



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Asropudin (2013:19), menyatakan bahwa “Komputer adalah alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer”.

Wahyudi (2012:3), menyatakan bahwa “Komputer adalah Peralatan (*device*) yang menerima data (*input*) dan menyimpan (*storage*) kemudian di proses (*process*) untuk menghasilkan data dalam bentuk lain (*output*)”.

Dari pengertian diatas, komputer adalah alat pemrosesan data elektronik yang membantu melakukan perintah-perintah atau proses-proses tertentu dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing komputer.

2.1.2. Pengertian Program

Sutarman (2012:03), menyatakan bahwa “Program adalah barisan perintah atau instruksi yang disusun sehingga dapat dipahami oleh komputer dan kemudian dijalankan sebagai barisan perhitungan numerik, dimana barisan perintah tersebut berhingga, berakhir, dan menghasilkan *output*”.

2.1.3. Pengertian Data

Asropudin (2013:22), menyatakan bahwa “Data adalah kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti”.

Ladjamudin (2013:8), menyatakan bahwa “Data adalah deskripsi dari sesuatu dan kejadian yang kita hadapi (*the description of things and events that we face*)”.



Dari pengertian diatas, data adalah kumpulan dari angka maupun karakter yang berasal dari sesuatu dan kejadian yang kita hadapi dan harus diolah terlebih dahulu untuk menjadi suatu informasi yang lebih berarti bagi penggunanya.

2.1.4. Pengertian Pengolahan Data

Sutarman (2012:4), menyatakan bahwa “Pengolahan data (*data processing*) adalah proses perhitungan/transmansi data *input* menjadi informasi yang mudah dimengerti ataupun sesuai dengan yang diinginkan”.

Ladjamudin (2013:9), menyatakan bahwa “Pengolahan Data adalah masa atau waktu yang digunakan untuk mendeskripsikan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Dari pengertian diatas, pengolahan data merupakan proses pengolahan data yang diinputkan untuk diubah menjadi informasi yang mudah dimengerti.

2.1.5. Pengertian Sistem

Hartono (2013:9), menyatakan bahwa “Sistem adalah suatu himpunan dari berbagai bagian atau elemen yang saling berhubungan secara terorganisasi berdasar fungsi-fungsinya menjadi satu kesatuan”.

Sutarman (2012:5), menyatakan bahwa “Sistem adalah kumpulan elemen yang saling berhubungan dan berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama”.

Dari pengertian diatas, sistem merupakan sekumpulan elemen yang saling berhubungan dan saling berinteraksi untuk menjalankan suatu proses pencapaian suatu tujuan utama.

2.1.6. Karakteristik Sistem

Ladjamudin (2013:3-5), menjelaskan bahwa sistem memiliki karakteristik atau sifat-sifat tertentu yang mempunyai komponen-komponen, batas sistem, lingkungan luar sistem, penghubung, masukan ,keluaran, pengolahan dan sasaran atau tujuan. Adapun karakteristik yang dimaksud adalah sebagai berikut :

**1. Komponen Sistem (*Components*)**

Suatu sistem *terdiri* dari sejumlah komponen yang saling berinteraksi, yang bekerja sama membentuk suatu kesatuan.

2. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lainnya atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan dan menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dan bersifat menguntungkan dan juga merugikan.

4. Penghubung Sistem (*Interface*)

Media yang menghubungkan antara satu subsistem dengan subsistem yang lainnya.

5. Masukan Sistem (*Input*)

Masukan adalah Energi yang dimasukkan ke dalam sistem, yang dapat berupa pemeliharaan (*maintanance input*) dan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi sub sistem yang lain.

7. Pengolahan Sistem (*Process*)

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki sasaran dan tujuan yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran, maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.



2.1.7. Metode Pengembangan Sistem

Sukanto et.al (2014:28), 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, pengodean, pengujian, dan pemeliharaan.

a. Analisis

Tahap analisis dilakukan secara intensif untuk menspesifikasikan kebutuhan sistem agar dapat dipahami sistem seperti apa yang dibutuhkan oleh *user*.

b. Desain

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

c. Pengkodean

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

d. Pengujian

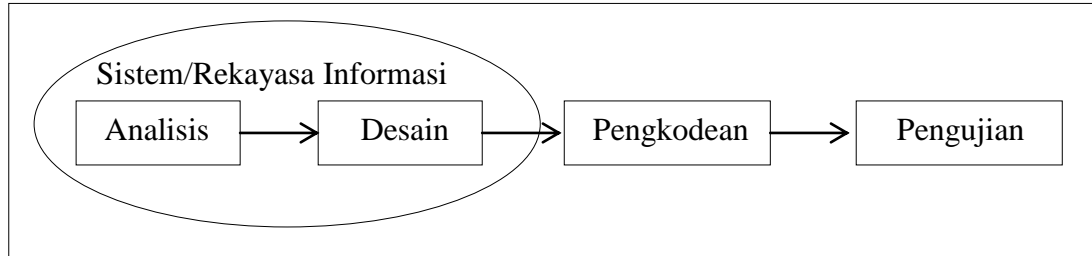
Tahap pengujian fokus pada sistem dari segi logika dan fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan (*error*) dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

e. Pemeliharaan

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



mulai dari analisis spesifikasi untuk perubahan sistem yang sudah ada, tapi tidak untuk sistem baru.



Gambar 2.1. Ilustrasi Model *Waterfall*

Sumber : Sukamto et.al (2014 : 29)

2.1.8. Pengertian Informasi

Sutarman (2012:4), “Informasi (*information*) adalah hasil kegiatan pengolahan data yang memberikan bentuk yang lebih berarti dari suatu kejadian”.

Kristanto (2008:7), menyatakan bahwa “Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima”.

Dari pengertian diatas, Informasi merupakan hasil kegiatan pengolahan data menjadi bentuk yang lebih berguna dan berarti dari suatu kejadian.

2.1.9. Pengertian Sistem Informasi Berbasis Komputer

Sutarman (2012:14), menjelaskan bahwa Sistem Informasi Berbasis Komputer atau *Computer-Based Information System* (CBIS) adalah sebuah sistem informasi yang menggunakan komputer dan teknologi telekomunikasi untuk melakukan tugas-tugas yang diinginkan. Komponen-komponen dasar dari sistem informasi, yaitu :

- a. *Hardware*: kumpulan peralatan seperti *processor*, *monitor*, *keyboard*, dan *printer* yang menerima data dan informasi, memproses data tersebut dan menampilkan data tersebut.
- b. *Software*: kumpulan program-program komputer yang memungkinkan *hardware* memproses data.



- c. *Database*: sekumpulan file yang saling berhubungan dan terorganisasi atau kumpulan *record-record* yang menyimpan data dan hubungan di antaranya.
- d. *Network*: sebuah sistem yang terhubung yang menunjang adanya pemakaian bersama sumber di antara komputer-komputer yang berbeda.
- e. *People*: elemen yang paling penting dalam sistem informasi, termasuk orang-orang yang bekerja dengan sistem informasi atau menggunakan *output*-nya.

2.1.10. Pengertian Bahasa Pemrograman

Sutarman (2012:159), menyatakan bahwa “Bahasa Pemrograman adalah *software* yang dipakai oleh para *programmer* (pembuat program/ *software*) untuk membuat/ menuliskan perintah-perintah atau program tertentu”.

2.2. Pengertian-pengertian Judul

2.2.1. Pengertian Aplikasi

Asropudin (2013:6), menyatakan bahwa “*Application* atau aplikasi merupakan *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya Microsoft Word, Microsoft Excel.”

Sutabri (2012:147), menyatakan bahwa “Aplikasi adalah alat terapan yang difungsikan secara khusus dan terpadu sesuai kemampuan yang dimilikinya”.

Dari pengertian diatas, Aplikasi merupakan alat terapan yang digunakan untuk mengerjakan tugas-tugas tertentu sesuai kemampuan yang dimilikinya.

2.2.2. Pengertian Sambungan

Yuniar (529), menyatakan bahwa “Sambungan adalah tambahan untuk memanjangkan; lanjutan, susulan”.

“Sambungan adalah hasil dari penyatuan beberapa bagian atau konstruksi dengan menggunakan suatu cara tertentu.”

(Diakses pada tanggal 30 Mei 2016 pukul 08:37 pada situs https://www.academia.edu/8111092/PENGERTIAN_SAMBUNGAN)



2.2.3. Pengertian Baru

Yuniar (78), menyatakan bahwa “Baru adalah permulaan, baru-baru ini, belum lama ini, beberapa waktu lalu; belum lama, belum pernah (dipakai atau dilihat, dsb); belum lama dibeli atau dimiliki; belum pernah didengar sebelumnya; segar karna baru dipetik (tentang buah); belum lama menikah; belum lama ditempati (tentang rumah)”.

2.2.4. Pengertian Perusahaan Daerah Air Minum (PDAM)

“PDAM atau Perusahaan Daerah Air Minum merupakan salah satu unit usaha milik daerah, yang bergerak dalam distribusi air bersih bagi masyarakat umum. PDAM terdapat di setiap provinsi, kabupaten, dan kotamadya di seluruh Indonesia. PDAM merupakan perusahaan daerah sebagai sarana penyedia air bersih yang diawasi dan dimonitor oleh aparat-aparat eksekutif maupun legislatif daerah”.

(Diakses pada tanggal 30 Mei 2016 pukul 08:40 pada situs <https://id.wikipedia.org/wiki/PDAM>)

2.2.5. Pengertian Aplikasi Pengolahan Data Sambungan Baru pada Perusahaan Daerah Air Minum (PDAM) Kabupaten Ogan Komering Ulu Baturaja

Dari uraian di atas dapat disimpulkan bahwa pengertian Aplikasi Pengolahan Data Sambungan Baru pada Perusahaan Daerah Air Minum (PDAM) Kabupaten Ogan Komering Ulu merupakan suatu aplikasi yang berfungsi untuk mengolah data sambungan baru Perusahaan Daerah Air Minum (PDAM) Kabupaten Ogan Komering Ulu sehingga pengolahan datanya akan lebih mudah dan mendapatkan hasil yang akurat.




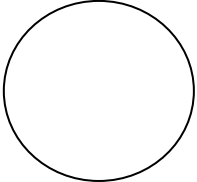
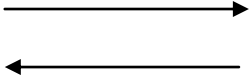

2.3. Teori Khusus

2.3.1. Pengertian DFD (*Data Flow Diagram*)

Ladjamudin (2013:64), Mengemukakan “*DFD* adalah model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil”.

Sutabri (2012:117), menyatakan bahwa “*Data flow diagram* ini adalah suatu *network* yang menggambarkan suatu sistem automat/komputerisasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya”.

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

No	Simbol	Keterangan
1.		<i>External Entity</i> Simbol ini digunakan untuk menggambarkan asal atau tujuan data.
2.		<i>Proses</i> Simbol ini digunakan untuk proses pengolahan data atau transformasi data.
3.		<i>Data Flow</i> Simbol ini digunakan untuk menggambarkan aliran data yang sedang berjalan
4.		<i>Data Store</i> Simbol ini digunakan untuk menggambarkan data <i>flow</i> yang sudah disimpan dan diarsipkan.

Sumber : Sutabri (2012:117)



Sukanto et.al (2014:72-73), tahapan-tahapan perancangan menggunakan DFD:


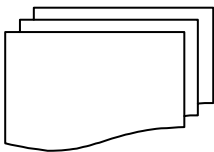
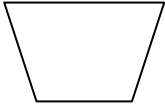

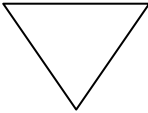
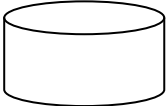
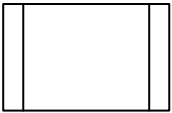
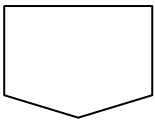
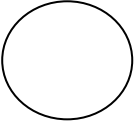
1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*
DFD Level 0 menggambarkan sistem yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun sistem orang lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara sistem yang akan dikembangkan dengan entitas 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 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 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 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.

2.3.2. Pengertian Blockchart


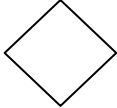


Kristanto (2008:75), mengemukakan “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *Blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

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

Tabel 2.2. Simbol-simbol dalam *Block Chart*

No	Simbol	Keterangan
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)
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



No	Simbol	Keterangan
10.		Terminasi yang menandakan awal dan akhir dari suatu aliran
11.		Pengambilan keputusan (<i>decision</i>)
12.		Layar peraga (monitor)
13.		Pemasukan data secara manual

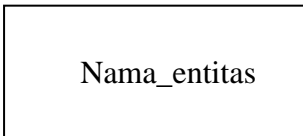
Sumber :Kristanto (2008 : 75)

2.3.3. Pengertian ERD (*Entity Relational Diagram*)

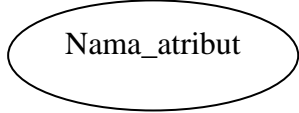
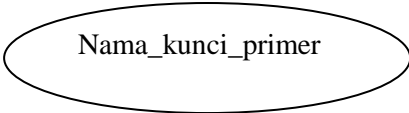
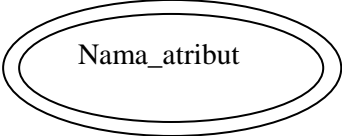
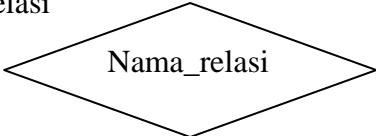
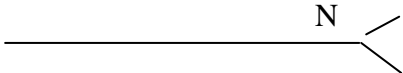
Menurut Sukamto et.al (2014:53), “ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional”.

Adapun simbol ERD menurut Sukamto dan Shalahuddin (2014:50-51) sebagai berikut:

Tabel 2.3. Simbol *Entity Relationship Diagram*(ERD)

No	Simbol	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 digunakan aplikasi komputer.



No	Simbol	Keterangan
2.	Atribut 	Field atau kolom data yang butuh disimpan dalam suatu entitas.
3.	Atribut Kunci <i>Primer</i> 	Field atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record, biasanya berupa id; kunci primer dapat lebih dari satu kolom tersebut dapat bersifat unik.
4.	Atribut multivalai/ <i>multivalue</i> 	Field 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 kerja
6.	Asosiasi/ <i>association</i> 	Penghubung antara relasi dan entitas di mana kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian.

Sumber : Sukanto et.al (2014:50-51)



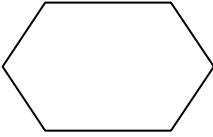

2.3.4. Pengertian *Flowchart*

Supardi (2013:58), “*Flowchart* merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program”.

Tabel 2.4. Simbol-simbol *Flowchart*

No	Simbol	Nama	Keterangan
1.		Input/Output	Simbol input/output (<i>input/output symbol</i>) digunakan untuk mewakili data input/output.
2.		Proses	Simbol proses digunakan untuk mewakili suatu proses.
3.		Garis Alir	Simbol garis alir (<i>flow lines symbol</i>) digunakan untuk menunjukkan arus dari proses.
4.		Penghubung	Simbol penghubung (<i>connector symbol</i>) digunakan untuk menunjukkan sambungan dari bagan alir yang terputus di halaman yang masih sama atau di halaman lainnya.
5.		Keputusan	Simbol keputusan (<i>decision symbol</i>) digunakan untuk suatu penyeleksian kondisi di dalam program.
6.		Proses Terdefinisi	Simbol proses terdefinisi (<i>predifined process symbol</i>) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.



No	Simbol	Nama	Keterangan
7.		Persiapan	Simbol persiapan (<i>preparation symbol</i>) digunakan untuk memberi nilai awal suatu besaran.
8.		Titik terminal	Simbol titik terminal (<i>terminal point symbol</i>) digunakan untuk menunjukkan awal dan akhir dari suatu proses.

Sumber : Supardi (2013:58-59)

2.3.5. Pengertian Kamus Data

Ladjamudin (2013 : 70), Mengemukakan ”Kamus data adalah katalog fakta tentang data dan kebutuhan”.

Sukamto et.al (2014:73), “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)”.

Tabel 2.5. Simbol-simbol dalam Kamus Data (*Data Dictionary*)

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

Sumber : Sukamto et.al (2014: 74)



2.3.6. Pengertian Daftar Kejadian (*Event List*)

Kristanto (2008:7), “Kejadian (*event*) adalah sesuatu yang terjadi pada saat tertentu”.

Kristanto (2008:70), menjelaskan bahwa daftar kejadian (*event list*) digambarkan dalam bentuk kalimat sederhana dan berfungsi untuk memodelkan kejadian yang terjadi dalam lingkungan sehari-hari dan membutuhkan tanggapan atau respon dari sistem. Adapun cara mendeskripsikan daftar kejadian, yaitu :

1. Pelaku adalah *entity* luar, jadi bukan sistem.
2. Menguji setiap *entity* luar dan mencoba mengevaluasi setiap *entity* luar yang terjadi pada sistem.
3. Hati-hati dengan kejadian yang spesifik, yang tak sengaja menyatu dalam paket yang sama.
4. Harus diingat bahwa kejadian yang dimodelkan bukan hanya interaksi normal antara sistem dengan *entity* luar, karena itu harus dievaluasi kebutuhan sistem untuk menanggapi kejadian yang gagal.
5. Setiap aliran keluaran sebaiknya merupakan respon dari kejadian.
6. Setiap kejadian yang tidak berorientasi pada waktu dalam daftar kejadian sebaiknya mempunyai masukan sehingga sistem dapat mendeteksi kejadian yang berlangsung.
7. Setiap kejadian sebaiknya menghasilkan keluaran langsung sebagai respon atau disimpan dalam berkas untuk bahan masukan.

2.4. Teori Program

2.4.1. Basis Data (*Database*)

Kadir (2013: 15), Mengemukakan “Database adalah suatu pengorganisasian data dengan tujuan memudahkan penyimpanan dan pengaksesan data”.

Yakub (2012:51), Mengemukakan “Basis data (*database*) merupakan kumpulan data yang saling berhubungan (punya relasi). Relasi biasanya ditunjukkan dengan kunci (*key*) dari tiap file yang ada”.



Ichwan (2011: 7), “Basis data (*database*) merupakan sebuah bangunan data yang terdiri dari kumpulan tabel yang saling berelasi satu dengan yang lainnya untuk memberikan suatu batasan tertentu”.

Adapun beberapa elemen basis data, diantaranya adalah sebagai berikut:

1. Tabel

Pada suatu basis data, tabel direpresentasikan menjadi suatu bentuk segiempat berupa matriks, yang terdiri dari kolom dan baris.

2. *Field*

Pada tabel basis data, kolom merupakan representasi untuk sebuah nama *field* yang pembacaan datanya dilakukan secara vertikal.

3. *Record*

Pada tabel basis data, baris merupakan suatu representasi untuk sebuah *record* yang pembacaan datanya dilakukan secara horizontal. Satu baris pada sebuah tabel merupakan data yang dimiliki oleh satu *record*. Nilai-nilai yang dimiliki oleh sebuah *record* merupakan gabungan dari semua *field* yang terdapat dalam tabel tersebut.

4. Kardinalitas

Kardinalitas merupakan batasan dari banyaknya hubungan, yang dapat dilakukan oleh himpunan entitas dalam melakukan relasi dengan himpunan entitas lainnya. Variasi kemungkinan untuk melakukan relasi yang dimiliki oleh kardinalitas terdiri dari empat macam, yaitu:

- a. Satu ke satu \longrightarrow (1:1).
- b. Satu ke banyak \longrightarrow (1:N).
- c. Banyak ke satu \longrightarrow (N:1).
- d. Banyak ke banyak \longrightarrow (N:N).

2.4.2. Pengertian *Dreamweaver CS3*

Madcoms (2008:1-5), Mengemukakan “*Dreamweaver* adalah sebuah HTML editor profesional untuk mendesain web secara visual dan mengelola situs atau halaman web”.

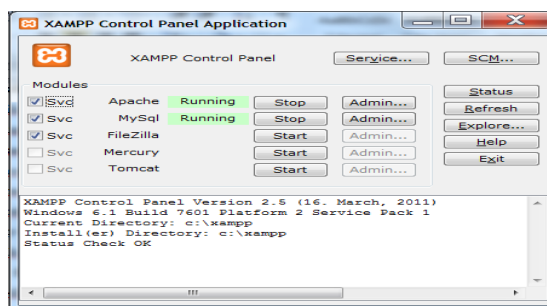


Adapun komponen-komponen yang disediakan dalam ruang kerja *Dreamweaver CS3* antara lain adalah *Insert Bar*, *Document Toolbar*, *Jendela Document*, *Panel Group*, *Tag Selector*, *Property Inspector*, dan *Site Panel*. Penjelasan dari komponen-komponen tersebut adalah :

- a. *Insert Bar*, berisi tombol-tombol untuk menyisipkan berbagai macam objek seperti image, tabel, dan layer ke dalam dokumen.
- b. *Document Toolbar*, berisi tombol-tombol dan menu pop-up yang menyediakan tampilan berbeda dari jendela dokumen.
- c. *Coding Window*, berisi kode-kode *HTML* dan tempat untuk menuliskan kode-kode pemrograman, misalnya *PHP* atau *ASP*.
- d. *Panel Group*, adalah kumpulan panel yang saling berkaitan satu sama lainnya yang dikelompokkan dibawah satu judul.
- e. *Property Inspector*, digunakan untuk melihat dan mengubah berbagai properti objek atau teks.
- f. *Jendela dokumen*, berfungsi untuk menampilkan dokumen dimana anda sekarang bekerja.
- g. *Ruler*, digunakan untuk memudahkan pengukuran saat anda mendesain suatu halaman web.
- h. *Site Panel*, digunakan untuk mengatur file-file dan folder-folder yang membentuk situs Web Anda.

2.4.3. Sekilas tentang XAMPP

Nugroho (2013:1), Mengemukakan ”XAMPP adalah paket program web lengkap yang dapat dipakai untuk belajar pemrogram web khususnya PHP dan MySQL.



Gambar 2.2. Tampilan *Svc Apache Running*

Sumber : Nugroho (2013:1)

2.4.4. Sekilas Tentang *HTML*

2.4.4.1. Pengertian *HTML*

Asropudin (2013:44), “*HTML* singkatan dari *Hyper Text Markup Language*, adalah bahasa komputer yang digunakan untuk membuat sebuah halaman *webpage*”.

Anhar (2010:40), mengemukakan bahwa “*HTML* merupakan sebuah bahasa pemrograman *web* yang memiliki sintak atau aturan dalam menuliskan script atau kode-kode, sehingga *browser* dapat menampilkan informasi dengan membaca kode-kode *HTML*”.

2.4.4.2. Menulis Kode *HTML*

Anhar (2010:41), “Program yang digunakan untuk membuat *document HTML* menggunakan *HTML editor*. Ada banyak *HTML editor* yang bisa digunakan, di antaranya: *Notepad*, *Ms. FrontPage*, dan *Dreamweaver*”.

Berikut ini contoh kode yang dituliskan di *Notepad* dan disimpan sebagai *document HTML*.



```

<html>
<head>
<title>Latihan Membuat Document HTML</title>
</head>
<body>
<p>ini Document HTMLku yang pertama</p>
<p>by: Aggi Dwi</p>
</body>

```

Gambar 2.3. Menulis Kode *HTML* di *Notepad*

Sumber : Anhar (2010: 41)

Setiap dokumen *HTML* harus diawali dan ditutup dengan tag *HTML* `<HTML></HTML>`. Tag *HTML* memberi tahu *browser* bahwa yang ada di dalam kedua tag tersebut adalah dokumen *HTML*. Bagian *Header* dari dokumen *HTML* diapit oleh tag `<HEAD></HEAD>` di dalam bagian ini biasanya dimuat tag *TITLE* yang menampilkan judul dari halaman pada *titlenya browser*. Bagian *Body* dari dokumen *HTML* diapit oleh tag `<BODY></BODY>`. Document *body* digunakan untuk menampilkan *text*, *image link*, dan semua yang akan di tampilkan pada halaman *web*.

2.4.5. Pengertian CSS

Madcoms (2010:141), Mengemukakan “*Cascading Style Sheets (CSS)* adalah suatu kumpulan kode-kode untuk memformat, yang mengendalikan tampilan isi dalam suatu halaman *web*”.

2.4.6. Pengertian JavaScript

Asporudin (2013:52), Mengemukakan “*Javascript* adalah bahasa script interpreter untuk menciptakan isi yang aktif dalam halaman *web*”.

Winarno (2014:134) menjelaskan bahwa cara menggunakan JavaScript adalah dengan menggunakan tag `<script>` dan `</script>`. JavaScript biasanya dituliskan pada bagian `<body>` dalam *HTML*. Contohnya :



```

<!Doctype html>

  <html>

  <body>

  <script>

  document.write("<h1>Ini adalah Heading </h1>");

  document.write("<p>Ini adalah paragraf.</p>");

```

Sumber : Winarno (2014:134)

2.4.7. Sekilas Tentang MySQL

2.4.7.1. Pengertian MySQL

Ichwan (2011:23), “MySQL adalah *Relational Database Management System* (RDBMS) yang didistribusikan secara gratis di bawah lisensi GPL (*General Public License*)”.

Kadir (2013:15), “MySQL adalah nama *database server*. *Database server* adalah *server* yang berfungsi untuk menangani *database*”.

Nugroho (2013:7), “MySQL adalah folder utama untuk *database MySQL* server. Di dalamnya terdapat subfolder data (lengkapnya; C:\xampp\mysql\data) untuk merekam semua nama database, serta subfolder bin yang berisi tools klien dan server *MySQL*”.



Gambar 2.4. Tampilan Logo MySQL

Sumber : Hakim (2014: 49)



2.4.7.2. Fungsi-fungsi MySQL

Kadir (2008:360-379), Sejumlah fungsi yang berawalan **mysql_** yang digunakan untuk mengakses *database server MySQL* dibahas di bawah ini:

1. Fungsi **mysql_connect**

Fungsi **mysql_connect()** digunakan untuk membuat hubungan ke *database MySQL* yang terdapat pada suatu *host*. Bentuk Pemanggilannya:
mysql_connect (*host, nama_pemakai, password*)

2. Fungsi **mysql_close**

Fungsi ini berguna untuk menutup hubungan ke *database MySQL*. Bentuk Pemanggilannya:

mysql_close (*pengenal_hubungan*)

3. Fungsi **mysql_select_db**

Fungsi ini berguna untuk memilih *database*, seperti kalau menggunakan perintah USE pada program klien *MySQL*. Format Pemanggilannya:

mysql_select_db (*database, pengenal_hubungan*)

4. Fungsi **mysql_query**

Fungsi **mysql_query** berguna untuk mengeksekusi permintaan terhadap sebuah tabel atau sejumlah tabel. Bentuk Pemanggilannya:

mysql_query (*permintaan, pengenal_hubungan*)

5. Fungsi **mysql_db_query**

Fungsi ini berguna untuk menjalankan suatu permintaan terhadap suatu *database*. Bentuk Pemanggilannya:

mysql_db_query (*database, permintaan, pengenal_hubungan*)

6. Fungsi **mysql_num_rows**

Fungsi ini berguna untuk memperoleh jumlah baris dari suatu hasil permintaan (query) yang menggunakan SELECT. Bentuk Pemanggilannya:

mysql_num_rows (*pengenal_hasil*)

7. Fungsi **mysql_affected_rows**

Fungsi ini berguna untuk memperoleh jumlah baris yang dikenai operasi INSERT, DELETE, dan UPDATE.. Bentuk Pemanggilannya:



Mysql_affected_rows (*[pengenal_hubungan]*)

8. Fungsi `mysql_num_fields`

Fungsi ini berguna untuk memperoleh jumlah kolom pada suatu hasil permintaan. Bentuk Pemanggilannya:

Mysql_num_fields (*pengenal_hasil*)

9. Fungsi `mysql_fetch_row`

Fungsi ini menghasilkan suatu *array* yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasil. Bentuk Pemanggilannya:

Mysql_fetch_row (*pengenal_hasil*)

10. Fungsi `mysql_fetch_array`

Fungsi ini mempunyai kegunaan serupa dengan **mysql_fetch_row()**. Hanya saja, setiap kolom akan disimpan dua kali pada *array* hasil. Yang pertama memiliki indeks angka (dimulai dari nol) dan yang kedua berindekskan nama kolom. Bentuk Pemanggilannya:

Mysql_fetch_array (*pengenal_hasil*)

11. Fungsi `mysql_fetch_field`

Fungsi ini berguna untuk memperoleh informasi suatu kolom. Bentuk Pemanggilannya:

Mysql_fetch_field (*pengenal_hasil, [nomor_kolom]*)

12. Fungsi `mysql_data_seek`

Fungsi ini berguna untuk memindah pointer pada suatu himpunan hasil supaya menunjuk ke baris tertentu. Format Pemanggilannya:

Mysql_data_seek (*pengenal_hasil, nomor_baris*)

13. Fungsi `mysql_field_seek`

Fungsi ini berguna untuk memindah pointer pada suatu himpunan hasil supaya menunjuk ke kolom tertentu. Format Pemanggilannya:

Mysql_field_seek (*pengenal_hasil, nomor_kolom*)

14. Fungsi `mysql_create_db`

Fungsi ini berguna untuk menciptakan *database* MySQL. Format Pemanggilannya:



Mysql_create_db (*database, [pengenal_hubungan]*)

15. Fungsi `mysql_drop_db`

Fungsi ini berguna untuk menghapus *database MySQL*. Format Pemanggilannya:

Mysql_drop_db (*database, [pengenal_hubungan]*)

16. Fungsi `mysql_list_dbs`

Fungsi ini berguna untuk menghasilkan daftar *database MySQL*. Format Pemanggilannya:

Mysql_list_dbs (*[pengenal_hubungan]*)

17. Fungsi `mysql_list_tables`

Fungsi ini berguna untuk memperoleh daftar nama tabel dalam suatu *database*. Format Pemanggilannya:

Mysql_list_tables (*database, [pengenal_hubungan]*)

18. Fungsi `mysql_list_fields`

Fungsi ini berguna untuk memperoleh daftar nama kolom dalam suatu *database*. Format Pemanggilannya:

Mysql_list_fields (*database, nama_tabel, [pengenal_hubungan]*)

2.4.8. Sekilas Tentang *PHP*

2.4.8.1. Pengertian Pemrograman *PHP*

Hakim (2014:2), “*PHP* (singkatan dari *Personal Home Page*) adalah bahasa pemrograman yang digunakan untuk membuat Aplikasi Web”.

Kadir (2013:120), menyatakan bahwa “*PHP* merupakan bahasa pemrograman yang ditujukan untuk membuat aplikasi *web*”.

Nugroho (2013:153), “*PHP* itu adalah bahasa program yang digunakan untuk membuat aplikasi berbasis web (website, blog, atau aplikasi web)”.



Gambar 2.5. Tampilan Logo *PHP*

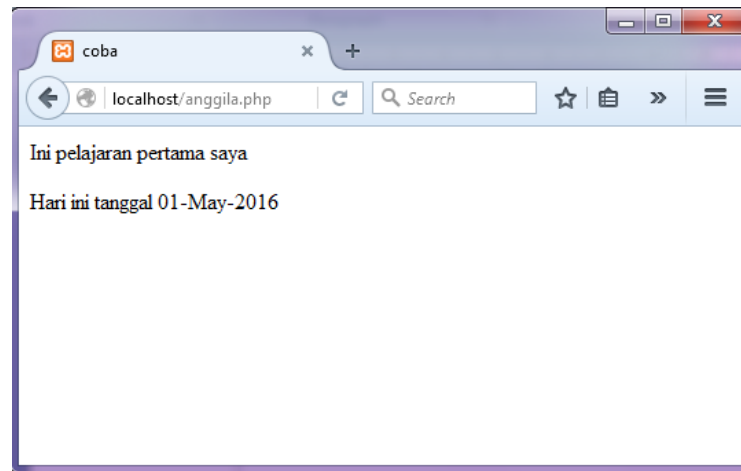
Sumber : Hakim (2014: 2)

2.4.8.2. Script *PHP*

Madcoms (2010:350-351), “Dalam penulisannya, *script PHP* tidak harus berdiri sendiri namun dapat disisipkan di antara kode *HTML*. *Script PHP* harus selalu diawali dengan `<? atau <?php` dan diakhiri dengan `?>`”.

Contoh penulisan *script PHP* berikut:

```
<html>
    <head>
    <title> coba </title>
    </head>
    <body>
    <p>Ini pelajaran pertama saya</p>
<?PHP
    echo "Hari ini tanggal " ;
    echo date("d-M-Y");
```

Gambar 2.6. Contoh *Script PHP*

Semua teks yang diketik setelah tanda buka *script* (`<?>`) dan tanda tutup *script* (`?>`) akan dieksekusi sebagai suatu *script PHP*. Beberapa cara untuk memberi keterangan di dalam *script PHP* adalah:

- a. Gunakan *tag* `/*` dan diakhiri *tag* `*/` apabila jumlah keterangan lebih dari 1 baris.
- b. Gunakan *tag* `//`, *tag* ini digunakan untuk keterangan yang hanya terdiri dari satu baris saja.
- c. Gunakan *tag* `#`, *tag* ini juga digunakan untuk 1 baris komentar saja.