



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

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

Menurut Siallagan (2009:1) menyatakan, “Komputer adalah sekumpulan alat elektronik yang saling bekerja sama, dapat menerima data (*input*), mengolah data (*process*), memberikan informasi (*output*), dan terkoordinasi di bawah kontrol program yang tersimpan dalam memorinya.”

Dari kedua pengertian tersebut, dapat disimpulkan bahwa komputer adalah suatu alat yang dipakai untuk mengolah data menurut prosedur yang telah dirumuskan.

2.1.2. Pengertian Internet

Menurut Ahmadi dan Hermawan (2013:2) menyatakan, “*Internet* adalah sistem informasi global berbasis komputer.”

Internet (kependekan dari *interconnection-networking*) adalah seluruh jaringan komputer yang saling terhubung menggunakan standar sistem global *Transmission Control Protocol / Internet Protocol Suite* (TCP/IP) sebagai protokol pertukaran paket (*packet switching communication protocol*) untuk melayani miliaran pengguna di seluruh dunia.

Dari definisi-definisi di atas, dapat disimpulkan bahwa *Internet* adalah sistem global dari seluruh jaringan komputer yang saling terhubung sehingga antara komputer dapat saling mengakses informasi dan bertukar data.



2.1.3. Metodologi Pengembangan Sistem

Menurut 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.

2.1.4. Pengertian Aplikasi

Menurut Asropudin (2013:6) menyatakan, “Aplikasi adalah *software* yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu, misalnya *Ms-Word*, *Ms-Excel*.”

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

Dari kedua pengertian tersebut, dapat disimpulkan bahwa aplikasi adalah suatu perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna.

2.1.5. Pengertian Penjualan

Menurut Kusnadi (2009,300) menyatakan “Penjualan (*sales*) adalah sejumlah uang yang dibebankan kepada pembeli atas barang atau jasa yang dijual”.



2.1.6. Pengertian Persediaan

Menurut Mulya (2010:214) menyatakan, “Persediaan adalah aktiva yang tersedia untuk dijual dalam kegiatan usaha normal perusahaan, aktiva dalam proses produksi dan atau dalam perjalanan atau dalam bentuk bahan atau perlengkapan untuk digunakan dalam proses produksi atau pemberian jasa”.

2.1.7. Pengertian Permintaan

Permintaan adalah sejumlah barang yang dibeli atau diminta pada suatu harga dan waktu tertentu yang berkaitan dengan keinginan konsumen akan suatu barang dan jasa yang ingin dipenuhi.

2.1.8. Pengertian Distributor

Menurut Salim (2013 : 162) menyatakan, “distributor adalah orang atau badan yang bertugas mendistribusikan sesuatu”.

2.1.9. Pengertian Web

Menurut Ahmadi dan Hermawan (2013:69) menyatakan, “*World Wide Web*, lebih terkenal disingkat WWW, adalah suatu ruang informasi yang dipakai oleh pengenal global yang disebut Pengidentifikasi Sumber Seragam untuk mengenal pasti sumber daya berguna.”

Pengertian *website* adalah suatu kumpulan dari beberapa tampilan halaman dari file dan dokumen secara *hypermedia* melalui media komputer dan jejaring untuk mempermudah komunikasi dan aktivitas pelayanan bisnis selama 24 jam dengan para konsumen dan rekan bisnis

Dari beberapa pengertian di atas, dapat disimpulkan bahwa *web/website* merupakan halaman situs sistem informasi yang dapat diakses secara cepat.

2.1.10. Pengertian PT. Semen Baturaja (Persero) Tbk

PT Semen Baturaja (Persero) adalah Badan Usaha Milik Negara (BUMN) yang bergerak pada bidang industri semen di wilayah Sumatera Bagian Selatan. Kapasitas produksi 1,250 juta ton per tahun dan lokasi pabrik yang terletak di



Baturaja, Palembang dan Panjang. Semen Baturaja selalu berusaha untuk menjaga kontinuitas dan stabilitas pasokan semen dalam negeri khususnya di Sumatera Bagian Selatan, karena Semen Baturaja merupakan produsen semen tunggal untuk wilayah sumatera bagian selatan.

2.1.11. Pengertian Aplikasi Penjualan dan Persediaan Produk Semen untuk Permintaan pada Distributor PT. Semen Baturaja (Persero)Tbk Lampung

Aplikasi Penjualan dan Persediaan Produk Semen untuk Permintaan Distributor pada PT. Semen Baturaja (Persero)Tbk Lampung adalah suatu aplikasi *e-commerce* dengan sistem informasi berbasis *web* yang dibuat dengan menggunakan bahasa pemrograman *PHP* dan *database MySQL* sehingga mempermudah dan mengoptimalkan dalam penjualan dan persediaan stock semen baturaja untuk para distributornya.

2.2. Teori Khusus

2.2.1. Pengertian DFD (Data Flow Diagram)

Sutabri (2012:117) menyatakan, “*Data Flow Diagram (DFD)* adalah salah satu *network* yang menggambarkan sistem automat/komputerisasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya.”

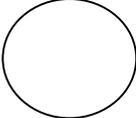
Sedangkan menurut Saputra (2013:118), “*Data Flow Diagram* yang disingkat *DFD* merupakan suatu diagram yang menggambarkan alir data dalam suatu entitas ke sistem atau sistem ke entitas.”

Dari kedua pengertian tersebut, dapat disimpulkan bahwa (*Data Flow Diagram*) *DFD* merupakan suatu cara atau metode untuk membuat rancangan sebuah sistem yang mana berorientasi pada alur data yang nantinya bergerak pada sebuah sistem.

Saputra (2013:119) juga menjelaskan 4 simbol *DFD* dalam versi Yourdon/De Marco sebagai berikut:



Tabel 2.1. Simbol Data Flow Diagram

No.	Simbol	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 : Saputra, 2013:119)

A.S dan Shalahuddin (2013:72) menjelaskan tahapan-tahapan perancangan dengan menggunakan *DFD* sebagai berikut:

1. *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. *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. DFD Level 2

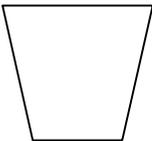
Modul-modul pada DFD Level 1 dapat di-*breakdown* menjadi DFD Level 2. Modul mana saja yang harus di-*breakdown* lebih detail tergantung pada tingkat kedetailan modul tersebut. Apabila modul tersebut sudah cukup detail dan rinci maka modul tersebut sudah tidak perlu untuk di-*breakdown* lagi. Untuk sebuah sistem, jumlah DFD Level 2 sama dengan jumlah modul pada DFD Level di atasnya. *Breakdown* pada level 3,4,5, dan seterusnya aturannya sama persis dengan DFD Level 1 atau Level 2.

2.2.2. Pengertian Block Chart

Kristanto (2008:75) menyatakan, “*Block chart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu.”

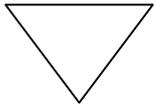
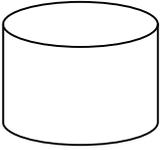
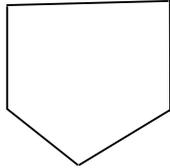
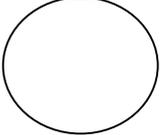
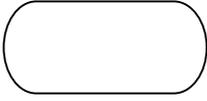
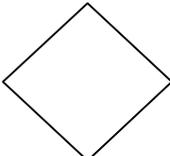
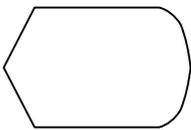
Kristanto (2008:75) juga menjelaskan tentang simbol-simbol yang sering digunakan dalam *block chart* dapat dilihat pada tabel berikut ini:

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.



Lanjutan Tabel 2.2. Simbol-simbol dalam Block Chart

No	Simbol	Keterangan
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsifkan (arsif manual)
6		Data penyimpanan (<i>data storage</i>).
7		Proses apa saja yang tidak terdefinisi termasuk aktifitas fisik.
8		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.
9		Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.
10		Terminasi yang menandakan awal dan akhir dari suatu aliran.
11		Pengambilan keputusan (<i>decision</i>).
12		Layar peraga (<i>monitor</i>).



Lanjutan Tabel 2.2. *Simbol-simbol dalam Block Chart*

No	Simbol	Keterangan
13		Pemasukan data secara manual.

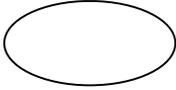
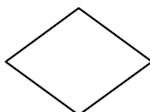
(Sumber : Kristanto, 2008:75)

2.2.3. Pengertian ERD (*Entity Relational Diagram*)

A.S dan Shalahuddin (2013:50) menjelaskan bahwa *Entity Relationship Diagram* (ERD) digunakan untuk pemodelan basis data relasional. ERD memiliki beberapa aliran notasi seperti notasi Chen (dikembangkan oleh Peter Chen), Barker (dikembangkan oleh Richard Barker), Ian (dikembangkan oleh Palmer, Harry Ellis), notasi Crow's Foot, dan beberapa notasi lainnya.

Berikut adalah simbol-simbol yang digunakan pada ERD dengan notasi Chen:

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

No	Simbol	Keterangan
1.		Entitas merupakan data inti yang akan disimpan
2.		Atribut atau <i>field</i> atau kolom data yang butuh disimpan dalam suatu entitas.
3.		Relasi, menghubungkan anatar entitas dan biasanya diawali dengan kata kerja.



Lanjutan Tabel 2.3. Simbol-simbol Entity Relationship Diagram (ERD)

No	Simbol	Keterangan
4.		Asosiasi, penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas satu dengan entitas yang lain disebut dengan kardinalitas. Misalkan ada kardinalitas 1 ke N atau sering disebut dengan <i>one to many</i> .

(Sumber : A.S dan Shalahuddin, 2013:50)

2.2.4. Pengertian *Flowchart*

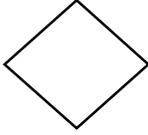
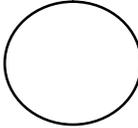
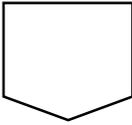
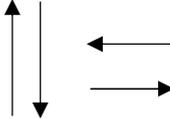
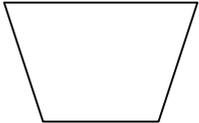
Ewolf community (2012:16) mengemukakan, “*Flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi, setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya”.

Adapun simbol-simbol *Flowchart* adalah sebagai berikut:

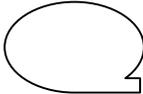
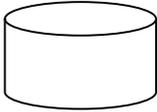
Tabel 2.3. Simbol-Simbol *Flowchart*

No.	Simbol	Arti
1.		Simbol Start atau End yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i>
2.		Simbol pemrosesan yang terjadi pada sebuah alur kerja
3.		Simbol <i>Input/Output</i> yang mendefinisikan masukan dan keluaran proses

Lanjutan **Tabel 2.3.** Simbol-Simbol *Flowchart*

No.	Simbol	Arti
4.		Simbol untuk memutuskan proses lanjutan dari kondisi tertentu
5.		Simbol konektor untuk menyambung proses pada lembar kerja yang sama
6.		Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
7.		Simbol untuk menghubungkan antar proses atau antar simbol
8.		Simbol yang menyatakan piranti keluaran, seperti layar monitor, <i>printer</i> , dll
9.		Simbol yang mendefinisikan proses yang dilakukan secara manual
10.		Simbol masukan atau keluaran dari atau ke sebuah dokumen

Lanjutan **Tabel 2.3.** Simbol-Simbol *Flowchart*

No.	Simbol	Arti
11.		Simbol yang menyatakan bagian dari program (subprogram)
12.		Simbol masukan atau keluaran dari atau ke sebuah pita magnetic
13.		Simbol database atau basis data

(Sumber: Ewolf Community, 2012:17)

2.2.5. Pengertian Kamus Data

A.S dan Shalahuddin (2013:73) menyatakan, "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)."

A.S dan Shalahuddin (2013:74) menjelaskan bahwa kamus data memiliki beberapa simbol untuk menjelaskan informasi tambahan, yaitu sebagai berikut:

Table 2.5. Simbol-simbol dalam Kamus Data

No	Simbol	Arti
1	=	disusun atau terdiri atas
2	+	AND atau dan
3	()	data opsional
4	[]	baik...atau...
5	*...*	batas komentar
6	{ }n	n kali diulang / bernilai banyak

(Sumber : A.S dan Shalahuddin, 2013:74)



2.2.6. Design Input dan Output

2.2.6.1. Input

Isa (2012:50) menyatakan, "Input adalah sesuatu yang dibutuhkan oleh suatu sub-sistem untuk melakukan fungsinya."

Input dapat berasal dari:

- a. Luar sistem yang baru (dari *internal* Sistem Informasi yang lain (misalnya *output* dari sistem pemesanan) atau dari eksternal sistem (misalnya dari *output* sistem paket aplikasi atau dari organisasi lain)).
- b. Sub-sistem yang lain, yang masih merupakan bagian dari sistem tersebut.
- c. Dari (pusat) kumpulan data/*database*.

2.2.6.2. Output

Isa (2012:50) menyatakan, "*Output* merupakan suatu bentuk keluaran yang dikehendaki dari *input* yang telah melalui pemrosesan (baik berupa laporan yang berbentuk *hardcopy*, maupun data/*file* yang akan menjadi *input* untuk sub-sistem lain)."

Output dapat ditujukan kepada:

- a. Sub-sistem yang lain (yang akan berfungsi sebagai *input* untuk sub-sistem tersebut baik yang berupa program maupun *spreadsheet*).
- b. *Internal* sistem yang lain (misalnya untuk sistem penggajian) atau eksternal sistem (misalnya data untuk sistem pembayaran gaji karyawan yang dilakukan oleh bank).
- c. *e-Commerce*.
- d. kumpulan data / *database*.



2.3. Teori Program

2.3.1. Basis Data (*Database*)

Priyadi (2014:2) menyatakan, “Secara umum, pengertian basis data adalah sekumpulan fakta berupa representasi tabel yang saling berhubungan dan disimpan dalam media penyimpanan secara digital.”

Priyadi (2014:3) juga menjelaskan beberapa elemen basis data, diantaranya adalah sebagai berikut:

1. *Tabel*

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

2. *Field*

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

3. *Record*

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

4. Kardinalitas

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



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

2.3.2. Sekilas Tentang *PHP*

2.3.2.1. Pengertian *PHP* (*Hypertext Preprocessor*)

Saputra (2013:1) menyatakan, “*PHP* adalah singkatan dari (*Hypertext Preprocessor*) yang merupakan suatu bahasa pemrograman yang hanya dapat berjalan pada sisi *server* (*Server Side Scripting*).”

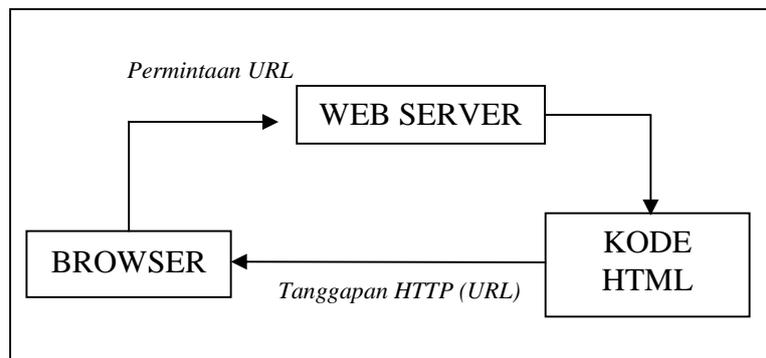
MADCOMS (2011:216) menyatakan, “*PHP* adalah salah satu bahasa pemrograman yang berjalan dalam sebuah *webserver* dan berfungsi sebagai pengolah data pada sebuah *server*.”

Dari kedua pengertian di atas, dapat disimpulkan bahwa *PHP* adalah salah satu bahasa pemrograman skrip yang dirancang untuk membangun aplikasi *web*.

2.3.2.2. Konsep Kerja *PHP*

Saputra (2013:4) menjelaskan tentang konsep kerja *PHP* adalah sebagai berikut:

1. Server membaca permintaan dari *client/browser*.
2. Kemudian dilanjutkan untuk mencari halaman/*page* pada *server*.
3. *Server* melakukan instruksi yang diberikan oleh *php* untuk melakukan modifikasi pada halaman/*page*.
4. Selanjutnya hasil modifikasi tersebut akan dikembangkan kepada *client/browser*.



(Sumber : Saputra, 2013:5)

Gambar 2.1. Cara Kerja PHP

2.3.2.3. Skrip PHP

Badiyanto (2013:32) menyatakan, "Skrip *PHP* merupakan sebuah bahasa *scripting* yang terpasang pada HTML. Berikut adalah contoh skrip *PHP* yang berada di dalam kode HTML:

```

<HTML>
<HEAD>
<TITLE> Latihan Pertama</TITLE>
</HEAD>
<BODY>
Selamat Belajar PHP<BR>
<?php
    Printf("Tgl. Sekarang: %s ", Date("d F y"));
?>
</BODY>
</HTML>
  
```

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



Gambar 2.2. Tampilan kode PHP pada browser

2.3.2.4. Tipe Data PHP

MADCOMS (2011:221) menjelaskan tentang tipe data *PHP*, yaitu:

1. Tipe Data *Integer*

Tipe data ini berisikan data semua bilangan bulat yang besarnya range sama dengan data pada bahasa C, yaitu antara **-2, 147, 483, 648** sampa **+2, 147, 483, 647** pada platform **32bit**. Apabila data di luar kisaran tersebut maka *PHP* secara otomatis mengkonversikan data tersebut dari tipe *Integer* menjadi tipe *Floating Point*.

2. Tipe Data *Floating Point*

Tipe data ini berisikan bilangan pecahan atau bilangan desimal. Kisaran data *floating* adalah antara **1.7E-308** sampai **1.7E+308**. Bentuk dari data ini adalah desimal ataupun dalam bentuk pangkat.

3. Variabel

Dalam program *PHP*, variabel digunakan untuk menyimpan data sementara baik jenis *string*, *integer*, maupun *array*. Variabel dinyatakan dengan tanda \$ di depan nama variabel. Nama variabel dapat berupa huruf, angka maupun garis bawah. Namun dalam penulisannya, variabel harus diawali dengan huruf atau



garis bawah (_), kemudian diikuti huruf atau angka. Penulisan variabel yang diawali angka tidak dibenarkan.

2.3.3. Sekilas Tentang *MySql*

2.3.3.1. Pengertian *MySql*

Saputra (2013:14), “*MySQL* merupakan *database storage engine* yang paling banyak digunakan oleh *web developer* karena sifatnya *free*, alias gratis.”

Sedangkan menurut Kadir (2008:342), “*MySQL* adalah salah satu jenis *database server* yang sangat populer yang bersifat *open source* dan termasuk jenis *RDBMS (Relational Database Management System)*.”

Dari kedua pengertian di atas, dapat disimpulkan bahwa *MySQL* adalah jenis *software* pada *Relational Database Management System (RDBMS)* pada sebuah manajemen *database* sebagai basis data.

2.3.3.2. Fungsi-Fungsi *MySQL*

Kadir (2008:360), menjelaskan tentang sejumlah fungsi yang berawalan *mysql_* yang digunakan untuk mengakses *database server MySQL* sebagai berikut:

Tabel 2.6. *Fungsi-Fungsi MySQL*

No	Nama Fungsi	Kegunaan
1	<i>MySQL_connect()</i>	Membuka koneksi ke <i>server MySQL</i>
2	<i>MySQL_close()</i>	Menutup koneksi <i>MySQL</i>
3	<i>MySQL_select_db()</i>	Membuka/mengaktifkan suatu basis data
4	<i>MySQL_query()</i>	Mengeksekusi <i>string query</i>
5	<i>MySQL_db_query()</i>	Melakukan <i>query</i>
6	<i>MySQL_num_fields()</i>	Menghitung jumlah <i>field</i> dalam tabel tertentu
7	<i>MySQL_fetch_assoc()</i>	Mengambil hasil <i>query</i> sebelumnya
8	<i>MySQL_fetch_result()</i>	Membebaskan <i>resource</i> dari hasil <i>query</i>
9	<i>MySQL_fetch_row()</i>	Menampilkan hasil <i>query</i> ke dalam <i>array</i>



Lanjutan Tabel 2.6. Fungsi-Fungsi MySQL

No	Nama Fungsi	Kegunaan
10	<i>MySQL_fetch_array()</i>	Menampilkan hasil <i>query</i> sebagai <i>array</i>
11	<i>MySQL_fetch_field()</i>	Menampilkan informasi <i>field</i> termasuk tipe datanya
12	<i>MySQL_affected_rows()</i>	Menampilkan jumlah baris yang dikenai operasi <i>SQL</i>
13	<i>MySQL_errno</i>	Menampilkan kode kesalahan (<i>error number</i>) dari perintah <i>SQL</i>
14	<i>MySQL_create_db()</i>	Membuat <i>database MySQL</i>
15	<i>MySQL_drop_db()</i>	Menghapus <i>database MySQL</i>
16	<i>MySQL_list_dbs()</i>	Menghasilkan seluruh daftar <i>database MySQL</i>
17	<i>MySQL_list_tables()</i>	Menampilkan seluruh tabel dalam <i>database</i> tertentu
18	<i>MySQL_list_fields()</i>	Menampilkan seluruh <i>field</i> dalam tabel tertentu

(Sumber : Kadir, 2008:360)

2.3.4. Pengertian Adobe Dreamweaver CS6

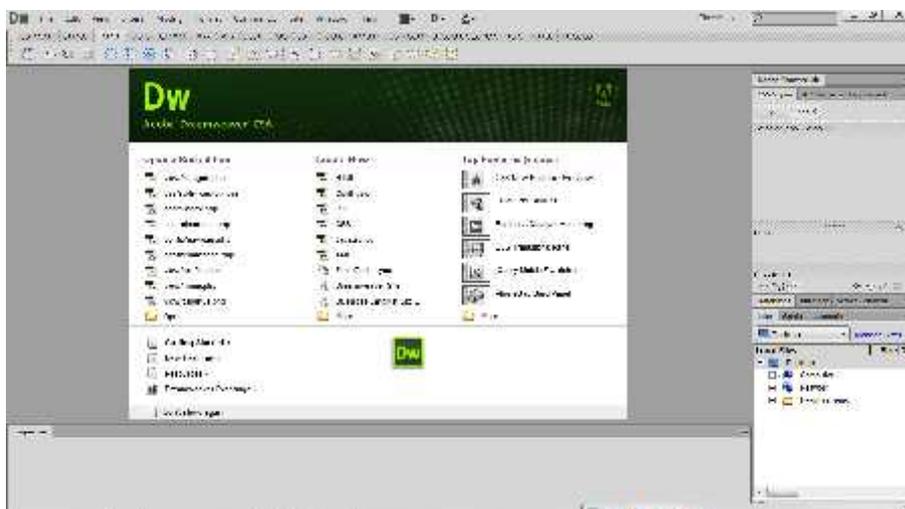
Adobe Dreamweaver CS6 merupakan program penyunting halaman *web* keluaran *Adobe Systems* yang dulu dikenal sebagai *Macromedia Dreamweaver* keluaran *Macromedia*. Program ini banyak digunakan oleh pengembang *web* karena fitur-fiturnya yang menarik dan kemudahan penggunaannya.

ELCOM (2013:1), "*Dreamweaver CS6* adalah sebuah editor profesional yang menggunakan *HTML* untuk mendesain *web* secara *visual* dan mengelola situs atau halaman *web* dengan beberapa kemampuan. *Adobe Dreamweaver CS6*



memiliki beberapa kemampuan. Versi ini bukan hanya *software* untuk desain *web* tetapi juga untuk menyunting kode serta pembuatan aplikasi *web* antara lain *JSP*, *PHP*, *ASP*, *XML*, dan *ColdFusion*.

Dari kedua pengertian di atas, dapat disimpulkan bahwa *Dreamweaver CS6* adalah salah satu program aplikasi yang digunakan untuk membangun sebuah *website*, baik secara grafis maupun menuliskan kode sumber secara langsung dengan tampilan yang lebih menarik dan kemampuan yang lebih banyak.



Gambar 2.3. Tampilan halaman Welcome Screen dari Dreamweaver CS6

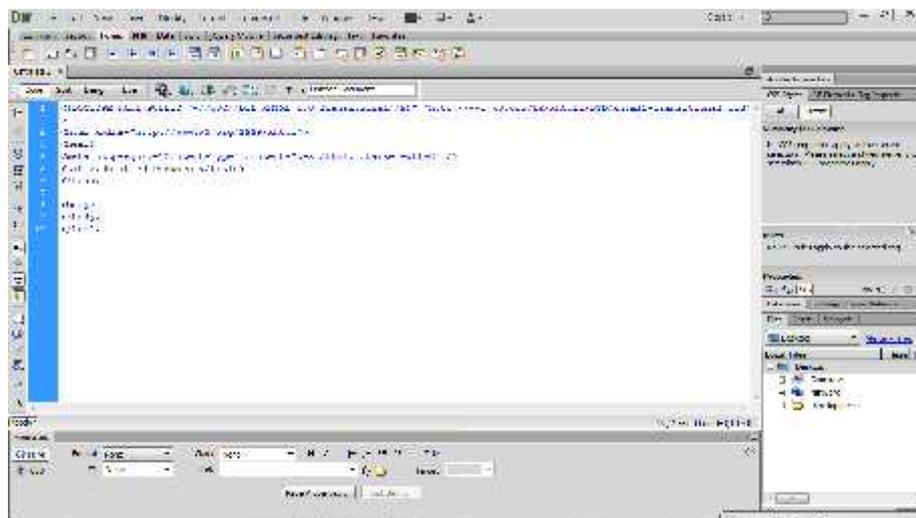
2.3.5. Ruang Kerja Dreamweaver CS6

ELCOM (2013:3) menjelaskan tentang ruang kerja *Dreamweaver CS6* adalah:

1. **Insert Bar**, berisi tombol-tombol untuk menyisipkan berbagai macam objek seperti *image*, *table*, dan *layer* ke dalam jendela dokumen.
2. **Document Toolbar**, berisi tombol-tombol dan menu pop-up yang menyediakan tampilan berbeda dari jendela dokumen.
3. **Coding Window**, berisi kode-kode *HTML* dan tempat untuk menuliskan kode-kode pemrograman, misalnya *PHP* atau *ASP*.



4. **Panel Group**, berisi kumpulan panel yang saling berkaitan satu sama lainnya yang dikelompokkan di bawah satu judul.
5. **Property Inspector**, digunakan untuk melihat dan mengubah berbagai properti objek atau teks.
6. **Jendela Dokumen**, digunakan untuk menampilkan dokumen saat dimana Anda sekarang bekerja.
7. **Ruler**, mempermudah ukuran dalam mendesain halaman *web*.
8. **Site Panel**, digunakan untuk mengatur *file-file* dan *folder-folder* yang membentuk situs *web* Anda.



Gambar 2.4. Ruang kerja dari Dreamweaver CS6

2.3.6. JQuery

JQuery adalah pustaka *JavaScript* kecil bersumber terbuka yang menekankan pada interaksi antara *Javascript* dan *HTML*. Pustaka ini dirilis pada Januari 2006 di BarCamp NYC oleh John Reig dan berlisensi ganda di bawah MIT dan GPL.

JQuery adalah library *JavaScript* yang dibuat untuk memudahkan pembuatan *website* dengan *HTML* yang berjalan di sisi *client*.

Dari definisi-definisi di atas, dapat disimpulkan bahwa *JQuery* adalah kumpulan kode atau fungsi *JavaScript* siap pakai, sehingga mempermudah dan mempercepat kita dalam membuat kode *JavaScript*.



2.3.7. JavaScript

Saputra (2012:3) menyatakan, “*JavaScript* adalah bahasa *scripting* yang handal yang berjalan di sisi *client*.”

JavaScript adalah bahasa skrip yang populer di *internet* dan dapat bekerja di sebagian besar penjelajah *web* populer seperti *Internet Explorer (IE)*, *Mozilla*, *Firefox*, *Netscape* dan *Opera*. Kode *JavaScript* dapat disisipkan dalam halaman *web* menggunakan tag *SCRIPT*.

JavaScript adalah bahasa pemrograman yang bisa disisipkan ke *HTML* seperti halnya *PHP* akan tetapi javascript berjalan di sisi *client*.

Dari definisi-definisi di atas, dapat disimpulkan bahwa *JavaScript* adalah sekumpulan perintah khusus yang digunakan untuk membuat sebuah halaman *web* yang lebih responsif dan interaktif.