BAB II
TINJAUAN PUSTAKA

2.1 Teori Umum

Adapun dalam penyusunan laporan kuliah praktek ini penulis akan menuliskan landasan teori, pengertian-pengertian yang terdapat pada penulisan laporan ini adalah sebagai berikut:

2.1.1. Pengertian Komputer

Hartono (2013:27), "komputer adalah sebuah mesin yang dapat dikendalikan melalui perintah (*programmable machine*) yang dirancang secara otomatis melakukan serangkaian urutan perhitungan (*arithmetic*) atau proses-proses yang diurutkan secara logis".

2.1.2. Pengertian Sistem

Hartono (2013:10), "sistem yakni suatu benda atau entitas (yaitu himpunan dari berbagai bagian atau komponen), dan sekaligus juga suatu proses atau metode atau cara untuk mencapai tujuan (yaitu saling berhubungan secara terorganisasi berdasar fungsi-fungsinya)".

2.1.3. Pengertian Informasi

Hartono (2013:9), "informasi adalah sehimpunan data yang telah diolah menjadi sesuatu yang memiliki arti dan kegunaan lebih luas".

2.1.4. Pengertian Sistem Informasi

Hartono (2013:20), "sistem informasi adalah rangkaian sistem yang terorganisasi dari sejumlah bagian/komponen yang secara bersama-sama berfungsi atau bergerak menghasilkan informasi".

2.1.5. Metodologi Pengembangan Sistem

Al-fatta (2007:25), "beberapa ahli membagi proses-proses pengembangan sistem ke sejumlah urutan yang berbeda-beda. Tetapi semuanya akan mengacu pada proses-proses standar berikut:

a. Analisis
b. Desain
c. Implementasi
d. Pemeliharaan

Pada perkembangannya, proses-proses standar tadi dituangkan dalam satu metode yang dikenal dengan nama *Systems Development Life Cycle* (SDLC) yang merupakan metodologi umum dalam pengembangan sistem yang memadai kemajuan usaha analisis dan desain".

2.1.6. Pengertian Data

Kristanto (2008:8), "data merupakan bentuk yang belum dapat memberikan manfaat yang besar bagi penerimanya, sehingga perlu suatu model yang nantinya akan dikelompokkan dan diproses untuk menghasilkan informasi".

2.2. Teori Khusus

Teori ini akan membahas mengenai desain sistem yang merupakan suatu fase dimana diperlukan suatu keahlian perencanaan untuk elemen-elemen komputer yang akan menggunakan sistem baru. Ada beberapa alat bantu yang digunakan dalam desain sistem yaitu:

2.2.1. Pengertian Basis Data (*Database)*

Sutarman (2012:15), "*database* adalah sekumpulan file yang saling berhubungan dan terorganisasi atau kumpulan *record-record* yang menyimpan data dan hubungan di antaranya",

2.2.2. Pengertian Diagram Aliran Data/*Data Flow Diagram* (DFD)

Kristanto (2008:61), "DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan darimana 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.

Al-Bahra (2013:64), Diagram Aliran Data merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil. Salah satu keuntungan menggunakan diagram aliran data adalah memudahkan pemakai atau user yang kurang menguasai bidang computer untuk mengerti sistem yang akan dikerjakan.

2.2.2.1. Diagram Konteks

Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh *input* ke sistem atau *output* dari sistem.

2.2.2.2. Diagram Nol/ *Zero*

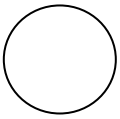
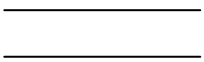
Diagram nol adalah diagram yang menggambarkan proses dari dataflow diagram. Diagram nol memberikan pandangan secara menyeluruh mengenai sistem yang ditangani, menunjukkan tentang fungsi-fungsi utama atau proses yang ada, aliran data dan eksternal entity.

2.2.2.3. Diagram Rinci (Level Diagram)

Diagram rinci adalah diagram yang menguraikan proses apa yang ada dalam diagram *zero* atau diagram level diatasnya.

Kristanto (2008:62), simbol DFD yang dipakai untuk menggambarkan data beserta proses transformasi data, antara lain:

Tabel 2.1 Simbol-simbol *data flow diagram*

No.	Nama	Simbol De Marco and Jourdan	Arti
1.	Entiti luar		Simbol ini digunakan untuk menggambarkan sumber atau tujuan dari aliran data dari atau ke sistem.
2.	Aliran data		Simbol ini digunakan untuk menggambarkan aliran data dari satu proses ke proses lainnya.
3.	Proses		Simbol ini digunakan untuk mentransformasikan data secara umum.
4.	Berkas/ tempat penyimpanan		Simbol ini digunakan untuk menyimpan data tau file.

Sumber: Perancangan Sistem Informasi dan Aplikasinya

Dalam penggambaran simbol DFD, ada beberapa peraturan yang harus diperhatikan sehingga dalam penggambarannya tidak terjadi kesalahan.

1. Antar entity luar tidak diijinkan terjadi hubungan relasi.
2. Tidak boleh ada aliran data antara entiti luar dengan *data store*.
3. Untuk alasan kerapian, entiti luas atau *data store* boleh digambar beberapa kali dengan tanda khusus, misalnya diberi nomor.
4. Satu aliran data boleh mengalirkan beberapa struktur data.
5. Bentuk anak panah aliran data boleh bervariasi.
6. Semua *obyek* harus mempunyai nama.
7. Aliran data selalu diawali dan diakhiri dengan proses.
8. Semua aliran data harus mempunyai tanda arah.

2.2.3. Pengertian *Flowchart*

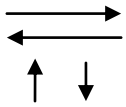
Al-Bahra (2013:263), *Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah.

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

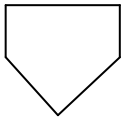
1. *Flow Direction Symbols (*Simbol penghubung/alur)

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

Tabel 2.2 *Flow Direction Symbols*

No.	Nama simbol	Simbol	Arti
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.

Lanjutan **Tabel 2.2** *Flow Direction Symbols*

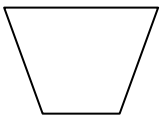
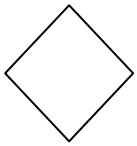
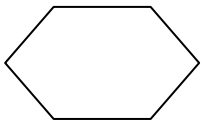
4.	Simbol *offline connector*		Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/ lembar yang berbeda

Sumber: Analisis dan Desain Sistem Informasi

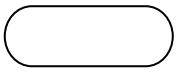
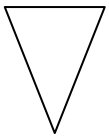
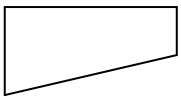
2. *Processing Symbols (*Simbol proses)

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

Tabel 2.3 *Processing Symbols*

No	Nama simbol	Simbol	Arti
1.	Simbol *offline connector*		Untuk menyatakan sambungan dari satu proses ke proses lainnya dalam halaman/ lembar yang berbeda.
2.	Simbol manual		Untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer (manual).
3.	Simbol *decisiom/* logika		Untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, ya/tidak.
4.	Simbol *predefined* proses		Untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk member harga awal.

Lanjutan **Tabel 2.3** *Processing Symbols*

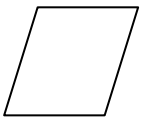
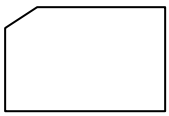
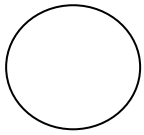
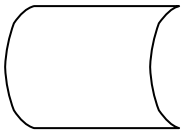
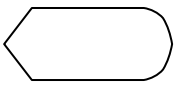
5.	Simbol terminal		Untuk menyatakan permulaan atau akhir suatu program.
6.	Simbol *keying operation*		Untuk menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai *keyboard.*
7.	Simbol *off-line storage*		Untuk menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
8.	Simbol manual *input*		Untuk memasukkan data secara manual dengan menggunakan *online keyboard.*

Sumber: Analisis dan Desain Sistem Informasi

3. *Input-output symbols (*Simbol *input-output*)

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

Tabel 2.4 *Input-output symbols*

No	Nama symbol	Simbol	Arti
1.	Simbol *input-output*		Untuk menyatakan proses *input* dan *output* tanpa tergantung dengan jenis peralatannya.
2.	Simbol *punched card*		Untuk menyatakan *input* berasal dari kartu atau *output* ditulis ke kartu.
3.	Simbol *magnetic-tape unit*		Untuk menyatakan *input* berasal dari pita *magnetic* atau *output* disimpan ke pita *magnetic.*
4.	Simbol *disk storage*		Untuk menyatakan *input* berasal dari *disk* atau *output* disimpan ke *disk.*
5.	Simbol *document*		Untuk mencetak laporan ke printer.
6.	Simbol *display*		Untuk menyatakan peralatan *output* yang digunakan berupa layar (video, komputer).

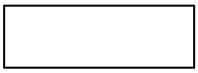
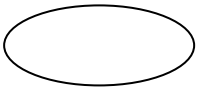
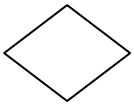
Sumber: Analisis dan Desain Sistem Informasi

2.2.4. Pengertian *Entity Relationship Diagram*/ Diagram Hubungan Antara Entitas (ERD)

Al-Bahra (2013:142), ERD adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak.

Adapun simbol-simbol dari *Entity Relationship Diagram* adalah sebagai berikut:

Tabel 2.5 Simbol-simbol *Entity Relationship Diagram* (ERD)

No.	Nama simbol	Simbol	Arti
1.	*Entitas*		Menyatakan *himpunan entitas/ entitas.*
2.	*Atribut*		Menyatakan *atribut (atribut* yang berfungsi sebagai *key* digarisbawahi).
3.	*Himpunan relasi*		Menyatakan *himpunan relasi/ relasi.*
4.	*Link*		Penghubung antara *himpunan relasi* dengan *himpunann entitas* dan *himpunan entitas* dengan *atributnya.*

Sumber: Analisis dan Desain Sistem Informasi

Entity Relationship Diagram memiliki elemen-elemen diagram hubungan entitas yaitu:

1. *Entity*

Entity digambarkan dengan sebuah bentuk persegi panjang. *Entity* adalah sesuatu apa saja yang ada di dalam sistem, nyata abstrak dimana data tersimpan atau dimana terdapat data.

2. *Relationship*

Relationship dapat digambarkan dengan sebuah bentuk belah ketupat. *Relationship* adalah hubungan alamiah yang terjadi antara entitas. Pada umumnya penghubung (*Relationship*) diberi nama dengan kata kerja dasar, sehingga memudahkan untuk melakukan pembacaan relasinya.

3. *Relationship Degree*

Relationship Degree adalah jumlah entitas yang berpartisipasi dalam suatu *relationship*. Derajat *relationship* yang sering dipakai di dalam *ERD* yaitu:

a. Unary Relationship

b. Binary Relationship

c. Tertary Relationship

4. *Atribut*

Secara umum atribut adalah sifat atau karakteristik dari tiap entitas manapun tiap *relationship*. Maksudnya, atribut adalah sesuatu yang menjelaskan apa yang sebenarnya yang dimaksud dengan entitas maupun *relationship*, sehingga sering dikatakan atribut adalah element adalah element dari setiap entitas dan *relationship*.

5. *Kardinalitas (Cardinality)*

Kardinalitas relasi menunjukkan jumlah maksimum tupel yang dapat belasi dengan entitas yang lain. *Kardinalitas* relasi menunjukan kepada hubungan maksimum yang terjadi dari entitas yang satu ke entitas yang lain dan begitu juga sebaliknya. Terdapat tiga macam kardinalitas relasi, yaitu:

a. *One To One*

Tingkat hubungan satu ke satu, dinyatakan dengan satu kejadian pada entitas pertama, hanya mempunyai satu hubungan denga satu kejadian pada entitas yang kedua dan sebaliknya. Yang berarti setiap tupel pada entitas A berhubungan dengan paling banyak satu tupel pada entitas B dan begitu juga sebaliknya setiap tupel pada entitas B berhubungan dengan paling banyak satu tupel pada entitas A.

b. *One To Many atau Many to One*

Tingkat hubungan satu ke banyak adalah sama dengan banyak ke satu. Tergantung dari arah mana hubungan tersebut dilihat. Untuk satu kejadian untuk entitas yang pertama dapat mempunyai banyak hubungan dengan kejadian pada entitas yang kedua. Sebaliknya satu kejadian pada entitas yang kedua hanya dapat mempunyai satu hubungan dengan satu kejadian pada entitas yang pertama.

c. *Many To Many*

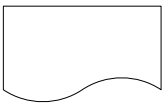
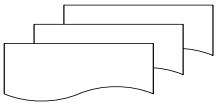
Tingkat hubungan banyak ke banyak terjadi jika tiap kejadian pada sebuah entitas akan mempunyai banyak hubungan dengan kejadian pada entitas yang lainnya. Baik dilihat dari sisi entitas yang pertama, maupun yang dilihat dari sisi kedua. Yang berarti setiap tupel pada entitas A dapat berhubungan dengan banyak tupel pada entitas B, dan demikian juga sebaliknya, dimana setiap tupel pada entitas B dapat berhubungan dengan banyak tupel pada entitas B.

2.2.5. Pengertian *Blockchart*

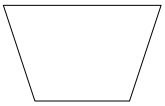
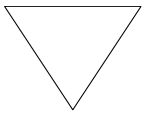
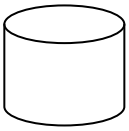
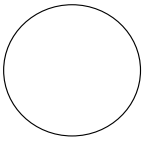
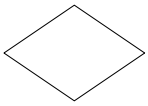
Kristanto (2008:75) *blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi.

Adapun simbol-simbol digunakan dalam *blockchart* adalah sebagai berikut:

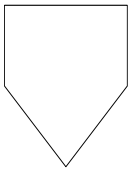
Tabel 2.6 Simbol-simbol *blockchart*

No.	Nama Simbol	Simbol	Arti
1.	*Document*		Simbol yang digunakan dalam menandakan suatu dokumen, bisa dalam bentuk surat, formulir, berkas atau cetakan.
2.	*Multi Document*		Suatu simbol yang digunakan dalam menandakan suatu multi dokumen, bisa dalam bentuk surat, formulir, berkas atau cetakan.

Lanjutan **Tabel 2.6** Simbol-simbol *blockchart*

3.	*Manual Operation*		Suatu simbol yang digunakan dalam suatu proses yang dilakukan secara manual.
4.	*Process*		Suatu proses yang dilakukan dengan menggunakan komputer.
5.	*Merge*		Simbol yang digunakan dalam dokumen yang menandakan dokumen diarsipkan (arsip manual).
6.	*Magnetik Disk*		Suatu simbol yang digunakan dalam data penyimpanan *(data storage).*
7.	*Tampilan*		Layar peraga (monitor)
8.	*Connector*		Suatu simbol yang digunakan dalam terminasi yang mewakili simbol-simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
9.	*Alternate Process*		Simbol yang digunakan dalam terminasi yang menandakan awal; atau akhir dari suatu aliran.
10.	*Decision*		Suatu simbol yang digunakan dalam pengambilan keputusan akhir.

Lanjutan **Tabel 2.6** Simbol-simbol *blockchart*

11.	*Off-Page Connector*		Suatu simbol yang digunakan dalam terminasi yang mewakili simbol-simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.

Sumber: Perancangan Sistem Informasi dan Aplikasinya

2.2.6. Kamus Data (*Data Dictionary*)

Kristanto (2008:72), Kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap field atau file di dalam sistem. Simbol-simbol yang ada dalam kamus data adalah sebagai berikut:

Tabel 2.7 Simbol-simbol dalam Kamus Data

No.	Simbol	Arti
1.	=	Terdiri atas
2.	+	Dan
3.	( )	Opsional
4.	[ ]	Memilih salah satu alternatif
5.	**	Komentar
6.	@	Identifikasi atribut kunci
7.	|	Pemisah alternatif simbol [ ]

Sumber: Perancangan Sistem Informasi dan Aplikasinya

2.3. Teori Judul

Teori ini akan membahas pengertian-pengertian yang terdapat pada judul laporan kuliah praktek ini yaitu, sebagai berikut:

2.3.1. Pengertian Aplikasi

Pipin (2013:6), "aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Ms-Word, Ms-Excel"*.

2.3.2. Pengertian Pendistribusian

Kamus Besar Bahasa Indonesia (2009:194), "pendistribusian adalah pengiriman atau penyaluran ke suatu atauke beberapa tempat".

2.3.3. Pengertian Penjualan

Kamus Besar Bahasa Indonesia (2009:336), "penjualan adalah proses, perbuatan, cara menjual barang kebutuhan pokok lancer dan mantap".

2.3.4. Pengertian Barang

Kamus Besar Bahasa Indonesia (2009:113), "benda adalah suatu yang berwujud benda cair, benda keras dan sebagainya".

2.3.5. Pengertian Elektronik

Kamus Besar Bahasa Indonesia (2009:216), "elektronik adalah pengertian mempraktekkan tenaga listrik pada berbagai alat".

2.3.6. Aplikasi Pendistribusian dan Penjualan Barang Elektronik pada PT Sharp Elektronik Indonesia Cabang Palembang

Pengertian Aplikasi Pendistribusian dan Penjualan Barang Elektronik pada PT Sharp Elektronik Indonesia Cabang Palembang adalah suatu program aplikasi yang digunakan untuk membantu pengoperasian proses pendistribusian dan penjualan barang elektronik pada PT Sharp Elektronik Indonesia Cabang Palembang

2.4. Teori Program

2.4.1. Mengenal *PHP (Hypertext Preprocessor)*

Madcoms (2011:216), PHP adalah salah satu bahasa pemrograman yang berjalan dalam sebuah webserver dan berfungsi sebagai pengolah data pada sebuah server.

Kode-kode bahasa PHP dalam penulisannya menyatu dengan tag-tag HTML dalam satu file. Kode PHP diletakkan antara tanda <? atau <?php dan diakhiri dengan tanda ?>, sebagai identitas bahasa pemrograman PHP.

2.4.2. Pengertian *Adobe Dreamweaver CS5*

Madcoms (2011:2), *Dreamweaver* merupakan software utama yang digunakan adalah sebuah *HTML* editor profesional untuk mendesain web secara visual dan mengelola situs atau halaman web. Saat ini terdapat *software* dari kelompok *Adobe* yang belakangan banyak digunakan untuk mendesain suatu situs web.

Salah satu versi dari *Adobe Dreamweaver* adalah *Adobe Dreamweaver CS5*. Pada *Adobe Dreamweaver CS5*, terdapat beberapa kemampuan bukan hanya sebagai *software* untuk desain web saja tetapi juga untuk menyunting kode serta pembuatan aplikasi web dengan menggunakan berbagai bahasa pemrograman web, antara lain : *JPS, PHP, ASP* dan *Coldfusion.*

Dreamweaver merupakan *software* utama yang digunakan oleh *Web Desainer* maupun *Web Programmer* dalam mengembangkan suatu situs web. Hal ini disebabkan ruang kerja, fasilitas dan kemampuan *Dreamweaver* yang mampu meningkatkan produktifitas dan efektivitas dalam desain maupun membangun suatu situs web.

2.4.2.1. Mengenal Elemen Kerja *Dreamweaver CS5*

Madcoms (2011:3-11), elemen-elemen kerja pada *Dreamweaver CS5* adalah sebagai berikut:

a. *Application Bar*

Berada dibagian paling atas jendela aplikasi *Dreamweaver CS5*. Baris ini berisi tombol *workspace* (workspace switcher), menu dan aplikasi lainnya.

b. *Toolbar Document*

Berisi tombol-tombol yang digunakan untuk mengubah tampilan jendela dokumen, sebagai contoh tampilan Design atau tampilan Code. Selain itu juga dapat digunakan untuk operasi-operasi umum, misalnya untuk melihat hasil sementara halaman web pada jendela *browser*.

c. Jendela Dokumen

Lembar kerja tempat membuat dan mengedit desain halaman web.

d. *Workspace Switcher*

Digunakan untuk mengubah tampilan ruang kerja *(workspace) Dreamweaver CS6*. Sebagai contoh mengubah tampilan menjadi tampilan *classic*, yaitu tampilan ruang kerja *Dreamwaver* versi sebelumnya.

e. *Panel Groups*

Kumpulan panel yang saling berkaitan, panel-panel ini dikelompokkan pada judul-judul tertentu bedasarkan fungsinya. Panel ini digunakan untuk memonitor dan memodifikasi pekerjaan. Secara default, panel *groups* berisi panel *Insert, CSS Styles, Asset, AP Element* dan *Files*.

f. *Tag Selector*

Terletak dibagian bawah jendela dokumen, satu baris dengan status bar. Bagian ini menampilkan hirarki pekerjaan yang sedang tetrpilih pada jendela dokumen, dapat juga digunakan untuk memilih objek pada jendela desain bedasarkan jenis atau kategori objek tersebut. *Tag selector* juga menampilakan informasi format dari bagian yang sedang aktif pada lembar kerja Design.

g. *Panel Property*

Digunakan untuk melihat dan mengubah berbagai properti objek atau teks pada jendela design. Properti untuk satu objek dengan objek lainnya selalu berbeda-beda. Jendela ini tidak dapat diuraikan pada tapilan jendela kode.

h. *Toolbar Coding*

Berisi tombol-tombol yang digunakan untuk melakukan operasi code-code standar. Jendela ini hanya tampil pada jendela *Code*.

i. *Panel Insert*

Berisi tombol-tombol untuk menyisipkan berbagai jenis objek, seperti image, tabel, atau objek media ke dalam jendela dokumen.

j. Panel Groups

Panel Groups adalah kumpulan panel yang saling berkaitan satu sama lainnya yang dikelompokkan dibawah satu judul, seperti: *Insert*, *Files*, *CSS Style* dan sebagainya.

k. *Panel Files*

Digunakan untuk mengatur file-file dan folder-folder yang membentuk situs web yang dibuat. Sebagai contoh mengcopy, memindahkan, atau mengganti nama file.

2.4.3. Mengenal *MySQL*

2.4.3.1. Pengertian *MySQL*

Kadir (2008:22), *SQL* adalah kependekan dari *Structured Query Language*. Bahasa ini merupakan standar yang digunakan untuk mengakses database relasional.

Kadir (2008:2-3), *MySQL* merupakan software yang tergolong sebagai *DBMS (Database Management System)* yang bersifat *Open Source*. *Open Source* menyatakan bahwa *software* ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat *MySQL*), selain tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa diperoleh dengan cara men-*download* (mengunduh) di internet secara gratis.

Sebagai software *DBMS, MySQL* memiliki sejumlah fitur seperti yang dijelaskan dibawah ini:

1. *Multiplatform*

   MySQL tersedia pada beberapa platform (*Windows*, *Linux*, *Unix*, dan lain-lain)

2. Andal, cepat, dan mudah digunakan

MySQL tergolong sebagai *database sever* (server yang melayani permintaan terhadap database) yang andal, dapat menangani *database* yang besar dengan kecepatan tinggi, mendukung banyak sekali fungsi untuk mengakses *database*, dan sekaligus mudah untuk digunakan. Berbagai pendukung tool juga tersedia. Perlu diketahui, *MySQL* dapat menangani sebuah tabel yang berukuran dalam terabyte (1 *terabyte* = 1024 *gigabyte*). Namun, ukuran yang sesungguhnya dapat bergantung pada batasan sistem operasi. Sebagai contoh pada sistem *Solaris* 9/10, batasan ukuran file sebesar 16 *terabyte*.

3. Jaminan keamanan akses

   MySQL mendukung pengamanan database dengan berbagai kriteria pengaksesan. Sebagai gambaran, dimungkinkan untuk mengatur *user* tertentu agar bisa mengakses data yang bersifat rahasia, sedangkan *user* lain tidak boleh. *MySQL* juga mendukung koneksivitas ke berbagai *software*. Sebagai contoh, *MySQL* bisa diakses melalui aplikasi berbasis *web*; misalnya dengan menggunakan *PHP*.

4. Dukungan *SQL*

   Seperti tersirat dalam namanya, *MySQL* mendukung perintah *SQL (Structured Query Language)*. Sebagaimana diketahui, *SQL* merupakan standar dalam pengaksesan *database* relasional. Pengetahuan akan *SQL* akan memudahkan siapapun untuk menggunakan *MySQL*.

Secara umum, *Database* merupakan koleksi data yang saling terkait, sehingga basis data dapat dianggap sebagai suatu penyusunan data yang terstruktur yang disimpan dalam media pengingat (*hard disk*) yang tujuannya adalah agar data tersebut dapat diakses dengan mudah dan cepat.

Sesungguhnya ada beberapa macam *database*, antara lain yaitu *database* hierarkis, *databse* jaringan, dan *databse* relasional. *Database* relasional pada saat ini telah diterapkan pada berbagai *platform*, dari *PC* hingga minikomputer.

2.4.3.2. Istilah Kunci Primer dan Kunci Asing

Kadir (2008:6-7), menurut konsep *database relasioal*, setiap tabel memiliki sebuah kunci primer (*primary key*), walaupun dalam praktiknya bisa saja

tidak memilikinya. Kunci primer adalah suatu nilai yang bersifat unik (tidak ada yang kembar) sehingga dapat digunakan untuk membedakan suatu baris dengan baris yang lain dalam sebuah tabel.

Pada *database* relasional, dikenal pula dengan istilah kunci asing (*foreign key*) atau sering juga disebut kunci tamu. Kunci tamu adalah sembarang kolom yang menunjuk ke kunci primer milik tabel lain.

2.4.3.3. Hubungan

Kadir (2008:7), pada model *database* relasional, kaitan atau asosiasi antara dua buah tabel disebut hubungan (*relationship*). Hubungan dapat berupa:

a. 1 - 1, yakni satu data pada suatu tabel berpasangan dengan hanya satu data pada tabel lain.
b. 1 - M, yakni satu data pada suatu tabel berpasangan dengan banyak data pada tabel lain.

2.4.3.4. Berbagai Tipe Data Pada *MySQL*

Kadir (2008:48-53), setiap kolom dalam tabel didefinisikan dengan melibatkan tipe kolom. Tipe kolom merupakan suatu bentuk kekangan data yang akan menentukan jenis data yang akan dimasukkan dalam suatu kolom. Secara garis besar *MySQL* menyediakan beberapa kategori tipe kolom, yaitu sebagai berikut:

a. Tipe Numeris

   Tipe numeris adalah tipe yang menyatakan bilangan. Berikut adalah tabel tipe numeris:

Tabel 2.8 Tipe Numeris

No.	Tipe Data	Keterangan	Kebutuhan Memori
1.	BIT	Menyatakan tipe bit. Satu bit dapat berupa bilangan 0 atau 1. 1. BIT identik dengan BIT(1) yang berarti 1 buah bit 2. BIT(2) berarti mengandung 2 bit Angka maksimal di dalam kurung yaitu 64, yang berarti 64 bit.	(*M*/7)/8 *byte*
2.	TINYINT[(*M*)]	Menyatakan data bilangan bulat yang berkisar antara -128 sampai 127. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan bilangan berkisar dari 0 sampai dengan 127.	1 *byte*
3.	BOOL atau BOOLEAN, atau TINYINT(1)	Untuk menyatakan nilai benar dan salah. Nilai nol dianggap salah dan nilai selain nol dianggap benar.	1 *byte*
4.	SMALLINT[(*M*)]	Menyatakan data bilangan bulat yang berkisar antara -32768 sampai 32767. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan bilangan berkisar dari 0 sampai dengan 65535.	2 *byte*
5.	MEDIUMINT[(*M*)]	Menyatakan data bilangan bulat yang berkisar antara -8388608 sampai 8388607. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan bilangan berkisar dari 0 sampai dengan 16777215	3 *byte*
6.	INT[(*M*)]atau INTEGER[(*M*)]	Menyatakan data bilangan bulat yang berkisar antara -2147483648 sampai 2147483647. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan bilangan berkisar dari 0 sampai dengan 4294967295.	4 *byte*

Lanjutan **Tabel 2.8** Tipe Numeris

7.	BEGINT[(*M*)]	Menyatakan data bilangan bulat yang berkisar antara -9223372036854775808 sampai 9223372036854775807. Bila ditambahkan kata UNSIGNED (berarti bilangan tidak bertanda), jangkauan bilangan berkisar dari 0 sampai dengan 18446744073709551615	6 *byte*
8.	FLOAT[(*M*,*D*)]	Tipe bilangan pecahan yang dapat menampung bilangan antara -3.402823466E+38 sampai dengan 1.175494351E-38, 0, dan 1.175494351E+38 sampai 3.402823466E-38. 1. *M* menyatakan jumlah digit desimal. 2. *D* menyatakan jumlah digit di belakang titik desimal. 3. Jika kata UNSIGNED disertakan di belakangnya, nilai negatif tidak diperkenankan.	4 *byte*
9.	DOUBLE[(*M*,*D*)] atau OUBLE PRECISION[(*M*,*D*)]	Tipe bilangan pecahan yang dapat menampung bilangan antara 1.79769348623157E+308 sampai dengan -2.2250738585072014E-308, 0, dan 2.2250738585072014E-308 sampai 1.79769348623157E+308. *1.* *M* menyatakan jumlah digit desimal. *2.* *D* menyatakan jumlah digit di belakang titik desimal. Jika kata UNSIGNED disertakan di belakangnya, nilai negatif tidak diperkenankan.	8 *byte*

Lanjutan **Tabel 2.8** Tipe Numeris

10.	FLOAT(*p*)	Tipe bilangan pecahan. Dalam hali ini, *p* menyatakan jumlah presisi dalam bit. 1. Jika *p* berkisar antara 0 sampai dengan 24, tipe data identik FLOAT tanpa nilai *M* ataupun *D*. 2. Jika *p* berkisar antara 25 sampai dengan 53, tipe data identik DOUBLE tanpa nilai *M* ataupun *D*.	4 *byte* jika 0 <= *p* <= 24, 8 *byte* jika 25 <= *p* <= 53
11.	DEC[(*M*)] atau DECIMAL[(*M*, *D*)]	Tipe bilangan dengan sejumlah titik desimal. *1.M* menyatakan jumlah digit desimal. *2.D* menyatakan jumlah digit di belakang titik desimal. *3.*Nilai terbesar untuk *M* adalah 65. *4.*Jika *D* tidak disertakan, dianggap *D* bernilai 0 (yang berarti tidak ada bagian pecahan). *5.*Cocok untuk menyimpan data yang memerlukan keakurasian tinggi misalnya uang.	

Sumber: Belajar Database Menggunakan MySQL

b. Tipe Tanggal dan Waktu

Tipe ini adalah tipe menyatakan tanggal dan waktu. Berikut adalah tabel tipe tanggal dan waktu:

Tabel 2.9 Tipe Tanggal dan Waktu

No.	Tipe Data	Keterangan	Kebutuhan Memori
1.	DATE	Menyatakan tipe untuk data tanggal. Tanggal yang didukung berkisar antara '1000-01-01' sampai dengan '999-12-31'. Perlu diketaui, notasi tanggal biasa ditulis dengan format tahun-bulan-tanggal.	3 *byte*
2.	TIME	Menyatakan tipe untuk data waktu yang berformat: 'jam:menit:detik'.	3 *byte*
3.	DATETIME	Menyatakan tipe data yang menggabungkan tanggal dan waktu. Format: 'tahun-bulan-tanggal jam:detik:menit.	8 *byte*
4.	TIMESTAMP	Tipe kolom yang nilainya akan diisi secara otomatis dengan tanggal dan jam saat ada operasi INSERT atau UPDATE.	4 *byte*
5.	YEAR[(2|4)]	Tipe untuk menyatakan tahun. Bila angka tidak disebutkanm tahun dianggap berformat 4 digit. Angka 2 berarti tahun menggunakan 2 digit. 1. Untuk format 4 digit, tahun dapat berkisar 1901 sampai dengan 2155 bisa juga 0000. 2. Untuk format 2 digit, tahun dapat berkisar 70 sampai dengan 69. 70 berarti 1970 dan 69 berarti 2069. 3. Nilai untuk YEAR(4) dapat diisi berupa string ataupun bilangan.	1 *byte*

Sumber: Belajar Database Menggunakan MySQL

c. Tipe String

String berarti deretan karakter. Berikut adalah tabel tipe string:

Tabel 2.10 Tipe String

No.	Tipe Data	Keterangan	Kebutuhan Memori
1.	CHAR(*M*)	String dengan panjang tetap yaitu *M* karakter. Bila string yang dimasukkan memiliki panjang kurang dari *M* maka sisanya diisi dengan spasi. Nilai terbesar untuk *M* adalah 255.	*M byte*, 0 <= *M* <= 255
2.	CHAR	Sinonim denga CHAR(1).	1 *byte*
3.	VARCHAR(*M*)	String dengan panjang variabel. *M* menyatakan jumlah maksimum karakter yang bisa disimpan. Nilai terbesar untuk *M* yaitu 65535.	*L*+1 *byte*, *L* <= *M* dan 0 <= *M* <= 65535
4.	BINARY(*M*)	Serupa dengan CHAR(*M*), tetapi disimpan dalam bentuk biner.	*M byte*, 0 <= *M* <= 255
5.	VARBINARY(*M*)	Serupa dengan VARCHAR(M), tetapi disimpan dalam bentuk biner.	*L*+1 *byte*, *L* <= *M* and 0 <= *M* <= 255
6.	TINYBLOB	BLOB dengan ukuran paling kecil.	*L*+1 *byte*, *L* < 2^8
7.	BLOB[(*M*)]	BLOB dengan ukuran lebih besar daripada TINYBLOB.	*L*+2 *byte*, *L* < 2^{16}
8.	MEDIUMBLOB	BLOB berukuran medium.	*L*+3 *byte*, *L* < 2^{24}
9.	LONGBLOB	BLOB berukuran paling besar.	*L*+4 *byte*, *L* < 2^{32}
10.	TINYTEXT	String teks dengan ukuran paling kecil.	*L*+1 *byte*, *L* < 2^8

Lanjutan **Tabel 2.10** Tipe String

11.	TEXT[(*M*)]	String teks dengan ukuran lebih besar	*L*+2 *byte*, *L* < 2^{16}
12.	MEDIUMTEXT	String teks berurutan medium.	*L*+3 *byte*, *L* < 2^{24}
13.	LONGTEXT	String teks berurutan paling besar.	*L*+4 *byte*, *L* < 2^{32}
14.	ENUM(‘nilai1’, ‘nilai2’,…)	Tipe kolom yang memungkinkan nilai kolom berupa salah satu string yang disebutkan pada ENUM. Jumlah elemen ENUM dapat mencapai 65535. Nilai untuk kolom dapat juga diisi dengan nomor indeks. Daftar berikutnya menunjukkan nilai dan indeksnya: 1. NULL (indeks NULL). 2. ‘’ (indeks 0). 3. ‘nilai 1’ (indeks 1). 4. ‘nilai 2’ (indeks 2), dan seterusnya.	1 atau 2 *byte*, bergantung pada jumlah nilai *enumerasi*
15.	SET(‘nilai1’, ‘nilai2’,…)	Merupakan suatu objek string yang bisa berisi nol, satu, atau beberapa nilai. Sebuah kolom bertipe SET maksimum berisi 64 elemen yang berbeda. Nilai antar-elemen ditulis dengan pisah koma.	1, 2, 3, 4, atau 8 *byte*, bergantung pada jumlah anggota himpunan

Sumber: Belajar Database Menggunakan MySQL