



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Data

Sutarman (2012:3), “Data adalah fakta dari sesuatu pernyataan yang berasal dari kenyataan, di mana pernyataan tersebut merupakan hasil dari pengukuran atau pengamatan”.

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

Kristanto (2008:7), mengatakan “Data merupakan bentuk yang menggambarkan kejadian yang sedang terjadi, dimana data tersebut diolah dan diterapkan dalam sistem menjadi input yang berguna dalam suatu sistem”.

2.1.2. Pengertian Pengolahan Data

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

Kristanto (2008:8), “Pengolahan data adalah waktu yang digunakan untuk menggambarkan perubahan bentuk data menjadi informasi yang memiliki kegunaan”.

Jadi, dapat disimpulkan bahwa pengolahan data adalah kegiatan penyimpanan data dan penanganan data dalam waktu tertentu yang menggambarkan perubahan bentuk menjadi informasi yang memiliki kegunaan.

2.1.3. Pengertian Sistem

Kristanto (2008:1), “Sistem adalah kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses masukan (*input*) yang ditujukan kepada system tersebut dan mengolah masukan tersebut sampai menghasilkan keluaran (*output*) yang diinginkan.



Fatta (2007:3), “Sistem adalah kumpulan dari bagian-bagian yang bekerja sama untuk mencapai suatu tujuan yang sama.

Sutabri (2012:22), “Sistem merupakan suatu bentuk intergrasi antara satu komponen dengan komponen lain karena system memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada didalam system tersebut.

2.1.3.1. Karakteristik Sistem

Sutabri (2012:20) menjelaskan bahwa sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik sistem antara lain:

a. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi , artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar atau sering disebut “supra sistem”.

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Batasan apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Dengan demikian, lingkungan luar tersebut harus tetap dijaga dan dipelihara. Lingkungan luar yang merugikan harus dikendalikan. Kalau tidak, maka akan mengganggu kelangsungan hidup sistem tersebut.



d. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.

e. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

f. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal yang menjadi *input* bagi subsistem lain.

g. Pengolah Sistem (*Proses*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. Sasaran Sistem

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat *deterministic*. Jika suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.3.2. Klasifikasi Sistem

Sutabri (2012:15) menjelaskan bahwa sistem dapat diklasifikasikan dari beberapa sudut pandang, seperti:

a. Sistem Abstrak dan Sistem Fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Contohnya Sistem Teologis, yaitu suatu sistem yang



berupa pemikiran tentang hubungan antara manusia dengan Tuhan. Sedangkan, sistem fisik adalah sistem yang ada secara fisik, seperti sistem komputer, sistem produksi, sistem penjualan, dan lain sebagainya.

b. Sistem Alamiah dan Sistem Buatan Manusia

Sistem Alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia. Contoh sistem alamiah adalah sistem perputaran bumi, terjadinya siang malam, dan pergantian musim. Sedangkan, sistem buatan manusia adalah sistem yang melibatkan hubungan manusia dengan mesin, yaitu *human machine system*.

c. Sistem Deterministik dan Sistem Probabilistik

Sistem Deterministik adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi. Contoh sistem deterministik adalah sistem komputer sebab sistem tersebut dapat dipastikan berdasarkan program komputer yang dijalankan. Sedangkan, sistem probabilistik adalah sistem yang kondisi masa depannya tidak dapat diprediksi, karena mengandung unsur probabilitas.

d. Sistem Terbuka dan Sistem Tertutup

Sistem Terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya, yang menerima masukan dan menghasilkan keluaran untuk subsistem lainnya. Sedangkan, sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa ada campur tangan dari pihak luar.

2.1.4. Pengertian Informasi

Sutanta (2011:13) menjelaskan bahwa informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan dasar dalam pengambilan keputusan yang dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang.



Sutabri (2012:22) menjelaskan bahwa informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan.

Sibero (2014:10) menjelaskan bahwa informasi adalah suatu hasil pengolahan data dari sistem komputer yang dibutuhkan pengguna yang memiliki manfaat bagi penggunanya.

2.1.5. Pengertian Manajemen

Coulter (2010:7), Manajemen adalah hal yang dilakukan oleh para manajer yang melibatkan aktivitas-aktivitas koordinasi dan pengawasan terhadap pekerjaan orang lain, sehingga pekerjaan tersebut dapat diselesaikan secara efisien dan efektif.

Dompas (2007:1), Manajemen adalah pengarahan menggerakkan sekelompok orang dan fasilitas dalam usaha untuk mencapai tujuan tertentu.

Daft (2003:5), Manajemen adalah pencapaian tujuan-tujuan organisasi dengan cara-cara yang efektif dan efisien melalui perencanaan, pengorganisasian, kepemimpinan dan pengendalian/ pengawasan sumber daya organisasi.

2.1.6. Pengertian Operasional

Prawirosentoro (2001:7), Operasional adalah berkaitan dengan perusahaan barang dan jasa, karena operasi pada prinsipnya pengelolaan atas proses seluruh operasi perusahaan”.

2.1.7. Pengertian Terminal

Menurut Kamus Indonesia (2009:633), Terminal adalah tempat berhentinya kendaraan atau stasiun.

Menurut Undang-Undang No. 14 tahun 1992, Terminal adalah prasarana transportasi jalan untuk keperluan memuat dan menurunkan orang dan atau barang serta mengatur kedatangan dan pemberangkatan kendaraan umum, yang merupakan salah satu wujud simpul jarinya transportasi.



2.1.8. Pengertian Petikemas

Suyono (2001:133), Petikemas adalah satu kemasan yang dirancang secara khusus dengan ukuran tertentu, dapat dipakai berulang kali, dipergunakan untuk menyimpan dan sekaligus mengangkat muatan yang ada didalamnya.

Menurut Koleangan (2008:6), Container adalah semua media dimana didalamnya dapat dimasukkan sesuatu barang, atau tempat mengisi barang.

2.1.9. Pengertian Sistem Informasi Manajemen Operasional Terminal Petikemas pada PT.Pelabuhan Indonesia II (Persero) cabang Palembang.

Sistem Informasi Manajemen Operasional Terminal Petikemas pada PT.Pelabuhan Indonesia II (Persero) cabang Palembang adalah suatu sistem informasi yang digunakan untuk membantu pendataan kegiatan manajemen operasional terminal petikemas pada PT.Pelabuhan Indonesia II (Persero) cabang Palembang.

2.2. Teori Khusus

2.2.1. Pengertian DFD (*Data Flow Diagram*)

Sutabri (2012:116-120), “Secara umum Data Flow Diagram adalah suatu *network* yang menggambarkan suatu system automat/komputerisasi, manualisasi, atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya.

DFD merupakan diagram yang digunakan untuk menggambarkan proses-proses yang terjadi pada sistem yang akan dikembangkan. Dengan model ini data-data yang terlibat pada masing-masing proses dapat didefinisikan.

Keuntungan penggunaan DFD adalah memungkinkan untuk menggambarkan sistem dari level yang paling tinggi kemudian menguraikannya menjadi level yang lebih rendah (dekomposisi). Sedangkan kekurangan


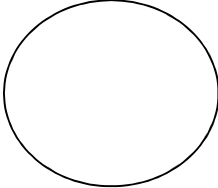
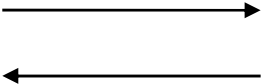



penggunaan DFD adalah tidak menunjukkan proses pengulangan (*looping*), proses keputusan dan proses perhitungan.

2.2.1.1. Simbol *Data Flow Diagram*

Simbol atau lambang yang digunakan dalam membuat *data flow diagram* ada 4 (empat) buah, yaitu sebagai berikut :

Tabel 2.1 Simbol-simbol dalam *Data Flow Diagram*

NO	Simbol	Keterangan
1		<i>EXTERNAL ENTITY</i> Simbol ini digunakan untuk menggambarkan asal atau tujuan data.
2		PROSES Simbol ini digunakan untuk proses pengolahan atau transformasi data.
3		<i>DATA FLOW</i> Simbol ini digunakan untuk menggambarkan aliran data yang berjalan.
4		<i>DATA STORE</i> Simbol ini digunakan untuk menggambarkan data <i>flow</i> yang sudah disimpan atau diarsipkan.



2.2.1.2. Aturan Main Data Flow Diagram

Bentuk rambu-rambu atau aturan main yang baku dan berlaku dalam penggunaan *data flow diagram* untuk membuat model sistem adalah sebagai berikut :

- 1) Di dalam *data flow diagram* tidak boleh menghubungkan antara satu *external entity* dengan *external entity* lainnya secara langsung.
- 2) Di dalam *data flow diagram* tidak boleh menghubungkan *data store* yang satu dengan *data store* yang lainnya secara langsung.
- 3) Di dalam *data flow diagram* tidak boleh menghubungkan *data store* dengan *external entity* secara langsung.
- 4) Setiap proses harus ada memiliki *data flow* yang masuk dan ada juga *data flow* yang keluar.

2.2.1.3. Teknik Menbuat Data Flow Diagram

Teknik atau cara yang lazim digunakan dalam membuat *data flow diagram* adalah :

- 1) Mulai dari yang umum atau tingkatan yang lebih tinggi, kemudian diuraikan atau dijelaskan sampai yang lebih detail atau tingkatan yang lebih rendah, yang lebih dikenal dengan istilah *TOP-DOWN ANALYSIS*.
- 2) Jabarkan proses yang terjadi di dalam *data flow diagram* sedetail mungkin sampai tidak dapat diuraikan lagi.
- 3) Peliharalah konsistensi proses yang terjadi didalam DFD, mulai dari diagram yang tingkatannya lebih tinggi sampai dengan diagram yang tingkatannya lebih rendah.
- 4) Berikan label yang bermakna untuk setiap simbol yang digunakan seperti:
 - a) Nama yang jelas untuk *EXTERNAL ENTITY*
 - b) Nama yang jelas untuk *PROSES*
 - c) Nama yang jelas untuk *DATA FLOW*
 - d) Nama yang jelas untuk *DATA STORE*



2.2.1.4 Tahapan *Data Flow Diagram*

Langkah-langkah di dalam membuat *data flow diagram* dibagi menjadi 3 (tiga) tahap atau tingkat konstruksi DFD, yaitu sebagai berikut:

1) DIAGRAM KONTEKS

Diagram ini dibuat untuk menggambarkan sumber serta tujuan data yang akan diproses atau dengan kata lain diagram tersebut digunakan untuk menggambarkan sistem secara umum/global dari keseluruhan system yang ada.

2) DIAGRAM NOL

Diagram ini dibuat untuk menggambarkan tahapan proses yang ada di dalam diagram konteks, yang penjabarannya lebih terperinci.

3) DIAGRAM DETAIL


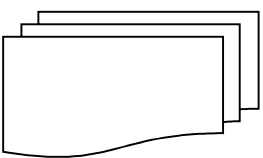
Diagram ini dibuat untuk menggambarkan arus data secara lebih mendetail lagi dari tahapan proses yang ada di dalam diagram nol.

2.2.2. Pengertian *Blockchart*

Kristanto (2008:68), “*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”.

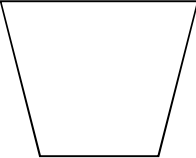

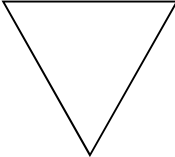
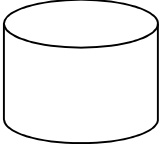

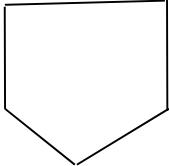
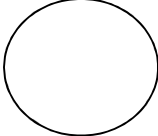
Kristanto (2008:75), “Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.2 Simbol-simbol dalam *Blockchart*

No.	Simbol	Keterangan
1.		Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan.
2.		Multi dokumen.


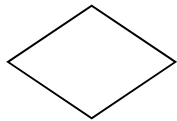

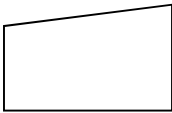


Lanjutan Tabel 2.2 Simbol-simbol dalam *Blockchart*

3.		Proses manual.
4.		Proses yang dilakukan oleh komputer.
5.		Menandakan dokumen yang diarsifkan(arsif manual)
6		Data penyimpanan(data storage)
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.



Lanjutan Tabel 2.2 Simbol-simbol dalam *Blockchart*

10		Terminasi yang menandakan awal dan akhir dari suatu aliran.
11		Pengambilan keputusan (<i>decision</i>).
12		Layar peraga (<i>monitor</i>).
13		Pemasukan data secara manual.

2.2.3. Pengertian ERD (*Entity Relational Diagram*)

Ladjamudin, “ERD (*Entity Relationship Diagram*) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”.

Ladjamudin, menjelaskan tentang elemen-elemen Diagram Hubungan Entitas (*Entity Relationship Diagram*), yaitu:

1. Entity

Pada E-R diagram, entity digambarkan dengan sebuah bentuk persegi panjang. Entity adalah sesuatu apa saja yang ada di dalam sistem, nyata maupun abstrak dimana data tersimpan atau dimana terdapat data. Entitas diberi nama dengan kata benda dan dapat dikelompokkan dalam empat jenis nama, yaitu orang, benda, lokasi, kejadian (terdapat unsur waktu di dalamnya).

2. Relationship

Pada E-R diagram, Relationship dapat digambarkan dengan sebuah bentuk belah ketupat. Relationship adalah hubungan alamiah yang terjadi antara entitas.



3. Relationship Degree

Relationship Degree atau Derajat Relationship adalah jumlah entitas yang berpartisipasi dalam satu Relationship.

4. Atribut

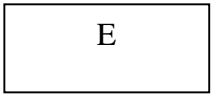
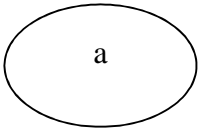
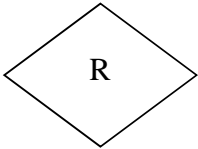

Secara umum atribut adalah sifat atau karakteristik dari tiap entitas maupun tiap Relationship. Maksudnya, atribut adalah sesuatu yang menjelaskan apa sebenarnya yang dimaksud entitas maupun Relationship, sehingga sering dikatakan atribut adalah elemen dari setiap entitas dan Relationship.

5. Kardinalitas.

Kardinalitas Relasi menunjukkan jumlah maksimum tupel yang dapat berelasi dengan entitas pada entitas yang lain.

Ladjamudin, menjelaskan notasi-notasi yang digunakan dalam ERD, yaitu:

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

No.	Simbol	Keterangan
1.		Himpunan Entitas/Entitas E, adalah Orang, tempat, atau benda dimana data akan dikumpulkan.
2.		Atribut a sebagai key, adalah properti dari entitas harus digunakan oleh minimal 1 proses bisnis dipecah dalam detail.
3.		Himpunan Relationship/Relasi R, Relationship, menunjukkan hubungan diantara sejumlah entitas.
4.		Link/Garis, sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.



2.2.4. Pengertian *Flowchart*

Ladjamudin, “*Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah”.

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

1. *Flow direction symbols*

Simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Berikut simbol *Flow direction symbols* seperti pada tabel dibawah ini:

Tabel 2.4 Simbol-simbol *Flow direction symbols*

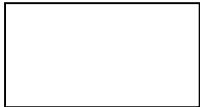
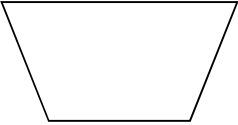
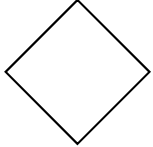

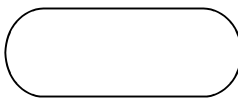
No.	Simbol	Nama Simbol	Keterangan
1.		Garis Alir (<i>Flow Line</i>)	Menyatakan jalannya arus suatu proses.
2		Communication Link	Digunakan untuk memberikan nilai awal pada suatu variabel atau counter.
3		Connector	Digunakan untuk menunjukkan hubungan arus proses yang terputus masih dalam halaman yang sama.
4		Offline Connector	Digunakan untuk menunjuk-kan hubungan arus proses yang terputus masih dalam halaman yang sama.



2. Processing symbols

Simbol yang menunjukkan jenis operasi pengolahan dalam suatu proses/prosedur. Berikut simbol *Processing symbols* seperti pada tabel dibawah ini:

Tabel 2.5 Simbol-simbol *Processing symbols*

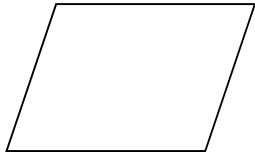

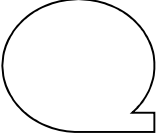


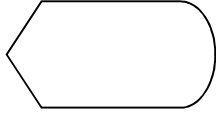
No.	Simbol	Nama Simbol	Keterangan
1.		<i>Process</i>	Menyatakan suatu tindakan (proses) yang dilakukan oleh komputer.
2.		<i>Manual</i>	Menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer.
3.		<i>Decision</i>	Menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban: ya/tidak.
4.		<i>Predefined Process</i>	Menyatakan penyediaan tempat penyimpanan suatu pengolahan untuk memberi harga awal.
5.		<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program.



3. *Input / Output symbols*

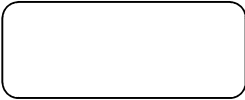
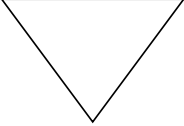

Simbol yang menunjukkan jenis peralatan yang digunakan sebagai media input atau output. Simbol-simbol tersebut adalah sebagai berikut:

Tabel 2.6 Simbol-simbol *Input / Output symbols*

No.	Simbol	Nama Simbol	Keterangan
1.		<i>Input/Output</i>	Menyatakan proses input atau output tanpa tergantung jenis peralatannya.
2.		<i>Punched Card</i>	Menyatakan input berasal dari kartu atau output ditulis ke kartu.
3.		<i>Magnetic Tape</i>	Menyatakan input berasal dari pita magnetis atau output disimpan ke pita magnetis.
4.		<i>Disk Storage</i>	Menyatakan input berasal dari disk atau output disimpan ke disk.
5.		<i>Document</i>	Digunakan untuk mencetak keluaran dalam bentuk dokumen (melalui printer).
6.		<i>Display</i>	Digunakan untuk mencetak keluaran dalam layar monitor.



Lanjutan Tabel 2.6 Simbol-simbol *Input / Output symbols*

6.		Keying Operation	Menyatakan segala jenis operasi yang diproses dengan menggunakan suatu mesin yang mempunyai keyboard.
7.		Offline Storage	Menunjukkan bahwa data dalam simbol ini akan disimpan ke suatu media tertentu.
8.		Manual Input	Memasukkan data secara manual dengan menggunakan online keyboard.

2.2.5. Pengertian Kamus Data

Kristanto (2008:66), menjelaskan "Kamus data adalah kumpulan elemen-elemen atau simbol-simbol yang digunakan untuk membantu dalam penggambaran atau pengidentifikasian setiap *field* atau *file* di dalam sistem".

Table 2.7 Simbol-simbol dalam Kamus Data

No	Notasi	Arti
1	=	Terdiri atas
2	+	AND atau dan
3	()	Opsional
4	[]	Memilih salah satu alternative
5	**	Komentar
6	@	identifikasi atribut kunci
7		Pemisah alternatif simbol []



2.3. Teori Program

2.3.1. Basis Data (*Database*)

Kristanto (2008:73) Database atau sering juga disebut “Basis data adalah sekumpulan informasi yang disimpan dalam komputer secara sistematis dan merupakan sumber informasi yang dapat diperiksa menggunakan suatu program komputer. Database berfungsi untuk menyimpan informasi atau data. Untuk mengenal database diperlukan *software* yang sering disebut dengan DBMS (*Database Management System*). Dengan *DBMS* pengguna atau *user* dapat membuat, mengolah, mengontrol, dan mengakses database dengan mudah, praktis dan efisien”. Ada beberapa istilah yang akan dipakai dalam database yaitu:

1. *Entity*, adalah konsep informasi yang direkam, meliputi orang, kejadian dan tempat.
2. *Field* (atribut), adalah sesuatu yang mewakili *entity*, misalnya untuk mahasiswa atributnya adalah NIM, nama, alamat dan sebagainya.
3. *Data Value*, adalah informasi yang tersimpan dalam setiap atribut.
4. *Record*, adalah kumpulan atribut yang saling berkaitan satu sama lain dan menginformasikan suatu *entity* secara lengkap.
5. *File*, adalah kumpulan *record* yang mempunyai panjang atribut yang sama tetapi berbeda dengan *data value*.
6. *Database*, adalah kumpulan *file* satu dengan *file* lainnya yang membentuk suatu informasi sistem secara keseluruhan.

2.3.2. Pengertian *Web Server*

Sibero (2013:11), *Web Server* adalah sebuah computer yang terdiri dari perangkat keras dan perangkat lunak.

Madcoms (2011:229), “*web server* adalah suatu program komputer yang mempunyai tanggung jawab atau tugas menerima permintaan HTTP (*HyperText Transfer Protocol*) dari komputer klien, yang dikenal dengan nama *web browser*, dan melayani mereka dengan menyediakan respon HTTP berupa konten data,



biasanya berupa halaman web yang terdiri dari dokumen HTML (*HyperText Markup Language*), dan objek yang terkait seperti gambar, suara, dan lain-lain.”

2.3.3. Pengertian XAMPP

Latief (2010:8), “XAMPP adalah perangkat lunak yang disediakan secara gratis, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Untuk menjalankan fungsinya sebagai *server* yang berdiri sendiri, yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penterjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. XAMPP adalah nama yang merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP*, dan *Perl*. Program ini tersedia dalam *GNU “General Public License”* dan bebas, merupakan *web server* yang mudah digunakan yang mampu melayani halaman dinamis.

Berikut adalah gambar tampilan awal XAMPP:



Gambar 2.1. Gambar Tampilan Awal XAMPP

2.3.4. Pengertian PHP (*Hypertext Preprocessor*)

Madcoms (2011:228), PHP merupakan singkatan dari *Hypertext Preprocessor*. Pada awalnya PHP merupakan kependekatan dari *Personal Home Page* (situs personal) dan PHP itu sendiri pertama kali dibuat oleh Rasmus



Lerdorf pada tahun 1995, dan pada saat PHP masih bernama FI (*Form Interpreter*), yang wujudnya berupa sekumpulan *script* yang digunakan untuk mengolah data *form* dari *web*. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum. PHP adalah sebuah bahasa *scripting* yang terpasang pada HTML. Sebagian besar sintaksnya mirip dengan bahasa pemrograman C, Java, asp dan Perl, ditambah beberapa fungsi PHP yang spesifik mudah dimengerti.

PHP digunakan untuk membuat tampilan *web* menjadi lebih dinamis. PHP bisa menampilkan atau menjalankan beberapa *file* dalam 1 *file* dengan cara di-*include* atau *require*. PHP itu sendiri sudah dapat berinteraksi dengan beberapa database walaupun dengan kelengkapan yang berbeda, yaitu seperti: DBM, FilePro (Personic, Inc), Informix, Ingres, InterBase, Microsoft Access, MSSQL, MySQL, Oracle, PostgreSQL, dan Sybase.

Dari uraian di atas maka dapat diambil 4 point utama tentang PHP:

- a. PHP adalah singkatan dari *Hypertext Preprocessor*.
- b. PHP adalah bahasa *scripting server-side*, artinya di jalankan di *server*, kemudian *outputnya* dikirim ke *client (browser)*.
- c. PHP digunakan untuk membuat aplikasi web.
- d. PHP mendukung banyak database (MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC, dll).

2.3.5. Cara Kerja PHP

Madcoms (2011:228), cara kerja PHP yaitu, pertama *client web browser* atau pengguna memakai komputer kemudian pengguna tersebut menjalankan *file* PHP itu di *web browser*, kemudian *file* PHP itu dikirim ke *web server*. *Web server* mengirimkannya lagi ke *Engine PHP* atau mesin PHP dan di dalam mesin PHP itu diproses, dan setelah diproses oleh mesin PHP maka akan berbentuk *file* HTML, dan *file* HTML ini akan dikirimkan ke *web server* dan *web server* akan memberikan ke pengguna.



2.3.6. Mengenal Tipe Data dan Variabel

Madcoms (2011:241), beberapa hal yang harus anda ketahui agar dapat memahami *script-script* PHP adalah tipe data dan variabel.

1. Mengenal Tipe Data

Tipe data dalam *script* PHP akan otomatis dikonversi sesuai karakternya tanpa harus menggunakan sebuah variabel. Berikut ini penjelasan dari beberapa tipe data dalam *script* PHP, yaitu:

a. Tipe data *String*

Dalam penulisannya, tipe data *String* menggunakan tanda kutip tunggal (‘ ’) atau menggunakan tanda kutip ganda (“ ”).

```
$data_string='Mengenal Tipe Data dalam PHP';
```

```
$data_string="Mengenal Tipe Data dalam PHP";
```

Perbedaan antara penggunaan dua kutip tersebut adalah pada saat penggunaan variabel. Jika menggunakan tanda petik tunggal, maka apabila sebuah variabel berisi data *string* dan berisi variabel yang lain, yang terjadi adalah nilai dari variabel tersebut akan dibaca atau akan tetap dicetak nama variabel itu sendiri.

b. Tipe data *Integer*

Yang termasuk tipe data *integer* adalah semua data yang berisikan bilangan bulat dan besarnya *range* sama dengan data pada bahasa C, yaitu antara -2,147,483,648 sampai +2, 147,483,647 pada *platform* 32 bit. Apabila data diluar kisaran tersebut maka PHP akan secara otomatis mengkonversikan data tersebut dari tipe *integer* menjadi tipe *Floating float*

c. Tipe data *Floating Float*

Yaitu tipe data yang berisi bilangan pecahan atau bilangan desimal. Kisaran data *floating* adalah antara 1.7E-308 sampai 1.7E+308. Data ini berbentuk desimal atau bentuk pangkat. Perhatikan contoh berikut:

```
$des=0.5;
```

```
$des1=5.0e-4;
```



d. Tipe data *Array*

Tipe data ini disebut juga data bertingkat, yaitu data yang mengandung beberapa data di dalamnya dan di-*index* atau dibaca berdasarkan data numerik atau *string*.

```
$data[ ]=20;
```

```
$data[ ]="Teknologi";
```

```
$data[ ]='Pemrograman';
```

e. Tipe data *Object*

Tipe data *Object* dapat berupa bilangan, variabel, ataupun fungsi. Tipe data ini dapat membantu *programmer* untuk membuat sebuah program. Data ini dapat disertakan ke dalam program, sehingga meringkas beberapa fungsi dan memperkecil ukuran *file*. Semakin kecil ukuran *file* semakin singkat waktu yang dibutuhkan untuk mengakses *file*.

f. Menenal Variabel

Variabel adalah tempat untuk menyimpan data. Data yang disimpan sewaktu-waktu bias saja anda panggil atau diganti dengan data lain. Mendefinisikan variabel cukup dengan sebuah kata (*akan menjadi nama variabel*) yang diawali dengan tanda \$, kemudian isikan nilai dari variabel tersebut.

Penulisan variabel ada aturannya namun cukup sederhana, yaitu gunakan kata apa saja yang didahului oleh huruf dan diberi tanda \$ di depan kata, tidak menggunakan spasi, tidak diawali dengan angka, dan tidak menggunakan karakter khusus (seperti: *, &, %, 3, dan lain-lain). Berikut ini 2 contoh variabel:

```
$file="hitung.txt";
```

```
$data_1=10;
```



2.3.7. Pengertian MySQL

Madcoms (2011:260), “Jenis database yang sangat populer dan digunakan pada banyak *website* di internet sebagai bank data adalah MySQL. MySQL menggunakan SQL dan bersifat gratis, selain itu MySQL dapat berjalan di berbagai *platform*, antara lain *Linux*, *Windows*, dan sebagainya.”

2.3.8. Dasar MySQL

2.3.8.1. Membuat Database Baru

Fungsi untuk membuat database :

- a) Fungsi yang digunakan untuk membuat database ini adalah

mysql_create_db().

Bentuk :

```
mysql_create_db(nama_database);
```

Keterangan :

Nama_database adalah string yang berisikan teks yang akan menjadi nama dari database.

Perhatikan contoh penulisannya sebagai berikut:

```
<?
//melakukan koneksi database
$koneksi=mysql_connect(“localhost”,”root”,”12345”);
//membuat database baru
$hasil=mysql_create_db(“smatiga”);
?>
```

2.3.8.2. Membuat Tabel

Database yang telah anda buat harus diisi dengan tabel untuk menampung data-data di dalamnya. Untuk membuat sebuah tabel, dapat menggunakan fungsi atau menggunakan jendela phpMyAdmin.

PHP tidak memiliki fungsi khusus untuk menangani field-field, sehingga untuk membuat tabel ini tetap menggunakan bentuk penulisan dari program



database itu sendiri. Untuk membuat sebuah tabel dapat dioperasikan dengan menggunakan fungsi **mysql_query()**.

Bentuk :

Mysql_Query(Variabel_Hasil);

Keterangan :

Variabel_hasil adalah variable yang berisi nilai hasil dari pembacaan data dalam database.

2.3.9. Pengertian Adobe Dreamweaver CS6

Madcoms (2011:3), “Adobe Dreamweaver CS6 adalah *software* terkemuka untuk membangun dan mengedit web dengan menyediakan kemampuan visual dan tingkat kode, yang dapat digunakan untuk membuat *website* berbasis standar dan desain untuk *desktop, mobile, smartphone, tablet*, dan perangkat lainnya”.

2.3.10. Fitur dalam Dreamweaver CS6

Madcoms (2011:4), Dreamweaver masih merupakan alat yang sangat ampuh untuk web *designer* profesional. Fitur baru yang ada di Dreamweaver CS6 terfokus untuk membantu mengembangkan sendiri proyek yang berbasis revolusi *multiscreen* atau beberapa tampilan. Dengan fasilitas *multiscreen* tentunya didukung dengan tambahan fitur baru juga yaitu JQuery Mobile yang memungkinkan membuat aplikasi Web untuk ponsel.

Berikut ini adalah beberapa fitur yang dapat ditemukan dalam Dreamweaver CS6:

a. *Layout fluid grid*

Anda dapat membuat desain web lintas *platform* dan lintas *browser* yang kompatibel menggunakan CSS3 berbasis sistem *layout fluid grid*. Pekerjaan akan lebih cepat dan lebih efisien seperti saat anda mengembangkan proyek-proyek yang menggunakan kode standar industri untuk berbagai macam perangkat dan komputer. Selain itu anda juga dapat membangun



desain web yang kompleks secara visual dan mengolah *layout* halaman tanpa harus berkuat dalam kode-kode yang memusingkan.

b. Integrasi dengan Adobe Bisnis *Catalyst*

Gunakan panel *Bisnis Catalyst Integration* dalam Dreamweaver untuk menghubungkan dan mengedit situs anda dengan menggunakan Adobe Bisnis *Catalyst* (tersedia secara terpisah). Anda dapat membangun situs *e-commerce* dengan solusi *host*.

c. Peningkatan *Support JQuery Mobile*

Digunakan untuk membangun aplikasi asli *mobile* atau ponsel pada *platform* IOS dan Android dengan menggunakan *support* yang telah diperbarui pada *JQuery Mobile*, anda dapat membangun aplikasi untuk menjangkau *audiens* ponsel sementara alur kerja pengembangan *mobile* anda akan lebih disederhanakan.

Anda dapat menambahkan pengembangan interaktivitas dengan menggunakan pengkodean *JQuery*, *JQuery* adalah standar industri di dalam *Library JavaScript*, sehingga mudah untuk menambahkan berbagai macam interaktivitas ke halaman web. Anda dapat dengan segera membuat halaman web *JQuery* dengan menggunakan *template* awal yang telah disediakan untuk ponsel. Gunakan juga *query* media untuk menulis dan membuat *style* terpisah untuk masing-masing perangkat tersebut.

d. *Support PhoneGap* diperbarui

Anda dapat membangun dan mengemas aplikasi asli untuk Android dan IOS dengan fungsi *PhoneGap* yang baru. Konvert HTML yang sudah pernah anda buat atau miliki menjadi aplikasi ponsel di dalam Dreamweaver, yaitu dengan menggunakan kerangka kerja (*framework*) *open source PhoneGap*.

Dengan diperbaruinya *support* untuk *Adobe PhoneGap* dibandingkan dengan versi sebelumnya, membuat Adobe Dreamweaver CS6 jauh lebih mudah untuk membangun dan membuat paket aplikasi asli untuk Android dan IOS. Anda dapat membuat aplikasi *mobile* dengan merencanakan atau merancang



kode HTML yang ada. Gunakan *emulator PhoneGap* untuk memeriksa desain anda.

e. Transisi CSS3 dan HTML5

Banyak sekali ditambahkan perubahan pada property CSS sebagai transisi untuk membawa desain web menjadi lebih hidup. Menjaga kontrol yang lebih besar pada desain web dengan mensiasati elemen halaman dan membuat efek yang lebih menawan.

Penggunaan *style* dalam panel CSS Dreamweaver CS6 telah diperbarui sehingga mendukung untuk CSS3. Pada tampilan *design* juga telah didukung oleh permintaan media, sehingga dapat menerapkan *style* atau gaya yang berbeda-beda saat anda menyesuaikannya dengan dimensi layar.

Dreamweaver CS6 juga mendukung HTML5 yang merupakan kode untuk masa depan yang mengisyaratkan dukungan dan perancangan (*rendering*) tampilan *design*.

Pada jenis tampilan *Live View* sekarang juga mendukung untuk `<video>` (dengan menggunakan media *QuickTime*) dan tag `<svg>`.

f. Fitur *Live View* diperbarui

Langkah pengujian halaman sebelum di-*publish* menggunakan fungsi *Live View* juga diperbarui. *Live View* pada Adobe Dreamweaver CS6 menggunakan versi terbaru dari mesin *rendering WebKit* untuk meningkatkan dukungan yang paling baik dalam HTML5.

Preview desain anda untuk beberapa perangkat juga didukung dengan *query media real-time*. Selain itu, kemampuan dan perangkat tambahan HTML5 dan CSS3 juga membuat langkah membangun aplikasi untuk perangkat *mobile* anda lebih efisien.

g. *Panel Multiscreen Preview* diperbarui

Anda dapat memeriksa tampilan layar dari proyek yang sedang dibangun untuk *smartphone*, *tablet*, dan *desktop* dengan menggunakan *Panel Multiscreen Preview* yang telah diperbarui. Panel tersebut telah ditingkatkan



kemampuannya sehingga memungkinkan anda untuk memeriksa secara acak konten HTML5.

Dengan dukungan *query media*, maka anda sebagai pengembang dapat mengatur *style* dan memvisualisasikan tampilan web anda secara acak untuk berbagai perangkat dalam satu panel.

Selain beberapa fitur utama di atas yang menjadi keunggulan Adobe Dreamweaver CS6, beberapa fitur berikut juga dapat anda temukan:

a. Integrasi *Adobe BrowserLab*

Dengan menggunakan *Adobe BrowserLab* maka anda dapat melihat *Preview* halaman web dinamis dan muatan lokal dengan tampilan ganda, diagnostik, dan sebagai alat perbandingan.

Dreamweaver terintegrasi dengan *Adobe BrowserLab*, yaitu sebuah layanan *Adobe CS Live online* yang secara akurat menguji konten web di seluruh *browser* dan sistem operasi.

b. Mendukung Integrasi CMS

Anda dapat menikmati fasilitas dukungan (*support*) untuk membuat dan menguji beberapa kerangka *Content Manajemen System (CMS)* seperti *Wordpress*, *Joomla!*, dan *Drupal*. *Fitur Dinamically Related Files* menyediakan akses langsung ke *file-file* terkait sebuah halaman, bahkan untuk halaman dinamis, dan *Live View Navigation* menawarkan *preview* yang akurat dari aplikasi yang dinamis.

c. Isyarat kode yang lebih spesifik

Dengan memanfaatkan isyarat kode pada *file* dan direktori non-standar di dalam Dreamweaver CS6, maka memungkinkan untuk ditingkatkannya dukungan isyarat pada *Library PHP* dan *framework CMS* pihak ketiga seperti *Wordpress*, *Drupal*, dan *Joomla!*.

d. Integrasi *Adobe Creative Suite*

Anda dapat menghemat waktu dan mengurangi jumlah langkah yang diperlukan untuk menyelesaikan proyek dengan integrasi cerdas di *Adobe Flash*



Professional, Fireworks, Photoshop Extended, dan layanan online *Adobe CS Live*.

- e. Komunitas Dreamweaver yang semakin diperluas
Anda dapat belajar dan berbagi dengan komunitas Dreamweaver dengan lebih luas lagi. Anda dapat mengambil banyak manfaat dari *Adobe Design Center* dan *Adobe Developer Connection* secara online, akan banyak pelatihan dan seminar, program sertifikasi pengembang, dan forum pengguna.
- f. Didukung oleh teknologi terkemuka
Dengan menggunakan Dreamweaver CS6, anda dapat merancang dan membangun web di dalam lingkungan kerja yang mendukung teknologi pengembangan web yang paling terkemuka, termasuk HTML, XHTML, CSS, XML, *JavaScript*, Ajax, PHP, perangkat lunak *Adobe ColdFusion*, dan ASP.
- g. Selalu menjadi yang terdepan dengan W3C validasi
Anda dapat menguji halaman web dengan lebih akurat pada *browser* dan sistem operasi yang berbeda, dengan adanya integrasi *Adobe BrowserLab* yang memberikan kode yang bersih dan dipadu dengan validasi secara *live* oleh *World Wide Web Consortium (W3C)*.
Gunakan layanan validasi online milik W3C untuk memastikan keakuratan desain web berbasis standar anda.
- h. Mendukung *Subversion*
- i. Simpan *file* anda lebih aman dengan dukungan untuk perangkat lunak *Subversion*, yang telah diperbarui untuk versi protocol 1,7.
- j. Inspeksi/ pengecekan CSS dan dukungn CSS yang komprehensif
- k. Dreamweaver CS6 merancang dan mengembangkan *website* dengan perangkat CSS yang sangat kuat. Secara visual menampilkan model kotak CSS tanpa memerlukan utilitas terpisah, dan mengurangi kebutuhan untuk manual mengedit kode CSS bahkan dalam *style sheet external*.
- l. Bantuan pengkodean yang cerdas dan isyarat kode *custom class PHP*



Anda dapat menuliskan kode dengan lebih jelas dan percaya diri, yaitu dengan memanfaatkan isyarat kode untuk *HTML*, *JavaScript*, dan *framework Ajax* seperti *Spry*, *JQuery*, dan *Ptototype*.

Anda juga dapat menambah wawasan secara langsung ke berbagai fungsi inti PHP, metode, dan objek dengan memanfaatkan isyarat kode PHP dinamis. Dengan langkah tersebut maka anda dapat membangun dan memelihara aplikasi PHP.

m. *Setup* situs yang sederhana

Di dalam Dreamweaver CS6 anda dapat membuat situs dengan mudah menggunakan kotak dialog *Site Definition Dreamweaver*. Tambahkan beberapa *server* dengan nama-nama baru untuk memanfaatkan bagian utilitas, jaringan, atau jenis situs lainnya.

n. *CSS Starter pages*

Anda dapat langsung mencoba untuk membangun desain *website* anda yang berbasis standar dengan *layout* halaman *CSS Starter* yang telah di-*update* dan disederhanakan. Dreamweaver CS6 menyediakan pilihan *template* yang lebih luas dengan menggabungkan praktik terbaik CSS dan dengan *selector* dan aturan yang lebih mudah dipahami.

o. Terintegrasi dengan konten FLV

Tambahkan *file* FLV ke halaman web dengan kemudahan *point-and-click* dan kode standar yang sesuai. Tampilkan *preview film* anda dengan kemampuan *playback* FLV di tampilan *Live View*.



2.4.11. Tampilan Awal Dreamweaver CS6

Jalankan Adobe Dreamweaver CS6 dengan cara klik tombol **Start > All Programs > Adobe Dreamweaver CS6**. Berikutnya akan tampil *Welcome Screen Dreamweaver CS6* seperti gambar berikut:



Sumber: Madcoms (2011:10)

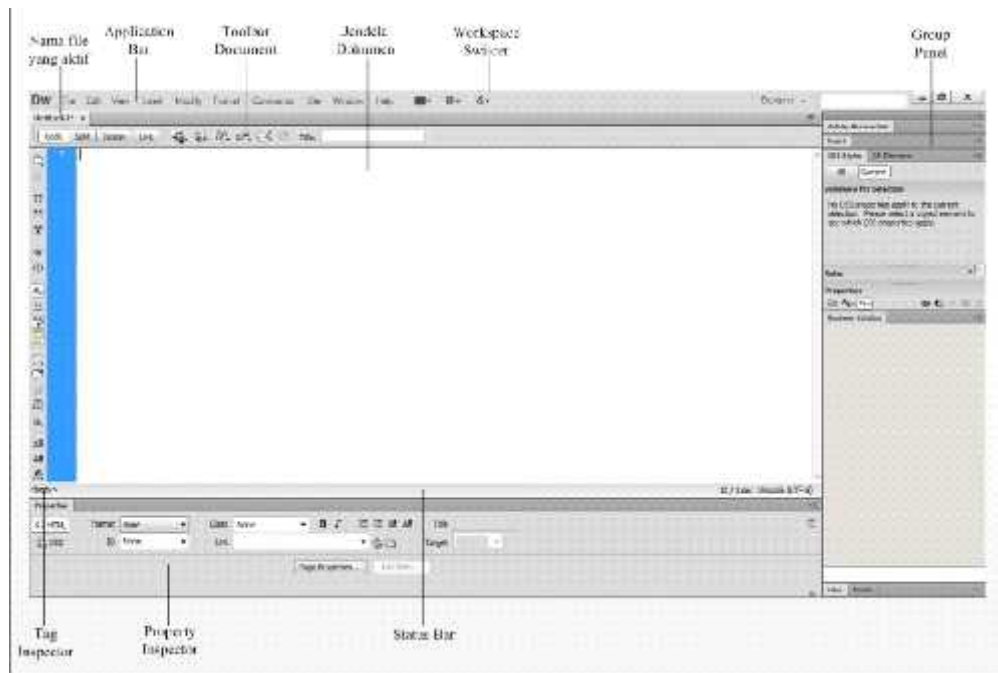
Gambar 2.2. Tampilan Awal Lembar Kerja Dreamweaver CS6

Welcome Screen Dreamweaver terdapat beberapa bagian yang sangat berguna untuk pekerjaan anda, yaitu sebagai berikut:

1. *Open a Recent Items*: bagian ini menampilkan *file* yang pernah terbuka sebelumnya dalam lembar kerja Dreamweaver CS6 anda. Klik ikon *Open* untuk membuka *file* lain.
2. *Create New*: bagian ini menampilkan beberapa pilihan jenis lembar kerja baru yang akan dibuka dalam Dreamweaver CS6, sebagai contoh jika ingin membuat *file* PHP baru, maka klik pilihan **PHP** dan sebagainya. Selain itu juga dapat digunakan untuk membuat *site* baru dan konfigurasinya dengan menggunakan pilihan **Dreamweaver Site**. Sedangkan untuk membuat lembar kerja baru dengan pilihan yang lebih banyak, klik pilihan **More**.



Standar lembar kerja yang nantinya akan anda gunakan saat bekerja dengan Adobe Dreamweaver CS6 adalah pilihan HTML, dengan tampilan seperti gambar berikut ini:



Sumber: Madcoms (2011:11)

Gambar 2.3. Tampilan Lembar Kerja Baru dengan Pilihan HTML

Top Features (Videos): bagian ini menampilkan fitur-fitur teratas atau terpopuler dalam Dreamweaver CS6 yang dapat dilihat dalam bentuk tampilan video. Klik pada salah satu daftar untuk melihat videonya dan terhubung langsung secara online ke www.adobe.com. *Getting Started*: bagian ini berisi link untuk mengikuti tuntunan penggunaan Dreamweaver CS6.



2.4.12. Alur Kerja di dalam Dreamweaver CS6

Madcoms (2011:29), sebuah proyek web yang kompleks biasanya memiliki alur kerja yang sistematis agar yang tercapai dapat optimal. Berikut ini adalah alur kerja pembuatan sebuah *website* dengan menggunakan Dreamweaver.

1. Membuat rancangan dan mencari lokasi situs

Tentukanlah kemana *file* akan diletakkan serta lakukan pengujian kebutuhan lokasi situs. Pertimbangkan juga kebutuhan teknis dari situs tersebut, seperti akses pemakai, *browser* yang digunakan, kebutuhan *plug-in*, serta pengaturan *download*. Setelah proses organisir dan pembuatan struktur, biasanya anda dapat menentukan lokasi situs.

2. Melakukan organisasi dan manajemen *file* situs

Di dalam panel *files*, anda dapat dengan mudah menambahkan, menghapus, dan mengganti nama *file* atau folder yang diperlukan. Panel *files* juga memiliki banyak *tool* untuk memajemen situs anda, antara lain memindahkan *file* menuju dan dari suatu *server remote*, mengatur proses *Check In* atau *Check out* untuk mencegah *file* yang berlebihan, dan menyamakan *file* pada *server remote* dan lokal.

Dengan panel *asset*, anda juga dapat dengan mudah mengorganisir *asset* di dalam suatu lokasi.

3. Membuat *layout* halaman web

Tambahkan *asset* dan unsur-unsur desain halaman web anda, seperti teks, *image*, *rollover image*, warna, *movie*, suara, *link HTML*, *jump menu*, dan sebagainya. Lakukan juga pemformatan seperti memberi judul atau mengganti *background* halaman. Dreamweaver juga menyediakan fasilitas untuk memaksimalkan hasil *website*, dan melakukan pengujian halaman untuk memastikan kecocokan dengan *browser* web yang berbeda.

4. Membuat halaman dengan kode program

Fasilitas kode merupakan langkah alternatif untuk mendesain sebuah halaman web. Dreamweaver tidak hanya memberikan fasilitas untuk membuat dan



mengedit desain visual, namun juga menyediakan fasilitas kode-kode program yang canggih.

5. Mengatur web menjadi aplikasi dengan isi yang dinamis

Banyak sekali halaman web yang mempersilahkan pengunjung untuk dapat melihat dan mengisi data dalam databasenya. Selain itu pengunjung juga diijinkan untuk menambah dan mengubah isi dari database. Untuk tujuan ini anda harus melakukan *setting web server*, juga memodifikasi halaman web sehingga terkoneksi dengan database.

6. Menciptakan halaman dinamis

Dalam dreamweaver, anda dapat menampilkan beberapa tipe isi yang dinamis, diantaranya menampilkan data *record* dari sebuah database, membuat *form* batasan, dan menggunakan komponen JavaBeans. Untuk menambahkan isi yang dinamis, anda cukup klik tahan geser ke dalam halaman web.

7. Menguji situs dan melakukan *publish*

Pengujian halaman yang telah anda buat, merupakan suatu keharusan dalam sebuah siklus pembuatan sebuah situs. Dan pada akhir siklus, maka anda harus melakukan *publish* untuk situs tersebut. Sebaiknya anda memiliki jadwal secara periodik untuk menguji apakah web anda tersebut masih dapat berfungsi dan berjalan dengan baik.