<BODY>
  <?php
    echo "Belajar bahasa Pemrograman PHP";
  ?>
</BODY>
</HTML>
```



Gambar 2.2 Contoh Script PHP



2.4.5.2 Tipe Data PHP

Tipe data merupakan jenis dari suatu data yang akan diproses oleh bahasa pemrograman. Yosef Murya (2014:26), menjelaskan beberapa tipe data dalam PHP, sebagai berikut :

1. **Integer** merupakan tipe data yang berguna untuk menyimpan bilangan bulat. Range bilangan integer adalah antara -2.147.4833.647 sampai dengan 2.147.483.647
2. **Double Floating** adalah tipe data yang berguna untuk menyimpan bilangan desimal. Range bilangan floating point antara 1e308 sampai dengan 1e308.
3. **Boolean** adalah tipe data yang paling sederhana, hanya berupa **TRUE** dan **FALSE**.
4. **String** adalah tipe data yang terdiri dari kata, bias berupa kata tunggal maupun kalimat. Penulisan string harus diapit dengan tanda petik, baik berupa petik tunggal ('...') maupun petik ganda ("...").
5. **Objek** adalah tipe data dibuat dengan tujuan agar para programmer terbiasa dengan OOP. Tipe data ini bias berupa bilangan.
6. **Array** merupakan **Tipe Compound Primitif**, terdapat pada bahasa pemrograman lain.
7. **Null** adalah tipe data yang tidak memuat apapun. Setiap variable yang diset menjadi tipe data Null, ini akan menjadikan variabel tersebut kosong.
8. **Resources** tipe data spesial yang satu ini dikhususkan untuk menyimpan *resources*, sumber atau alamat.

2.4.6 Pengertian JavaScript

Abdulloh (2018:193), "*Javascript* merupakan Bahasa pemrograman web yang pemrosesannya dilakukan di sisi *client*. Karena berjalan di sisi *client*. *Javascript* dapat dijalankan hanya dengan menggunakan browser."

Bratadinata (2013:5), "*JavaScript* adalah bahasa pemrograman yang dibuat mengikuti spesifikasi standar yang disebut *ECMAScript* dan saat ini versi termmodern dari *ECMAScript* adalah versi 5."



Jadi dapat penulis simpulkan pengertian *javascript* adalah Bahasa pemrograman yang pemrosesannya dilakukan di sisi *client-side* yang disebut *ECMAScript* dan saat ini versi termodern nya adalah versi 5.”

2.4.6.1 Dasar JavaScript

Cara menggunakan JavaScript adalah dengan dimasukkan di antara kode HTML menggunakan tag `<script>` dan `</script>`. Javascript bisa diletakkan di tag `<body>` ataupun tag `<head>` dari kode HTML. Untuk memasukkan javascript wanda harus menggunakan tag `<script>`, tag `<script>` dan `</script>` menentukan dimana javascript harus dimulai dan diakhiri.

Baris diantara tag `<script>` dan `</script>` ini berisi data Javascript contohnya seperti berikut :

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <script>
      var a = “ Selamat Datang! ” ;
      function selamat () {
        var b = “ Selamat Pagi! ” ;
        document.write (a);
      }
      selamat () ;
      document.write (b) ;
    </script>
  </body>
</html>
```