



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

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

Sutarman (2012:86), “komputer adalah suatu rangkaian peralatan elektronik yang bekerja secara bersama-sama”. Komputer dapat melakukan rangkaian pekerjaan secara otomatis melalui instruksi (program) yang diberikan dan alat pengolah data menjadi informasi melalui proses tertentu.

Wahyudi (2012:3), “komputer adalah peralatan (*device*) yang menerima data (*input*) dan menyimpan (*storage*) kemudian diproses (*process*) untuk menghasilkan data dalam bentuk lain (*output*)”.

2.1.2. Pengertian Teknologi

Sunardi et.al (2011:4) mengemukakan, “Secara bahasa, teknologi dapat diartikan sebagai suatu cara atau metodologi yang menerapkan pengetahuan-pengetahuan atau peralatan-peralatan teknik”.

Menurut Kamus Besar Bahasa Indonesia (2013:1422), teknologi adalah metode ilmiah untuk mencapai tujuan praktis.

2.1.3. Pengertian Data

Kristanto (2008:7) mengemukakan, “Data merupakan kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata”.

McLeod dalam Yakub (2012:5) mengemukakan, “Data adalah kenyataan yang menggambarkan adanya suatu kejadian (event), data terdiri dari fakta (fact) dan angka yang secara relatif tidak berarti bagi pemakai”.



Hartono dalam Hendelman et.al, (2013:15) mengemukakan, “Data adalah hasil pengukuran dan pencatatan terhadap fakta tentang sesuatu, keadaan, tindakan atau kejadian”.

2.1.4. Pengertian Pengolahan Data

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

2.1.5. Pengertian Informasi

Kristanto (2008:7) mengemukakan, “Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima”.

Sutabri (2012:22) mengemukakan, “Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan”.

Hartono dalam Davis (2013:15) mengemukakan, “Informasi adalah data yang telah diolah menjadi suatu bentuk yang berguna bagi penerimanya dan memiliki nilai bagi pengambilan keputusan saat ini atau di masa yang akan datang”.

2.2. Teori Judul

2.2.1. Pengertian Aplikasi

Asropudin (2013:7), “aplikasi (*application*) adalah software yang dibuat oleh suatu perusahaan komputer untuk mengerjakan tugas-tugas tertentu”.

Kamus Besar Bahasa Indonesia, “aplikasi adalah program komputer atau perangkat lunak yang didesain untuk mengerjakan tugas tertentu”.

Hendrayudi (2009:143), “aplikasi adalah kumpulan perintah program yang dibuat untuk melakukan pekerjaan-pekerjaan tertentu (khusus)”.

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



2.2.2. Pengertian Integrasi

Menurut Kamus Besar Bahasa Indonesia (2009:362), Integrasi adalah Penyumlahan dari rangkaian diferensial, ditunjuk dengan simbol; proses yang digunakan untuk menggabungkan dan mengatur materi-materi organis, Psikologis, atau social kedalam keseluruhan yang kompleks pada taraf yang lebih tinggi

2.2.3. Pengertian Monitoring

Monitoring (bahasa Indonesia: pemantauan) adalah pemantauan yang dapat dijelaskan sebagai kesadaran (awareness) tentang apa yang ingin diketahui, pemantauan berkadar tingkat tinggi dilakukan agar dapat membuat pengukuran melalui waktu yang menunjukkan pergerakan ke arah tujuan atau menjauh dari itu. Monitoring akan memberikan informasi tentang status dan kecenderungan bahwa pengukuran dan evaluasi yang diselesaikan berulang dari waktu ke waktu, pemantauan umumnya dilakukan untuk tujuan tertentu, untuk memeriksa terhadap proses berikut objek atau untuk mengevaluasi kondisi atau kemajuan menuju tujuan hasil manajemen atas efek tindakan dari beberapa jenis antara lain tindakan untuk mempertahankan manajemen yang sedang berjalan.

<http://id.wikipedia.org/wiki/Monitoring>

2.2.4. Pengertian Gudang

Menurut Kamus Besar Bahasa Indonesia (2009:295), Gudang adalah Rumah atau bangsal menyimpan barang-barang.

2.2.5. Pengertian PT

Sempal (2013:12), “Perseroan terbatas adalah organisasi bisnis yang memilikibadan hukum resmi yang dimiliki oleh minimal dua orang dengan tanggung jawab yanghanya berlaku pada perusahaan tanpa melibatkan harta pribadi atau perseorangan yangada di dalamnya”.

Syahrizal et.al (2012:1), “Perseroan Terbatas adalah badan hukum yangmerupakan persekutuan modal yang dilakukan oleh minimal dua orang



dengantangung jawab yang hanya berlaku pada perusahaan serta, tanpa melibatkan hartapribadi atau perseorangan yang ada di dalamnya (para pemegang saham), didirikanberdasarkan perjanjian, melakukan kegiatan usaha dengan modal dasar yang seluruhnyaterbagai dalam saham dan memenuhi persyaratan yang ditetapkan dalam Undang-Undang Republik Indonesia Nomor 40 Tahun 2007 Tentang Perseroan Terbatas sertaperaturan pelaksanaanya”

2.2.6. Pengertian PLN

Perusahaan Listrik Negara (disingkat PLN) adalah sebuah BUMN yang mengurus semua aspek kelistrikan yang ada di Indonesia. Direktur Utamanya adalah Nur Pamudji, menggantikan Dahlan Iskan Dirut sebelumnya yg di lantik menjadi menteri BUMN

Ketenagalistrikan di Indonesia dimulai pada akhir abad ke-19, ketika beberapa perusahaan Belanda mendirikan pembangkitan tenaga listrik untuk keperluan sendiri. Pengusahaan tenaga listrik untuk kepentingan umum dimulai sejak perusahaan swasta Belanda NV. NIGM memperluas usahanya di bidang tenaga listrik, yang semula hanya bergerak di bidang gas. Kemudian meluas dengan berdirinya perusahaan swasta lainnya.

[http://id.wikipedia.org/wiki/Perusahaan Listrik Negara](http://id.wikipedia.org/wiki/Perusahaan_Listrik_Negara)

2.2.7. Pengertian Aplikasi Integrasi Monitoring Barang pada Gudang Pusat PT.PLN (Persero) Pembangkitan Sumbagsel Sektor Pembangkitan Keramasan”.

Pengertian dari Aplikasi Integrasi Monitoring Barang pada Gudang Pusat PT.PLN (Persero) Pembangkitan Sumatera Bagian Selatan Sektor Pembangkitan Keramasan,adalah suatau perangkat lunak yang digunakan untuk menggabungkan pengawasan dan meminta terhadap barang atau material yang terdapat di PT.PLN (Persero) Pembangkitan Sumbagsel Sektor Pembangkitan Keramasan, untuk mempermudah dan tidak banyak membuang waktu dalam memantau barang dan material di gudang tersebut.



2.3. Teori Program

2.3.1. Sekilas Tentang PHP

2.3.1.1. Pengertian PHP

Wahana (2012:76), “*PHP* merupakan bahasa pemrograman berbasis web yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis”.

Anhar (2010:3), “*PHP* merupakan script yang digunakan untuk membuat halaman website yang dinamis”.

Kadir (2013:17), “*PHP* adalah bahasa pemrograman yang ditujukan untuk kepentingan pembuatan aplikasi web”.

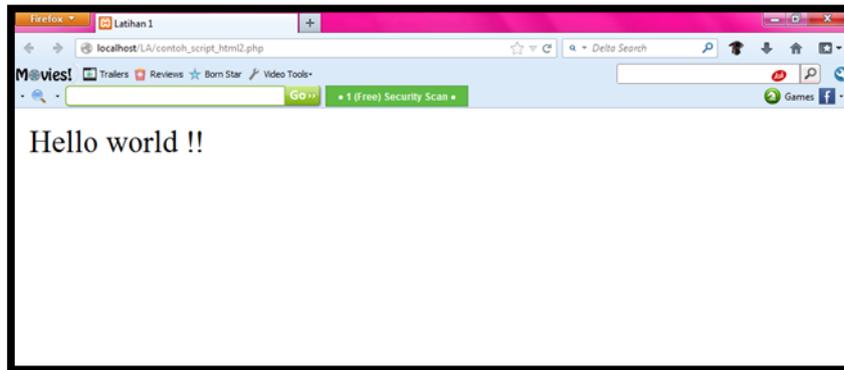
Madcoms (2010:341), “bahasa pemrograman *PHP* adalah bahasa pemrograman yang bekerja dalam sebuah *web server*. *Script-script PHP* dibuat harus tersimpan dalam sebuah *server* dan dieksekusi atau diproses dalam *server* tersebut”.

2.3.1.2. Script PHP

Script PHP berkedudukan sebagai tag dalam bahasa *HTML*. Sebagaimana diketahui, *HTML (Hypertext Markup Language)* adalah bahasa standar untuk membuat halaman-halaman *web*. Sebagai contoh, berikut adalah kode *HTML* (disimpan dengan ekstensi *.htm* atau *.html*):

```
<html>
  <head>
    <title>Latihan 1</title>
  </head>
  <body>
    Hello world !!
  </body>
</html>
```

Bila dijalankan melalui *browser*, kode diatas akan menampilkan hasil seperti berikut:



Gambar 2.1 Tampilan program HTML pada *browser*

2.3.1.3. Teknik Penulisan Script *PHP*

Madcoms (2010:350) menyatakan, “*Script PHP* harus selalu diawali dengan `<? Atau <?php` dan diakhiri dengan `?>`. sedangkan untuk menampilkan ke dalam *browser*, digunakan fungsi `echo”data”;`. Semua teks yang diketik setelah tanda buka *script* (`<?`) dan tanda tutup *script* (`?>`) akan dieksekusi sebagai suatu *script PHP*. Contoh penulisan *script PHP* adalah sebagai berikut:

Tabel 2.1. *Script Dasar PHP*

<code><?</code> <code>?></code>	Tempat penulisan <i>script PHP</i>
atau	
<code><?php</code> <code>?></code>	Tempat penulisan <i>script PHP</i>
Dan untuk menampilkan ke <i>browser</i>	
<code><?php</code> Echo “Welcome to PHP”; <code>?></code>	

Di dalam *script PHP* juga dapat membuat komentar dan komentar tersebut tidak akan dieksekusi sebagai sebuah *script*. Beberapa cara untuk memberi keterangan di dalam *script PHP* adalah sebagai berikut:

- a. Gunakan *tag* `/*` dan akhiri *tag* `*/` apabila jumlah keterangan lebih dari 1 baris.



- b. Gunakan *tag //*, *tag* ini digunakan untuk keterangan yang hanya terdiri dari 1 baris saja.
- c. Gunakan *tag #*, *tag* ini juga digunakan untuk 1 baris komentar saja.

Dalam penulisannya, baris perintah *script PHP* selalu diakhiri dengan menuliskan tanda titik koma (;). Satu baris *script PHP* tidak harus berada dalam satu baris, melainkan dapat menuliskan perintah *script PHP* lebih dari satu baris.

2.3.2. Sekilas Tentang MySQL

2.3.2.1. Pengertian MySQL

Wahana (2010:5), “MySQL adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan *multi-user*”.

Kadir (2013:412), “MySQL adalah nama sebuah *database server* yang menangani akses *database* yang selalu dalam bentuk pernyataan *SQL (Structured Query Language)* yaitu suatu bahasa yang digunakan untuk mengakses *database relasional*”.

Anhar (2010:21), “MySQL (*My Structure Query Language*) adalah sebuah perangkat lunak sistem manajemen basis data *SQL (Database Management System)* atau DBMS dari sekian banyak DBMS, seperti *Oracle, MS SQL, Postagre SQL*, dan lain-lain”.

2.3.2.2. Fungsi-Fungsi MySQL

Untuk dapat menggunakan semua perintah *SQL* pada *PHP*, kita memerlukan fungsi *MySQL*. Fungsi ini berguna untuk mengantarkan perintah *SQL* pada *PHP* menuju ke *server* sehingga perintah tersebut dapat dieksekusi oleh semua *server MySQL*.

Fungsi-fungsi *MySQL* yang berawalan *mysql_* yang digunakan untuk mengakses *database server MySQL* adalah sebagai berikut:

Tabel 2.2. Fungsi-fungsi *MySQL* dalam *PHP*

No.	Fungsi MySQL	Penggunaan
1	<i>Mysql_connect()</i>	Membuat hubungan ke <i>database MySQL</i> yang terdapat pada suatu host



No.	Fungsi MySQL	Penggunaan
2	<i>Mysql_close()</i>	Menutup hubungan ke <i>database MySQL</i>
3	<i>Mysql_select_db()</i>	Memilih <i>database</i>
4	<i>Mysql_query()</i>	Mengeksekusi permintaan terhadap sebuah tabel atau sejumlah tabel
5	<i>Mysql_db_query()</i>	Menjalankan suatu permintaan terhadap suatu <i>database</i>
6	<i>Mysql_num_rows()</i>	Memperoleh jumlah baris dari suatu hasil permintaan yang menggunakan <i>SELECT</i>
7	<i>Mysql_affected_rows()</i>	Memperoleh jumlah baris yang dikenai operasi <i>INSERT, DELETE, UPDATE</i>
8	<i>Mysql_num_fields()</i>	Memperoleh jumlah kolom pada suatu hasil permintaan
9	<i>Mysql_fetch_row()</i>	Menghasilkan <i>array</i> /baris yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasil
10	<i>Mysql_fetch_array()</i>	Menghasilkan <i>array</i> /baris yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasil yang akan disimpan dua kali pada <i>array</i> hasil
11	<i>Mysql_fetch_field()</i>	Menghasilkan informasi suatu kolom
12	<i>Mysql_data_seek()</i>	Memindahkan pointer pada suatu himpunan hasil supaya menunjuk ke baris tertentu
13	<i>Mysql_field_seek()</i>	Memindahkan pointer pada suatu himpunan hasil supaya menunjuk ke kolom tertentu
14	<i>Mysql_create_db()</i>	Membuat <i>database MySQL</i>
15	<i>Mysql_drop_db()</i>	Menghapus <i>database MySQL</i>
16	<i>Mysql_list_dbs()</i>	Menghasilkan daftar <i>database MySQL</i>
17	<i>Mysql_list_tables()</i>	Memperoleh daftar nama tabel dalam suatu <i>database</i>



18	<i>Mysql_list_fields()</i>	Memperoleh daftar nama kolom dalam suatu <i>database</i>
19	<i>Mysql_fetch_assoc()</i>	Mendapatkan array baris dari suatu <i>recordset</i>
20	<i>Mysql_fetch_lengths()</i>	Mendapatkan panjang baris pada setiap isi <i>field</i>
21	<i>Mysql_fetch_object()</i>	Menghasilkan baris dari <i>recordset</i> sebagai sebuah objek
	No.	Fungsi MySQL
22	<i>Mysql_field_len()</i>	Mendapatkan informasi panjang maksimum <i>field</i> dalam sebuah <i>recorset</i>
23	<i>Mysql_field_name()</i>	Mendapatkan informasi nama <i>field</i> dalam <i>recordset</i>
24	<i>Mysql_get_client_info()</i>	Mendapatkan informasi tentang <i>MySQL client</i>
25	<i>Mysql_get_host_info()</i>	Mendapatkan informasi tentang <i>MySQL client</i>
26	<i>Mysql_get_server_info()</i>	Mendapatkan informasi tentang <i>MySQL server</i>
27	<i>Mysql_info()</i>	Mendapatkan informasi tentang <i>query</i> terakhir
28	<i>Mysql_ping()</i>	Memeriksa koneksi <i>server</i> dan akan mencoba untuk melakukan koneksi ulang jika koneksi terputus
29	<i>Mysql_result()</i>	Mendapatkan nilai dari <i>field</i> dalam sebuah <i>recordset</i>
30	<i>Mysql_stat()</i>	Menghasilkan informasi status <i>server</i> terkini

(Sumber: Abdul Kadir (2008:360) dan eWolf Community (2012:157-168))

2.3.2.3. Keistimewaan MySQL

MySQL memiliki beberapa keistimewaan, antara lain:

1. Portabilitas. MySQL dapat berjalan stabil pada berbagai sistem operasi seperti *Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga*, dan masih banyak lagi.
2. *Multi-user*. MySQL dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.



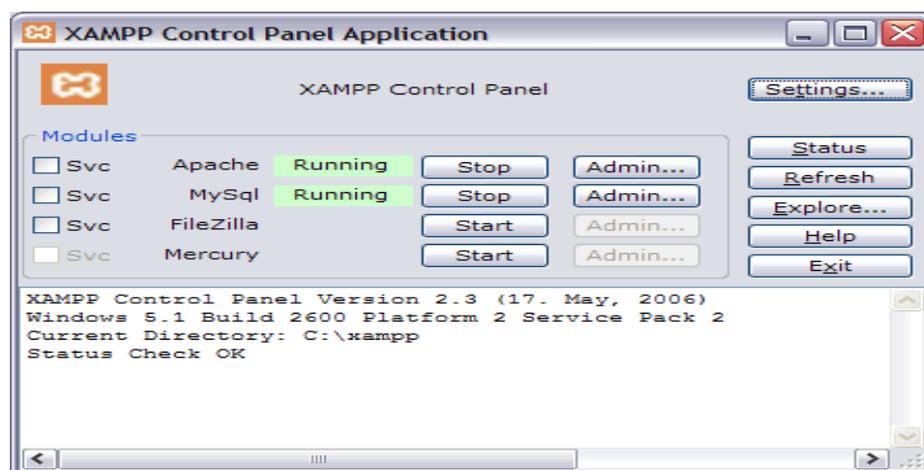
3. Perangkat lunak sumber terbuka. *MySQL* didistribusikan sebagai perangkat lunak sumber terbuka, dibawah lisensi GPL sehingga dapat digunakan secara gratis.
 4. *Performance tuning*. *MySQL* memiliki kecepatan yang menakjubkan dalam menangani *query* sederhana, dengan kata lain dapat memproses lebih banyak *SQL* per satuan waktu.
 5. Ragam tipe data. *MySQL* memiliki ragam tipe data yang sangat kaya, seperti *signed / unsigned integer, float, double, char, text, date, timestamp*, dan lain-lain.
 6. Keamanan. *MySQL* memiliki beberapa lapisan keamanan seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.
 7. Skalabilitas dan Pembatasan. *MySQL* mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
 8. Konektivitas. *MySQL* dapat melakukan koneksi dengan klien menggunakan *protokol TCP/IP, Unix socket (UNIX)*, atau *Named Pipes (NT)*.
 9. Lokalisasi. *MySQL* dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.
 10. Antar Muka. *MySQL* memiliki antar muka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
 11. Klien dan Peralatan. *MySQL* dilengkapi dengan berbagai peralatan (*tool*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk *online*.
 12. Struktur tabel. *MySQL* memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE*, dibandingkan basis data lainnya seperti *PostgreSQL* ataupun *Oracle*.
-



13. Perintah dan Fungsi. *MySQL* memiliki operator dan fungsi secara penuh yang mendukung perintah *Select* dan *Where* dalam perintah (*query*).

Adapun langkah-langkah menjalankan *MySQL* menggunakan *XAMPP* yaitu:

1. Buka *XAMPP Control Panel Application.exe* pada folder *root XAMPP* lalu mengkopi *shortcut* nya pada *desktop* atau dijadikan *start menu*.
2. Sebelum melakukan sesuatu, pastikan tidak ada yang diatur secara otomatis untuk berjalan sebagai *service*, Anda perlu memeriksa di dua tempat. Pertama, pastikan *check box "SVC"* tidak dicentang untuk setiap item komponen *XAMPP*. Kedua, klik tombol *Settings* di *Control Panel*, dan pastikan bahwa tidak ada kotak yang tercentang di *Setting Area*. Klik *OK* untuk kembali ke tampilan utama *Control Panel*.
3. Untuk mengaktifkan *Apache* dan *MySQL on / off*, cukup klik *start/stop* masing-masing tombol untuk masing-masing komponen. Tombol akan mengubah keadaan tergantung pada status komponen. Gambar menunjukkan *Control Panel* bahwa *Apache* dan *MySQL* berjalan.

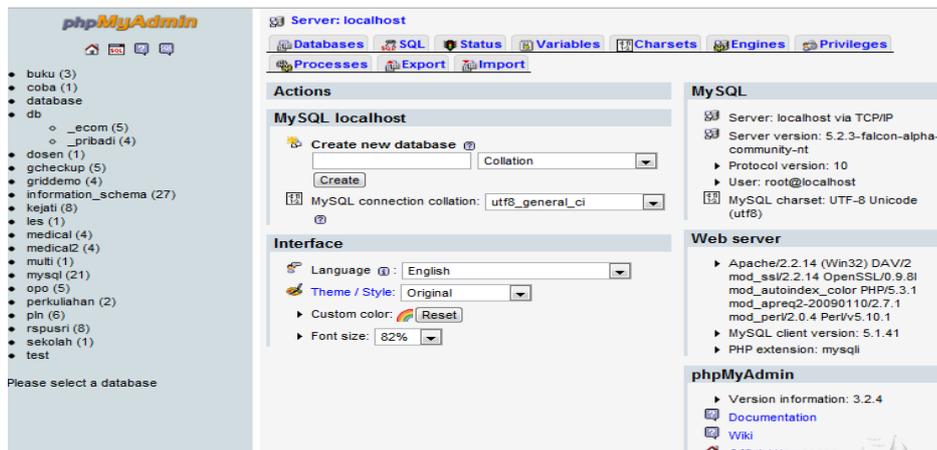


Gambar 2.2 *XAMPP Control Panel Application*

1. Setelah itu anda dapat langsung membuka *XAMPP* melalui *browser* dengan mengetikan <http://localhost/phpmyadmin/>



2. Pilih *Create New Database* untuk membuat *database* baru menggunakan *MySQL*.



Gambar 2.3 Tampilan XAMPP Php MyAdmin pada browser

2.3.3. Sekilas Tentang Adobe Dreamweaver CS5

2.3.3.1. Pengertian Adobe Dreamweaver CS5

Madcoms (2010:1), “*dreamweaver* merupakan *software* aplikasi yang digunakan sebagai *HTML* editor profesional untuk mendesain *web* secara visual. Aplikasi ini dikenal dengan istilah *WYSIWYG* (*What You See Is What Tou Get*), yang intinya adalah kita tidak harus berurusan dengan *tag-tag HTML* untuk membuat sebuah situs. *Dreamweaver* juga memberikan keleluasaan untuk menggunakannya sebagai media penulisan bahasa pemrograman *web*”.

Adobe terus mengeluarkan beberapa varian *Dreamweaver* yang semakin berkembang, salah satunya yaitu *Dreamweaver CS5*. Ada beberapa fitur baru yang dapat ditemukan pada versi ini, diantaranya: integrasi dengan *Adobe Bussiness Catalyst*, integrasi *Adobe Browser Lab*, Integrasi dengan *CMS*, pengecekan *CSS*, dan lain-lain.

2.3.3.2. Starting Page



Gambar 2.4. Tampilan awal *Adobe Dreamweaver CS5*

Jendela diatas menampilkan 4 menu pilihan, yaitu:

1. *Open a Recent Item*

Menu ini menampilkan daftar nama dokumen yang pernah dibuka oleh *Adobe Dreamweaver CS5*. Dokumen tersebut dapat dibuka dengan cara memilih nama dokumen yang ada.

2. *Create New*

Menu ini digunakan untuk membuat dokumen *web* yang baru. Dokumen *web* dapat dibuat dengan beberapa tipe yang disediakan oleh *Adobe Dreamweaver CS5*.

3. *Top Features* (video)

Merupakan fitur yang disediakan *Dreamweaver* melalui *internet*.

4. *Help online*

Merupakan menu pilihan untuk menampilkan fasilitas *help* secara *online* melalui *internet*.

2.3.3.3. Ruang Kerja *Adobe Dreamweaver CS5*

Saat pertama kali menjalankan *Dreamweaver* dengan, akan ditampilkan sebuah kotak dialog *Workspace Setup* yang digunakan untuk memilih tampilan ruang kerja.

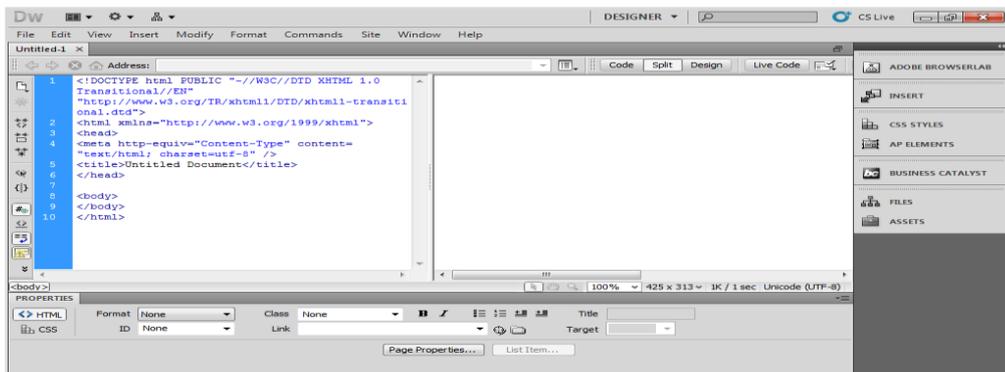
1. ***Designer Workspace***, adalah sebuah penggabungan ruang kerja menggunakan MDI (*Multiple Document Interface*), dimana semua jendela *document* dan panel-panel tergabung di dalam satu jendela aplikasi yang benar, dengan sisi panel group di sebelah kanan.

2. ***Coder Workspace***, adalah ruang kerja yang tergabung sama, tapi letak sisi panel group berada di sebelah kiri.

Ruang kerja *Dreamweaver CS5* memiliki komponen-komponen yang memberikan fasilitas dan ruang untuk menuangkan kreasi saat bekerja, seperti



Application Bar, Toolbar Document, Document Window, Workspace Switcher, Panel Groups, Tag Selector, Property Inspector, Toolbar Browser Navigation.



Gambar 2.5. Tampilan Ruang Kerja *Dreamweaver CS5*

1. **Application Bar**, berada di bagian paling atas jendela aplikasi *Dreamweaver CS5*. Baris ini berisi tombol *workspace* (*workspace switcher*), *CS Live*, menu, dan aplikasi lainnya.



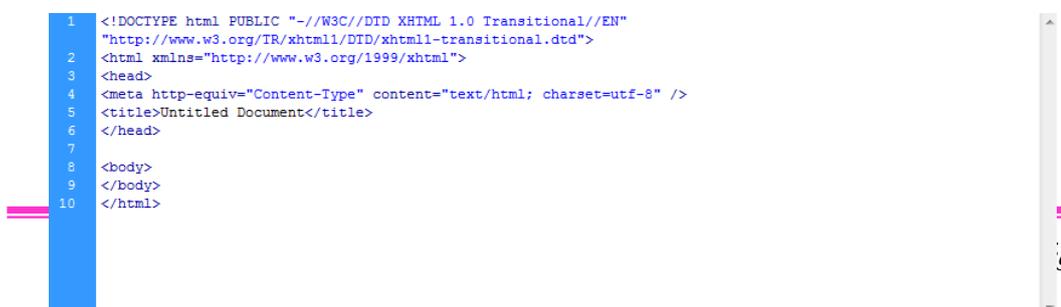
Gambar 2.6. *Application Bar* pada *Dreamweaver CS5*

2. **Toolbar Document**, berisi tombol-tombol yang digunakan untuk mengubah tampilan jendela dokumen, sebagai contoh tampilan *design* atau tampilan kode. Selain itu juga dapat digunakan untuk operasi-operasi umum, misalnya untuk melihat hasil sementara halaman *web* pada jendela *browser*.



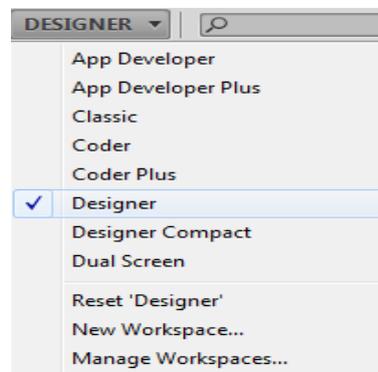
Gambar 2.7. *Toolbar Document* pada *Dreamweaver CS5*

3. **Document Window** (Jendela Dokumen), lembar kerja tempat membuat dan mengedit desain halaman *web*.



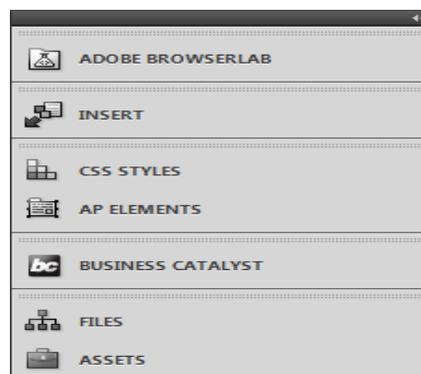
Gambar 2.8. *Document window* pada *Dreamweaver CS5*

4. **Workspace Switcher**, digunakan untuk mengubah tampilan ruang kerja (*workspace*) *Dreamweaver CS5*



Gambar 2.9. *Workspace Switcher* pada *Dreamweaver CS5*

5. **Panel Groups**, adalah kumpulan panel yang saling berkaitan, panel-panel ini dikelompokkan pada judul-judul tertentu berdasarkan fungsinya. Panel ini digunakan untuk memonitor dan memodifikasi pekerjaan.



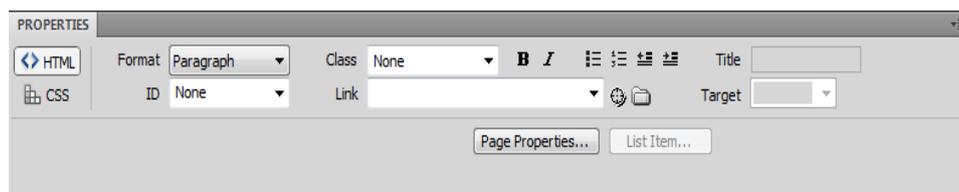
Gambar 2.10. *Panel Groups* pada *Dreamweaver CS5*

6. **Tag Selector**, terletak di bagian bawah jendela dokumen, satu baris dengan *status bar*. Bagian ini menampilkan hirarki pekerjaan yang sedang terpilih pada jendela dokumen, dapat juga digunakan untuk memilih objek pada jendela desain berdasarkan jenis atau kategori objek tersebut. *Tag Selector* juga menampilkan informasi format dari bagian yang sedang aktif pada lembar kerja desain.



Gambar 2.11. *Tag Selector* pada *Dreamweaver CS5*

7. **Property Inspector**, digunakan untuk melihat dan mengubah berbagai *property* objek atau teks pada jendela desain. *Property* untuk satu objek dengan objek lainnya selalu berbeda-beda. Jendela ini tidak dapat diuraikan pada tampilan jendela kode.



Gambar 2.12. *Property Inspector* pada *Dreamweaver CS5*

8. **Toolbar Browser Navigation**, toolbar ini merupakan *toolbar* baru yang ada di dalam *Dreamweaver CS5* dan letaknya tepat berada di atas jendela dokumen. Toolbar ini berisi tombol-tombol yang digunakan sebagai navigasi di dalam *browser*.



Gambar 2.13. *Toolbar Browser Navigation* pada *Dreamweaver CS5*



2.4. Teori Khusus

2.4.1. Pengertian Basis Data (*Database*)

Kristanto (2008:79), “basis data (*database*) adalah kumpulan data yang dapat digambarkan sebagai aktivitas dari satu atau lebih organisasi yang berelasi”.

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

Ladjamudin (2012:21), “*database* adalah kumpulan file yang saling terintegrasi, yang dapat diakses oleh siapapun dengan *software* aplikasi”.

Kadir (2013:411), “*database* adalah istilah yang menyatakan kumpulan data yang disimpan dalam bentuk yang mudah diakses oleh pemakai”.

2.4.2. *Data Flow Diagram* (DFD)

Kristanto (2008:55), “*data flow diagram*/DFD adalah suatu model logika atau proses yang dibuat untuk menggambarkan dari mana asal data dan ke mana tujuan data yang keluar dari sistem, di mana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut”.

Al-Fatta (2007:105), “*data flow diagram*/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 diidentifikasi.

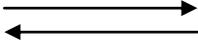
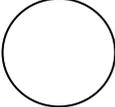
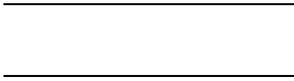
Shalahuddin (2013:70), “*data flow diagram* (DFD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.

Ada beberapa simbol DFD yang dipakai untuk menggambarkan data beserta proses transformasi data, antara lain:



1. Teknik Yourdan Dan De Marco

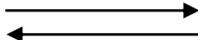
Tabel 2.3 Simbol- Simbol *Data Flow Diagram* Yourdan & De Marco

Simbol	Keterangan
	<i>External Entity</i> , yaitu merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan <i>input</i> atau menerima <i>output</i> dari sistem.
	<i>Data Flow</i> , digunakan untuk menggambarkan aliran data dari satu proses ke proses lainnya.
	<i>Process</i> , digunakan untuk mentransformasikan data secara umum
	<i>Data Store</i> , digunakan untuk menyimpan data seperti: suatu <i>file</i> suatu arsip, suatu kotak, suatu tabel dan suatu agenda.

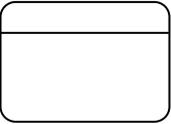
(Sumber: Al Fatta, 2007:107)

2. Teknik Gane dan Sarson

Tabel 2.4 Simbol- Simbol *Data Flow Diagram* Gane & Sarson

Simbol	Keterangan
	<i>External Entity</i> , yaitu merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan <i>input</i> atau menerima <i>output</i> dari sistem.
	<i>Data Flow</i> , digunakan untuk menggambarkan aliran data dari satu proses ke proses lainnya.
Simbol	Keterangan

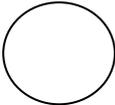
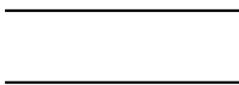


	<p><i>Process</i>, digunakan untuk mentransformasikan data secara umum.</p>
	<p><i>Data Store</i>, digunakan untuk menyimpan data seperti: suatu <i>file</i>, suatu arsip, suatu kotak, suatu tabel dan suatu agenda.</p>

(Sumber: Al Fatta, 2007:107)

Ada beberapa simbol DFD yang dipakai untuk menggambarkan data beserta proses transformasi data, antara lain:

Tabel 2.5 Simbol- Simbol *Data Flow Diagram* menurut Shalahuddin

Notasi	Keterangan
	<p>Proses atau fungsi atau prosedur. Nama yang diberikan pada sebuah proses biasanya berupa kata kerja.</p>
	<p><i>Field</i> atau basis data atau penyimpanan (<i>storage</i>). Nama yang diberikan pada sebuah penyimpanan biasanya berupa kata benda.</p>
	<p><i>Entitas</i> atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang. Nama yang diberikan biasanya berupa kata benda.</p>
	<p>Aliran data; merupakan data yang dikirim antarproses, dari penyimpanan ke proses atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Nama yang diberikan biasanya berupa kata benda, dapat diawali dengan kata data misalnya "data_siswa" atau tanpa kata data misalnya "siswa".</p>

(Sumber: Shalahuddin, 2013:71)



2.4.3 Kamus Data (*Data Dictionary*)

Jogiyanto (2005:725), “kamus data atau *data dictionary* atau disebut juga dengan istilah *systems data dictionary* adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi”.

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

Shalahuddin (2013:73), “kamus data (*data dictionary*) adalah kumpulan daftar elemen data yang mengalir pada sistem sehingga masukan dan keluaran dapat dipahami secara umum (memiliki standar cara penulisan)”.

Adapun simbol-simbol dalam kamus data adalah sebagai berikut:

Tabel 2.6 Simbol-simbol Kamus Data (*Data Dictionary*)

Simbol	Keterangan
=	Disusun, terdiri atas, mendefinisikan, diuraikan menjadi, artinya
+	Dan
()	Opsional/Pilihan (boleh Ya atau Tidak)
[]	Salah satu alternative
*	Komentar, catatan, keterangan
@	Identifikasi atribut kunci, petunjuk (<i>key field</i>)
	Pemisah alternatif simbol []

(Sumber: Kristanto, 2008:66)

2.4.4 Pengertian *Flowchart*

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

Flowchart digunakan untuk mempermudah penyusunan program. Dengan menggunakan *flowchart*, logika pemrograman lebih dapat dipahami dan dianalisis, sehingga dapat menentukan kode-kode pemrograman yang sesuai dengan pekerjaannya.

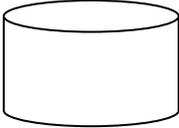


Berikut adalah beberapa simbol standar yang sering digunakan dalam pemrograman komputer, yaitu:

Tabel 2.7 Simbol-simbol *Flowchart*

Simbol	Nama Simbol	Keterangan
	Garis Alir (<i>Flow Line</i>)	Simbol untuk menghubungkan antar proses atau antar simbol
	<i>Connector</i>	Simbol konektor untuk menyambung proses pada lembar kerja yang sama
	<i>Offline Connector</i>	Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
	<i>Process</i>	Simbol pemrosesan yang terjadi pada sebuah alur kerja
	<i>Manual</i>	Simbol yang mendefinisikan proses yang dilakukan secara manual
	<i>Decision</i>	Simbol untuk memutuskan proses lanjutan dari kondisi tertentu
	<i>Terminal</i>	Simbol <i>start</i> atau <i>end</i> yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i>
	<i>Input/Output</i>	Simbol <i>input/output</i> yang mendefinisikan masukan dan keluaran proses
	<i>Magnetic Tape</i>	Simbol masukan atau keluaran dari atau ke sebuah pita magnetik
	<i>Document</i>	Simbol masukan atau keluaran dari atau ke sebuah dokumen



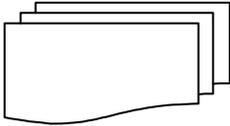
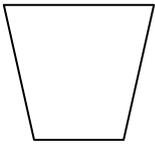
Simbol	Nama Simbol	Keterangan
	<i>Display</i>	Simbol yang menyatakan piranti keluaran, seperti layar monitor, printer, dll
	<i>Data Storage</i>	Simbol database atau basis data
	<i>Sub Program</i>	Simbol yang menyatakan bagian dari program (sub program)

(Sumber: eWolf Community, 2012:17)

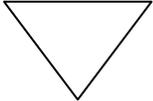
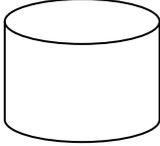
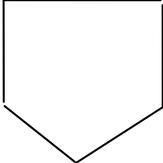
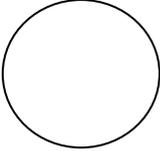
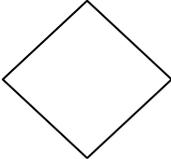
2.4.5 Pengertian *Blockchart*

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. Simbol-simbol yang sering digunakan dalam *blockchart* dapat dilihat pada tabel berikut ini:

Tabel 2.8 Simbol-simbol dalam *Blockchart*

Simbol	Keterangan
	Menandakan dokumen, bisa dalam bentuk surat, formulir, buku/bendel/berkas atau cetakan
	Multi dokumen
	Proses manual.



Simbol	Keterangan
	Proses yang dilakukan oleh komputer.
	Menandakan dokumen yang diarsifkan(arsif manual)
	Data penyimpanan(<i>data storage</i>)
	Proses apa saja yang tidak terdefinisi termasuk aktifitas fisik
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang ain
	Terminasi yang mewakili simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama
	Terminasi yang menandakan awal dan akhir dari suatu aliran
	Pengambilan keputusan (<i>decision</i>)
	Layar peraga (<i>monitor</i>)



Simbol	Keterangan
	Pemasukan data secara manual

(Sumber: Kristanto, 2008:68)

2.4.6 Entity Relational Diagram (ERD)

Al Fatta (2007:121), “*entity relationship diagram/ERD* adalah gambar atau diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis”. Dalam entitas digunakan untuk menghubungkan antar entitas yang sekaligus menunjukkan antar data.

1. Entitas

Entitas bisa berupa orang, kejadian, benda di mana data akan dikumpulkan.

2. Relationship

- a. Hubungan antar 2 entitas.
- b. Entitas pertama dalam *relationship* disebut entitas induk, entitas kedua disebut sebagai entitas anak.
- c. *Relationship* harus memiliki nama yang berupa kata kerja.
- d. *Relationship* berjalan 2 arah.

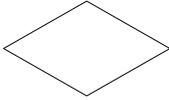
3. Atribut

- a. Informasi yang diambil tentang sebuah entitas.
- b. Hanya yang digunakan oleh organisasi yang dimasukkan dalam model.
- c. Nama atribut harus merupakan kata benda.



Simbol-simbol yang digunakan dalam ERD, yaitu:

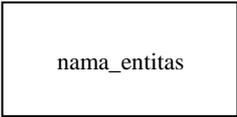
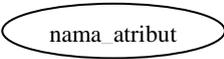
Tabel 2.9 Simbol-simbol *Entity Relationship Diagram (ERD)*

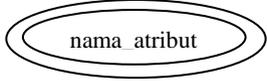
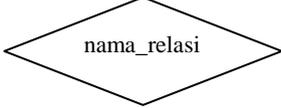
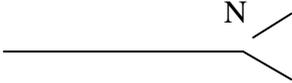
Nama Simbol	Simbol	Keterangan
<i>Entitas</i>		Orang, tempat atau benda memiliki nama tunggal.
<i>Relationship</i>		Menunjukkan hubungan antar 2 entitas. Dideskripsikan dengan kata kerja.
<i>Atribut</i>		Properti dari entitas harus digunakan oleh minimal 1 proses bisnis dipecah dalam detail.
<i>Link</i>		Penghubung antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atributnya.

(Sumber: Hanif Al Fatta, 2007:124)

Simbol-simbol yang digunakan dalam ERD, yaitu:

Tabel 2.10 Simbol- Simbol *Entity Relationship Diagram (ERD)*

Simbol	Deskripsi
Entitas / <i>Entity</i> 	Entitas merupakan data inti yang akan tersimpan; bakal tabel pada basis data; benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer; penamaan entitas biasanya lebih ke kata benda dan belum merupakan nama tabel.
Atribut 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas.

Simbol	Deskripsi
Atribut Kunci Primer 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses <i>record</i> yang diinginkan; biasanya berupa id; kunci primer dapat lebih dari satu kolom, asalkan kombinasi dari beberapa kolom tersebut dapat bersifat unik (berbeda tanpa ada yang sama).
Atribut Multinilai / <i>Multivalued</i> 	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas yang dapat memiliki nilai lebih dari satu.
Relasi 	Relasi yang menghubungkan antarentitas; biasanya diawali dengan kata kerja.
Asosiasi / <i>Association</i> 	Penghubung antara relasi dan entitas di mana di kedua ujungnya punya <i>multiplicity</i> kemungkinan jumlah pemakaian. Kemungkinan jumlah maksimum keterhubungan antara entitas yang lain disebut dengan kardanitas. Misalkan ada kardanitas 1 ke N (sering disebut dengan one to many) menghubungkan entitas A dan B.

(Sumber: Shalahuddin, 2013:50)

Ada beberapa pedoman yang perlu diperhatikan untuk membuat ERD, diantaranya:

- a. Entitas harus memiliki banyak kejadian atau realitas.
- b. Hindari penggunaan atribut yang tidak perlu.
- c. Berilah label yang jelas untuk semua komponen.
- d. Pasangkan kardinalitas dan modalitas yang jelas dan benar.
- e. Pecah atribut menjadi level serendah mungkin yang diperlukan.
- f. Level harus menrefleksikan istilah-istilah bisnis yang umum.
- g. Asumsi harus disebutkan dengan jelas.

