

## BAB II

### LANDASAN TEORI

#### 2.1. Konsep Dasar Web

##### 2.1.1. Pengenalan *Internet*

Menurut M. Shalahuddin (2012:3), internet atau *internetwork* adalah sekumpulan jaringan berbeda yang saling berhubungan bersama sebagai satu kesatuan dengan menggunakan berbagai macam protokol, salah satunya adalah protokol TCP/IP (*Transmission Control Protocol/Internet Protocol*).

TCP/IP adalah protokol yang paling banyak digunakan di *internet*. Protokol TCP/IP merupakan cara standar untuk memaketkan dan mengalamatkan data komputer (sinyal elektronik) sehingga data tersebut dapat dikirim ke komputer terdekat atau keliling dunia dan tiba dalam waktu yang cepat tanpa rusak dan hilang.

Menurut Budi Sutedjo, Dharma Octomo (2010a:23), di Indonesia jaringan internet mulai dikembangkan pada tahun 1983 di Universitas Indonesia berupa UINet oleh Dr. Joseph F.P. Luhukay. Ketika itu, ia baru menamatkan program doktoral Filosofi Ilmu Komputer di Amerika Serikat. Jaringan dibangun selama empat tahun. Pada tahun yang sama, Luhukay pun mulai mengembangkan *University Network* (Uninet) di lingkungan Departemen Pendidikan dan Kebudayaan. Uninet merupakan jaringan komputer dengan jangkauan lebih luas dan meliputi Universitas Indonesia, Institut Teknologi Bandung, Institut Pertanian Bogor, Universitas Gadjah Mada, Institut Teknologi Surabaya, Universitas Hasanudin, dan Ditjen Dikti.



Menurut Budi Sutedjo, Dharma Oetomo (2010b:24), fasilitas yang tersedia di internet meliputi :

1. *E-mail*

*E-mail* merupakan jenis layanan di internet yang paling populer, yaitu layanan surat elektronik yang dapat digunakan untuk mengirim atau menjawab pesan, mengirimkan *file* sebagai bagian dari berita *e-mail*, dan berlangganan berita kepada grup diskusi yang diminati.

2. *Mailing List*

Penggunaan *e-mail* dapat dikembangkan menjadi layanan *Mailing List* atau sering disingkat *milis*. Layanan ini bermanfaat untuk berlangganan informasi dari suatu grup diskusi yang diminati atau sarana diskusi yang mengasyikkan.

3. *Internet Relay Chat (Chatting)*

Layanan *chatting* ini merupakan sarana yang murah bagi para pengakses untuk berkomunikasi secara tekstual, jadi, pengakses berkomunikasi melalui tulisan yang diketik. Kemudian, rekan pengakses akan membalas dalam bentuk tulisan pula.

4. *USENET* dan *Newsgroup*

Jenis layanan ini berupa *Bulletin Board Service (BBS)* dalam bentuk pesan. Setiap pengakses internet dapat bergabung untuk saling bertukar informasi satu sama lainnya. *Newsgroup* adalah sarana konferensi elektronik jarak jauh bagi para pengakses internet berdasarkan kompetensi atau bidang minat tertentu.



5. *File Transfer Protocol*

Melalui layanan internet ini, para pengakses dapat pula mengirimkan *file* yang berisi tulisan, gambar, animasi, musik, atau *game* kepada para rekannya. FTP memungkinkan pengakses untuk menyalin *file* secara elektronik dari satu komputer ke komputer lainnya di dalam internet.

6. *Telnet*

*Telnet* adalah fasilitas yang seolah-olah menghubungkan langsung komputer pengakses dengan komputer rekannya melalui internet.

7. *Web Browser*

Untuk mengakses sebuah *file* internet diperlukan suatu program aplikasi yang disebut *web browser*. Program aplikasi ini mempunyai kemampuan untuk menampilkan suatu *web page* yang ditulis dalam format HTML. Dua contoh program aplikasi angkatan pertama adalah *Lynx* dan *Mosaic*. Saat ini, *browser* yang paling banyak digunakan adalah *Internet Explorer* dan *Mozilla Firefox*.



8. *Word Wide Web*

Layanan multimedia internet atau yang dikenal *Word Wide Web* (WWW) merupakan aplikasi internet yang paling diminati para pengakses. WWW disukai karena mencakup sumber daya multimedia, antara lain suara, gambar, video, audio, dan animasi. Oleh karena itu, aplikasi demikian menjadi semacam sarana untuk menyajikan informasi yang interaktif dan atraktif. Layanan WWW lebih dikenal dengan sebutan *web*.

Situs *web* pertama yang muncul di dunia maya sekaligus menandai dimulainya era *World Wide Web* alias '*www*' dibuat oleh Sir Timothy John

"Tim" Berners Lee pada 6 Agustus 1991. Beliau saat itu mengorbitkan <http://nxoc01.cern.ch> sebagai situs pertama dan kemudian diikuti dengan <http://nxoc01.cern.ch/hypertext/WWW/TheProject.html> yang menjadi halaman *web* pertama. Sayangnya situs ini sudah mati namun salinannya masih tersimpan. Selanjutnya untuk menampilkan situs *web* tersebut, beliau menggunakan komputer NeXTcube sebagai *server web* yang menjadikan komputer seharga USD 6.500 itu sebagai *server web* pertama di dunia.

Tahun 1994, Sir Tim Berners-Lee mendirikan *World Wide Web Consortium* (W3C) di Massachusetts Institute of Technology dan sampai sekarang masih menjabat sebagai ketua. Sumbangsih beliau dalam dunia maya sangat besar karena *World Wide Web* yang merupakan hasil karyanya tidak dipatenkan sehingga sampai saat ini masih bisa digunakan secara bebas. Atas jasa-jasanya itu, Sir Tim Berners-Lee dianugerahi gelar kehormatan KBE oleh Ratu Elizabeth II pada 16 Juli 2004.



#### 9. *Uniform Resource Locator (URL)*

Seorang pengguna internet bukan hanya bisa mengakses *web page* yang berupa dokumen HTML saja, tetapi juga dapat mengakses dokumen-dokumen yang disediakan oleh *FTP*, *Gopher*, *Usenet News*, dan sebagainya. Karena jenis dokumen yang bisa diakses bermacam-macam maka dibutuhkan suatu cara untuk menunjuk dengan tepat dan cepat ke dokumen yang diinginkan. URL (*Uniform Resource Locators*) adalah jawaban masalah tersebut. Pada penggunaan WWW, penunjuk suatu sumber informasi menggunakan metode *Uniform Resource Locator (URL)* yang merupakan konsep penamaan lokasi

standar dari suatu *file*, direktori, komputer, lokasi komputernya dan untuk menentukan alamat yang akan kita pakai untuk mengakses internet khususnya *web*. URL mengantarkan *browser* ke alamat yang dituju. Jadi, jika kita mengakses *web* terlebih dahulu kita menulis URL sebagai definisi alamat yang dituju seperti [http://www.yahoo.com/iklan\\_mini/index.htm](http://www.yahoo.com/iklan_mini/index.htm), dimana:

*http* : adalah jenis protocol yang digunakan

[www.yahoo.com](http://www.yahoo.com) : adalah alamat *host* yang digunakan

*iklan\_mini* : adalah direktori atau *path* yang akan dituju

*index.htm* : adalah nama *file* yang akan diakses

10. *Hypertext Transfer Protocol (HTTP)*

Menurut M. Shalahuddin (2012a:4), HTTP (*Hypertext Transfer Protocol*) merupakan aturan pengiriman informasi yang berupa *hypertext* (teks pada komputer yang memungkinkan *user* saling mengirimkan informasi (*request-respons*)). HTTP adalah protokol atau aturan standar untuk mengakses dokumen HTML (*Hypertext Markup Language*) pada *web*. HTTP dijalankan dengan cara klien mengirimkan permintaan (*request*) kepada *server* dan *server* membalas permintaan klien dengan respon yang diminta.

11. *Hypertext Markup Language (HTML)*

Menurut M. Shalahuddin (2012b:19), HTML atau *Hypertext Markup Language* adalah suatu format data yang digunakan untuk membuat dokumen *hypertext* (teks pada komputer yang memungkinkan *user* saling mengirimkan informasi). Dokumen HTML harus disimpan dengan ekstensi *.htm* atau *.html*. HTML memiliki *tag-tag* yang telah didefinisikan untuk



membuat halaman *web*. Penulisan *tag-tag* HTML dapat menggunakan huruf besar atau huruf kecil, karena HTML tidak *case sensitive* (membedakan huruf besar dan huruf kecil memiliki maksud berbeda).

#### 12. *Homepage (file)*

Menurut M. Shalahuddin (2012c:6), *website* adalah informasi di *word wide web* yang disimpan dalam *file* yang berbeda-beda sebagai halaman *web*. *Homepage* merupakan halaman awal dari sebuah situs *web*.

#### 13. *Web Hosting*

Menurut M. Shalahuddin (2012d:7), *web hosting* mengizinkan perorangan atau organisasi membuat sebuah *website* yang dapat diakses melalui *word wide web*. *Web hostis* merupakan organisasi yang menyediakan tempat di *server*-nya untuk perorangan atau organisasi meletakkan semua *file website*-nya dan menyediakan koneksi dengan internet agar dapat diakses melalui *internet*.



#### 14. *Domain dan DNS Server*

Jika dianalogikan sebagai dunia nyata, domain adalah alamat untuk mengakses rumah sebuah *file*. Sementara rumah disini adalah sebuah komputer yang berfungsi sebagai *webserver (host)*.

DNS (*Domain Name Server/Service*) atau yang sering disebut 'Domain' saja berfungsi untuk mempermudah pengguna internet saat melakukan akses ke *file* untuk menerjemahkan *IP address* komputer. *Domain* pertama kali dibuat pada tahun 1984 yang terdiri dari *domain top level* seperti [dot]COM, .ORG, .EDU, .GOV, .MIL dan ccTLD.

Tanggal 15 Maret 1985, *domain komersial* pertama (.com) didaftarkan atas nama Symbolics.com oleh perusahaan komputer Symbolics Inc., menyusul kemudian 5 domain selanjutnya yaitu bbn.com (24 April 1985), think.com (24 Mei 1985), mcc.com (11 Juli 1985) dec.com, (30 September 1985) northrop.com (7 November 1985) Bulan April 1985, *domain* yang diperuntukkan bagi institusi pendidikan (.edu) pertama kali terdaftar atas nama cmu.edu, purdue.edu, rice.edu dan ucla.edu. Juni 1985, *domain* yang diperuntukkan bagi institusi pemerintahan (.gov) pertama kali terdaftar atas nama css.gov. Juli 1985, *domain* yang diperuntukkan bagi organisasi (.org) pertama kali terdaftar atas nama mitre.org. Hingga Juni 2010, menurut statistik Domaintools tercatat ada 119.928.149 *domain* aktif dan 87.818.851 di antaranya adalah domain (.com).



### 2.1.2. Pengenalan Web Programming

*Web* dikategorikan menjadi dua yaitu *web* statis dan *web* dinamis atau interaktif. *Web* statis adalah *web* yang berisi atau menampilkan informasi-informasi yang sifatnya statis (tetap), sedangkan *web* dinamis adalah *web* yang menampilkan informasi serta dapat berinteraksi dengan user yang sifatnya dinamis. Sehingga untuk membuat *web* dinamis dibutuhkan kemampuan pemrograman *web*. Dalam pemrograman *web* ada dua kategori, yaitu:

#### 1. Server – Side Programming

Perintah-perintah program (*script*) dijalankan di *web browser*, kemudian hasilnya dikirimkan kedalam *browser* dalam bentuk HTML biasa seperti ASP (*Active Server Pages*), PHP, JSP (*Java Server Pages*) dan CFM (*Cold Fusion*).

## 2. Client – Side Programming

Perintah program dijalankan di *web browser*, sehingga ketika *client* meminta dokumen yang mengandung *script*, maka *script* tersebut akan di *download* dari *server*-nya kemudian dijalankan di *browser* yang bersangkutan. Seperti *JavaScript*, *Vbscrip* dan HTML.

### 2.1.3. Pengenalan PHP

Menurut Kadir (2011a:2), PHP merupakan singkatan dari *PHP Hypertext Preprocessor*. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server. Hasilnya yang dikirimkan ke klien, tempat pemakai menggunakan *browser*. Secara khusus, PHP dirancang untuk membentuk aplikasi *web* dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, Anda bisa menampilkan isi *database* ke halaman *web*. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), Cold Fusion, ataupun Perl. Namun, perlu diketahui bahwa PHP sebenarnya bisa dipakai secara *command line*. Artinya, skrip PHP dapat dijalankan tanpa melibatkan *web server* maupun *browser*. Kelahiran PHP bermula saat Rasmus Lerdorf membuat sejumlah skrip Perl yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. *Script-script* ini selanjutnya dikemas menjadi *tool* yang disebut "Personal Home Page". Paket inilah yang menjadi cikal-bakal PHP. Pada tahun 1995, Rasmus menciptakan PHP/FI Versi 2. Pada versi inilah pemrograman 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 saat ini PHP cukup populer sebagai peranti pemrograman *Web*, terutama di lingkungan Linux. Walaupun demikian, PHP sebenarnya juga dapat berfungsi pada *server-server* yang berbasis UNIX, Windows, dan Macintosh. Pada awalnya, PHP dirancang untuk diintegrasikan dengan *web server* Apache. Namun, belakangan PHP juga dapat bekerja dengan *web server* seperti PWS (*Personal Web Server*), IIS (*Internet Information Server*), dan Xitami. Untuk mencoba PHP, kita tidak perlu menggunakan komputer berkelas *server*. Hanya dengan sebuah komputer biasa, Anda bisa mempelajari dan mempraktikkan PHP. PHP bersifat bebas dipakai. Anda tidak perlu membayar apa pun untuk menggunakan perangkat lunak ini alias *free*. Anda dapat mendownloadnya melalui situs <http://www.php.net>, PHP tersedia dalam bentuk kode biner maupun kode sumber yang lengkap. Bagaimana halnya kalau yang diminta adalah sebuah halaman PHP? Prinsipnya serupa dengan kode HTML. Hanya saja, ketika berkas PHP yang diminta didapatkan oleh *web server*, isinya segera dikirimkan ke mesin PHP dan mesin inilah yang memproses dan memberikan hasilnya (berupa kode HTML) ke *web server*. Selanjutnya, *web server* menyampaikan ke klien. Salah satu kelebihan dari PHP adalah mampu berkomunikasi dengan berbagai *database* yang terkenal. Dengan demikian, menampilkan data yang bersifat *dinamis*, yang diambil dari *database*, merupakan hal yang mudah untuk *diimplementasikan*. Itulah sebabnya sering dikatakan bahwa PHP sangat cocok untuk membangun halaman-halaman *web* dinamis. Pada saat ini PHP sudah dapat berkomunikasi dengan berbagai *database* meskipun dengan kelengkapan yang berbeda-beda. Beberapa di antaranya *BASE*,



*DBM, FilePro(Personix, Inc), Informix, Ingres, InterBase, Microsoft Access, MSOL, MySQL, Oracle, PostgresSQL, Sybase.*

### 1. Tag-tag PHP

- a. Menggunakan sintaks `<?php ... ?>`

Contoh :

```
<?php echo"Ini adalah script PHP"; ?>
```

- a. Menggunakan sintaks `<? ... ?>`

Contoh :

```
<? echo"Ini merupakan cara kedua penulisan script PHP"; ?>
```

- b. Menggunakan sintaks `<script language="php"> ... </script>`

Contoh :

```
<script language="php">
```

echo"Ini merupakan cara ketiga penulisan script PHP"; </script>.

- c. Menggunakan sintaks `<% ... %>`

Contoh :

```
<% echo"Ini merupakan cara keempat penulisan script PHP"; %>.
```

Jika saat penulisan *script* PHP panjang dan kompleks, sebaiknya diberikan komentar atau penjelasan pada bagian-bagian yang dianggap penting. Dan semua tulisan yang berada di dalam tag komentar tidak akan di proses. Ada tiga cara dalam penulisan komentar dalam *script* PHP, yaitu:

1. Komentar *C style*, yang diawali dengan tag pembuka `/*` dan diakhiri dengan tag penutup `*/`, seperti pada contoh berikut :

```
<?php
```

```
/* menampilkan kata apa kabar*/
echo"apa kabar";
?>
```

Komentar C++ *style*, dengan menuliskan tag // pada awal baris komentar, seperti pada contoh berikut :

```
<?php
echo" apa kabar";//menampilkan kata apa kabar
?>
```

2. Komentar Bourne Shell *style*, dengan menuliskan tag # pada baris kode

PHP yang dimaksud, seperti pada contoh berikut :

```
<?php
echo" apa kabar";#menampilkan kata apa kabar
?>
```



## 2. Variabel PHP

Variabel digunakan untuk menyimpan data sementara dan nilainya bisa berubah-ubah setiap kali program dijalankan. Adapun aturan dalam penulisan variabel adalah sebagai berikut :

- a. Variabel di mulai dengan tanda \$
- b. Harus dimulai dengan huruf atau *underscore* (tidak boleh dimulai dengan angka).
- c. Mengenal perbedaan huruf besar dan huruf kecil (*Case sensitif*).

Contoh penulisan variabel yang benar :

```
$Fantoe
```

`$_Fantoe`

`$Fantoe1`

Contoh penulisan variabel yang salah :

`$1Fantoe.`

`$_Fantoe.`

Untuk mendefinisikan nilai sebuah variabel diikuti dengan tanda sama dengan (=) dan diikuti oleh ekspresi. Ekspresi adalah sebuah pernyataan yang mengandung nilai (*value*) untuk sebuah variabel. Contoh penerapannya adalah sebagai berikut :

`$namavariabel="ekspresi string"`

Jika ekspresi berupa data integer, kita tidak usah menggunakan tanda kutip ganda (") atau tanda kutip tunggal ('). Contoh penulisan variabel dan ekspresi :

<?

`$nama="Andhika";`

`$nim="12105880";`

`$kelas="12.6F.31";`

`$semester="Semester VI";`

`$pekerjaan="Mahasiswa";`

`$panggilan="Dhika";`

`echo "Nama :$nama <br>";`

`echo "NIM :$nim <br>";`

`echo "Kelas :$kelas <br>";`

`echo "Semester :$semester <br>";`



```

echo"Pekerjaan :$pekerjaan <br>";
echo"Panggilan :$panggilan oke!!<br>";
echo" <br>";
?>

```

Sebagian besar *script* PHP menggunakan variabel dengan lingkup tunggal (*single scope*), yang artinya sebuah variabel dideklarasikan dan dioperasikan pada satu tubuh *script*. Lingkup tunggal ini dapat di perluas menggunakan fungsi `include()` atau `require()` untuk menyertakan variabel yang di deklarasikan dalam *file* berbeda. Contoh :

```

<?
For($a=1;$a<4;$a++)
{ include("contohphp1.php"); }
?>

```

Pada contoh diatas, *script* `contohphp2.php` menyertakan *script* `contohphp1.php` dengan menggunakan fungsi `include()`.



### 3. Tipe data pada PHP

PHP mengenal beberapa macam tipe data, antara lain :

#### a. Integer

*Integer* merupakan tipe data yang meliputi semua bilangan bulat (...., -3, -2, 1, 0, 1, 2, ...). Nilai maksimum tipe data *integer* adalah 2147483647 dan nilai minimumnya adalah -2147483647.

#### b. Floating point

Tipe data *floating point* merupakan tipe data yang mencakup bilangan pecahan.

c. *String*

Tipe data *string* adalah tipe data yang mencakup sekumpulan karakter, baik berupa abjad (a – z , A – Z ) maupun angka (0 – 9). Penulisan *string* dalam PHP selalu diawali dengan tanda petik ganda ( " " ) atau tanda petik tunggal ( ' ' ).

d. *Boolean*

Tipe data *boolean* mengekspresikan nilai *TRUE* dan *FALSE*. Tipe data *boolean* biasanya digunakan dalam sintaks *condition* atau statement *if*. Nilai default dari tipe data ini adalah *TRUE*.

#### 2.1.4. Entity Relationship Diagram

Yang dimaksud dengan *entity relation diagram* (ERD) adalah penggambaran relasi antar entitas secara keseluruhan dengan menggunakan grafik atau gambar.

*Entity Relation Diagram* dapat digambarkan dengan simbol-simbol yang digunakan oleh *tools* PowerDesigner. Berikut simbol-simbol yang digunakan yaitu :

1. *Entity*:

*Entity* adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat.

2. Atribut

*Entity* mempunyai elemen yang disebut atribut, dan berfungsi mendeskripsikan karakter *entity*. Setiap ERD bisa terdapat lebih dari satu atribut.



### 3. Relasi *One To One*

Notasi yang menyatakan hubungan antara entitas yang satu dengan yang lain bersifat satu ke satu.

### 4. Relasi *One To Many*

Notasi yang menyatakan hubungan antara entitas yang satu dengan yang lain bersifat satu ke banyak.

Pada *tools* PowerDesigner yang digunakan penulis dalam perancangan dan pembuatan sistem, simbol-simbol yang digunakan pada ER diagram konvensional berbeda dengan simbol-simbol yang digunakan oleh *tools* Power Designer. Pada tabel dibawah ini merupakan simbol-simbol ER diagram yang digunakan oleh penulis dalam pembuatan sistem dengan menggunakan *tools* PowerDesigner.



#### 2.1.5 Pengenalan MySQL

*MySQL* (*My Structure Query Language*) menurut Nugroho (2008:91) merupakan “sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan *DBMS* (*DataBase Management System*) yang handal karena menggunakan sistem *SQL* (*Structure Query Language*)”. *SQL* dipublikasikan oleh E.F. Codd (1970) mengenai model *relational*. Kemudian pada tahun 1974, D. Chamberlin dan R.F. Boyce mengembangkan bahasa *query* untuk memanipulasi dan mengekstrak data dan basisdata relational. Ada beberapa sasaran *SQL*:

1. Menciptakan basisdata dan struktur relasi.
2. Melakukan manajemen data tingkat dasar.

3. Membentuk *query* sederhana dan kompleks.
4. Melakukan tugas-tugas dengan seminimal mungkin memakai struktur dan *sintaks* perintah yang relatif mudah untuk dipelajari. *DBMS* yang menggunakan bahasa *SQL* :
  - a. *MySQL*.
  - b. *MSQL*.
  - c. *PosgreSQL*.
  - d. *Oracle*.
  - e. *SQL Server 97, 2000, dll*.
  - f. *Interbase*.
5. Program-program yang mendukung *MySQL*
  - a. *PHP (Page Hipertext Preprosesor)*.
  - b. *Borland Delphi, Borland C++ Builder*.
  - c. *Visual Basic 5.0/6.0 dan .Net*.
  - d. *Visual Foxpro*.
  - e. *Cold Fusion*.
6. Pengelompokan *Statement MySQL* :
  - a. *Data Definition Language (DDL)*
    - 1) *Create*
      - a) Pembuatan *Database*  
*Sintaksnya adalah create database (nama\_database);*
      - b) Pembuatan Tabel  
*Sintaksnya adalah create table            nama\_tabel*



(nama\_kolom1,tipe\_data\_kolom1,nama\_kolom2,tipe\_data\_kolom2,...);

c) Pembuatan *Index*

*Sintaksnya* adalah `create [unique] index nama_index on nama_tabel(nama_kolom).`

d) Pembuatan *View*

*Sintaksnya* adalah `create view nama_view[(nama_kolom1,...)] as select statement1 [with check option].`

2) *Drop*

a) Menghapus *Database*

*Sintaksnya* adalah `drop database nama_database;`

b) Menghapus *Tabel*

*Sintaksnya* adalah `drop table nama_table;`

c) Menghapus *Index*

*Sintaksnya* adalah `drop index nama_index;`

d) Menghapus *View*

*Sintaksnya* adalah `drop view nama_view;`

3) *Alter Table* (Merubah Struktur Tabel)

a) Menambah Kolom

*Sintaksnya* adalah `alter table nama_tabel add (nama_kolom, jenis_kolom before(nama_kolom);`

b) Merubah Jenis Kolom

*Sintaksnya* adalah `alter table nama_tabel modify (nama_kolom`



*jenis\_kolom size\_kolom*);

c) Menghapus Kolom

*Sintaksnya* adalah *alter table nama\_tabel drop*(nama\_kolom, jenis\_kolom);

b. *Data Manipulation Language (DML)*

1) *Insert*

*Sintaksnya* adalah *insert into nama\_tabel [(nama\_kolom1,...)]*.

2) *Update*

*Sintaksnya* adalah *update nama\_tabel set nama\_kolom = value\_1 where kondisi*.

3) *Delete*

*Sintaksnya* adalah *delete from nama\_tabel where kondisi*.

Menurut Kadir (2008b:348), MySQL adalah salah satu jenis *database server* yang sangat terkenal. Kepopulerannya disebabkan MySQL menggunakan SQL sebagai bahasa dasar untuk mengakses *databasenya*. Selain itu, ia bersifat *Open Source* (kita tidak perlu membayar untuk menggunakannya) pada perbagai platform (kecuali untuk jenis Enterprise, yang bersifat komersial). Perangkat lunak MySQL sendiri bisa diunduh dari <http://www.mysql.com>. MySQL termasuk jenis RDBMS (*Relational Database Management System*). Itulah sebabnya, istilah seperti tabel, baris, dan kolom digunakan pada MySQL. Pada MySQL, sebuah *database* mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom. Salah satu *database server* yang cukup dikenal saat ini adalah MySQL. Adapun keunggulan dari MySQL adalah:



1. Mampu menangani jutaan *user* dalam waktu yang bersamaan.
2. Mampu menampung lebih dari 50.000.000 *record*.
3. Sangat cepat mengeksekusi perintah.
4. MySQL menggunakan SQL dan bersifat *free* (gratis).
5. MySQL dapat berjalan di berbagai *platform*, antara lain Linux, Windows, dan lain sebagainya.
6. Memiliki *user privilege system* yang mudah dan efisien.
7. MySQL juga menyediakan dukungan *open source*. Setiap pengguna MySQL diizinkan mengubah *source* untuk keperluan pengembangan atau menyelaraskan spesifikasi *database* sesuai kebutuhan.

Fungsi MySQL dalam sintaks PHP yaitu PHP berfungsi sebagai *Application Programming Interface* (API) yang menghubungkan *programmer* dengan MySQL. API dapat diibaratkan seseorang dengan dua muka, satu muka PHP menghadap *programmer* untuk berkomunikasi dengannya, dan satu muka yang lain menghadap MySQL untuk menyampaikan pesan dari *programmer*. Berikut beberapa fungsi MySQL yang sering digunakan beserta deskripsi singkat masing-masing fungsi tersebut:

#### 1. *mysql\_connect()*

Sebelum melakukan operasi apapun di MySQL, hal pertama yang harus dilakukan adalah membuka koneksi dengan MySQL. Fungsi yang digunakan untuk membuka koneksi dengan MySQL adalah *mysql\_connect()*.

#### 2. *mysql\_close()*

Digunakan untuk mengakhiri koneksi ke MySQL. Fungsi yang digunakan adalah

*mysql\_close()*.

**3. *mysql\_create\_db()***

Selain digunakan untuk melakukan koneksi ke MySQL, *username* dan *password* yang diberikan oleh *administrator server* adalah juga merupakan izin terbatas untuk membuat dan menggunakan *database* sendiri. Untuk membuat *database* yang hanya dapat diakses oleh *username* dan *password*, digunakan fungsi *mysql\_create\_db()* setelah koneksi ke MySQL dibuka.

**4. *mysql\_db\_query()***

Perintah yang diberikan kepada MySQL untuk melakukan sesuatu disebut dengan *query*. *Query* dikirim kepada *database* yang sedang aktif.

**5. *mysql\_select\_db()***

Jika suatu *website* menggunakan satu *database*, fungsi *mysql\_select\_db()* dapat digunakan memilih *database* yang sedang aktif. Dengan memilih *database* yang aktif tersebut dapat menghemat *resource*. Tidak perlu mengulang-ulang fungsi *mysql\_select\_db()* untuk mengirimkan *query* kepada *database* yang aktif.

**6. *mysql\_query()***

Fungsi *mysql\_query()* digunakan untuk mengirimkan *query* kepada *database* aktif yang sudah dipilih dengan fungsi *mysql\_select\_db()*

**7. *mysql\_fetch\_row()***

Fungsi *mysql\_fetch\_row()* digunakan untuk mengambil hasil *query* dari *database* sebagai data *numeric array*. Fungsi ini akan mengambil data baris per baris (per *record*) dari *database* secara berulang-ulang sampai nilai kondisi *query* menjadi *FALSE*. Nilai kondisi *query* diperoleh dari fungsi *mysql\_query(\$query)*.



### 8. *mysql\_fetch\_array()*

Hampir sama dengan *mysql\_fetch\_row()*, *mysql\_fetch\_array()* digunakan untuk mengambil hasil *query* dari *database* sebagai data *associative array*.

### 9. *mysql\_num\_rows()*

Fungsi *mysql\_num\_row()* digunakan untuk menghitung jumlah *row* (blok data atau *record*) hasil *query*.

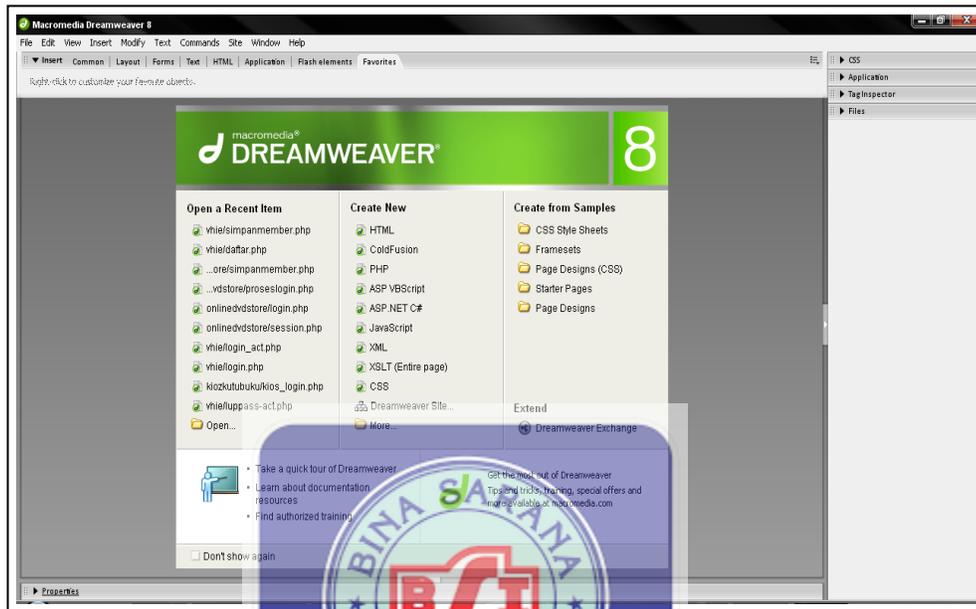
## 2.1.6. Pengenalan *Macromedia Dreamweaver 8*

Macromedia Dreamweaver 8 menurut Nugroho (2008:1) adalah "suatu bentuk program editor *web* yang dibuat oleh macromedia dan merupakan *software* yang dikenal sebagai *software web authoring tool*, yaitu *software* untuk desain dan *layout* halaman *web*. Versi terbaru Dreamweaver memiliki kemampuan bukan hanya sebagai *software* untuk desain *web* saja, tetapi juga untuk editing kode serta pembuatan aplikasi *web* dengan menggunakan berbagai bahasa pemrograman *web*, antara lain *ASP (Java Server Pages)*, *PHP*, *ASP (Active Server Pages)* dan *ColdFusion*.



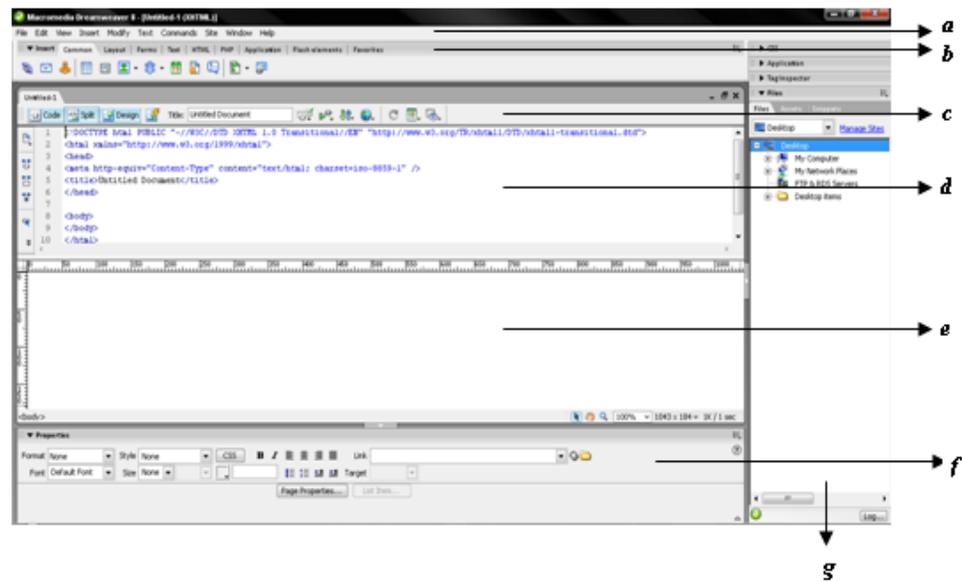
Macromedia Dreamweaver 8 mempunyai sifat *WYSIWYG (What You See Is What You Get)* yang artinya apa yang kita lihat pada halaman desain, maka semuanya itu akan kita peroleh pada *browser*. Dengan kelebihan ini sehingga seorang pembuat program (*programmer*) atau pembuat desain *web* (*web designer*) dapat langsung melihat hasil buaatannya tanpa harus membukanya pada *browser*.

Untuk lebih jelas perhatikan gambar di bawah ini :



**Gambar II.1. Tampilan Macromedia Dreamweaver 8**

Berikut ini adalah bagian-bagian yang terdapat pada jendela kerja Macromedia Dreamweaver 8 :



Gambar II.2. Jendela Kerja *Macromedia Dreamweaver 8*

Keterangan :

1. *Menu Bar*

Menampilkan perintah-perintah untuk merancang dan mengolah dokumen *website*.

2. *Insert Bar*

Menampilkan tombol-tombol untuk memasukkan berbagai jenis objek seperti gambar, *form*, tabel, *layer* dan lainnya kedalam dokumen *website* yang sedang dibuat. *Insert bar* dibuat untuk memudahkan *user* agar dapat melengkapi halaman *web* yang sedang dibuat dengan cepat.

3. *Document Tool Bar*

Menampilkan tombol-tombol dan menu *pop-up* yang menyediakan tampilan berbeda dari komponen *window* (seperti *Design view* dan *Code view*), pilihan menampilkan *ruler*, *grid* dan beberapa operasi umum, seperti



*preview* di *browser*.

#### 4. *Code Window*

Menampilkan *script* program yang digunakan di dalam *web*.

#### 5. *Design Window*

Menampilkan *design web* yang sedang dibuat.

#### 6. *Property Inspector*

Digunakan untuk mengetahui atau mengubah *properties* dari objek yang dimasukkan ke dalam dokumen seperti mengubah jenis, warna dan ukuran huruf.

#### 7. *Panel Site*

Memberikan fasilitas untuk mengantar *file* dan *folder* yang membentuk situs *web*. *Panel* ini berfungsi sebagai *site explorer*, kita dapat membuka atau mengaktifkan situs lama yang pernah kita buat, melihat isi situs, menghapus dokumen *web*, mengganti nama dokumen *web* atau membuat *file* baru.



#### 2.1.7. Pengenalan CSS (*Cascading Style Sheet*)

Prasetio (2012:41) “CSS (*Cascading Style Sheet*) adalah suatu teknologi yang digunakan untuk memperindah tampilan halaman website (situs). Singkatnya dengan menggunakan CSS mudah mengubah secara keseluruhan sekaligus memformat ulang situs.

CSS mempunyai 2 bagian utama yaitu *selectors* dan deklarasi. Yang dimaksud *selectors* biasanya terdiri dari properti dan nilai. Properti sendiri adalah atribut *style* yang ingin anda ubah, dan setiap properti memiliki nilai.

Contoh:

```
P {color : red; text-align : center;}
```

Selanjutnya adalah CSS comment. Comment digunakan untuk menjelaskan kode anda dan membantu ketika mengedit *scrip*. Namun, comment akan diabaikan oleh *browser* jadi kita harus menyembunyikannya. Comment ditandai dengan “/\*” dan diakhiri dengan “\*/”.

Contoh:

```
/* ini adalah comment */
```

```
P
{
  Text-align : center;
}
```



Materi CSS selanjutnya adalah tentang id dan class. Id digunakan untuk membuat *style* dan elemen yang unik. Id selector menggunakan atribut id elemen dan didefinisikan dengan “#” html.

Contoh:

```
#belajar
{
  Text-align : center;
  Color : red;
}
```

CSS juga memungkinkan kita untuk menyatukan elemen-elemen html secara bersamaan didalam sebuah kelas (*classes*) dan menerapkannya aturan style-sheet kedalam sebuah class. Selector class menggunakan atribut html dan didefinisikan

dengan “.”.

Contoh:

```
.center {text-align : center;}
```

### 2.1.8. Pengenalan Xampp

Menurut Nugroho (2008:2) “XAMPP adalah suatu bundel *web server* yang populer digunakan untuk coba-coba di *windows* karena kemudahan instalasinya. Bundel program *open source* tersebut berisi antara lain server web Apache, interpreter PHP, dan basis data MySQL”. Setelah menginstall XAMPP, kita bisa memulai pemrograman PHP di komputer sendiri maupun mencoba menginstall aplikasi-aplikasi web.



### 1.2. Peralatan Pendukung (Tools Program)

Menurut Prihatna (2005:51) Struktur Navigasi adalah “Susunan menu atau hirarki dari suatu situs yang menggambarkan isi dari setiap halaman dan link atau navigasi tiap halaman pada suatu situs *web*”. Struktur navigasi suatu situs *web* sangat di pengaruhi oleh tujuan dari situs *web* yang akan dibuat. Struktur navigasi dapat digolongkan menurut kebutuhan akan objek, kemudahan pemakaian, keinteraktifitasnya, dan kemudahan membuatnya yang berpengaruh terhadap waktu pembuatan suatu situs *web*. Bentuk dasar dari struktur navigasi adalah sebagai berikut:

#### 1. Satu alur (*Linear*)

*Linear* merupakan struktur yang hanya mempunyai satu rangkaian cerita yang berurut. Dengan kata lain struktur ini hanya dapat menampilkan satu demi satu tampilan layar secara berurut menurut urutannya. Tampilan yang ada di tampilan

pada struktur jenis ini adalah satu halaman sebelumnya atau satu halaman sesudahnya dan tidak dapat menampilkan dua halaman sebelumnya atau dua halaman sesudahnya. Salah satu yang terpenting dari struktur ini adalah tidak diperkenankan terjadinya percabangan.



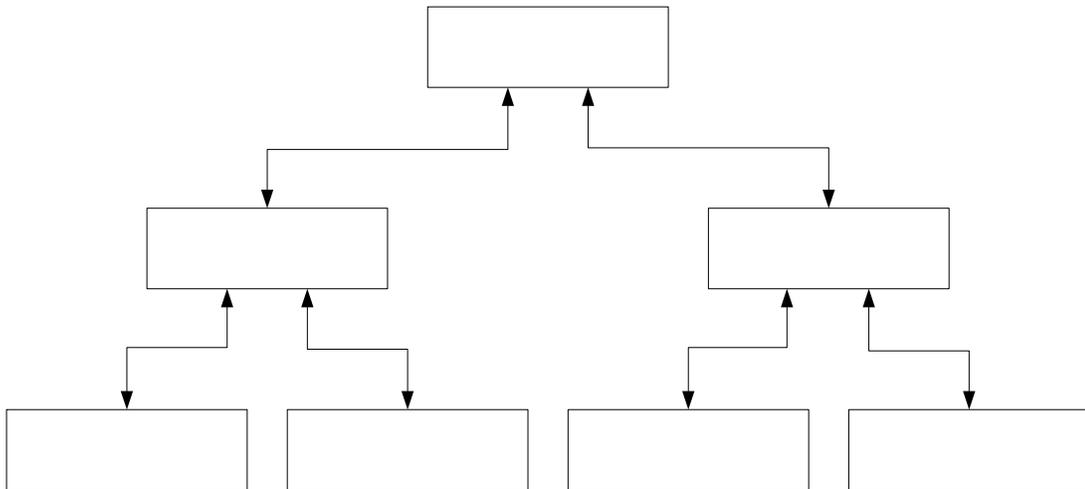
Sumber :

**Gambar II.3. Struktur Navigasi *Linear***

## 2. Hirarki (*Hierarchical*)

Struktur *Hierarchi* ini percabangan untuk menampilkan data berdasarkan kriteria tertentu. Tampilan pada menu pertama akan disebut sebagai *Master Page* (halaman utama kesatu), halaman utama ini akan mempunyai halaman percabangan yang dikatakan *Slave Page* (halaman pendukung). Jika salah satu halaman pendukung dipilih atau diklikkan, maka tampilan tersebut akan bernama *Master Page* (halaman utama kedua), dan seterusnya. Yang terpenting dari struktur penjejakan ini tidak diperkenankan adanya tampilan secara *linear*.



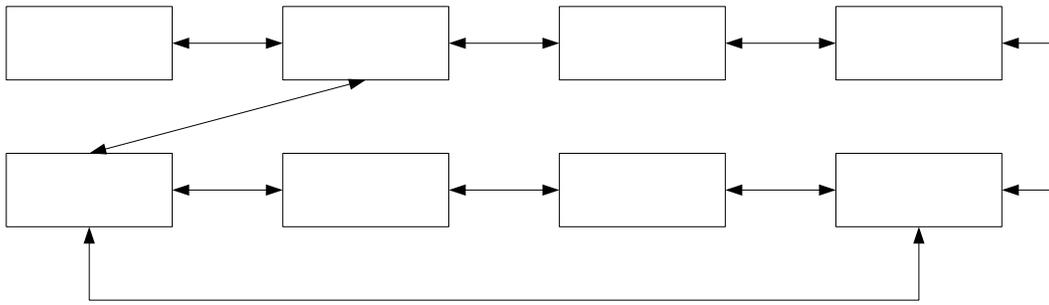


Sumber :

**Gambar II.4. Struktur Navigasi *Hierarchical***

3. Tidak Berurut (*Non Linear*)

Struktur penjejakan *Non Linear* (tidak berurut) merupakan pengembangan dari struktur penjejakan *Linear*. Pada struktur ini diperkenankan membuat penjejakan bercabang. Pemakai bebas menelusuri *website* tanpa dibatasi oleh suatu rute dimana kontrol navigasi dapat mengakses ke semua halaman manapun. Percabangan yang dibuat pada struktur *Non Linear* ini berbeda dengan percabangan pada struktur *Hierarchi*, karena pada percabangan *Non Linear* ini walaupun terdapat percabangan, tetapi tiap-tiap tampilan mempunyai kedudukan yang sama tidak ada *Master Page* dan *Slave Page*.

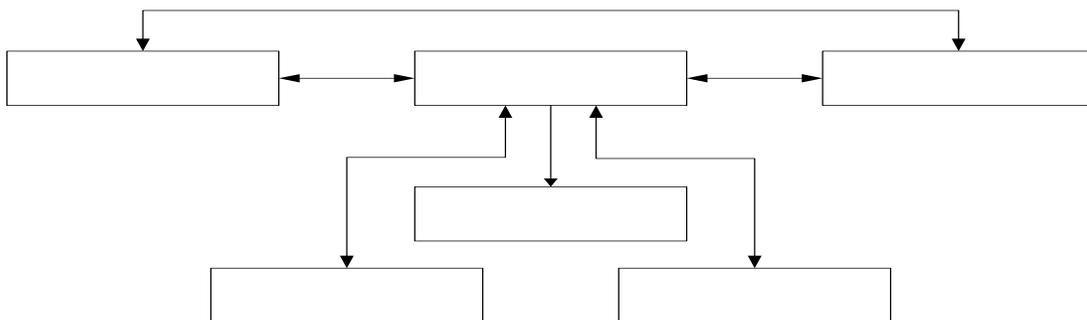


Sumber :

**Gambar II.5. Struktur Navigasi *Non Linear***

#### 4. Campuran (*Composite*)

*Composite* (campuran) atau disebut juga struktur penjejakan bebas merupakan gabungan dari ketiga struktur sebelumnya yaitu *Linear*, *Non Linear* dan *Hierarchi*. Jika suatu tampilan membutuhkan percabangan, maka dapat dibuat percabangan, dan bila dalam percabangan tersebut terdapat suatu tampilan yang sama kedudukannya maka dapat dibuat struktur *Linear* dalam percabangan tersebut.



Sumber :

**Gambar II.6. Struktur Navigasi *Composite***