



BAB II

TINJAUAN PUSTAKA

2.1 Teori Umum

2.1.1 Pengertian Komputer

Sujatmiko (2012:70) Komputer adalah sebarang mesin yang melakukan tiga hal yaitu menerima input terstruktur, memprosesnya sesuai hukum-hukum yang ditentukan, memproduksi hasilnya sebagai output.

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

Dari beberapa pendapat di atas dapat disimpulkan bahwa komputer adalah alat bantu manusia yang digunakan sebagai proses pengolahan informasi untuk menerima, mengolah, menyimpan dan mengontrol data kemudian di proses data untuk menghasilkan data dalam bentuk lain (*output*) yang berguna atau bermanfaat.

2.1.2 Pengertian Aplikasi

Sujatmiko (2012:23) Aplikasi adalah program komputer yang dibuat oleh suatu perusahaan komputer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu, misalnya *Ms-word*, *Ms-excel*.

Asropudin (2012:6) Aplikasi adalah software yang dibuat oleh suatu perusahaan computer untuk mengerjakan tugas-tugas tertentu, misalnya MS-Word, Ms-Excel.

Dari beberapa pendapat di atas dapat disimpulkan bahwa aplikasi adalah sebuah program komputer yang dibuat untuk menolong manusia dalam melaksanakan tugas tertentu.

2.1.3 Pengertian Pengolahan Data

Kristanto (2011:8) Pengolahan Data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan. Semakin banyak data dan kompleksnya aktivitas pengolahan data



dalam suatu organisasi, baik itu organisasi besar maupun organisasi kecil, maka metode pengolahan data yang tepat sangat dibutuhkan.

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

Dari beberapa pendapat di atas dapat disimpulkan bahwa pengolahan data merupakan bagian yang amat penting dalam metode ilmiah, karena pengolahan data, data tersebut dapat diberi arti dan makna yang berguna dalam memecahkan masalah penelitian.

2.1.4 Pengertian Data

Ladjamudin (2013:8) Data adalah deskripsi dari sesuatu kejadian yang dihadapi.

Sujatmiko (2012:76) Data adalah kumpulan dari angka-angka maupun karakter-karakter yang tidak memiliki arti. Data dapat diolah sehingga menghasilkan informasi.

Yakub (2012:5) Menurut (McLeod, 2004) Data adalah kenyataan yang menggambarkan adanya suatu kejadian, data terdiri dari fakta dan angka yang secara relatif tidak berarti bagi pemakai.

Dari beberapa pendapat di atas dapat disimpulkan bahwa data adalah kumpulan dari angka, huruf, dan karakter yang tidak mempunyai makna untuk mendeskripsikan tentang sesuatu kejadian yang dihadapi.

2.1.5 Pengertian Penjualan

Indrajani (2015:62) menjelaskan, kegiatan penjualan terdiri atas penjualan barang dan jasa, baik secara kredit maupun secara tunai. Dalam transaksi penjualan kredit, jika pesanan dari pelanggan telah dipenuhi dengan pengiriman barang atau penyerahan jasa, untuk jangka waktu tertentu perusahaan memiliki piutang kepada pelanggannya. Dalam sistem penjualan secara tunai, barang atau jasa baru diserahkan oleh perusahaan kepada pembeli jika perusahaan telah menerima kas dari pembeli.



2.1.6 Pengertian Sistem

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

Sutabri (2012:15) Sistem merupakan suatu bentuk integrasi antara satu komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi di dalam sistem tersebut.

Dari beberapa pendapat di atas dapat disimpulkan bahwa sistem adalah jaringan kerja yang saling berhubungan membentuk satu kesatuan untuk mencapai sasaran atau tujuan tertentu.

2.1.7 Pengertian Program

Sutarman (2012:3) 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.

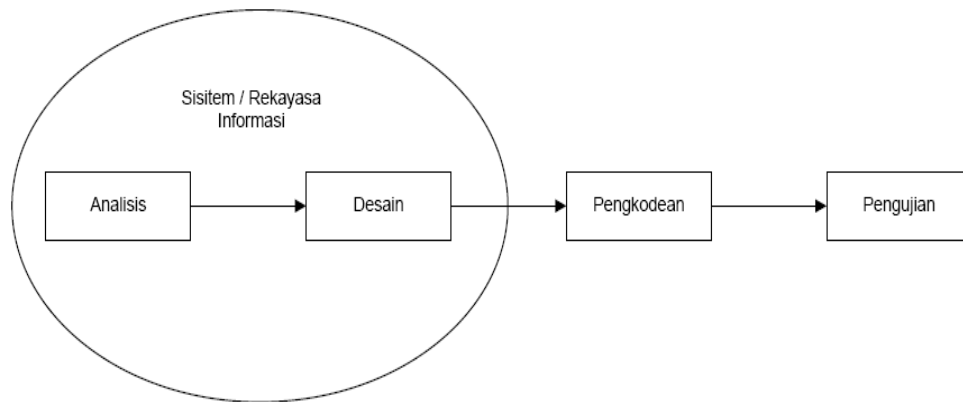
2.1.8 Pengertian Cahaya Murni Sriwindo

PT Cahaya Murni Sriwindo merupakan perusahaan yang bergerak di bidang penjualan *furniture* dengan menjual berbagai macam produk.

Produk yang dihasilkan dan dijual dari PT Cahaya Murni Sriwindo ini sendiri yaitu meliputi kursi plastik, *spring bed*, kasur busa, dan produk rumah tangga lainnya. PT Cahaya Murni Sriwindo juga merupakan cabang dari Cahaya Buana Group (CBG) yang berpusat di Bogor.

2.1.9 Metode Pengembangan Sistem

Menurut Sukamto dan Shalahudin (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. Berikut adalah gambar model air terjun :



Gambar 2.1 Ilustrasi Model Waterfall

a. Analisis Kebutuhan Perangkat Lunak

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 pengodean. Tahap ini mentranslasi kebutuhan sistem dari tahap analisis kebutuhan ke representasi desain agar dapat diimplementasikan menjadi program pada tahap selanjutnya.

c. Pengodean

Pada tahap pengodean, 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 Pendukung (*support*) atau Pemeliharaan (*maintenance*)

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.

2.2 Teori Khusus

2.2.1 Pengertian DFD (*Data Flow Diagram*)

Kristanto (2008:61) menyatakan, “*Data flow diagram* (DFD) adalah suatu model logika atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut”.


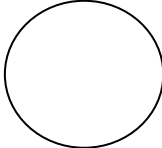
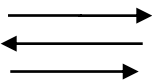
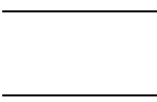
Ladjamudin (2013:64) menyatakan, “*Data flow diagram* (DFD) merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil.”

Sukanto dan Shalahuddin (2014:69) menjelaskan, *Data Flow Diagram* (DFD) awalnya dikembangkan oleh Chris Gane dan Trish Sarson pada tahun 1979 yang termasuk dalam *Structured Systems Analysis and Design Methodology* (SSADM) yang ditulis oleh Chris Gane dan Trish Sarson. Sistem yang dikembangkan ini berbasis pada dekomposisi fungsional dari sebuah sistem. Informasi yang ada di dalam perangkat lunak dimodifikasi dengan beberapa transformasi yang dibutuhkan. *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*). DFD dapat digunakan untuk merepresentasikan sebuah sistem atau perangkat lunak pada beberapa level abstraksi. DFD dapat dibagi menjadi beberapa level yang lebih detail untuk merepresentasikan aliran informasi atau fungsi yang lebih detail. DFD menyediakan mekanisme untuk pemodelan fungsional ataupun

pemodelan aliran informasi. Oleh karena itu, DFD lebih sesuai digunakan untuk memodelkan fungsi-fungsi perangkat lunak yang akan diimplementasikan menggunakan pemrograman terstruktur karena pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur-prosedur.

Adapun simbol-simbol atau notasi-notasi yang menggambarkan *Data Flow Diagram* (DFD), sebagai berikut :

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

No	Yourdon/De Marco	Nama Simbol	Keterangan
1.		Entitas Eksternal	Entitas eksternal dapat berupa orang unit terkait yang berinteraksi dengan sistem, tetapi di luar sistem.
2.		Proses	Orang, unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
3.		Aliran Data	Aliran data dengan arah khusus dari sumber ke tujuan.
4.		Data Store	Penyimpanan data atau tempat data di-refer oleh proses.

(Sumber: Sukamto dan Shalahuddin, 2014:71-72)

Berikut 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 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. Pada satu diagram DFD sebaiknya jumlah modul tidak boleh lebih dari 20 buah. Jika lebih dari 20 buah modul, diagram akan terlihat rumit dan susah untuk dibaca sehingga menyebabkan sistem yang dikembangkan juga menjadi rumit.

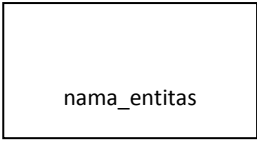
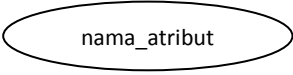
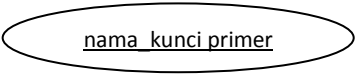
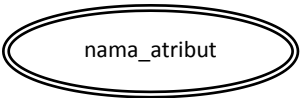
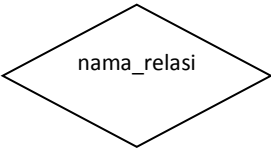
2.2.2 Pengertian *Entity Relationship Diagram (ERD)*

Yakub (2012:60) ERD merupakan suatu model jaringan yang menggunakan susunan data yang disimpan pada sistem secara abstrak. ERD juga menggambarkan hubungan antara satu entitas yang memiliki sejumlah atribut dengan entitas yang lain dalam suatu sistem yang terintegrasi. ERD digunakan oleh perancang sistem untuk memodelkan data yang nantinya akan dikembangkan menjadi basis data (*database*).

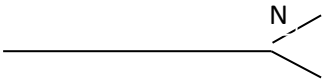
Sukamto dan Shalahuddin (2014:53) menyatakan, “*Entity Relationship Diagram (ERD)* adalah bentuk paling awal dalam melakukan perancangan basis data relasional.”

Adapun simbol-simbol yang digunakan dalam ERD (*Entity Relationship Diagram*) yaitu :

Tabel 2.2 Simbol-simbol ERD (*Entity Relationship Diagram*)

No.	Simbol	Keterangan
1.	Entitas/ <i>Entity</i> 	Entitas merupakan data inti yang akan tersimpan; 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.
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.
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 	<i>Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja.</i>

Lanjutan Tabel 2.2 Simbol-simbol ERD (*Entity Relationship Diagram*)

6.	Asosiasi/Association 	Penghubung antara relasi dan entitas di mana dikedua ujungnya punya multiplicity kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas yang lain disebut dengan one to many menghubungkan entitas A dan entitas B.

(Sumber: Sukanto dan Shalahuddin, 2014:50-51)

2.2.3 Pengertian *Flowchart Diagram*

Ladjamudin (2013:263) menyatakan, “*Flowchart* adalah bagan–bagan yang mempunyai arus yang menggambarkan langkah–langkah penyelesaian suatu masalah”.

2.2.3.1 Simbol-simbol *Flowchart*

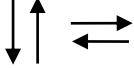

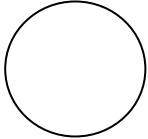
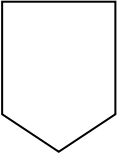
Flowchart disusun dengan simbol. Simbol ini dipakai sebagai alat bantu menggambarkan proses didalam program. Simbol-simbol yang digunakan dapat dibagi menjadi 3 (tiga) kelompok, yakni sebagai berikut :

1. *Flow Direction Symbols* (Simbol penghubung atau alur)
2. *Processing Symbols* (Simbol proses).
3. *Input-output Symbols* (Simbol input output)

2.2.3.2 *Flow Direction Symbols*

Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga *connecting line*. Simbol-simbol tersebut adalah sebagai berikut :

Tabel 2.3 Simbol-simbol *flowchart*


No.	Simbol	Keterangan
1.		Simbol arus / flow Untuk menyatakan jalannya arus suatu proses.
2.		Simbol Communication link Untuk menyatakan bahwa adanya transisi suatu data/informasi dari suatu lokasi ke lokasi lainnya.
3.		Simbol Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang sama.
4.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/lembar yang berbeda.

(Sumber: Ladjamudin, 2013:266)

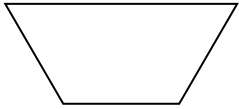
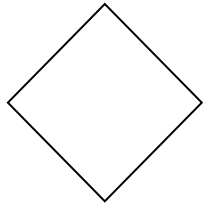

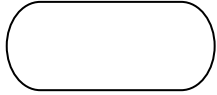

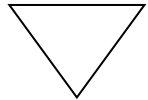

2.2.3.3 Processing Symbols

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses atau prosedur. Simbol-simbol tersebut adalah sebagai berikut :

Tabel 2.4 *Processing Symbols*

No.	Simbol	Keterangan
1.		Simbol Offline Connector Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman atau lembar yang berbeda.

Lanjutan Tabel 2.4 *Processing Symbols*



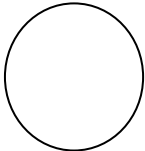


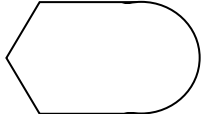
2.		<i>Simbol Manual</i> Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).
3.		<i>Simbol Decision/logika</i> Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/ tidak.
4.		<i>Simbol Predefined Proses</i> Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
5.		<i>Simbol Terminal</i> Untuk menyatakan permulaan atau akhir suatu program.
6.		<i>Simbol Keying Operation</i> Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
7.		<i>Simbol Offline Storage</i> Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
8.		<i>Simbol Manual Input</i> Untuk memasukkan data secara manual dengan menggunakan online keyboard.

(Sumber: Ladjamudin, 2013:267)

2.2.3.4 Input-output Symbols

Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media *input* atau *output*. Simbol-simbol tersebut adalah sebagai berikut :

Tabel 2.5 *Input-output Symbols*

No.	Simbol	Keterangan
1.		<i>Simbol Input-output</i> Untuk menyatakan proses input dan output tanpa tergantung dengan jenis peralatannya.
2.		<i>Simbol Punched Card</i> Untuk menyatakan input berasal dari kartu atau output dituliskan ke kartu.
3.		<i>Simbol Magnetic-tape Unit</i> Untuk menyatakan input berasal dari pita magnetic atau output disimpan ke pita magnetic.
4.		<i>Simbol Disk Storage</i> Untuk menyatakan input berasal dari <i>disk</i> atau <i>output</i> disimpan ke <i>disk</i> .
5.		<i>Simbol Document</i> Untuk mencetak laporan ke <i>printer</i> .
6.		<i>Simbol Display</i> Untuk menyatakan peralatan <i>output</i> yang digunakan berupa layar (video, komputer).



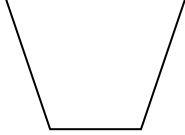

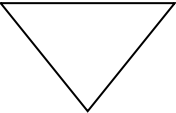
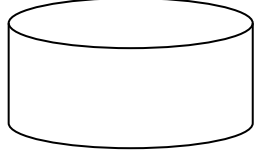
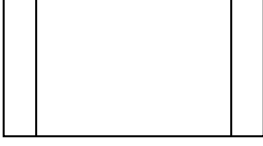
(Sumber: Ladjamudin, 2013:268)

2.2.4 Pengertian *Blockchart Diagram*

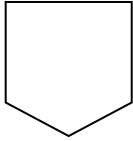
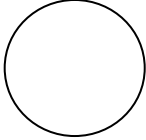
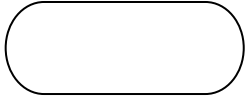
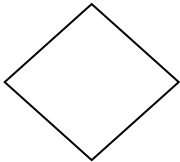
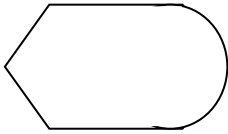

Kristanto (2008:75), *Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

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

Tabel 2.6 Simbol-simbol dalam *Blockchart*

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 computer
5.		Menandakan dokumen yang diarsipkan (arsip manual)
6.		Data penyimpanan (data storage)
7.		Proses apa saja yang tidak terdefinisi termasuk aktivitas fisik

Lanjutan Tabel 2.6 Simbol-simbol dalam *Blockchart*

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

(Sumber: Kristanto, 2008:75-77)

2.2.5 Pengertian Kamus Data

Sukamto dan Shalahuddin (2014:73) menjelaskan, kamus data (*data dictionary*) dipergunakan untuk memperjelas aliran data yang digambarkan pada DFD. 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 dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur. Kamus data biasanya berisi :

- a. Nama – nama dari data
- b. Digunakan pada – merupakan proses-proses yang terkait data

c. Deskripsi – merupakan deskripsi data

Informasi tambahan – seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data.

Kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan sebagai berikut :

Tabel 2.7 Simbol-simbol dalam Kamus Data (*Data Dictionary*)

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

(Sumber : Sukamto dan Shalahuddin, 2014:74)

Kamus data pada DFD nanti harus dapat dipetakan dengan hasil perancangan basis data yang dilakukan sebelumnya. Jika ada kamus data yang tidak dapat dipetakan pada tabel hasil perancangan basis data berarti hasil perancangan basis data dengan perancangan dengan DFD masih belum sesuai, sehingga harus ada yang diperbaiki baik perancangan basis datanya dan DFD nya.

2.3 Teori Program

2.3.1 Pengertian Basis Data (*Database*)

Sujatmiko (2012:76), Basis Data merupakan basis data atau representasi kumpulan fakta yang saling berhubungan disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.

Kristanto (2011:73) Basis Data merupakan kumpulan data, yang dapat digambarkan sebagai aktifitas dari satu atau lebih organisasi yang berelasi.



Yakub (2012:55) Basis Data merupakan koleksi dari data-data yang terorganisasi dengan cara sedemikian rupa sehingga data tersebut mudah disimpan dan dimanipulasi.

Priyadi (2014:3) juga menjelaskan beberapa elemen basis data, yaitu :

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

Dari beberapa pendapat di atas dapat disimpulkan bahwa basis data adalah kumpulan data informasi yang disimpan didalam komputer secara sistematis sehingga dapat diperiksa, diolah atau dimanipulasi menggunakan suatu program komputer untuk memperoleh informasi dari *database* tersebut.



2.3.2 Pengertian XAMPP

Risnandar, dkk (2013:53) XAMPP adalah suatu program yang digunakan sebagai server untuk mengakses fungsi yang ada dalam halaman website tersebut agar bisa diakses oleh user.

2.3.3 Pengertian MySQL

Rosa dan Shalahuddin (2013:46), “SQL (*Structured Query Language*) adalah bahasa yang digunakan untuk mengelola data pada RDBMS (*Relational Database Management System*). Suatu database terdiri dari banyak tabel.. SQL awalnya dikembangkan berdasarkan teori aljabar relasional dan kalkulus.”

Risnandar, dkk (2013:92), “MySQL merupakan basis data yang bersifat *open source* sehingga banyak digunakan di dunia. Walaupun gratis, MySQL tetap berkualitas dan sudah cukup memberikan performa yang memadai.”

Kesimpulannya, MySQL adalah bahasa yang digunakan untuk mengelola data pada RDBMS. RDBMS adalah kependekan dari *Relational Database Management System*. RDBMS adalah program yang melayani sistem basis data yang entitas utamanya terdiri dari tabel-tabel yang mempunyai relasi dari satu tabel ke tabel yang lain yang bersifat *open source* sehingga banyak digunakan di dunia.

2.3.4 Pengertian PHP

Sujatmiko (2012:213), “PHP adalah bahasa pemrograman yang bisa kita gunakan untuk membuat aplikasi web.”

Risnandar, dkk (2013:57), “PHP (*Hypertext Preprocessing*) merupakan bahasa pemrograman yang biasa digunakan untuk membuat halaman HTML.”

Kesimpulannya, PHP adalah bahasa pemrograman berbasis *server-side* yang bisa kita gunakan untuk membuat aplikasi web yang disisipkan pada HTML, yang dijalankan di *server*, dan juga bisa digunakan untuk membuat aplikasi dekstop.

2.3.5.1 Tipe Data PHP

Risnandar dkk (2013:60), PHP mempunyai tipe data dasar yaitu:

1. *Integer*



Integer merupakan tipe data yang mencakup semua bilangan bulat. Range bilangan *integer* adalah antara -2.147.483.647 sampai dengan 2.147.483.647.

Contoh *integer* :

\$biaya = 1000;

\$a = 1234;

\$a = -123;

2. *Floating Point*

Floating point merupakan tipe data yang mencakup semua bilangan desimal (bilangan yang memiliki angka dibelakang koma). Range bilangan *floating point* antara $1e-308$ sampai dengan $1e308$ (10^{-308} s.d. 10^{308}).

Contoh *floating point* :

\$a = 1.234;

\$a = 1.2e3;

3. *Character*

Character merupakan tipe yang digunakan untuk menyimpan data-data yang berupa karakter (1 huruf). Penulisannya biasanya diapitnya dengannya tandanya kutipnya satunya ('...').

Contoh *character* :

\$abjad = 'a';

\$abjad = 'b';

Jika karakter-karakter berkumpul membentuk suatu kata, kalimat atau paragraf maka tipe tadi tidak bisa lagi disebut sebagai *character* tapi bertipe *string*. *String* merupakan tipe data tersendiri dan tidak dapat dikelompokkan menjadi tipe data dasar. Penulisannya biasa diapit dengan tanda kutip 2 ("...").

Contoh *string* :

\$nama = "gus";

\$salamat = "Jalan Tanjung";

4. *Boolean*

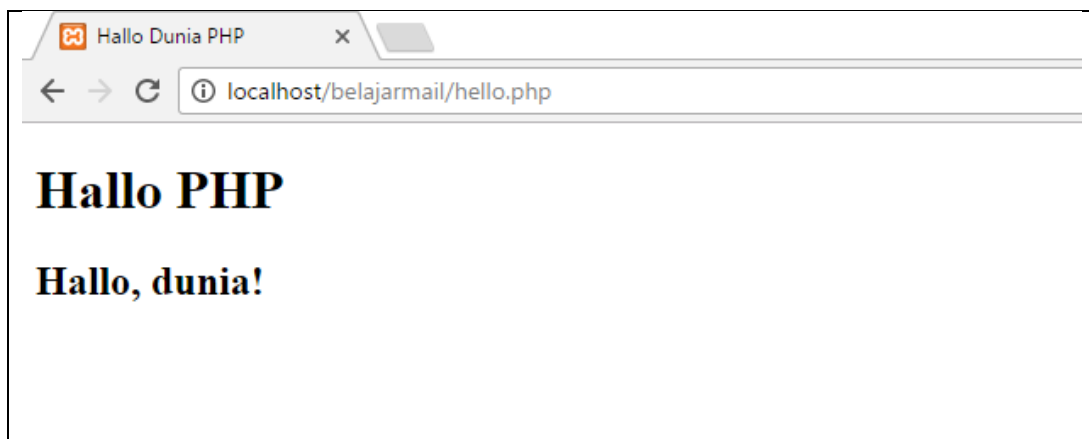
Boolean menyimpan data yang bernilai logika (benar atau salah, 0 atau 1, ya atau tidak).

2.4.5.2. Skrip PHP

Kode (*Script*) PHP yang sering disebut dengan istilah *embedded script* yaitu *script PHP* yang disisipkan diantara *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*:

```
<!DOCTYPE html>
<html>
<head>
  <title>Hallo Dunia PHP</title>
</head>
<body>
  <h1>Hallo PHP</h1>
  <?php
    echo "<h2>Hallo, dunia!</h2>";
    phpInfo();
  ?>
</body>
</html>
```

Bila dijalankan melalui browser, kode tersebut membentuk tampilan seperti berikut:



Gambar 2.2 Tampilan PHP sederhana