

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Pengertian dan definisi sistem pada berbagai bidang berbeda-beda, tetapi meskipun istilah sistem yang digunakan bervariasi, semua sistem pada bidang-bidang tersebut mempunyai beberapa persyaratan umum, yaitu sistem harus mempunyai elemen, lingkungan, interaksi antar elemen, interaksi antara elemen dengan lingkungannya, dan yang terpenting adalah sistem harus mempunyai tujuan yang akan dicapai.

Berdasarkan persyaratan ini, sistem dapat didefinisikan sebagai seperangkat elemen yang digabungkan satu dengan lainnya untuk suatu tujuan bersama. Kumpulan elemen terdiri dari manusia, mesin, prosedur, dokumen, data atau elemen lain yang terorganisir dari elemen-elemen tersebut. Elemen sistem selain berhubungan satu sama lain, berhubungan juga dengan lingkungannya untuk mencapai tujuan yang telah ditentukan sebelumnya. Terdapat beberapa definisi sistem yaitu : [BET99]

“Sebuah sistem terdiri dari bagian-bagian saling berkaitan yang beroperasi bersama untuk mencapai beberapa sasaran atau maksud”.
(Gordon B. Davis : 1984)

“Sistem adalah himpunan dari unsur-unsur yang saling berkaitan sehingga membentuk suatu kesatuan yang utuh dan terpadu”. (Ramond Mcleod 2001).

2.2 Konsep Dasar Informasi

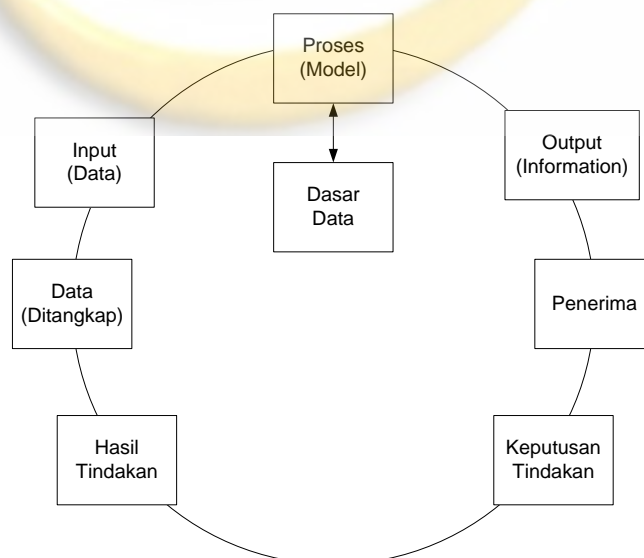
Pengertian informasi secara umum dapat didefinisikan sebagai hasil dari pengolahan data dalam suatu bentuk yang lebih berguna dan lebih berarti bagi penerimanya yang menggambarkan suatu kejadian-kejadian yang nyata yang digunakan untuk pengambilan keputusan. Sedangkan informasi menurut Raymond Mcleod, :

“Informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi si penerima dan bermanfaat bagi pengambilan keputusan saat ini atau mendatang”. (Ramond Mcleod 2001).

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu. Di dalam dunia bisnis, kejadian-kejadian yang sering terjadi adalah transaksi perubahan dari suatu nilai yang disebut transaksi. Kesatuan nyata adalah berupa suatu obyek nyata seperti tempat, benda dan orang yang betul-betul ada dan terjadi.

Data merupakan bentuk yang masih mentah, belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu metode untuk menghasilkan informasi. Data dapat berbentuk simbol-simbol semacam huruf, angka, bentuk suara, sinyal, gambar, dsb.

Data yang diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus informasi ini dapat digambarkan sebagai berikut : [FAI12]



Gambar 2.1. Siklus Informasi (Ramond Mcleod 2001)

2.3 Konsep Dasar Sistem Informasi

Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambilan keputusan. Pengertian sistem informasi menurut McLeod :

“Sistem Informasi merupakan sistem yang mempunyai kemampuan untuk mengumpulkan informasi dari semua sumber dan menggunakan berbagai media untuk menampilkan informasi”.

2.4 Konsep Database

2.4.1 Pengantar Database

Basis data yaitu representasi dari fakta dunia yang mewakili suatu obyek yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya. Pengertian basis data menurut Chou:

“Basis data dapat dianggap tempat untuk sekumpulan berkas data terkomputerisasi, dengan tujuan utama memelihara informasi dan membuat informasi tersebut tersedia saat dibutuhkan”.

2.4.2 Manfaat Manajemen Database

Manajemen *database* adalah alat bantu bagi para pemrogram untuk menalar dan men-*capture* realitas dunia bisnis menjadi satu sistem informasi yang terstruktur dan terkelola. Secara lebih spesifik manfaat dari manajemen *database* adalah : [KAD99]

1. Mempermudah akses data

Untuk dapat mengakses data yang banyak dengan jumlah transaksi yang sering digunakan, maka diperlukan pengelolaan data berbasis manajemen

database yang dilengkapi program aplikasi yang mampu mengakses data yang diperlukan secara mudah dan akurat.

2. Mengurangi redundansi dan inkonsistensi data

Sistem manajemen *database* dibangun berdasarkan *database relational* yang saling berhubungan antara tabel satu dengan yang lain dengan menggunakan satu *field* yang disebut *field* kunci, adanya *field* kunci ini dapat mengatasi adanya redundansi data dan inkonsistensi data.

3. Mengisolasi data untuk distansarisasikan

Sistem manajemen *database* mampu menstrukturkan dan menyeragamkan data.

4. Mempermudah penggunaan

5. Mengamankan data

Sistem manajemen *database* akan membangun sistem keamanan data untuk membatasi akses pemakai.

6. Menjaga Integritas data

7. Menciptakan independensi data

2.5 Pengantar Dasar Web

2.5.1 Pengertian Download

Download adalah proses transmisi sebuah *file* dari sebuah sistem komputer ke sistem komputer yang lainnya. Dari internet, *user* yang melakukan proses *download* adalah proses dimana seorang user meminta / *request* sebuah *file* dari sebuah komputer lain (*website*, *server* atau yang lainnya) dan menerimanya. Dengan kata lain, *download* adalah transmisi data dari internet ke komputer *client*/pemakai.

Arti istilah *download* dianggap berkaitan erat dengan menerima *file* dari komputer lain ke komputer Anda. Kegiatan penerimaan data (berupa *file*) dari komputer lainnya ke komputer lokal yang terhubung dalam sebuah *network*.

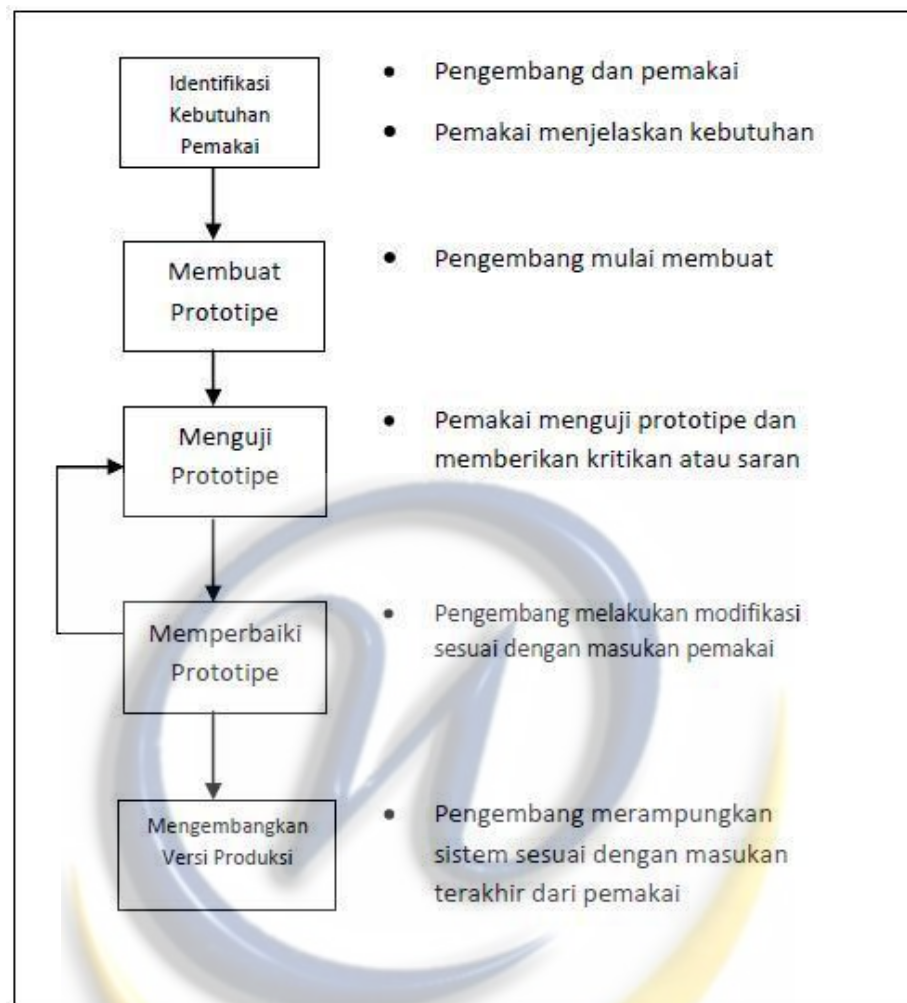
2.5.2 Pengertian Upload

Upload adalah suatu proses transmisi sebuah *file* dari sebuah sistem komputer ke sistem komputer lainnya dengan arah yang berlawanan dengan *download*. Dari internet, *user* yang melakukan proses *upload* adalah proses dimana *user* mengirimkan *file* ke komputer lain yang memintanya. *User* yang men-*share* gambar dan foto dengan *user* yang lainnya di *bulletin board system* (BBS), mengunggah file ke BBS. *File Transfer Protocol* (FTP) adalah contoh Internet protokol untuk proses *download* dan *upload file*.

Arti istilah *upload* dianggap berkaitan erat dengan mengirim *file* dari komputer Anda ke komputer lain. Kegiatan pengiriman data (berupa *file*) dari komputer lokal ke komputer lainnya yang terhubung dalam sebuah *network*. Secara umum *upload* dapat diartikan proses transmisi data dari komputer *client*/pemakai ke internet. [KAD02]

2.6 Metode Rekayasa Perangkat Lunak

Desain penelitian ini dimodelkan dengan menggunakan model proses *prototype*. Model proses *prototype* merupakan suatu metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat sesuatu program dengan cepat dan bertahap sehingga dapat segera dievaluasi oleh pemakai (*user*). Berikut gambar dari metode pengembangan yang digunakan penulis :



Gambar 2.2 Mekanisme Pengembangan Sistem dengan *Prototype*
(Sumber: Abdul Kadir. 2003)

Dari pengertian metode *prototype* diatas, penulis akan memberikan beberapa alasan mengapa penulis menggunakan metode pengembangan sistem dengan *prototype*, yaitu dikarenakan penulis akan lebih mudah dalam merancang sistem yang diinginkan dan dapat diterima oleh *user* sebagai pemakai, penulis menginginkan perancangan sistem yang telah dihasilkan kemudian dipresentasikan kepada *user* dan *user* diberikan kesempatan untuk memberikan masukan-masukan sehingga sistem informasi yang dihasilkan betul-betul sesuai dengan yang diinginkan.

Metode *prototype* dirancang agar dapat menerima perubahan-perubahan dalam rangka menyempurnakan *prototype* yang sudah ada sehingga pada akhirnya dapat menghasilkan sistem informasi yang dapat diterima dan memberikan gambaran bagaimana penggunaan sistem tersebut kepada pemakai setelah sistem tersebut disetujui. Berikut adalah langkah-langkah penulis dalam merancang sebuah sistem yang menggunakan mekanisme pengembangan sistem dengan *prototype*, langkah-langkah antara lain :


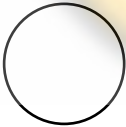
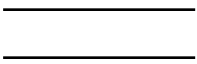
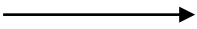
1. Penulis akan mengidentifikasi kebutuhan *User*, supaya penulis bisa merancang sistem yang akan dibangun sesuai dengan yang diharapkan *User*. Sebelum pada tahap perancangan, penulis menganalisis sistem dengan cara melakukan pengumpulan data yaitu dengan *research method* (metode penelitian)/ observasi, dan *interview* (wawancara) dan dengan cara *literature* yaitu dengan dokumentasi terhadap kebutuhan yang diinginkan pemakai.
2. Pada tahap kedua, penulis membuat *prototype* sistem tersebut untuk memperlihatkan kepada *user* model sistem yang akan dirancang.
3. Pada tahap ketiga, penulis melakukan uji coba sistem yang telah dirancang untuk memastikan bahwa sistem tersebut dapat digunakan dengan baik dan benar, sesuai kebutuhan pemakai.
4. Pada tahap keempat, penulis akan melakukan implementasi dan *maintenance*, guna menunjang performa maksimal terhadap kinerja sistem. Atau bahkan harus dilakukan beberapa perbaikan, dan setelah perbaikan sistem itu selesai dikerjakan, penulis akan kembali lagi pada tahap ketiga yaitu melakukan pengujian *prototype* kembali. [KAD03]

2.7 Diagram Alir Data (*Data Flow Diagram*)

Untuk menganalisa sistem yang dibuat dilakukan pemodelan dengan menggunakan *Data Flow Diagram (DFD)*. *DFD* adalah suatu teknik pemodelan secara grafis yang menggambarkan aliran data dalam sistem serta fungsi-fungsi (proses) yang terlibat dalam transformasi aliran data tersebut.

Selain dapat memberikan informasi tambahan yang digunakan selama tahap analisis, *DFD* juga digunakan untuk merepresentasikan sistem atau perangkat lunak pada berbagai tingkatan abstraksi. Artinya, *DFD* dapat dibagi menjadi beberapa level yang menggambarkan penambahan aliran informasi dan fungsionalitas yang lebih rinci. *DFD* level 0 (diagram konteks) merepresentasikan hubungan perangkat lunak/sistem dengan lingkungan yang berkaitan dengan sistem tersebut sebagai satu proses dengan data masukan dan keluaran digambarkan sebagai panah yang masuk dan keluar proses. Selanjutnya pada level yang lebih tinggi (1,2,..dst), proses tersebut dipecah-pecah untuk memperoleh aliran data dan proses yang lebih rinci. Komponen *Data Flow Diagram* menurut Yourdan dan DeMarco yaitu : [KAD03]

Tabel 2.1 Simbol-simbol pada Diagram Alir Data (*Data Flow Diagram*)

No.	Simbol	Keterangan
1.		<i>External entity</i> (kata benda), sebuah elemen sistem (misalnya perangkat keras, seseorang, program yang lain) atau sistem yang lain yang menghasilkan informasi bagi transformasi oleh perangkat lunak atau menerima informasi yang dihasilkan oleh PL.
2.		Proses (kata kerja), menggambarkan kegiatan yang dilakukan oleh sistem, dapat berupa prosedur yang memanipulasi atau mengolah data.
3.		<i>Data Store</i> (kata benda), merupakan tempat penyimpanan data untuk direferensi atau diolah lagi lebih lanjut.
4.		<i>Data Flow</i> (kata benda), menggambarkan aliran data yang menunjukkan transportasi data atau informasi.

2.8 Implementasi

2.8.1 PHP (*Hypertext Preprocessor*)

PHP yang merupakan singkatan dari *Hypertext Preprocessor*, adalah bahasa pemrograman yang mana filenya diletakkan di server dan seluruh prosesnya dikerjakan di server, kemudian hasilnya yang dikirimkan ke klien, tempat pemakai menggunakan *browser* (lebih dikenal dengan istilah *server-side scripting*). PHP bekerja didalam sebuah dokumen HTML (*Hypertext Markup Language*) untuk dapat menghasilkan isi dari sebuah halaman *web* sesuai permintaan. Dengan PHP, kita dapat merubah situs kita menjadi sebuah aplikasi berbasis *web*, tidak lagi hanya sekedar sekumpulan halaman statik, yang jarang diperbaharui. Menggunakan PHP Karena PHP bersifat tidak memiliki ketergantungan terhadap berbagai *platform*, jadi PHP dapat dijalankan dalam *platform* apapun, baik itu Unix, Windows ataupun Macintosh. Kelebihan lain dari PHP adalah kemudahan melakukan pengkodean, karena perintah-perintah PHP mirip dengan perintah-perintah C selain itu kemudahan dari PHP adalah dapat dengan mudah dihubungkan dengan aplikasi *database* (melakukan *query*), seperti MySQL dan PostgreSQL.

PHP bersifat *free* (bebas dipakai). Kita tidak perlu membayar apapun untuk menggunakan perangkat lunak ini. Kita dapat mendownload PHP melalui situs resminya yaitu www.php.net. Untuk versi Windows, kita dapat memperoleh kode binernya, dan untuk versi Linux, kita mendapatkan kode sumbernya secara lengkap.

Kelahiran PHP bemula saat Rasmus Lerdorf membuat sejumlah skrip Perl yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi *tool* yang disebut "*Personal Home Page*". Kumpulan *tool* inilah yang nantinya menjadi cikal-bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI versi 2. Pada versi inilah pemrogram dapat menempelkan kode terstruktur di dalam *tag* HTML. Yang menarik kode PHP juga bisa berkomunikasi dengan *database* dan melakukan perhitungan-perhitungan yang kompleks sambil jalan. Pada awalnya, PHP dirancang untuk diintegrasikan dengan *webserver* Apache. Namun

belakangan ini, PHP juga dapat bekerja dengan *webserver* seperti PWS (Personal Web Server), IIS (*Internet Information Server*) dan Xitami. Yang membedakan PHP dengan bahasa pemrograman lain adalah adanya *tag* penentu, yaitu diawali dengan “<?” atau “<?php” dan diakhiri dengan “?””. Jadi kita bebas menempatkan skrip PHP dimanapun dalam dokumen HTML yang telah kita buat.

Pada bulan Oktober 2000 PHP merilis versi 4.0 dengan mengintegrasikan Zend Engine, dan merupakan versi pengembangan dari PHP 3 *Script Engine*. Keunggulannya dari sifatnya yang *server-side* tersebut antara lain :

1. Tidak diperlukan kompatibilitas *browser* atau harus menggunakan *browser* tertentu, karena *server*-lah yang akan mengerjakan skrip PHP. Hasil yang dikirimkan kembali ke *browser* apapun.
2. Dapat memanfaatkan sumber-sumber aplikasi yang dimiliki oleh *server*, misalnya koneksi ke *database*.
3. Skrip tidak dapat “diintip” dengan menggunakan fasilitas view HTML source.

Adapun kelebihan lain dari PHP yaitu dapat “melakukan” semua aplikasi program CGI, seperti mengambil nilai *form*, menghasilkan halaman *web* yang dinamis, mengirim dan menerima *cookie*. PHP juga dapat berkomunikasi dengan layanan-layanan yang menggunakan protokol IMAP, SNMP, NNTP, POP3, HTTP, dan lain-lain.

Namun tampaknya kelebihan PHP yang paling signifikan adalah kemampuannya untuk melakukan koneksi dengan berbagai macam *database*. Saat ini, *database* yang didukung PHP adalah : Adabas D, InterBase, PostgreSQL, dBase, FrontBase, Solid, Empress, mSQL, Sybase, FilePro(read-only), Direct MS-SQL, Velocis, IBM DB2, MySQL, Unix dbm, Informix, Semua *database* yang mempunyai provider ODBC, Ingres, Oracle (OCI7 and OCI8). [RAS02]

2.8.2 MySQL

MySQL dikembangkan oleh sebuah perusahaan Swedia bernama MySQL AB, yang pada saat itu bernama TcX Data Konsult AB, sejak sekitar 1994 – 1995. MySQL versi 1.0 dirilis Mei 1996 secara terbatas kepada empat orang. Baru di bulan Oktober versi 3.11.0 dilepas ke publik. Versi pertama ini hanya berjalan di Linux dan Solaris serta sebagian besar masih belum terdokumentasi itu berangsur-angsur diperbaiki dan ditambah fitur demi fiturnya. Barulah di versi-versi akhir 3.22-sepanjang 1998-1999 MySQL menjadi semakin populer dan dilirik orang. Kalau di versi ini MySQL mulai diadopsi banyak orang dan meningkat jumlah penggunaanya, maka di versi 5.0.41-lah terjadi banyak peningkatan dari sisi teknologi.

MySQL adalah salah satu jenis *database server* yang sangat terkenal. MySQL menjadi sangat populer karena MySQL bersifat *free* (tidak perlu membayar dalam menggunakannya) pada berbagai *platform* (unix/windows). Untuk mendapatkan MySQL dapat di-download dari <http://www.MySQL.org> atau www.MySQL.com. MySQL termasuk jenis *RDBMS (Relational Database Management System)*. Pada MySQL, sebuah *database* mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom.

Pada saat instalasi, secara *default* MySQL akan membuat sebuah *database* bernama MySQL. Salah satu isi dari *database* ini adalah tabel *user*, tabel ini berisi nama dan *password user* yang dapat mengakses data pada *database* yang dibuat di MySQL. *Database* ini juga berisi hak-hak yang diberikan pada setiap *user*. [DWI03]

2.8.3 Macromedia Dreamweaver 8

Macromedia Dreamweaver 8 merupakan sebuah program aplikasi yang dapat mengedit HTML secara visual dan mengelola *website* serta *pages*. Kemampuan yang disediakan program aplikasi Macromedia Dreamweaver 8 antara lain *Interface* Macromedia Dreamweaver 8, tampilan panel-panel, dokumen baru, teks editor yang terintegrasi, *panel CSS style*, *templates*, petunjuk kode, *panel snippets*, editor *tag* dan *web developer*.



Gambar 2.3 Interface Macromedia Dreamweaver 8

Macromedia Dreamweaver 8 dilengkapi dengan kemampuan *client-side scripting*. Beberapa *script server side* dapat dikenali oleh Macromedia Dreamweaver 8, misalnya JSP, ASP, PHP, ASP.Net dan CFML. Berikut ini adalah kemampuan-kemampuan yang disediakan oleh program aplikasi Macromedia Dreamweaver 8: [KAD02]

a) *Interface* Macromedia Dreamweaver 8

Macromedia Dreamweaver 8 tampil dengan *interface* standar yang lebih konsisten dengan produk-produk *author Web* Macromedia lainnya. *Interface* ini meliputi tampilan panel, penamaan menu, ikon, *shortcut keyboard*, *launcher bar* dan yang lainnya.

b) Tampilan Panel-panel

Panel-panel di dalam Macromedia Dreamweaver 8 ditempelkan pada jendela kerja sehingga memudahkan *user* bekerja. Selain itu tampilan panel-panel tersebut bisa dimaksimalkan (ditampilkan secara lengkap) maupun diminimalkan untuk menghemat area kerja.

c) Dokumen Baru

Macromedia Dreamweaver 8 memberikan fasilitas baru di dalam pembuatan dokumen baru, yaitu kotak dialog *New Document*, yang memungkinkan *user* memilih atau menentukan *layout* halaman maupun kode yang akan digunakan.

d) Teks *Editor* yang Terintegrasi

Fasilitas *Code View* maupun *Code Inspector* tampil lebih terintegrasi *Code View* menyediakan fitur-fitur sebagai *autoediting*, *live syntax coloring* untuk HTML dan Javascript, *toolbar* dengan kode yang lebih terorientasi. Fitur-fitur ini memudahkan penulisan kode sehingga semudah seperti bekerja dengan *Design View* (bekerja secara visual)

e) Panel *CSS Style*

Panel *CSS Style* tampil lebih lengkap dengan *Style sheet* yang memudahkan pengeditan halaman. Selain itu panel *CSS Style* lebih memudahkan *user* di dalam membedakan *style* yang ditentukan secara lokal dengan *style* yang ditentukan secara eksternal.

f) *Templates*

Fasilitas *Templates* yang disediakan Macromedia Dreamweaver 8 memungkinkan *user* lebih mengontrol *layout* halaman.

g) Petunjuk Kode

Pada saat akan melakukan pengkodean, menu *pop-up* akan muncul memberikan petunjuk tentang kode yang sesuai dengan *item* yang dipilih *user*.

h) Panel *Snippets*

Panel *Snippets* memungkinkan penyimpanan, pengaturan, dan penggunaan kembali kode-kode.

i) *Tag Editor*

Pada jendela *Code View*, *Tag Editor* memungkinkan pengeditan *property* dari tag individual termasuk *HTML tag*, *CFML tag* dan *ASP.NET tag*.

j) *Web Developer*

Macromedia Dreamweaver 8 memberikan fitur-fitur baru untuk *Web Developer* yang di dalamnya termasuk *ColdFusion 8 Support*, *ASP.NET Support*, *PHP Support*, *Web Services Introspectios*.

