BAB 2

TINJAUAN PUSTAKA

2.1 Sejarah Dinas Pendidikan Kota Cimahi

Pembentukan Dinas Pendidikan Kota Cimahi ditetapkan dengan Peraturan Daerah Nomor 8 tahun 2008 tentang Dinas Daerah Kota Cimahi dan Peraturan Walikota Cimahi Nomor 23 tahun 2008 tanggal 26 Nopember 2008 tentang Tugas Pokok, Fungsi, Tata Kerja dan Uraian Tugas Jabatan Struktural di Lingkungan Dinas Pendidikan. Dinas Pendidikan Kota Cimahi merupakan perangkat daerah sebagai unsur pelaksana teknis di bidang pendidikan dipimpin oleh seorang Kepala yang berada di bawah dan bertanggung jawab kepada Walikota Cimahi. Dinas Pendidikan mempunyai tugas pokok melaksanakan sebagian urusan Pemerintahan Daerah, di bidang Pendidikan. Dalam melaksanakan tugas pokok tersebut Dinas Pendidikan mempunyai fungsi:

a. Perumusan kebijakan teknis bidang Pendidikan, Pemuda dan Olahraga;
b. Penyelenggaraan sebagian urusan pemerintahan dan pelayanan umum di bidang Pendidikan, Pemuda dan Olahraga;
c. Pembinaan dan pelaksanaan tugas di bidang Pendidikan, Pemuda dan Olahraga, meliputi Pendidikan Dasar, Pendidikan Menengah, Pendidikan Non Formal, Pemuda dan Olahraga Peningkatan Mutu Pendidik dan Tenaga Kependidikan (PMPTK) dan Sarana Pendidikan, dan urusan kesekretariatan;

d. Pelaksanaan tugas lain yang diberikan oleh Walikota sesuai dengan tugas pokok dan fungsinya.

2.1.1 Visi Misi Dinas Pendidikan Kota Cimahi

1. Visi Dinas Pendidikan Kota Cimahi

Terselenggaranya layanan pendidikan yang merata dan berkualitas melalui profesionalisme tata kelola menuju masyarakat Cimahi berakhlak mulia, cerdas, dan mandiri.

2. Misi Dinas Pendidikan Kota Cimahi

a) Meningkatkan akses dan kemudahan memperoleh layanan pendidikan bagi masyarakat;

b) Meningkatkan kualitas layanan pendidikan;

c) Mengoptimalkan peran dan fungsi kelembagaan dalam layanan pendidikan.

2.1.2 Struktur Organisasi Dinas Pendidikan Kota Cimahi

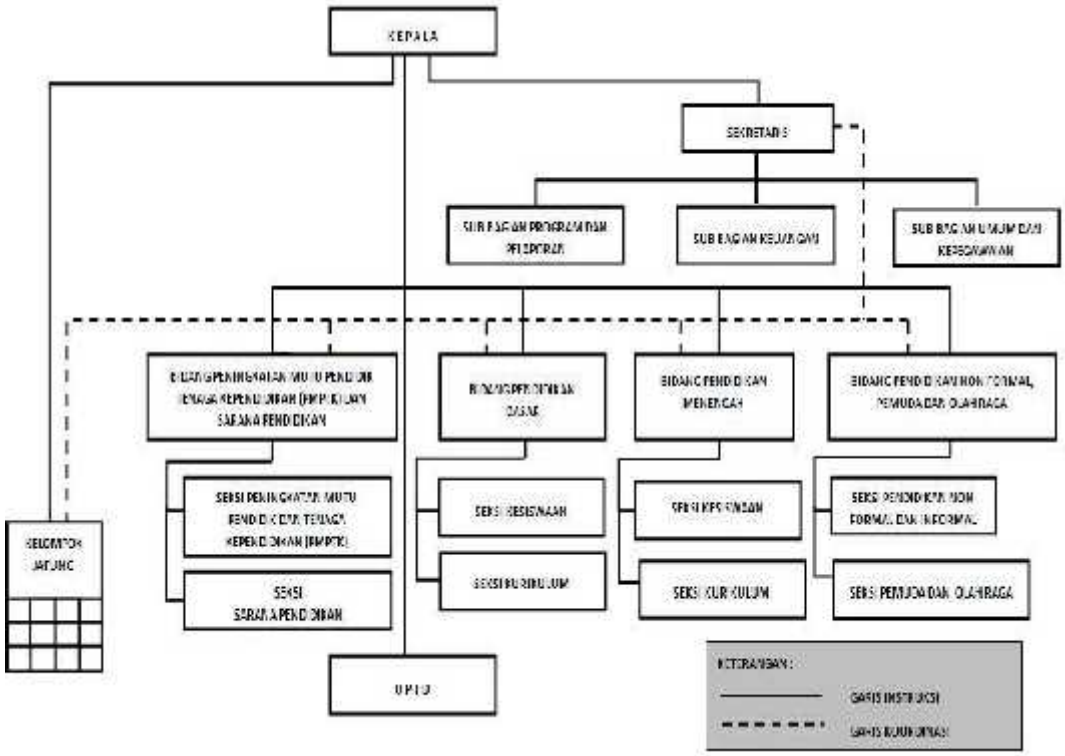
Susunan struktur Organisasi Dinas Pendidikan terdiri dari :

1. Kepala Dinas;
2. Sekretariat, membawahi :
   a. Sub Bagian Program dan Pelaporan;
   b. Sub Bagian Keuangan;
   c. Sub Bagian Umum, Perlengkapan dan Kepegawaian.
3. Bidang Pendidikan Dasar, membawahi :
   a. Seksi Kesiswaan;

b. Seksi Kurikulum.

4. Bidang Pendidikan Menengah, membawahi :
   a. Seksi Kesiswaan;
   b. Seksi Kurikulum.
5. Bidang Pendidikan Non Formal, Pemuda dan Olahraga;
   a. Seksi Pendidikan Non Formal;
   b. Seksi Pemuda dan Olahraga.
6. Bidang Peningkatan Mutu Pendidik dan Tenaga Kependidikan (PMPTK) dan Sarana Pendidikan, membawahi :
   a. Seksi Peningkatan Mutu Pendidik dan Tenaga Kependidikan (PMPTK);
   b. Seksi Sarana Pendidikan.
7. Unit Pelaksana Teknis Sanggar Kegiatan Belajar (SKB);
8. Kelompok Jabatan Fungsional.

Sedangkan bagan struktur organisasi Dinas Pendidikan Kota Cimahi dapat dilihat dalam gambar 2.1.



Gambar 2.1 Struktur Organisasi Dinas Pendidikan Kota Cimahi

2.2 Sekolah Menengah Atas (SMA) yang Ada di kota Cimahi

Di lingkungan Dinas Pendidikan kota Cimahi, ada 16 sekolah menengah atas (SMA) yang terdiri dari 6 SMA negeri dan 10 SMA swasta. Sekolah-sekolah tersebut bisa dilihat pada tabel berikut :

Tabel 2.1 Daftar Sekolah SMA di Cimahi

No.	No Induk	Sekolah	Alamat
1.	20.22.4113	SMAN 1 Cimahi	Jl. Pacinan No. 22A Cimahi
2.	20.22.4139	SMAN 2 Cimahi	Jl. KPAD Sriwijaya IX No. 45 A Cimahi 40524
3.	20.22.4114	SMAN 3 Cimahi	Jl. Pasantren No.161 Cimahi
4.	20.22.4109	SMAN 4 Cimahi	Jl. Kihapit Barat No. 323 Cimahi Selatan 40532

5.	20.22.4112	SMAN 5 Cimahi	Jl. Pacinan No. 23 Cimahi
6.	20.22.4108	SMAN 6 Cimahi	Jl. Melong Raya No. 172 Cimahi
7.	20.22.4115	SMA Budi Luhur	Jl. KPAD Kebon Rumput No.1 Cimahi
8.	20.22.4110	SMA Kartika Siliwangi 4	Jl. Dr.Sam Ratulangi D-26 Cimahi
9.	20.22.4107	SMA Muhammadiyah 1	Jl. Raya Cibabat No. 7B Cimahi Cimahi
10.	20.22.4106	SMA Pasundan 1 Cimahi	Jl. Terusan No.32 Cimahi
11.	20.22.4105	SMA Pasundan 2 Cimahi	Jl. Melong Raya No. 4 Perumnas Cijerah II Cimahi 40534
12.	20.22.4121	SMA Pasundan 3 Cimahi	Jl. Encep Kartawiria No. 97/A Cimahi 40512
13.	20.22.4122	SMA Putra Mandiri	Jl. Sangkuriang No.36 Ex.804 Cimahi
14.	20.22.4124	SMA Santa Maria 3	Jl. Gatot Subroto No.6 Cimahi
15.	20.22.4138	SMA Tut Wuri Handayani	Jl. Encep Kartawiria No. 93 Cimahi
16.	20.22.4137	SMA Warga Bakti	Jl. Raya Cibeber No.148

2.3 Sejarah Ujian Nasional

Ujian Nasional (UN) yang memimbulkan sikap pro kontra di kalangan masyarakat sebenarnya telah dilaksanakan sejak tahun 1965, namanya mengalami Evolusi sampai akhirnya bernama Ujian Nasional.

Pada jenjang pendidikan dasar dan menengah, sistem ujian nasional telah mengalami beberapa kali perubahan dan penyempurnaan, perkembangan ujian nasional tersebut dapat kita lihat di bawah ini :

1) Periode 1965 - 1971, pada periode ini, sistem ujian akhir disebut dengan Ujian Negara, berlaku untuk hampir semua mata pelajaran. Bahkan ujian dan pelaksanaannya ditetapkan oleh pemerintah pusat dan seragam untuk seluruh wilayah di Indonesia.

2) Periode1972 - 1979, pada tahun 1972 ditetapkan sistem ujian sekolah dimana setiap atau sekelompok sekolah menyelenggarakan ujian akhir sekolah masing-masing. Soal dan pemrosesan hasil ujian semuanya ditentukan oleh masing-masing sekolah/kelompok sekolah. Pemerintah pusat hanya menyusun dan mengeluarkan pedoman yang bersifat umum.
3) Periode 1980-2000, untuk meningkatkan dan mengendalikan mutu pendidikan serta diperolehnya nilai yang memiliki makna yang "sama" dan dapat dibandingkan antar sekolah ,maka sejak tahun 1980 dilaksanakan ujian akhir nasional yang dikenal dengan sebutan Evaluasi Belajar Tahap Akhir Nasional. Dalam EBTANAS dikembangkan sejumlah perangkat soal yang "paralel" untuk setiap mata pelajaran , dan penggandaan soal dilakukan di daerah.
4) Periode 2001-2004, sejak tahun 2001, EBTANAS diganti dengan penilaian hasil belajar secara nasional dan kemudian berubah nama menjadi Ujian Akhir Nasional, sejak tahun 2002. Perbedaan yang menonjol antara UAN dan EBTANAS adalah dalam cara menentukan kelulusan siswa, terutama sejak tahun 2003. Dalam EBTANAS kelulusan siswa ditentukan oleh kombinasi nilai semester 1, nilai semester 2, dan nilai EBTANAS murni, sedangkan kelulusan siswa pada UAN ditentukan oleh nilai mata pelajaran secara individual.
5) Periode 2005-sekarang, untuk mendorong tercapainya target wajib belajar pendidikan yang bermutu, pemerintah menyelenggarakan Ujian

Nasisonal untuk SMP / MTs / SMPLB dan SMA / SMK / MA / SMALB / SMKLB.

6) Periode 2008-sekarang, untuk mendorong tercapainya target wajib belajar pendidikan yang bermutu, mulai tahun ajaran 2008 /2009 pemerintah menyelenggarakan Ujian Akhir Sekolah Berstandar Nasional untuk SD / MI / SDLB.

2.3.1 *Tryout* Ujian Nasional

Ujian nasional merupakan ujian akhir siswa kelas VI, IX, dan XII untuk bisa melanjutkan pendidikan ke tahap yang lebih tinggi lagi. Jadi sudah jelas, apabila ada siswa yang tidak lulus dalam ujian nasional, maka siswa tersebut tidak bisa melanjutkan pendidikan ke jenjang berikutnya.

Untuk mempersiapkan diri dalam menghadapi ujian nasional, sudah menjadi rahasia umum bila setiap sekolah selalu mengadakan *tryout* (uji coba) mengerjakan soal-soal prediksi ujian nasional. *Tryout* ujian nasional bisa juga disebut simulasi dari ujian nasional yang sesungguhnya. Para siswa diberikan soal-soal dan harus dikerjakan layaknya ujian nasional yang sebenarnya.

Tryout ujian nasional tersebut dimaksudkan agar para siswa bisa mempersiapkan diri lebih baik, dan mengetahui bagaimana ujian nasioal itu berlangsung.

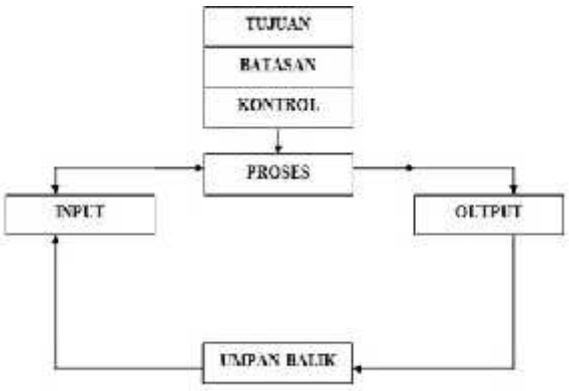
2.4 Pengertian Sistem

Kata sistem mengandung arti kumpulan dari komponen-komponen yang memiliki unsur keterkaitan antara satu dan yang lainnya. Sistem adalah kumpulan elemen-elemen yang saling berkaitan dan bertanggung jawab memproses masukan (*input)* sehingga menghasilkan keluaran *(output).*

Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan,berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu,dengan kata lain kumpulan dari elemen-elemen yang berinteraksi untuk tujuan tertentu.

Elemen-elemen yang terdapat dalam sistem meliputi : tujuan sistem, batasan sistem, kontrol, i*nput, p*roses,*output*, dan umpan balik.

Hubungan antara elemen-elemen dalam sistem dapat dilihat pada gambar 2.2 di bawah ini :



Gambar 2.2 Elemen-elemen dalam sistem

2.5 Karakteristik Sistem

Suatu Sistem mempunyai karateristik atau sifat-sifat tertentu, yaitu mempunyai komponen-komponen (*components)*, batas sistem (*boundary*),

lingkungan luar sistem (*environments*), penghubung (*interface),* masukan (*input),* keluaran (*output),* pengolah (*process),* dan sasaran (*objectives),* atau tujuan (*goal).*

1) **Kompenen Sistem (*components)***

   Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Setiap sistem selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu sistem yang lebih besar yang disebut dengan s*uprasystem.*

2) **Batas Sistem (*boundary)***

   Batas Sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan.

3) **Lingkungan Luar Sistem *(environments)***

   Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang memepengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan atau pun merugikan sistem tersebut. Lingkungan luar yang menguntungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan agar tidak mengganggu kelangsungan hidup dari sistem.

4) Penghubung Sistem (*interface)*

Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem yang lainya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainya. Keluaran dari satu subsistem merupakan masukan bagi subsistem lainya dengan melaui penghubung. Dengan melaui penghubung satu subsistem dapat beritegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

5) Masukan Sistem (*input)*

Masukan adalah energi yang dimasukan kedalam sistem. Masukan dapat berupa masukan perawatan dan masukan sinyal.

6) Keluaran Sistem (*output)*

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

7) Pengolahan Sistem

Suatu sistem dapat mempunyai satu bagian pengolah yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi.

8) Sasaran Sistem (*objectives)*

Suatu sistem pasti mempunyai tujuan *(goal*) atau sasaran (*objectives*). Kalau suatu sistem tidak mempunyai tujuan atau sasaran, maka operasi sistem tidak ada gunanya. Sasaran dari sistem sangat menetukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem.

2.6 Perancangan Sistem

2.6.1 Diagram Konteks

Diagram konteks adalah suatu alat atau metode penggambaran suatu sistem informasi secara global, baik sistem informasi yang berbasis komputer atau tidak berbasis komputer. Diagram konteks terdiri dari sebuah simbol proses yang mewakili keseluruhan proses dalam sistem dan minimal sebuah *external entity* (entitas luar) yang merupakan sumber atau tujuan data dari sistem tersebut dan aliran data yang menggambarkan aliran suatu masukan ataupun keluaran dari sistem tersebut. Berdasarkan notasi Yourdon proses digambarkan dengan lingkaran, entitas luar dengan persegi panjang, dan aliran data digambarkan dengan garis yang diberi mata panah.

2.6.2 Diagram Arus Data

DAD (Diagram Arus Data) merupakan alat yang digunakan pada metodologi pengembangan sistem yang tersruktur (*Structured Analysis and Design*). DAD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa

mempertimbangkan lingkungan fisik dimana data tersebut akan disimpan. Untuk mewakili arus data dalam suatu sistem digunakan notasi atau simbol sehingga sangat membantu dalam komunikasi dengan pemakai sistem untuk memahami sistem secara logika.

Beberapa simbol yang sering digunakan di DAD untuk maksud mewakili :

1. External entity (kesatuan luar) atau boundary (batas sistem)

Setiap sistem pasti mempunyai batas sistem (*boundary*) yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima input dan menghasilkan output kepada lingkungan luarnya. Kesatuan luar (*external entity*) merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan input atau menerima output dari sistem. Suatu kesatuan luar dapat disimbolkan dengan suatu notasi kotak sebagai berikut :

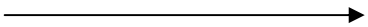


Gambar 2.3 Notasi kesatuan luar

2. Data flow (arus data)

Arus data menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem. Arus data ini mengalir diantara proses, simpanan data dan kesatuan luar. Arus data di DAD diberi simbol

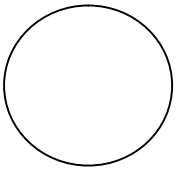
suatu panah. Arus data sebaiknya diberi nama yang jelas dam mempunyai arti. Nama dari arus data dituliskan diatas garis panahnya sebagai berikut :



Gambar 2.4 Notasi Arus data

3. Process (proses)

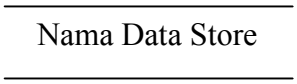
Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk kedalam proses untuk dihasilkan arus data yang akan keluar dari proses. Suatu proses data ditunjukkan dengan simbol lingkaran. Setiap proses harus diberi penjelasan yang lengkap meliputi identifikasi proses, nama proses dan pemroses.



Gambar 2.5 Notasi Proses

4. Data Store (Simpanan data)

Simpanan data merupakan simpanan dari data yang dapat berupa suatu file atau database di sistem komputer. Simpanan data di DAD dapat disimbolkan dengan sepasang garis horizontal paralel.



Gambar 2.6 Notasi Simpanan data

2.6.3 Kamus Data

Kamus data (*data dictionary*) adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Kamus data dibuat pada tahap analisis maupun pada tahap perencanaan sistem. Pada tahap analisis, kamus data dapat digunakan sebagai alat komunikasi antara analis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem. Kamus data harus memuat hal-hal berikut ini:

1. Nama arus data
2. Panjang karakter
3. Tipe data
4. Deskripsi field

2.7 Konsep *database*

Database terdiri dari dua kata, yaitu *data* dan *base*. *Base* merupakan tempat untuk berkumpul, sedangkan *data* adalah fakta-fakta yang dapat disimpan dan mewakili suatu objek seperti manusia, hewan, peristiwa, konsep, keadaan, barang, dan sebagainya yang direkam dalam bentuk angka, huruf, symbol, teks, gambar ataupun kombinasinya. Secara umum *database* dapat diartikan sebagai kumpulan data tentang suatu benda atau kejadian yang disimpan dan saling berhubungan satu sama lain.

2.7.1 Perancangan *Database*

Database adalah kumpulan dari *item* data yang saling berhubungan satu dengan lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dan dengan *software* untuk melakukan manipulasi untuk kegunaan tertentu. Beberapa istilah yang berhubungan dengan database, yaitu:

1. *Entity*

   Entity adalah suatu objek nyata dan informasinya akan direkam. Pada sistem pemesanan *online* misalnya, *entity* adalah anggota, produk, admin.

2. *Atribute*

   Setiap *entity* memiliki *Atribute* atau sebutan untuk mewakili suatu *entity*. Misalnya, anggota memiliki *atribute* id_anggota, nama, *password*, alamat.

3. Data *Value*

   Data aktual atau informasi yang disimpan pada tiap data elemen atau *atribute*. *Atribute* nama anggota menunjukkan tempat dimana informasi nama anggota disimpan, sedang data *value* adalah isi data nama anggota tersebut.

4. *Record*

   Record adalah kumpulan elemen-elemen yang saling berkaitan menginformasikan tentang suatu *entity* secara lengkap. Satu *record* mewakili satu data atau informasi tentang pesan misalnya, id_pesan, no_anggota, kd_produk, tgl_pesan, jam_pesan.

5. *File*

   File adalah kumpulan *record-record* sejenis yang mempunyai panjang elemen yang sama, *atribute* yang sama, namun berbeda-beda data *value*-nya.

Tujuan dalam pengelolaan data dalam database adalah mengelompokkan data-data sesuai dengan fungsi dan untuk memenuhi sejumlah tujuan organisasi/perusahaan, oleh sebab itu pengembangan sistem database berhubungan dengan masalah-masalah pada file-file database yang harus dihindari yaitu:

1. Redudansi dan inkonsistensi data.
2. Kesulitan pengaksesan data.
3. Isolasi data untuk standarisasi.
4. *Multiple user.*
5. *Security.*
6. Masalah integritas.
7. Masalah data *independence.*

2.7.2 Normalisasi

Normalisasi adalah suatu proses untuk membuat data yang tidak normal menjadi data yang normal (tanpa redudansi). Bentuk data yang tidak normal/ data mentah biasa juga disebut *unnormalized form. unnormalized form* merupakan kumpulan data yang tidak mengikuti aturan tertentu, tidak lengkap atau terduplikasi.

Hal-hal yang menjadi dasar proses normalisasi adalah sebagai berikut:

1) Lossie-Join Decompotion

Dilakukan dekomposisi memiliki pengertian, yaitu melakukan pemilahan terhadap suatu tabel menjadi beberapa tabel dengan mempertimbangkan Ketergantungan Fungsional. Pada prinsipnya,

dekomposisi agar setiap tabel yang digunakan hanya memiliki 1 KF (lebih tepatnya KF Minimum), karena sebuah KF bisa saja diuraikan menjadi sejumlah KF. Dan ini dapat menjadi salah satu indikasi tentang baik/buruknya sebuah tabel.

2) Dependency Preservation

Dependency Preservation (dapat diartikan sebagai Pemeliharaan ketergantungan) merupakan kriteria kedua yang harus dicapai ketika melakukan perubahan data, maka harus bisa dijamin agar perubahan tersebut tidak menghasilkan inkonsistensi data yang mengakibatkan Ketergantungan Fungsional yang sudah benar menjadi tidak terpenuhi.

2.7.3 Model Ketergantungan Entitas (E*ntity Relationship Diagram*)

ERD merupakan pemodelan konseptual yang mendeskripsikan hubungan antar data dalam bentuk diagram. ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data, pada dasarnya ada 3 macam simbol yang digunakan yaitu:

a. Entitas *(Entity)*

Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain.

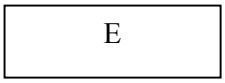
b. Atribut *(Attributes/Properties)*

Atribut merupakan bagian yang mendeskripsikan karakteristik dari entitas.

c. Hubungan

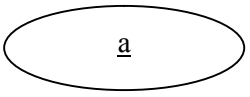
Sebagaimana halnya entitas maka dalam hubunganpun harus dibedakan antara hubungan atau bentuk hubungan antar entiti dengan isi dari hubungan itu sendiri. Model *Entity Relationship* yang berisi komponen-komponen himpunan entitas dan himpunan relasi, yang masing-masing dengan atribut yang merepresentasikan seluruh fakta, dan digambarkan dengan lebih sistematis dengan menggunakan diagram E-R. Notasi simbolik di dalam diagram E-R yang digunakan adalah:

1) Persegi panjang yang menyatakan himpunan entitas.



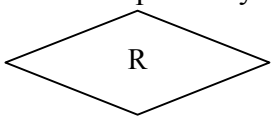
Himpunan entitas E.

2) Lingkaran atau *elips* yang menyatakan atribut.



Atribut a sebagai key.

3) Belah ketupat menyatakan himpunan relasi.



Himpunan relasi R.

4) Garis sebagai penghubung antara himpunan relasi (menyatakan *cardinality ratio* dan *participation constraint)* dengan himpunan entitas dan himpunan entitas dengan atributnya.

———————— *Partial Participation* (keikutsertaan sebagian anggota entitas).

———————— *Total Participation* (keikutsertaan semua anggota entitas).

Hubungan antar entitas *(Entity Relationship)* dapat dikategorikan dalam 3 macam hubungan, yaitu:

1) Satu ke Satu *(One to One)*

   Berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B dan begitu juga sebaliknya.

2) Satu ke Banyak *(One to Many)*

   Berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B. Tetapi sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak satu entitas pada himpunan entitas A.

3) Banyak ke Banyak *(Many to Many)*

   Berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B. Demikian juga sebaliknya.

2.8 Model Pengembangan Perangkat Lunak

2.8.1 Model Sekuensial Linear (*Waterfall*)

Model sekuensial linier untuk *software engineering,* sering disebut juga dengan **siklus kehidupan klasik** atau **model air terjun (*waterfall*)**. Model ini mengusulkan sebuah pendekatan kepada perkembangan software yang sistematik dan sekuensial yang mulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan. Dimodelkan setelah siklus rekysa konvensional, model sekuensial linier melingkupi aktivitas – aktivitas sebagai berikut :

1. Rekayasa dan pemodelan sistem/informasi

Karena sistem merupakan bagian dari sebuah sistem yang lebih besar, kerja dimulai dengan membangun syarat dari semua elemen sistem dan mengalokasikan beberapa subset dari kebutuhan ke software tersebut. Pandangan sistem ini penting ketika software harus berhubungan dengan elemen-elemen yang lain seperti software, manusia, dandatabase. Rekayasa dan anasisis sistem menyangkut pengumpulan kebutuhan pada tingkat sistem dengan sejumlah kecil analisis serta disain tingkat puncak. Rekayasa informasi mancakup juga pengumpulan kebutuhan pada tingkat bisnis strategis dan tingkat area bisnis.

2. Analisis kebutuhan Software

Proses pengumpulan kebutuhan diintensifkan dan difokuskan, khusunya pada software. Untuk memahami sifat program yang dibangun, analis harus memahami domain informasi, tingkahlaku, unjuk kerja, dan interface yang diperlukan. Kebutuhan baik untuk sistem maupun software didokumentasikan dan dilihat lagi dengan pelanggan.

3. Desain

Desain software sebenarnya adalah proses multi langkah yang berfokus pada empat atribut sebuah program yang berbeda; struktur data, arsitektur software, representasi interface, dan detail (algoritma) prosedural. Proses desain menterjemahkan syarat/kebutuhan ke dalam sebuah representasi software yang dapat diperkirakan demi kualitas sebelum dimulai pemunculan kode. Sebagaimana persyaratan,

desain didokumentasikan dan menjadi bagian dari konfigurasi software.

4. Generasi Kode

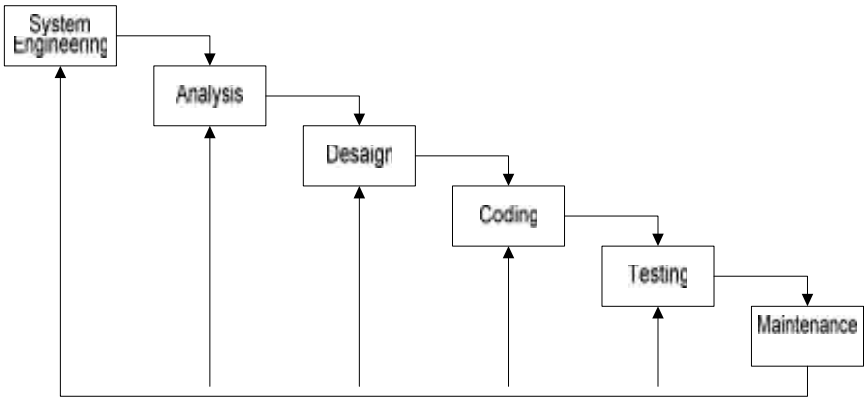
Desain harus diterjemahkan kedalam bentuk mesin yang bisa dibaca. Langkah pembuatan kode melakukan tugas ini. Jika desain dilakukan dengan cara yang lengkap, pembuatan kode dapat diselesaikan secara mekanis.

5. Pengujian

Sekali program dibuat, pengujian program dimulai. Proses pengujian berfokus pada logika internal software, memastikan bahwa semua pernyataan sudah diuji, dan pada eksternal fungsional, yaitu mengarahkan pengujian untuk menemukan kesalahan – kesalahan dan memastikan bahwa input yang dibatasi akan memberikan hasil aktual yang sesuai dengan hasil yang dibutuhkan.

6. Pemeliharaan

Software akan mengalami perubahan setelah disampaikan kepada pelanggan (perkecualian yang mungkin adalah software yang dilekatkan). Perubahan akan terjadi karena kesalahan – kesalahan ditentukan, karena software harus disesuaikan untuk mengakomodasi perubahan–perubahan di dalam lingkungan eksternalnya (contohnya perubahan yang dibutuhkan sebagai akibat dari perangkat peripheral atau sistem operasi yang baru), atau karena pelanggan membutuhkan perkembangan fungsional atau unjuk kerja. Pemeliharaan software mengaplikasikan lagi setiap fase program sebelumnya dan tidak membuat yang baru lagi.



Gambar 2.7 Model Waterfall

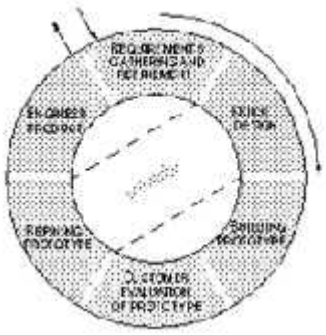
Masalah yang kadang terjadi ketika model sekuensial linier diaplikasikan adalah :

1. Jarang sekali proyek nyata mengikuti aliran sekuensial yang dianjurkan oleh model. Meskipun model linier bisa mengakomodasi iterasi, model ini melakukannya dengan cara tidak langsung. Sebagai hasilnya, perubahan–perubahan dapat menyebabkan keraguan pada saat tim proyek berjalan.
2. Kadang – kadang sulit bagi pelanggan untuk menyatakan semua kebutuhannya secara eksplisit. Model linier sekuensial memerlukan hal ini dan mengalami kesulitan untuk mengakomodasi ketidakpastian natural yang ada pada bagian awal beberapa proyek.
3. Pelanggan harus bersifat sabar. Sebuah versi kerja dari program – program kerja itu tidak akan diperoleh sampai akhir waktu proyek dilalui. Sebuah kesalahan besar, jika tidak terdeteksi sampai program yang bekerja tersebut dikaji ulang, bisa menjadi petaka.
4. Pengembang sering melakukan penundan yang tidak perlu. Sifat alami dari siklus kehidupan klasik membawa kepada *blocking state* di mana

banyak anggota tim proyek harus menunggu tim yang lain untuk melengkapi tugas yang saling memiliki ketergantungan. Blocking state cenderung menjadi lebih lazim pada awal dan akhir sebuah proses sekuensial linier.

2.8.2 Model *Prototype*

Prototyping paradigma dimulai dengan pengumpulan kebutuhan. Pengembang dan pelanggan bertemu dan mendefinisikan obyektif keseluruhan dari software, mengidentifikasi segala kebutuhan yang diketahui, dan area garis besar diman definisi lebih jauh merupakan keharusan kemudian dilakukan "perancangan kilat". Perancangan kilat berfokus pada penyajian dari aspek – aspek software tersebut yang akan nampak bagi pelanggan atau pemakai (contohnya pendekatan input dan format output). Perancangan kilat membawa kepada konstruksi sebuah prototipe. Prototipe tersebut dievaluasi oleh pelanggan/pemakai dan dipakai untuk menyaring kebutuhan pengembangan software. Iterasi terjadi pada saat prototipe disetel untuk memenuhi kebutuhan pelanggan, dan pada saat yang sama memungkinkan pengembang untuk secara lebih baik memahami apa yang harus dilakukannya.



Gambar 2.8 Prototipe Paradigma

Prototipe bisa juga menjadi masalah karena alasan sebagai berikut:

1) Pelanggan melihat apa yang tampak sebagai versi software yang bekerja tanpa melihat bahwa prototipe itu dijalin bersama – sama "dengan permen karet dan *baling wire*", tanpa melihat bahwa di dalam untuk membuatnya bekerja, kita belum menyantumkan kualitas software secara keseluruhan atau kemampuan pemeliharaan untuk jangka waktu yang panjang. Ketika diberi informasi bahwa produk harus dibangun lagi agar tingkat kualitas yang tinggi bisa dijaga, pelanggan akan meneriakan kecurangan dan permintaan agar dipakai "beberapa campuran" untuk membuat prototipe menjadi sebuah produk yang bekerja yang lebih sering terjadi, sehingga manajemen pengembangan software menjadi penuh dengan belas kasihan.
2) Pengembang sering membuat kompromi – kompromi implementasi untuk membuat prototipe bekerja dengan cepat. Sistem operasi atau bahasa pemrograman yang tidak sesuai bisa dipakai secara sederhana karena mungkin diperoleh dan dikenal; algoritma yang tidak efisien secara sederhana bisa diimplementasikan untuk mendemonstrasikan kemampuan. Setelah selang waktu tertentu, pengembang mungkin mengenali pilihan – pilihan tersebut dan melupakan semua alasan mengapa mereka tidak cocok. Pilihan yang kurang ideal telah menjadi bagian integral dari sebuah sistem.

Meskipun berbagai masalah bisa terjadi, prototipe bisa menjadi paradigma yang efektif bagi *Software Engineering.* Kuncinya adalah mendefinisikan aturan

main pada saat awal; yaitu pelanggan dan pengembang keduanya harus setuju bahwa prototipe dibangun untuk berfungsi sebagai mekanisme pendefinisian kebutuhan. Prototipe kemudian disingkirkan (paling tidak sebagian), dan software aktual direkayasa dengan tertuju kepada kualitas dan kemampuan pemeliharaan.

2.8.3 Model *Rapid Aplication Development* (RAD)

Rapid Aplication Development (RAD) adalah sebuah model proses perkembangan software sekuensial linier yang menekankan siklus perkembangan yang sangat pendek. Model RAD ini merupakan sebuah adaptasi "kecepatan tinggi" dari model sekuensial linier di mana perkembangan cepat dicapai dengan menggunakan pendekatan kontruksi berbasis komponen. Jika kebutuhan dipahami dengan baik, proses RAD memungkinkan tim pengembangan menciptakan "sistem fungsional yang utuh" dalam periode waktu yang sangat pendek (kira-kira 60 sampai 90 hari). Karena dipakai terutama pada aplikasi sistem konstruksi, pendekatan RAD melingkupi fase – fase sebagai berikut :

1) Bussiness modeling

Aliran informasi di antara fungsi – fungsi bisnis dimodelkan dengan suatu cara untuk menjawab pertanyaan – pertanyaan berikut : informasi apa yang mengendalikan proses bisnis? Informasi apa yang di munculkan? Siapa yang memunculkanya? Ke mana informasi itu pergi? Siapa yang memprosesnya?

2) Data modeling

Aliran informasi yang didefinisikan sebagai bagian dari fase bussiness modelling disaring ke dalam serangkaian objek data yang dibutuhkan untuk

menopang bisnis tersebut. Karakteristik (disebut atribut) masing–masing objek diidentifikasi dan hubungan antara objek – objek tersebut didefinisikan.

3) Prosess modelling

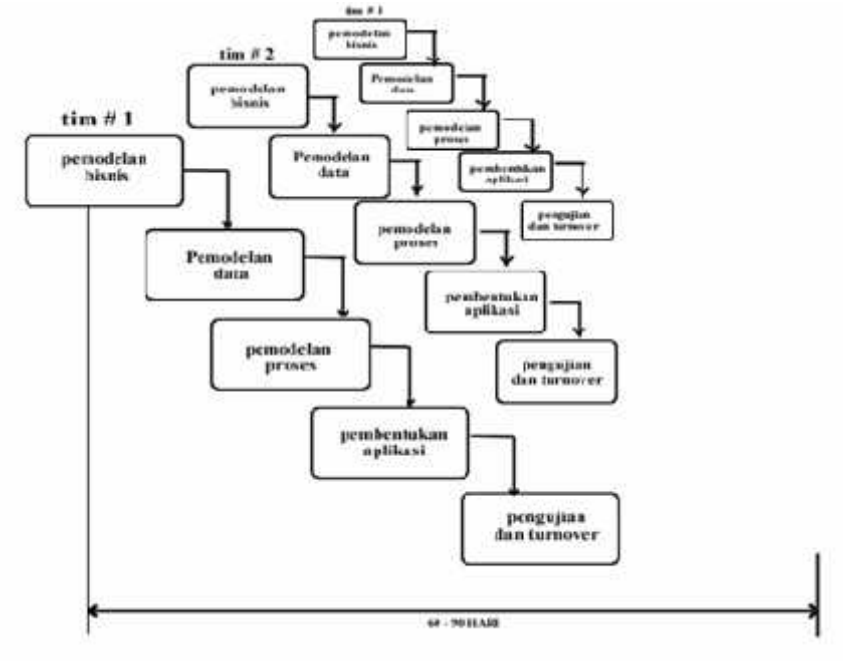
Aliran informasi yang didefinisikan di dalam fase data modeling ditransformasikan untuk mencapai aliran informasi yang perlu bagi implementasi sebuah fungsi bisnis. Gambaran pemrosesan diciptakan untuk menambah, memodifikasi, menghapus, atau mendapatkan kembali sebuah objek data.

4) Aplication generation

RAD mengasumsikan pemakaian teknik generasi ke empat. Selain menciptakan perangkat lunak dengan menggunakan bahasa pemrograman generasi ketiga yang konvensional, RAD lebih banyak memproses kerja untuk memkai lagi komponen program yang ada (pada saat memungkinkan) atau menciptakan komponen yang bisa dipakai lagi (bila perlu). Pada semua kasus, alat – alat bantu otomatis dipakai untuk memfasilitasi konstruksi perangkat lunak.

5) Testing and turnover

Karena proses RAD menekankan pada pemakaian kembali, banyak komponen program telah diuji. Hal ini mengurangi keseluruhan waktu pengujian. Tetapi komponen baru harus di uji dan semua interface harus dilatih secara penuh.



Gambar 2.9 Model RAD

Kekurangan model RAD adalah :

1) Bagi proyek yang besar tetapi berskala, RAD memerlukan sumber daya manusia yang memadai untuk menciptakan jumlah tim RAD yang baik.
2) RAD menuntut pengembangan dan pelanggan memiliki komitmen di dalam aktivitas *rapid-fire* yang diperlukan untuk melengkapi sebuah sistem, di dalam kerangka waktu yang sangat diperpendek. Jika komitmen tersebut tidak ada, proyek RAD akan gagal.

2.8.4 Model Spiral

Model spiral (*spiral model*) adalah model proses software yang evolusioner yang merangkai sifat iteratif dari prototipe dengan cara kontrol dan aspek sistematis dari model sekuensial linier. Model ini berpotensi untuk pengembangan versi pertambahan software secara cepat. Di dalam model spiral, software dikembangkan di dalam suatu deretan pertambahan. Selama awal iterasi, rilis

inkremental bisa merupakan sebuah model atau prototipe kertas. Selama iterasi berikutnya, sedikit demi sedikit dihasilkan versi sistem rekayasa yang lebih lengkap.

Model spiral dibagi menjadi sejumlah aktifitas kerangka kerja, disebut juga wilayah tugas, di antara tiga sampai enam wilayah tugas, yaitu :

1) Komunikasi Pelanggan

Tugas – tugas yang dibutuhkan untuk membangun komunikasi yang efektif di antara pengembangan dan pelanggan.

2) Perencanaan

Tugas – tugas yang dibutuhkan untuk mendefinisikan sumber – sumber daya, ketepatan waktu, dan proyek informasi lain yang berhubungan.

3) Analisis Risiko

Tugas – tugas yang dibutuhkan untuk menaksir risiko – risiko, baik manajemen maupun teknis

4) Perekayasaan

Tugas – tugas yang dibutuhkan untuk membangun satu atau lebih representasi dari aplikasi tersebut.

5) Konstruksi dan peluncuran

Tugas – tugas yang dibutuhkan untuk mengkonstruksi, menguji, memasang (instal) dan memberikan pelayanan kepada pemakai (contohnya pelatihan dan dokumentasi).

6) Evaluasi pelanggan

Tugas – tugas yang dibutuhkan untuk memperoleh umpan balik dari pelnggan dengan didasarkan pada evaluasi representasi software, yang dibuat selama masa perekayasaan, dan diimplementasikan selama masa pemasangan



Gambar 2.10 Model spiral yang disesuaikan untuk siklus hidup bagian dalam.

Kekurangan model spiral adalah sulitnya untuk meyakinkan konsumen (khusunya dalam situasi kontrak) bahwa pendekatan evolusioner bisa dikontrol. Model spiral memerlukan keahlian penaksiran risiko yang msuk akal , dan sangat bertumpu pada keakhlian ini untuk mencapai keberhasilan. Jika resiko mayor tidak ditemukan dan diatur, pasti akan terjadi masalah. Akhirnya model itu sendiri masih baru dan belum dipergunakan secara luas seperti paradigma sekuensial dan prototipe

2.9 Konsep *E-Learning*

2.9.1 Pengertian *E-Learning*

Istilah *e-Learning* dapat didefinisikan sebagai sebuah bentuk teknologi informasi yang diterapkan dalam bidang pendidikan dalam bentuk sekolah maya. Definisi *e-Learning* sendiri sebenarnya sangatlah luas bahkan portal yang menyediakan informasi tentang suatu topik dapat tercakup dalam lingkup *e-Learning* ini. Namun, istilah *e-Learning* lebih tepat ditujukan sebagai usaha untuk sebuah transformasi proses belajar mengajar yang ada disekolah kedalam bentuk digital yang dijembatani oleh teknologi internet.

2.9.2 Keuntungan dan Keterbatasan *E-Learning*

1. **Keuntungan *e-Learning***

   Keuntungan yang diperoleh dengan *e-Learning* :

   a) Biaya

   Kelebihan pertama adalah *e-learning* dapat mengurangi biaya. Perusahaan/sekolah tidak perlu lagi menyediakan ruang kelas beserta isinya, kebutuhan pengajar dan lain sebagainya. Di sisi lain peserta tidak perlu mengeluarkan biaya seperti yang diperlukan jika mengikuti pendidikan secara konvensional seperti transportasi, buku-buku dan sebagainya.

   b) Fleksibilitas Waktu

   Dengan *e-Learning* peserta dapat menyesuaikan waktu belajar tanpa harus datang tepat waktu seperti belajar konvensional,*e-Learning* lebih menyesuaikan jadwal yang dimiliki peserta.

c) Fleksibilitas Tempat

Adanya *e-Learning* membuat peserta dapat mengakses dari mana saja dengan mudah,asal terhubung melalui internet,terlebih semakin banyak tempat-tempat umum yang menyediakan fasilitas internet gratis menggunakan teknologi *hot spot.*

d) Fleksibilitas kecepatan pembelajaran

Setiap orang memiliki kecepatan belajar yang berbeda-beda,dengan *e-Learning* kecepatan belajar dapat diatur sesuai kecepatan masing-masing,bagi yang cepat dapat langsung ke pelajaran selanjutnya,sedangkan yang lebih lambat dapat diulang sampai mengerti lalu baru ke pelajaran selanjutnya.

e) Distribusi yang lebih cepat

Untuk memberikan sebuah materi pelajaran baru,pengajar tidak lagi perlu masuk ke setiap kelas membagi-bagikan materi tersebut,pengajar cukup meng-*upload* melalui halaman situs yang disediakan,lalu setiap peserta tinggal men-*download*.Kelebihan ini tentu akan lebih memperlancar distribusi pelajaran.

2. Keterbatasan *E-Learning*

a) Budaya

Beberapa orang merasa tidak nyaman belajar melalui komputer,terlebih e-Learning menuntut lebih pada budaya *self-learning* (belajar sendiri) dimana ia harus memotivasi diri sendiri agar mau belajar.

b) Investasi

Seperti yang telah diketahui sebelumnya bahwa *e-Learning* akan mengurangi biaya dengan cukup signifikan,namun investasi awal yang harus dikeluarkan cukup besar untuk sampai pada tahap implementasi.Selain biaya yang akan diperlukan untuk sistem *e-Learning* itu sendiri,jika dukungan infrastruktur belum memadai maka investasi yang diperlukan akan lebih besar lagi.

2.10 Konsep Internet

2.10.1 Pengertian Internet

Internet pada dasarnya adalah kumpulan-kumpulan yang saling berhubungan dengan komputer yang lain melalui koneksi yang tersedia. Jarak fisik antar komputer sangat jauh (dapat antarkota, negara, bahkan sampai lintas benua).

2.10.2 Sejarah Internet

Teknologi *internet*, pada awalnya digunakan hanya untuk keperluan pertahanan yang dirintis oleh lembaga riset Departemen Pertahanan Amerika. Lembaga riset tersebut menginginkan agar komputer-komputer yang ada dapat saling berhubungan satu dengan yang lain untuk kepentingan militer. Sistem jaringan komputer yang dimiliki oleh lembaga riset ini juga berhubungan dengan kalangan universitas, dengan harapan agar jaringan komputer ini dapat semakin besar dan berkembang. Kira-kira pada pertengahan tahun 1970, salah satu universitas yang bekerja sama dengan Lembaga Riset Departemen Pertahanan

Amerika, yaitu *Standford University*, mulai mengembangkan standardisasi jaringan komputer tersebut menjadi sebuah protokol (pengatur hubungan antar-komputer) yang mana protokol tersebut dinamakan sebagai protokol TCP/IP (*Transmission Control Protocol/Internet Protocol*). TCP/IP inilah yang sekarang menjadi protokol di *internet*. Sebenarnya fungsi utama TCP/IP adalah untuk menjembatani tiap komputer yang memiliki sistem operasi dan juga *hardware* yang berbeda-beda.

2.10.3 Fasilitas Internet

Berikut adalah beberapa layanan yang disediakan oleh internet yang berbasis pada protokol TCP/IP:

1) WWW (*World Wide Web*)

Layanan ini adalah layanan yang sering kita gunakan. Layanan WWW inimenggunakan protokol yang dinamakan protokol HTTP (*HyperText Tranfer Protocol*).

2) *File Transfer* (Pengiriman File)

FTP ini memungkinkan kita untuk mengirimkan atau menerima file ke atau dari komputer jaringan. Oleh karena masalah keamanan data, FTP seringkali memerlukan username dan *password* tertentu, meskipun banyak juga FTP *server* yang dapat melalui *anonymous* (pengguna biasa dan tanpa *password*, permintaan *password* dapat diisi dengan alamat *email*).

3) *Remote Login* (Telnet)

Telnet memungkinkan pengguna komputer dapat melakukan login ke dalam suatu komputer di dalam jaringan. Ketika kita melakukan telnet,

secara tidak langsung kita telah menjadi pengguna yang sah dari komputer tersebut.

4) *Electronic Mail* (*Email*)

Layanan yang digunakan untuk mengirim dan menerima *email*. Untuk pengiriman email digunakan protokol SMTP (*Simple Mail Transfer Protocol*) dan untuk membaca email digunakan protokol POP3 (*Post Office Protocol3*).

5) NFS *(Network File System)*

Sebuah pelayanan akses file-file jarak jauh yang memungkinkan klien-klien untukmengakses file- file pada jaringan jarak jauh.

6) *Name Server*

Daftar nama alamat yang digunakan pada internet untuk menjelaskan mengenai alamat asli *host internet* (berupa nomor IP mesin).

7) *Remote Execution*

Memungkinkan pengguna komputer untuk menjalankan suatu program di komputer yang lain.

2.11 Pengertian www (*World Wide Web)*

Web atau WWW *(World Wide Web*) yaitu sebuah metode baru yang berjalan di dunia internet yang akhir-akhir ini berkembang dengan cepat. Dengan adanya media ini, seseorang dapat menciptakan puluhan bahkan ratusan aplikasi yang berjalan di bawah web (*under web).*

WWW pertama kali dikembangkan oleh **CERN** yang merupakan suatu badan laboratorium fisika partikel Eropa. WWW diciptakan dengan tujuan untuk menyediakan suatu bentuk informasi di internet yan memungkinkan pemakai internet mendapatkan informasi-informasi di mana saja dan kapan saja dengan mudah dan cepat. Dengan tujuan ini, **CERN** mengembangkan suatu bentuk dokumen yang dapat dengan mudah ditampilkan di berbagai tipe perangkat dan di dalamnya terdapat link atau hubungan ke dokumen lain. Jadi, dari suatu hubungan WWW pemakai dapat langsung menuju ke dokumen-dokumen atau alamat-alamat lain di internet secara langsung melalui link-link yang telah disediakan. Setelah dimunculkan pertama kalinya, WWW mengalami perkembangan yang pesat sampai sekarang.

World wide web (lazim disebut web) adalah fasilitas di internet yang mampu menampilkan informasi secara grafis dan interaktif, yang tersusun dari rangkaian hubungan dari satu sumber (dokumen, alamat, web, dan lain-lain) dengan sumber yang lain di internet dan memungkinkan siapapun untuk berada 24 jam/hari di internet.

Web bekerja pada dua hal penting, yaitu *software browser* dan *software server*. Kedua *software* ini bekerja seperti sebuah *client* dan *server*. *Web browser* bertindak sebagai client yang mengirim permintaan melalui jaringan kerja kepada *web server,* yang kemudian memberikan informasi kepada komputer anda. *Web server* bertindak sebagai *server* untuk fasilitas *world wide web* dan dapat diakses oleh seluruh pemakai internet.

Dokumen-dokumen di WWW merupakan suatu dokumen yang disusun dengan format HTML *(hyper text markup language)*, dan untuk mengakses atau menampilkan dokumen tersebut, pamakai internet harus menggunakan program aplikasi *client*nya atau *web browser*. Saat ini, sudah banyak *web browser* yang beredar, diantaranya nestcape navigator, Microsoft internet explorer, Opera, Modzilla dan lain-lain.

2.12 Perangkat Lunak Pengembang

2.12.1 HTML (*Hypertext markup language)*

HTML (*Hyper Text Markup Language*) adalah Format dokumen yang digunakan dalam *World Wide Web* (WWW). HTML merujuk pada tampilan halaman, jenis, huruf, elemen grafis, juga link hypertext ke dokumen lain di Internet.HTML adalah bentuk program yang sederhana yang sebenarnya merupakan dokumen dalam bentuk ASCII atau text biasa yang diterjemahkan oleh browser menjadi halaman web.

2.12.2 PHP (*Pretext Hyper-Processor)*

PHP adalah bahasa scripting yang menyatu dengan HTML dan dijalankan pada *server side*. Artinya semua sintaks yang kita berikan akan sepenuhnya dijalankan pada server sedangkan yang dikirimkan ke *browser* hanya hasilnya saja.PHP menyatu dengan bahasa HTML untuk membuat halaman web yang menarik.PHP mampu berjalan diatas beberapa *platfrom* seperti Windows,Unix serta varian Linux.Beberapa kelebihan PHP antara lain dapat membuat situs yang interaktif dengan forum diskusi,*guestbook* dan sebagainya,koneksitas yang baik

dengan bermacam-macam *database* seperti Oracle, MySQL, PostgreSQL dan lain-lain.

Kode PHP diawali dengan tag <? atau <?php dan ditutup dengan tag ?>. File yang berisi tag HTML dan kode PHP diberi ekstensi. Berdasarkan ekstensi ini, pada saat file diakses, *server* akan tahu bahwa file tersebut mengandung kode PHP. *Server* akan menerjemahkan kode dan menghasilkan *output* dalam bentuk tag HTML yang akan dikirim ke *browser* klien yang mengakses file tersebut.

Contoh skrip PHP:

```
<?php
echo "ini contoh skrip PHP";
?>
```

Ketika skrip tersebut dijalankan di *serve*r, *server* akan mengembalikan ke *browser client* dalam bentuk tag HTML. Sehingga ketika dilihat hasilnya di *browser* menggunakan menu *view source*, hasilnya adalah sebagi berikut:

```
<HTML>
<BODY>ini contoh skrip PHP</BODY>
</HTML>
```

Dalam menuliskan *script* atau program PHP, ada dua cara yang sering digunakan, yaitu *embedded script* dan *non-embedded script*. *Embedded script* yaitu menuliskan *script* PHP di antara tag-tag HTML. Sedangkan *non-embedded script* merupakan pembuatan program murni PHP, di mana tag-tag HTML yang diletakkan di dalam *script* PHP. Spasi tidak berpengaruh pada penulisan baris perintah PHP. Untuk penulisan fungsi-fungsi, PHP tidak membedakan antara huruf kecil dan huruf besar. Contoh perintah ‘echo’ sama dengan perintah ‘ECHO’ atau ‘Echo’.

1. Tipe Data

PHP mengenal lima tipe data, yaitu *Integer, Floating Point, String, Array, dan Objects*. Tipe data dari sebuah *variable* akan ditentukan otomatis oleh PHP bergantung pada operasi yang sedang dilakukan menggunakan *variable* tersebut.

a. *Integer*, meliputi semua bilangan bulat. Besarnya *range* data *Integer* yaitu antara –2.147.483.648 sampai 2.147.483.647. PHP akan secara otomatis mengkonversi data *Integer* menjadi *Floating Point* jika berada di luar *range* tersebut.

b. *Floating Point*, meliputi semua bilangan pecahan atau bilangan desimal. *Range* tipe data *Floating Point* yaitu antara 1,7E-308 sampai 1,7E+308.

c. *String*, sebuah data dengan tipe data *String* dinyatakan dengan mengapitnya menggunakan tanda petik tunggal (‘ ‘) maupun tanda petik ganda (“ “).

d. *Array*, adalah sebuah data yang mengandung satu atau lebih data, dan dapat diindeks berdasarkan *numeric* maupun *string*. Data yang dikandung oleh sebuah data *Array* dapat dari tipe data lainnya, bahkan dapat juga bertipe *Array.*

e. *Objects*, adalah sebuah tipe data yang dapat berupa sebuah bilangan, *variable*, atau bahkan sebuah fungsi. Objects dibuat dengan tujuan untuk membantu programmer yang terbiasa dengan *Object Oriented Programming.*

Dalam pemrograman PHP, tipe data tidak ditentukan secara eksplisit, tipe data ditetapkan sesuai konteksnya pada saat digunakan. Sebagai contoh, sebuah *variable* $a dideklarasikan bertipe *Integer*, tetapi akan berubah secara otomatis menjadi *Floating Point* jika diberi nilai bilangan pecahan.

2. Perubah (*variable*) dan Konstanta

Perubah atau yang biasa disebut *variable* dinyatakan dengan tanda $ di belakang nama *variable*. Nama *variable* dapat terdiri atas angka, huruf, atau *underscore*. Penamaan *variable* bersifat *case sensitive*, artinya penggunaan huruf besar dan kecil dibedakan. Jadi $a tidak sama dengan $A. Deklarasi sebuah *variable* selalu diikuti dengan pemberian nilai dari *variable* tersebut, sehingga jika sebuah *variable* belum memiliki nilai, tidak perlu dideklarasikan terlebih dahulu.

Konstanta mirip sebuah *variable*, hanya nilainya tetap dan tidak perlu memakai tag $. Konstanta dideklarasikan menggunakan fungsi *define()*. Konstanta dideklarasikan hanya satu kali dan nilai tidak dapat diubah atau didefinisikan lagi pada keseluruhan program.

3. Statement

Statement (Statemen) berfungsi sebagai rangka dari bagian program. Aliran program diatur dengan statemen-statemen kontrol. PHP mengenal dua jenis statemen kontrol, yaitu statemen kondisional dan loop.

a. Statemen kondisional mengatur aliran program berdasarkan pada kondisi tertentu yang ditetapkan. Untuk masalah dengan satu atau

dua percabangan dapat digunakan *if* dan *else*, untuk *multiple* alternatif dapat digunakan *elseif* dan *switch.*

b. *Loop* adalah proses sebuah proses eksekusi operasi program secara berulang-ulang sampai ditemui kondisi untuk mengakhiri eksekusi tersebut. PHP mempunyai dua macam *loop*, yaitu *while loop* dan *for loop*.

4. *Include dan Require*

Include dan *require*, keduanya digunakan untuk memanggil dan mengeksekusi *file* yang ditentukan. Dengan kedua statemen tersebut dapat dibuat fungsi-fungsi, konstanta ataupun perintah operasi biasa dalam sebuah *file* terpisah yang dapat dipanggil dari *file* program lain.

5. Fungsi

Fungsi adalah sekumpulan perintah operasi program yang dapat menerima argumen *input* dan dapat memberikan hasil *output* yang dapat berupa sebuah nilai ataupun sebuah hasil operasi. Fungsi dideklarasikan dengan statemen *function* diikuti nama fungsi dan beberapa *input variable* jika ada. Nama fungsi tidak boleh sama dengan nama *built-in function* yang telah dipunyai oleh PHP, jika sama maka akan keluar pesan kesalahan seperti berikut:

```
Fatal error: Can't redeclare already declared function in
filename on line …
```

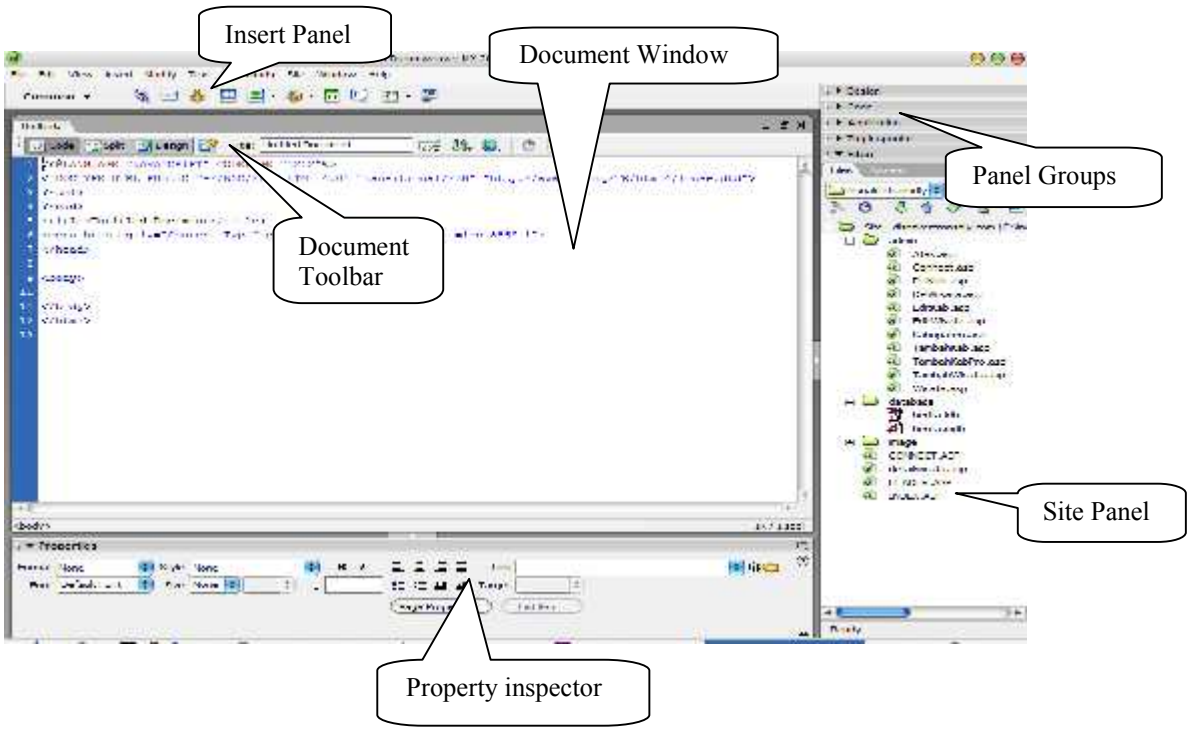
2.12.3 Macromedia Dreamweaver MX

Dreamweaver MX adalah suatu bentuk program editor web yang dibuat oleh Macromedia. Dengan program ini, seorang programer web dapat dengan mudah membuat dan mendesain web.

Dreamweaver adalah editor yang lengkap digunakan untuk membuat animasi sederhana yang berbentuk layer. Dengan adanya program ini, *programer*tidak akan susah dalam menulis *script-script* format HTML, PHP, ASP, maupun bentuk program lainnya.

Sebagai editor, dreamweaver MX mempunyai sifat yang WYSIWYG *(what you see is what you get)*, artinya apa yang kamu lihat akan kamu peroleh. Dengan kelebihan ini, seorang programer dapat langsung melihat hasil buatannya tanpa harus dibuka di *browser*. Seperti program editor-editor lainnya, Dreamweaver MX juga memiliki dua bentuk layer, yaitu bentuk halaman design dan halaman code. Hal ini akan mempermudah dalam menambahkan *script* berbasis PHP maupun Javascript. Dreamweaver selain mendukung pembuatan web yang berbasis HTML, juga mendukung program-program web yang lain, seperti PHP, ASP, Perl, Javascript dan lain-lain.

Versi sebelum dreamweaver MX adalah dreamweaver versi 3 dan 4. Yang membedakan dreamweaver MX dengan versi sebelumnya adalah bentuk tool-toolnya yang diimplementasikan menjadi icon yang mempermudah dalam penggunaannya. Bantuan pada halaman code pun dimiliki oleh dreamweaver MX sehingga membantu *programer* jika lupa. Dreamweaver juga mendukung format bahasa pemrograman yang ada.



Gambar 2.11 Lembar Kerja Macromedia Dreamweaver MX

Beberapa komponen yang ada pada gambar diatas adalah:

1) ***Document Window*** berfunsi untuk menampilkan dokumen dimana anda sekarang bekerja.
2) ***Insert Bar*** mengandung tombol-tombol untuk menyisipkan barbagai macam objek seperti *image*, tabel dan layer ke dalam dokumen.
3) ***Document Toolbar*** berisi tombol-tombol dan menu pop-up yang menyediakan tampilan berbeda dari Document Window.
4) **Kelompok Panel** adalah kumpulan panel yang saling berkaitan satu sama lain, yang dikelompokan dibawah satu judul.
5) **Tag Selector** berfungsi untuk menampilkan hirarki tag disekitar pilihan yang aktif pada *Design View*.

6) **Property Inspector** digunakan untuk melihat dan mengubah berbagai *property object* atau teks.
7) **Site Panel** digunakan untuk mengatur file dan folder yang membentuk situs web anda.

2.12.4 *Databas*e MySQL

MySQL adalah salah satu jenis *database server* yang sangat terkenal, kepopulerannya disebabkan MySQL menggunakan SQL (*Structure Query Language*) sebagai dasar untuk mengakses *database*nya. Selain itu, MySQL bersifat free pada berbagai *platform* (kecuali pada Windows, yang bersifat *shareware* atau anda perlu membayar setelah melakukan evaluasi dan memutuskan untuk digunakan untuk keperluan produksi) atau tidak di cekal.

MySQL termasuk jenis RDBMS (*Relational Database Management System*). Itulah sebabnya istilah seperti tabel, baris dan kolom digunakan di dalam MySQL. Sebuah *database* mengandung satu atau sejumlah tabel, tabel terdiri atas sejumlah baris an setiap baris mengandung satu atau beberapa kolom. Kelebihan dari MySQL adalah:

1) Menggunakan bahasa *query standart* yang dimiliki SQL(*Struktur Query Language*).
2) API (*Application Programming Interface*) dengan C, C++, Eifell, Java, Perl, PHP, Phyton dan Tcl.
3) Dukungan sepenuhnya untuk operator dan fungsi pada bagian SELECT dan WHERE dari *query.*

4) Dapat mencampurkan tabel dari *database* yang berbeda pada *query* yang sama.
5) Mampu menangani *database* dengan ukuran besar, MySQL dapat memuat 50.000.000 record dan 60.000 tabel.
6) Memungkinkan alias pada tabel dan kolom, dan sebagainya.

Program-program aplikasi pendukung MySQL, antara lain:

1) PHP (Page Hypertext Preprosesor)
2) Visual Delphi
3) Visual Basic
4) Cold Fusion, dll

Selain kelebihan yang dimiliki oleh MySQL, ada beberapa *feature* yang dimiliki oleh system database lain yang belum disediakan oleh MySQL.

Beberapa *feature* yang belum disediakan oleh MySQL diantaranya:

1) Sub select (sub query)
2) Perintah SELECT INTO TABLE

   MySQL tidak men-support Oracle SQL Extension: SELECT INTO TABLE… Untuk memecahkan permasalahan tersebut, dapat menggunakan CREATE TABLE…SELECT…dalam MySQL.
3) *Stored procedure* dan *trigger*

   Stored procedure adalah sekumpulan perintah SQL yang dapat dikompilasi dan disimpan di server. Sekali dilakukan, *client* tidak perlu mengeluarkan keseluruhan *query* tetapi dapat mengacu pada *stored procedure*. Hal ini memungkinkan kinerja yang lebih karena *query*

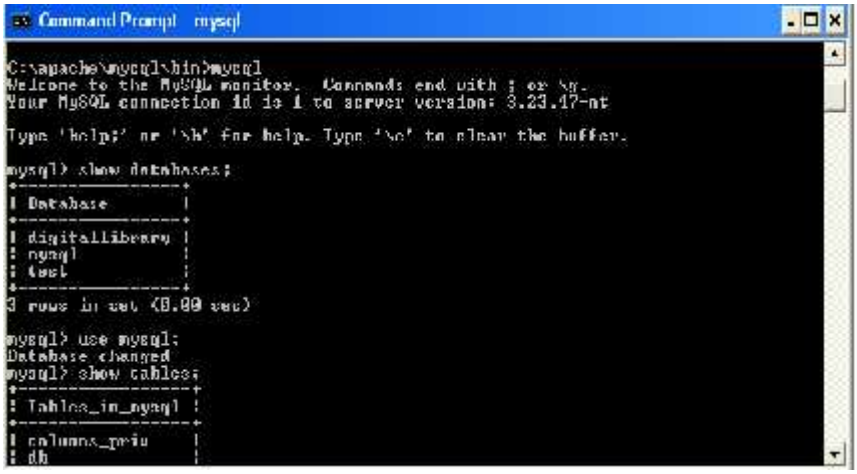
hanya perlu di-parsing sekali dan lebih sedikit informasi yang harus dikirim antara *client* dan *server*.

Sebuah *trigger* adalah *stored procedure* yang dipanggil saat suatu event tertentu terjadi.

MySQL memiliki layer utama yang seperti layer DOS yaitu memiliki prompt utama yang disebut mysql>

Untuk data-data MySQL di simpan di folder C:\apache\mysql\data.

Contoh Layar MySQL adalah:



Gambar 2.12 Lembar Kerja MySQL

Beberapa Perintah dasar yang sering digunakan pada MySQL antara lain:

1) CREATE DATABASE : Membuat *Database* baru
2) DROP DATABASE : Menghapus *database*
3) CREATE TABLE : Membuat tabel baru
4) DESC TABLE : Deskripsi tabel (kolom)
5) ALTER TABLE : Melakukan modifikasi tabel.
6) DROP TABLE : Menghapus tabel.
7) DELETE : Menghapus isi tabel.

8) INSERT INTO : Memasukkan data dlm tabel.
9) SELECT : Menampilkan isi tabel.
10) UPDATE : Mengubah data pada suatu *field/*kolom.

Tipe Data MySQL

Beberapa tipe data yang didukung uleh MySQL dan sering digunakan adalah sebagai berikut:

1) TEXT /BLOB :String dengan max 2^{23} karakter.
2) CHAR/VARCHAR :String dengan panjang antara 1 sampai 255 karakter.
3) DATE :Tanggal dengan format YYYY-MM-DD.
4) TIME : Waktu dengan format HH:MM:SS.
5) DATETIME :Tanggal dan waktu dengan format YYY-MM-DD HH:MM:SS.
6) FLOAT : Bilangan floating-point.
7) INTEGER : Bilangan integer.
8) ENUM : Enumerasi.
9) TIMESTAMP : Tanggal dan waktu dengan format YYYYMMDDHHMMSS.

2.12.5 XAMPP

XAMPP merupakan singkatan dari **X** (empat sistem operasi apapun), **A**pache, **M**ySQL, **P**HP, **P**erl. XAMPP merupakan tool yang menyediakan paket perangkat lunak ke dalam satu buah paket. Dalam paketnya sudah terdapat

Apache (web server), MySQL (database), PHP (server side scripting), Perl, FTP server, phpMyAdmin dan berbagai pustaka bantu lainnya. Dengan menginstall XAMPP maka tidak perlu lagi melakukan instalasi dan konfigurasi web server Apache, PHP dan MySQL secara manual. XAMPP akan menginstalasi dan mengkonfigurasikannya secara otomatis.

Versi XAMPP yang ada saat ini adalah Versi 1.7.3 yang terdiri atas :

1. Apache 2.2.14 (IPv6 enabled) + OpenSSL 0.9.8l
2. MySQL 5.1.41 + PBXT engine
3. PHP 5.3.1
4. phpMyAdmin 3.2.4
5. Perl 5.10.1
6. FileZilla FTP Server 0.9.33
7. Mercury Mail Transport System 4.72

Berikut ini penjelasan tentang web server Apache, PHP, MySQL, phpMyAdmin dan Perl:

1. Apache sudah berkembang sejak versi pertamanya. Sampai saat ditulisnya artikel ini versi terakhirnya yang ada yaitu Apache ver 2.2.41. Apache bersifat open source, artinya setiap orang boleh menggunakannya, mengambil dan bahkan mengubah kode programnya. Tugas utama apache adalah menghasilkan halaman web yang benar kepada peminta, berdasarkan kode PHP yang dituliskan oleh pembuat halaman web. Jika diperlukan juga berdasarkan kode PHP yang dituliskan, maka dapat saja

suatu database diakses terlebih dahulu (misalnya dalam MySQL) untuk mendukung halaman web yang dihasilkan.

2. PHP, Bahasa pemrograman PHP merupakan bahasa pemrograman untuk mebuat web yang bersifat server-side scripting. PHP memungkinkan kita untuk membuat halaman web yang bersifat dinamis. PHP dapat dijalankan pada berbagai macam Operating System (OS), misalnya Windows, Linux dan Mac OS. Selain Apache, PHP juga mendukung beberapa web server lain, misalnya Microsoft IIS, Caudium, PWS dan lain-lain. PHP dapat memanfaatkan database untuk menghasilkan halaman web yang dinamis. Sistem manajemen database yang sering digunakan bersama PHP adalah MySQL. Namun PHP juga mendukung system manajemen Database Oracle, Microsoft Acces, Interbase, d-Base, PostgreSQL dan sebagainya. Hingga kini PHP sudah berkembang hingga versi ke 5. PHP 5 mendukung penuh Object Oriented Programing (OOP), integrasi XML, mendukung semua ekstensi terbaru MySQL, pengembangan web services dengan SOAP dan REST, serta ratusan peningkatan kemampuan lainnya dibandingkan versi sebelumnya. Sama dengan web server lainnya PHP juga bersifat open source sehingga setiap orang dapat menggunakannya dengan gratis.
3. MySQL, perkembangannya disebut SQL yang merupakan kepanjangan dari Structured Query Language. SQL merupakan bahasa terstruktur yang khusus digunakan untuk mengolah database. SQL pertama kali didefinisikan oleh American National Standards Institute (ANSI) pada

tahun 1986. MySQL adalah sebuah sistem manajemen database yang bersifat open source. MySQL adalah pasangan serasi dari PHP. MySQL dibuat dan dikembangkan oleh MySQL AB yang berada di Swedia. MySQL dapat digunakan untuk membuat dan mengola database beserta isinya. Kita dapat memanfaatkan MySQL untuk menambahkan, mengubah dan menghapus data yang berada dalam database. MySQL merupakan sisitem manajemen database yang bersifat at relational. Artinya data-data yang dikelola dalam database akan diletakkan pada beberapa tabel yang terpisah sehingga manipulasi data akan menjadi jauh lebih cepat. MySQL dapat digunakan untuk mengelola database mulai dari yang kecil sampai dengan yang sangat besar. MySQL juga dapat menjalankan perintah-perintah Structured Query Language (SQL) untuk mengelola database-database yang ada di dalamnya. Hingga kini, MySQL sudah berkembang hingga versi 5. MySQL 5 sudah mendukung trigger untuk memudahkan pengelolaan tabel dalam database.

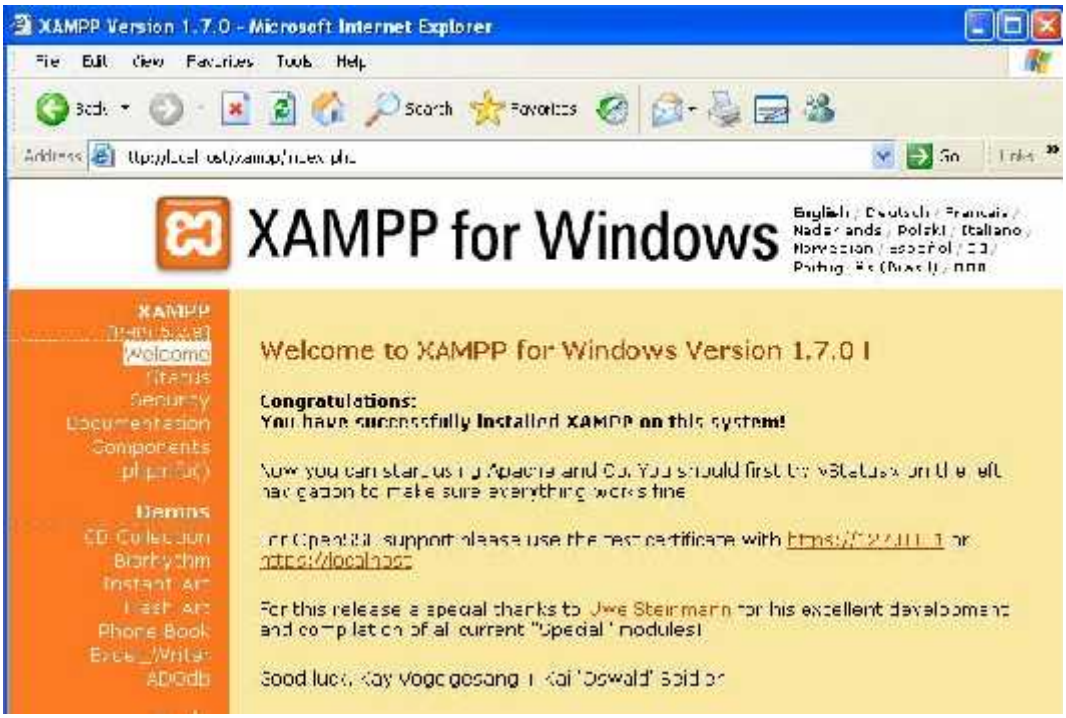
4. PhpMyAdmin, Pengelolaan database dengan MYSQL harus dilakukan dengan mengetikkan baris-baris perintah yang sesuai (command line) untuk setiap maksud tertentu. Jika anda ingin membuat database, ketikkan baris perintah yang sesuai untuk membuat database. Jika kita ingin menghapus tabel, ketikkan baris perintah yang sesuai untuk menghapus tabel. Hal tersebut tentu cukup menyulitkan karena kita harus hafal dan mengetikkan perintahnya satu persatu. Banyak sekali perangkat lunak yang dapat dimanfaatkan untuk mengelola data base dalam MySQL, salah satunya

adalah phpMyAdmin. Dengan phpMyAdmin kita dapat membuat tabel, mengisi data dan lain-lain dengan mudah tanpa harus hafal perintahnya. Untuk mengaktifkan phpMyAdmin langkah-langkahnya adalah : yang pertama setelah XAMP kita terinstall, kita harus mengaktifkan web server Apache dan MySQL dari control panel XAMPP. Yang kedua, jalankan browser kesayangan Anda (IE, Mozilla Firefox atau Opera) lalu ketikkan alamat web berikut: http://localhost/phpmyadmin/ pada address bar lalu tekan Enter. Langkah ketiga apabila telah nampak interface (tampilan antar muka) phpMyAdmin anda bisa memulainya dengan mengetikkan nama database, nama tabel dan seterusnya.

5. Perl adalah bahasa pemrograman untuk segala keperluan, dikembangkan pertama kali oleh Larry Wall di mesin Unix. Perl dirilis pertama kali pada tanggal 18 Desember 1987 ditandai dengan keluarnya Perl 1. Pada versi-versi selanjutnya, Perl tersedia pula untuk berbagai sistem operasi varian Unix (SunOS, Linux, BSD, HP-UX), juga tersedia untuk sistem operasi seperti DOS, Windows, PowerPC, BeOS, VMS, EBCDIC, dan PocketPC. Dukungan terhadap pemrograman berbasis obyek (object oriented programming/OOP) ditambahkan pada Perl 5, yang pertama kali dirilis pada tanggal 31 Juli 1993. Proyek pengembangan Perl 6 dimulai pada tahun 2000, dan masih berlangsung hingga kini tanpa tanggal yang jelas kapan mau dirilis. Ini dikatakan sendiri oleh Larry Wall dalam satu pidatonya yang dikenal dengan seri The State of the Onion. Dua di antara karakteristik utama Perl adalah penanganan teks dan berbagai jalan pintas untuk

menyelesaiakn persoalan-persoalan umum. Perl sangat populer digunakan dalam program-program CGI (Common Gateway Interface) dan berbagai protokol Internet lainnya. Seperti diketahui, TCP/IP sebagai basis bagi semua protokol Internet yang dikenal sekarang ini menggunakan format teks dalam komunikasi data. Seperti juga bahasa populer lainnya, Perl menerima banyak kritikan. Meski banyak di antaranya hanya berupa mitos, atau berlebih-lebihan, tapi terdapat juga sejumlah kritikan yang valid. Salah satunya adalah, sintaksnya susah dibaca, karena banyak menggunakan simbol-simbol yang bukan huruf dan angka.

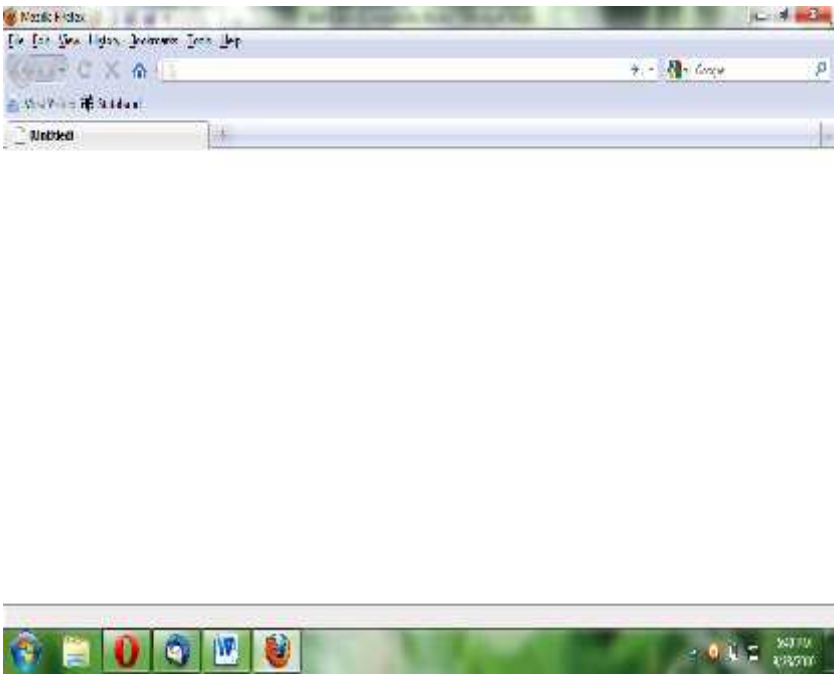
Berikut adalah tampilan antarmuka dari XAMPP.



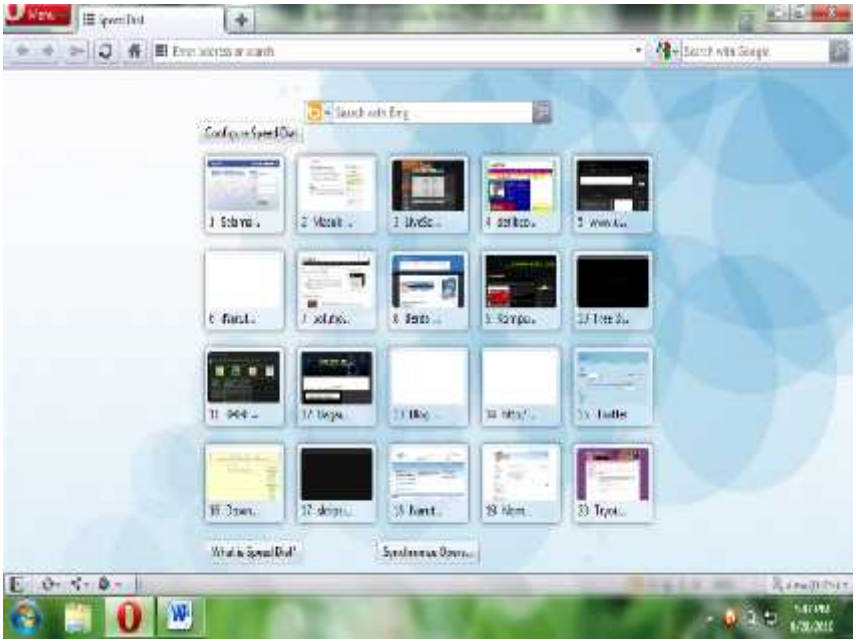
Gambar 2.13 Tampilan XAMPP

2.12.6 Web *Browser*

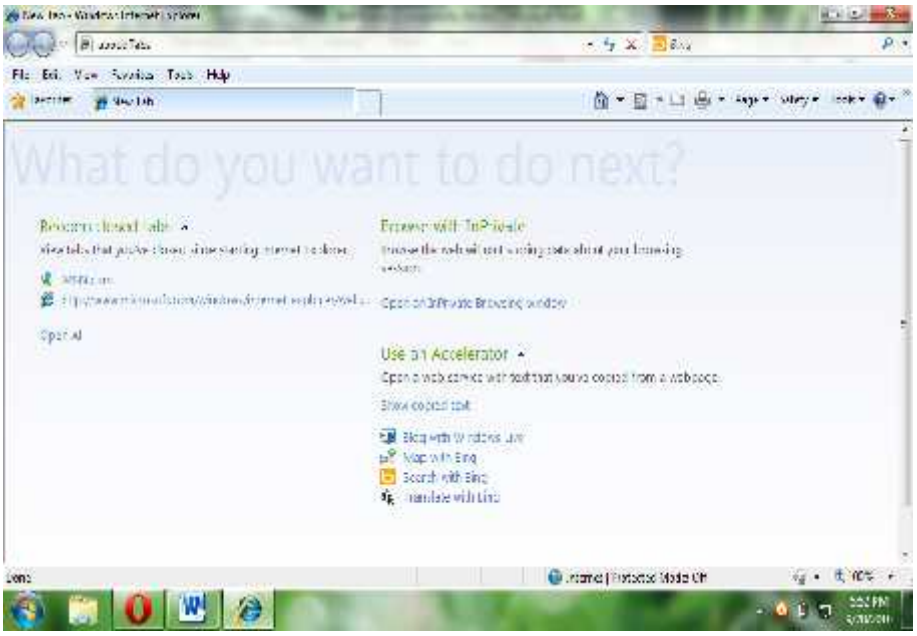
Web browser adalah program aplikasi yang digunakan untuk menjelajahi dunia *world wide web (www).* Aplikasi ini dapat menterjemahkan bahasa pemprograman web seperti PHP, ASP, Javascript, XML dan lain-lain. *Web Browser* yang terkenal antara lain Internet Explorer, Mozilla Firefox, Opera, dan lain-lain.



Gambar 2.14 Tampilan awal Mozzila Firefox



Gambar 2.15 Tampilan Awal Opera



Gambar 2.16 Tampilan Awal Internet Explorer