



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Sujatmiko (2012:156), mengemukakan bahwa komputer merupakan mesin yang dapat mengolah data digital dengan mengikuti serangkaian perintah atau program.

Asropudin (2013:19) menjelaskan 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.

Hartono (2013:27), mengemukakan bahwa komputer merupakan sebuah mesin yang dapat dikendalikan melalui perintah yang dirancang untuk secara otomatis melakukan serangkaian urutan penghitungan atau proses-proses yang diurutkan secara logis.

2.1.2. Pengertian Data

Kadir (2003:29), mengatakan bahwa data adalah deskripsi tentang benda, kejadian, aktivitas, dan transaksi, yang tidak mempunyai makna atau tidak berpengaruh secara langsung kepada pemakai.

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

Hal yang sama dikemukakan oleh Sutabri (2012:16), data adalah bahan mentah untuk diolah yang hasilnya kemudian menjadi informasi.

2.1.3. Pengertian Sistem

Davis (dalam Al-Bahra, 2005:3), mendefinisikan sistem sebagai bagian-bagian yang saling berkaitan yang beroperasi bersama untuk mencapai beberapa sasaran atau maksud.



Selain itu Sutabri (2012:10), mengemukakan bahwa suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang teroganisir, saling berinteraksi, saling bergantung satu sama lain, dan terpadu.

Hal yang sama dikemukakan oleh Kadir (2014:7), sistem didefinisikan sebagai kumpulan prosedur yang saling berkaitan dan saling terhubung untuk melakukan tugas bersama-sama.

2.1.4. Pengertian Informasi

Davis (dikutip, Kadir 2003:31) mengatakan bahwa informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang.

Sementara Sutabri (2012:22), Informasi adalah data yang telah diklasifikasi atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan.

Sibero (2013:10), Informasi adalah sekumpulan data yang memiliki maksud dan tujuan serta dapat memberikan keterangan akurat yang diperlukan dalam pengambilan keputusan.

2.1.5. Pengertian Sistem Informasi

Sutabri (2012:38), mengemukakan bahwa sistem informasi suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan laporan-laporan yang diperlukan oleh pihak tertentu.

Sementara Al-bahra (2005:13), mendefinisikan bahwa sistem informasi merupakan suatu sistem yang dibuat oleh manusia yang terdiri dari komponen-komponen dalam organisasi untuk mencapai suatu tujuan yaitu menyajikan informasi.

yaqub (2012:17) menjelaskan bahwa system informasi juga merupakan sekumpulan dari komponen-komponen dalam organisasi yang berhubungan dengan proses penciptaan dan aliran informasi.



2.2. Teori Judul

2.2.1. Pengertian Aplikasi

Sujatmiko (2012:23), menyatakan bahwa aplikasi adalah program komputer yang dibuat oleh suatu perusahaan computer untuk membantu manusia dalam mengerjakan tugas-tugas tertentu.

Sementara itu Nufus (2016 : 34) , mengemukakan bahwa aplikasi adalah karya hias dalam seni jahit-menjahit dengan menempelkan (menjahitkan) guntingan-guntingan kain yang dibentuk seperti bunga (buah, binatang, dan sebagainya) pada kain lain sebagai hiasan.

2.2.2. Administrasi

Husaini (2013:2) Berpendapat bahwa Administrasi dalam Bahasa Belanda mengandung pengertian sempit yaitu pekerjaan yang berhubungan dengan ketatausahaan (surat-menyurat) yang mengola disini meliputi perencanaan, pengorganisasian, pengarahan, pemotivasian, pengkoordinasian, penganggaran, pelaporan dan pengendalian.

2.2.3. Surat

Dyah dan Novita (2016:72) Mengemukakan bahwa surat adalah wujud pribadi pihak tertentu untuk menyampaikan informasi yang tidak mungkin disampaikan secara langsung. Oleh karena itu menyusun surat harus hati-hati dan cermat dengan adanya surat pihak lain dapat menilai “siapakah kita”. Surat merupakan bukti tertulis yang sewaktu-waktu dapat digunakan sebagai bahan bukti.

2.2.4. Pengertian Aplikasi Tertib Administrasi Surat-Menyurat di Perusahaan Daerah Air Minum (PDAM) Tirta Musi Unit Rambutan Palembang

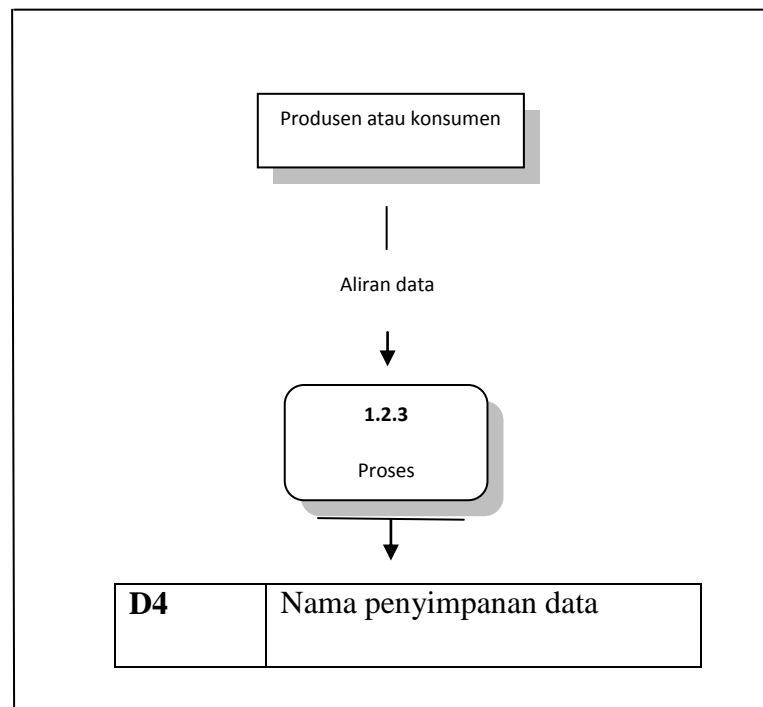
Aplikasi Tertib Administrasi Surat-Menyurat di Perusahaan Daerah Air Minum (PDAM) Tirta Musi Unit Rambutan Palembang adalah suatu perangkat lunak (*software*) berbasis *website* untuk mempermudah menyimpan arsip dari surat-menyurat dari surat masuk sampai surat keluar.



2.3. Teori Khusus

2.3.1. Pengertian DFD (*Data Flow Diagram*)

Rosa 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. System yang dikembangkan ini berbasis pada dekomposisi fungsional dari sebuah system.” Berikut adalah contoh DFD yang dikembangkan oleh Chris Gane dan Trish Sarson:



Gambar 2.1 Contoh DFD yang dikembangkan Chris Gane dan Trish Sarson

Edward Yourdon dan Tom DeMarco memperkenalkan metode yang lain pada tahun 1980-an di mana mengubah persegi dengan sudut lengkung (pada DFD Chris Gane dan Trish Sarson) dengan lingkaran untuk menotasikan. DFD Edward Yourdon dan Tom DeMarco populer digunakan sebagai model analisis system perangkat lunak untuk system perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur.

Informasi yang ada di dalam perangkat lunak dimodifikasi dengan beberapa



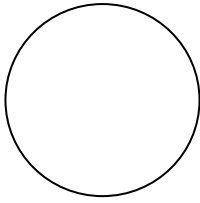
transformasi yang dibutuhkan. *Data Flow Diagram* (DFD) atau dalam bahasa Indonesia menjadi Diagram Alir Data (DAD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*).

DFD dapat digunakan untuk merepresentasikan sebuah system 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.

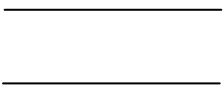
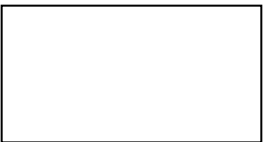

DFD tidak sesuai untuk memodelkan system perangkat lunak yang akan dibangun menggunakan pemrograman berorientasi objek. Paradigma pemrograman terstruktur dan pemrograman berorientasi objek merupakan hal yang berbeda. Jangan mencampuradukkan pemrograman terstruktur dan pemrograman berorientasi objek.

Rosa dan Shalahuddin (2014:71) menjelaskan, “notasi- notasi pada DFD (Edward Yourdon dan Tom DeMarco) adalah sebagai berikut:

Tabel 2.1 Notasi-notasi pada DFD (Edward Yourdon dan Tom DeMarco)

NO	Notasi	Keterangan
1.		Proses atau fungsi atau prosedur; pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya menjadi fungsi atau prosedur di dalam kode program.



2.		<p><i>File</i> atau basisdata atau penyimpanan (<i>storage</i>); pada pemodelan perangkat lunak yang akan diimplementasikan dengan pemrograman terstruktur, maka pemodelan notasi inilah yang harusnya dibuat menjadi tabel-tabel basis data yang dibutuhkan, tabel-tabel ini juga harus sesuai dengan perancangan tabel-tabel pada basis data (<i>Entity Relationship Diagram (ERD)</i>, <i>Conceptual Data Model (CDM)</i>, <i>Physical Data Model (PDM)</i>)</p>
3.		<p>Entitas luar (<i>external entity</i>) atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang yang emakai/berinteraksi dengan perangkat lunak yang dimodelkan atau system lain yang terkait dengan aliran data dari system yang dimodelkan.</p> <p>Catatan:</p> <p>Nama yang digunakan pada masukan (<i>input</i>) atau keluaran (<i>output</i>) biasanya berupa kata benda</p>
4.		<p>Aliran data; merupakan data yang dikirim antar proses, dari penyimpanan ke proses, atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>)</p> <p>Catatan:</p> <p>Nama yang digunakan pada aliran data biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data siswa”.</p>



Berikut ini adalah tahapan-tahapan perancangan dengan menggunakan DFD:

1. Membuat DFD Level 0 atau sering disebut juga *Context Diagram*

DFD Level 0 menggambarkan system yang akan dibuat sebagai suatu entitas tunggal yang berinteraksi dengan orang maupun system lain. DFD Level 0 digunakan untuk menggambarkan interaksi antara system yang akan dikembangkan dengan entitas luar.

2. Membuat DFD Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam system 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 system, 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 system yang dikembangkan juga menjadi rumit.



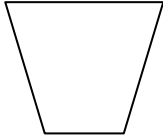
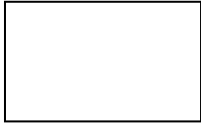
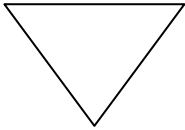
2.3.2. Pengertian Blockchart

Kristanto (2011:68) menjelaskan, “*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.”

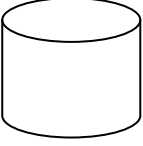

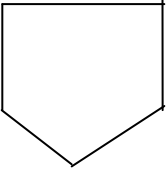
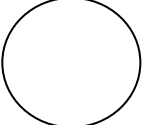
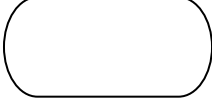
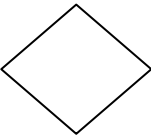


Kristanto (2011:68) menjelaskan, “Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

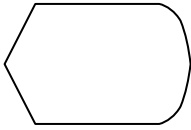

Tabel 2.2 Simbol-simbol dalam *Block Chart*

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



6.		Data penyimpanan (<i>Storage</i>)
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.		Terminal yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10.		Terminal 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.

2.3.3. Pengertian ERD (*Entity Relationship Diagram*)

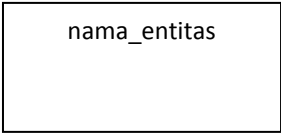
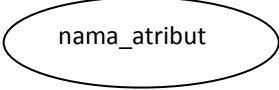
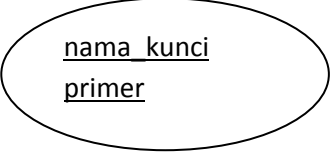
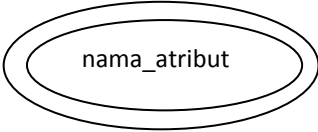
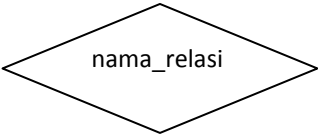
Rosa dan Shalahuddin (2014 : 50) menjelaskan, tentang pemodelan awal basis data yang paling banyak digunakan adalah:

menggunakan *Entity Relationship Diagram* (ERD). ERD dikembangkan berdasarkan teori himpunan dalam bidang matematika. ERD digunakan untuk pemodelan basis data relasional. Sehingga jika penyimpanan basis data menggunakan OODBMS maka perancangan basis data tidak perlu menggunakan ERD. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker, Ian Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lain. Namun yang banyak digunakan adalah notasi dari Chen.


Rosa dan Shalahuddin(2014 : 71) menjelaskan, “Berikut adalah symbol-simbol yang digunakan pada ERD dengan notasi Chen:



Tabel 2.3 Symbol-simbol ERD dengan notasi Chen

NO	Simbol	Deskripsi
1.	Entitas / <i>entity</i> 	Entitas merupakan data inti yang akan disimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi computer; 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; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama)
4.	Atribut multivalai / <i>multivalue</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu
5.	Relasi 	Relasi yang menghubungkan antar entitas; biasanya diawali dengan kata kerja



6.	Asosiasi / <i>association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas
----	--	--


ERD biasanya memiliki hubungan *binary* (satu relasi menghubungkan dua buah entitas). Beberapa metode perancangan ERD menoleransi hubungan relasi *ternary* (satu relasi menghubungkan tiga buah relasi) atau *N-ary* (satu relasi menghubungkan banyak entitas), tapi banyak metode perancangan ERD yang tidak mengizinkan hubungan *ternary* atau *N-ary*.

2.3.4. Pengertian *Flowchart*


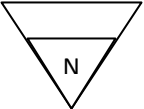
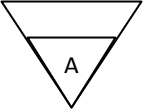
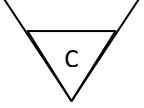



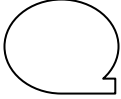
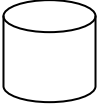
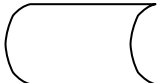
Supardi (2013:51) mengemukakan bahwa, bagan alir (*flowchart*) merupakan bagan (*chart*) yang menunjukkan alir (*flow*) didalam program atau prosedur sistem secara logika.

Ladjamudin (2013:211) mengemukakan bahwa, *flowchart* adalah bagan – bagan yang mempunyai arus yang menggambarkan langkah – langkah penyelesaian suatu masalah. *Flowchart* merupakan cara penyajian dari suatu algoritma.







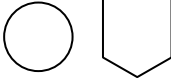
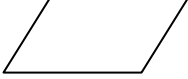
Tabel 2.4. Simbol-simbol *Flowchart*

No	Nama Simbol	Simbol	Fungsi
1.	Simbol dokumen		Menunjukkan dokumen input untuk proses manual, mekanik atau komputer.

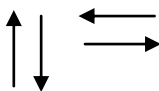
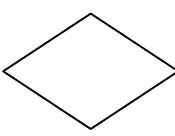


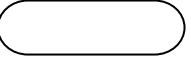


2.	Simbol kegiatan manual		Menunjukkan pekerjaan manual.
3.	Simbol simpanan offline		File non-komputer yang diarsip urut angka (<i>numerical</i>)/
			File non-komputer yang diarsip urut huruf (<i>alphabetical</i>).
			File non-komputer yang diarsip urut tanggal (<i>cronological</i>).
4.	Simbol kartu plong		Menunjukkan <i>input/output</i> yang menggunakan kartu plong (<i>punched card</i>).
5.	Simbol proses		Menunjukkan kegiatan proses dari operasi program komputer.
6.	Simbol operasi luar		Menunjukkan operasi yang dilakukan di luar proses operasi komputer.
8.	Simbol pita magnetic		Menunjukkan <i>input/output</i> menggunakan pita magnetik.
9.	Simbol <i>hard disk</i>		Menunjukkan <i>input/output</i> menggunakan <i>hard disk</i> .
10.	Simbol <i>diskette</i>		Menunjukkan <i>input/output</i> menggunakan <i>diskette</i> .



11.	Simbol drum magnetic		Menunjukkan <i>input/output</i> menggunakan drum magnetik.
12.	Simbol pita kertas berlubang		Menunjukkan <i>input/output</i> menggunakan pita kertas berlubang.
13.	Simbol <i>keyboard</i>		Menunjukkan <i>input</i> menggunakan <i>on-line keyboard</i> .
14.	Simbol <i>display</i>		Menunjukkan <i>output</i> yang ditampilkan di monitor.
15.	Simbol pita kontrol		Menunjukkan penggunaan pita kontrol (<i>control tape</i>) dalam <i>batch control total</i> untuk pencocokan di proses <i>batch processing</i> .
16.	Simbol hubungan komunikasi		Menunjukkan proses transmisi data melalui channel komunikasi.
18.	Simbol penghubung		Menunjukkan penghubung ke halaman yang masih sama atau ke halaman lain.
19.	Simbol <i>input/output</i>		Simbol <i>input/output</i> (<i>input/output symbol</i>) digunakan untuk mewakili data <i>input/output</i> .



20.	Simbol garis alir		Simbol garis alir (<i>flow lines symbol</i>) digunakan untuk menunjukkan arus dari proses.
21.	Simbol keputusan		Simbol keputusan (<i>decision symbol</i>) digunakan untuk suatu penyeleksian kondisi didalam program.
22.	Simbol proses terdefinisi		Simbol proses terdefinisi (<i>predifined prosesmsymbol</i>) digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan di tempat lain.
23.	Simbol persiapan		Simbol persiapan (<i>preparation symbol</i>) digunakan untuk memberi nilai awal suatu besaran.
24.	Simbol titik terminal		Simbol titik terminal (<i>terminal point symbol</i>) digunakan untuk awal dan akhir dari suatu proses.

2.3.5. Pengertian Kamus Data

Rosa 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 system 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
- d. Informasi tambahan – seperti tipe data, nilai data, batas nilai data, dan komponen yang membentuk data

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

Tabel 2.5 Simbol-simbol dalam 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

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, perancangan DFD-nya, atau keduanya.

2.4. Teori Program

2.4.1. Sekilas Tentang Basis Data (*Database*)

Rosa dan Shalahuddin (2014:43) menjelaskan, “System basis data adalah system terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat.

Yakub (2012:51-52), Basis data (database) merupakan kumpulan data yang saling berhubungan (punya relasi).

Rusdiana dan irfan (2014:302) menjelaskan, Database adalah kumpulan informasi yang disimpan dalam komputer secara sistematis untuk memperoleh informasi dari basis data.

Berikut ini Karakteristik database dalam *Database Management System* (DBMS) Rusdiana dan irfan (2014:303) memiliki tiga karakteristik utama, yaitu :

- a. Data yang sama dapat diakses secara serempak oleh berapa pengguna untuk berbagai kegunaan yang berbeda.
- b. Data tidak bergantung pada struktur penyimpanan atau cara membaca data dari program aplikasi, atau data bersifat transparan terhadap program aplikasi.
- c. Dapat Memiliki integritas (akurasi dan validasi) yang terkendali.

2.4.2. Pengertian *MySQL*

Raharjo (2015:16), “MySQL merupakan *software* RDBMS (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 berbarengan (*multi-threaded*)”.



Winarno dkk (2014:102) juga berpendapat bahwa MySQL adalah software database. MySQL merupakan tipe data relasional yang artinya MySQL menyimpan datanya dalam bentuk table -tabel yang saling berhubungan.

Hendry (2015:7), MySQL merupakan turunan salah satu konsep utama dalam basis data yang telah ada sebelumnya, yaitu SQL (*Structured Query Language*). SQL adalah sebuah konsep pengoperasian basis data, terutama untuk pemilihan atau seleksi dan pemasukkan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Berdasarkan teori para ahli di atas dapat disimpulkan bahwa, MySQL adalah sebuah software database yang diolah dengan sangat cepat dan dapat menampung data dalam jumlah yang sangat besar serta diakses oleh user.

Berikut tipe-tipe data pada *MySQL*:

Tabel 2.6. Tipe Data pada *MySQL*

NO	Tipe Data	Ukuran (Bytes)
1.	TINYINT	1
2.	SMALLINT	2
3.	MEDIUM	3
4.	INT	4
5.	BIGINT	8
6.	FLOAT(p)	4
7.	FLOAT	4
8.	DOUBLE	8
9.	REAL	8



10.	DECIMAL (L, D)	$L + 1$
11.	NUMERIC (L, D)	$L + 1$
12.	BIT (L)	8
13.	DATE	3
14.	TIME	3
15.	DATETIME	8
16.	TIMESTAMP	4
17.	YEAR	1
18.	CHAR (L)	L
19.	BINARY(L)	$L + 1$
20.	VARCHAR (L)	$L + 1$
21.	VARBINARY (L)	$L + 1$
22.	TINYBLOB	$L + 1$
23.	TINYTEXT	$L + 1$
24.	BLOB	$L + 2$
25.	TEXT	$L + 2$
26.	MEDIUMBLOB	$L + 3$
27.	MEDIUMTEXT	$L + 3$
28.	LOB	$L + 4$
29.	LONGTEXT	$L + 4$



30.	ENUM ('nilai1',...)	1 – 2
31.	SET ('nilai1',...)	1 – 8

Keistimewaan MySQL

Hendry (2015:7—9), MySQL memiliki beberapa keistimewaan, antara lain:

1. Portabilitas
MySQL dapat berjalan stabil pada berbagai sistem operasi, seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.
2. Perangkat lunak sumber terbuka
MySQL didistribusikan sebagai perangkat lunak sumber terbuka, di bawah lisensi GPL sehingga dapat digunakan secara gratis.
3. *Multi-user*
MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. *Performance tuning*
MySQL memiliki kecepatan yang menakjubkan dalam menangani query sederhana. Dengan kata lain, dapat memproses lebih banyak SQL per satuan waktu.
5. Ragam tipe data
MySQL memiliki ragam tipe data yang sangat kaya, seperti signed/unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
6. Perintah dan Fungsi
MySQL server memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).
7. Keamanan
MySQL memiliki beberapa lapisan keamanan seperti level subnetmask, nama host, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.



8. Skalabilitas dan Pembatasan

MySQL mampu menangani basis data dalam skala cukup besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu table, serta 5 miliar baris. Selain itu, batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

9. Konektivitas

MySQL dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, Unix socket (Unix), atau Named Pipes (NT).

10. Lokalisasi

MySQL dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk di dalamnya.

11. Antarmuka

MySQL memiliki antarmuka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

12. Klien dan Peralatan

MySQL dilengkapi dengan berbagai peralatan (*tool*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada, disertakan petunjuk online.

13. Struktur Tabel

MySQL memiliki struktur tabel lebih fleksibel dalam menangani ALTER TABLE, dibandingkan basis data lainnya seacam PostgreSQL ataupun Oracle.

2.4.3. Sekilas Tentang PHP

Prasetio (2015:130) mengemukakan bahwa, PHP (PHP:Hypertext Preprocessor) adalah bahasa script yang ditanam di sisi sever. Proses PHP dijalankan di server sat sebuah halaman dibuka dan mengandung kode PHP, prosesor itu akan menerjemahkan dan mengeksekusikan semua perintah dalam halaman tersebut, dan kemudian menampilkan hasilnya ke browser sebagai halaman html biasa.



Winarno dkk (2014:49) PHP atau *PHP Hypertext Preprocessor*, adalah sebuah bahasa pemrograman web berbasis server (server-side) yang mampu memarsing kode PHP dari kode web dengan ekstensi .php, sehingga menghasilkan tampilan website yang dinamis di sisi client (browser).

Sujatmiko (2012:213), PHP adalah Bahasa pemrograman yang bisa kita gunakan untuk membuat aplikasi web. PHP banyak digunakan karena bisa kita ubah dan kita gunakan secara bebas.

Sidik (2017:4), PHP merupakan secara umum dikenal sebagai bahasa pemrograman script – script yang membuat dokumen HTML secara *on the fly* yang dieksekusi di server web, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML. dikenal juga sebagai bahasa pemrograman server side.

2.4.4. Sekilas tentang Dreamweaver

2.4.4.1. Pengertian Dreamweaver

Sadeli (2014:12), Dreamweaver merupakan suatu perangkat lunak web editor keluaran *adobe system* yang digunakan untuk mengembangkan dan mendesain suatu website dengan fitur-fitur yang menarik dan kemudahan dalam penggunaannya.

Madcoms (2012:51) menjelaskan, Dreamweaver adalah *HTML* editor profesional untuk mendesain web secara visual dan mengelola situs atau halaman web.

2.4.4.2 Tampilan Ruang Kerja Adobe Dreamweaver CS 3

Agung (2013:3-4) menjelaskan Ruang kerja Dreamweaver CS3 memiliki komponen-komponen yang memberikan fasilitas dan ruang untuk menuangkan kreasi pada saat bekerja yaitu sebagai berikut :

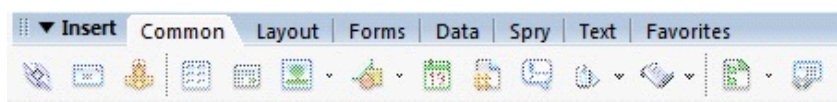


1. **WelcomeScreen**, adalah layar pada *workspace* yang digunakan untuk memudahkan pengguna membuat *file* yang pernah digunakan.
2. **Menu**, adalah media yang digunakan untuk melakukan atau menjalankan perintah pada *Dreamweaver*. Menu berisi perintah untuk pembuatan, pengaturan, manipulasi dan konfigurasi pada *Dreamweaver*.

File Edit View Insert Modify Text Commands Site Window Help

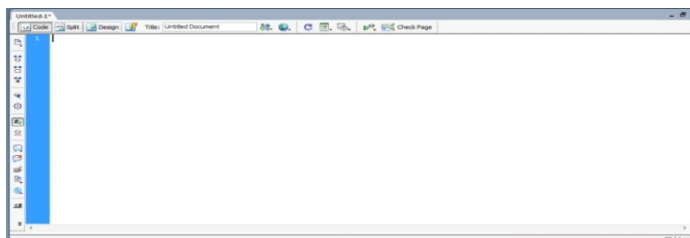
Gambar 2.2. Tampilan Menu *Adobe Dreamweaver CS3*

3. **InsertBar**, adalah suatu bentuk *toolbar* yang berisi perintah untuk menambahkan suatu elemen pada dokumen.



Gambar 2.3. Tampilan Insert Bar *Adobe Dreamweaver CS3*

4. **DocumentWindow**, adalah editor *text* atau visual pada ruang kerja. *DocumentWindow* menyajikan bentuk dokumen secara kode maupun visual.



Gambar 2.4. Tampilan *DocumentWindow* *Adobe Dreamweaver CS3*

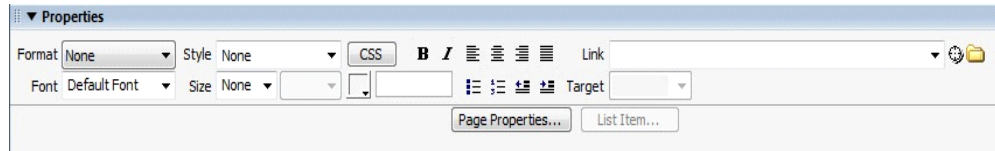
5. **PanelGroup**, adalah bagian ruang kerja *Dreamweaver* yang berisikan kumpulan *panel*. *PanelGroup* terdiri dari kumpulan beberapa *panelcontrol*



Gambar 2.5. Tampilan *Panel Group* *Adobe Dreamweaver CS3*



6. **PropertyInspector**, digunakan untuk menampilkan informasi *property* elemen *HTML* tertentu yang sedang dipilih.



Gambar 2.6. Tampilan *PropertyInspector* Adobe Dreamweaver CS3

2.4.5. Pengertian XAMPP

Sadeli (2014:4) mengemukakan bahwa, Xampp adalah program yang berisi paket apache, MySQL, dan phpMyAdmin.



Gambar 2.7 Halaman download xampp

Adapun langkah-langkah untuk menginstall program xampp pada komputer sebagai berikut .

1. Klik 2x (Double Klik) file instalasinya, selanjutnya anda akan diminta untuk menentukan bahasa yang digunakan saat instalasi.



Gambar 2.8 Menjalankan file instalasi



2.4.6. Sekilas Tentang CSS

Madcoms (2012:51) menjelaskan bahwa, *Cascading Style Sheet (CSS)* adalah kumpulan kode-kode yang digunakan untuk mengendalikan tampilan isi suatu halaman web.

Suryana dan Koesheryatin (2014:101) menjelaskan bahwa, CSS (*Cascading Style Sheet*) adalah suatu bahasa stylesheet yang digunakan untuk tampilan suatu website baik tata letaknya, jenis huruf, warna dan semua yang berhubungan dengan tampilan.