

BAB II

TINJAUAN PUSTAKA

II.1 Kecerdasan Buatan

Kecerdasan buatan adalah suatu ilmu yang mempelajari cara membuat komputer melakukan sesuatu seperti yang dilakukan oleh manusia (Minsky, 1989). Defenisi lain diungkapkan oleh H. A Simon [1987]. Kecerdasan buatan (artificial intelligence) merupakan kawasan penelitian, aplikasi dan instruksi yang terkait dengan pemrograman komputer untuk melakukan suatu hal yang dalam pandangan manusia adalah cerdas.

Rich dan Knight [1991] mendefenisikan kecerdasan buatan (AI) sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia.

Sementara Ensiklopedi Britannica mendefenisikan kecerdasan buatan (AI) sebagai cabang dari ilmu komputer yang dalam merepresentasi pengetahuan lebih banyak menggunakan bentuk simbol-simbol daripada bilangan, dan memproses informasi berdasarkan metode heuristic atau dengan berdasarkan sejumlah aturan.

Ada tiga tujuan kecerdasan buatan, yaitu: membuat komputer lebih cerdas, mengerti tentang kecerdasan, dan membuat mesin lebih berguna. Yang dimaksud kecerdasan adalah kemampuan untuk belajar atau mengerti dari pengalaman, memahami pesan yang kontradiktif dan ambigu, menanggapi dengan cepat dan baik atas situasi yang baru, menggunakan penalaran dalam memecahkan masalah

serta menyelesaikannya dengan efektif (Winston dan Prendengast, 1994). (sumber Kusri 2006 hal 3 – 4).

II.2 Konsep Kecerdasan Buatan

Ada beberapa konsep yang harus dipahami dalam kecerdasan buatan, di antaranya :

1. Turing Test – Metode Pengujian Kecerdasan

Turing test merupakan sebuah metode pengujian kecerdasan yang dibuat oleh alat turing. Proses uji ini melibatkan seorang penanya (manusia) dan dua obyek yang ditanyai. Yang satu adalah seorang manusia dan satunya adalah seorang mesin yang akan diuji.

2. Pemroses Simbolik

Komputer semula didesain untuk memproses bilangan/angka-angka (pemroses numeric). Sementara manusia dalam berfikir dan menyelesaikan masalah lebih bersifat simbolik. Tidak didasarkan pada sejumlah rumus atau melakukan komputasi matematis.

3. Heuristic

Istilah *heuristic* diambil dari bahasa Yunani yang berarti menemukan. Heuristic merupakan suatu strategi untuk melakukan proses pencarian (*search*) ruang problem secara selektif, yang memadukan proses pencarian yang kita lakukan disepanjang jalur yang memiliki kemungkinan sukses paling besar.

4. Penarikan kesimpulan (*Inferencing*)

AI mencoba membuat mesin memiliki kemampuan berpikir atau mempertimbangkan (*reasoning*) kemampuan berpikir (*reasoning*) termasuk didalamnya proses penarikan kesimpulan (*inferencing*) berdasarkan fakta-fakta dan aturan dengan menggunakan metode heuristic atau metode pencarian lainnya.

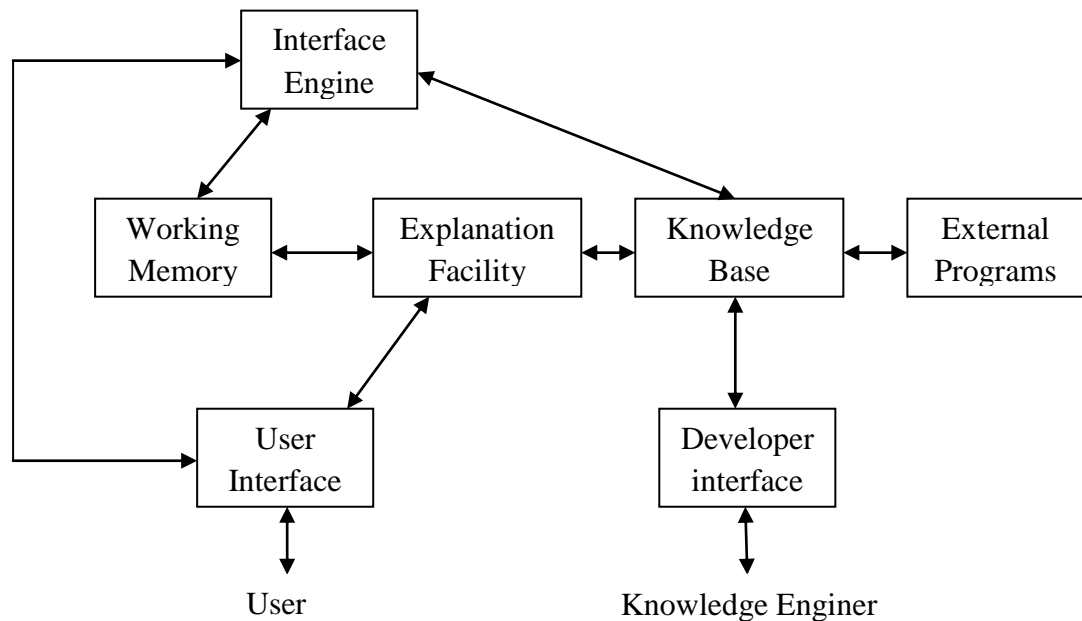
5. Pencocokan Pola (*Pattern Matching*)

AI bekerja dengan metode pencocokan pola (*pattern matching*) yang berusaha untuk menjelaskan obyek, kejadian atau proses, dalam hubungan logic atau komputasional.

(Sumber Kusrini 2006 : hal 5-6).

II.2.1 Arsitektur Sistem Pakar

Arsitektur sistem pakar dapat dilihat pada gambar II.1 di bawah ini dimana sebuah sistem pakar terdiri dari tiga modul utama, yaitu: *knowledge base*, *working memory* dan *inference engine* yang merupakan bagian utama dari sebuah sistem pakar. Sedangkan bagian-bagian selain ketiga komponen utama itu adalah : *user interface*, *developer interface*, *explanation facility*, dan *external programs*.(Jurnal Informatika, Andreas Handoyo, M. Isa Irwan ,Vol.5 No.1. Mei 2004: 33 – 34).



Gambar II.1 Arsitektur Sistem Pakar

(Sumber : Jurnal Informatika)

Keterangan :

1. *Knowledge base* adalah representasi pengetahuan dari seorang atau beberapa pakar yang diperlukan untuk memahami, memformulasikan dan memecahkan masalah. Dalam hal ini digunakan untuk memecahkan masalah-masalah yang terjadi pada komputer. *Knowledge base* ini terdiri dari dua elemen dasar, yaitu fakta dan *rules*.
2. *Inference engine* merupakan otak dari sistem pakar yang mengandung mekanisme fungsi berpikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar. Mekanisme ini yang menganalisis suatu masalah tertentu dan kemudian mencari solusi atau kesimpulan yang terbaik.
3. *Working Memory* merupakan tempat penyimpanan fakta-fakta yang diketahui dari hasil menjawab pertanyaan.

4. *User/Developer interface*. Semua *software* pengembangan sistem pakar memberikan *interface* yang berbeda bagi *user* dan *developer*. *User* akan berhadapan dengan tampilan yang sederhana dan mudah sedangkan *developer* akan berhadapan dengan *editor* dan *source code* waktu mengembangkan program.
5. *Explanation facility* memberikan penjelasan saat mana *user* mengetahui apakah alasan yang diberikan sebuah solusi.
6. *External programs*. Berbagai program seperti *database*, *spreadsheets*, *algorithms*, dan lainnya yang berfungsi untuk mendukung sistem.

II.2.1 Pohon AND-OR

Banyak tipe sistem pakar menggunakan runut balik (*backward chaining*) untuk mendapatkan solusi dan permasalahan. PROLOG merupakan contoh baik dari runut balik yang mencoba untuk menyelesaikan permasalahan dengan memisahkannya ke dalam sub permasalahan yang paling kecil dan menyelesaikannya.

Salah satu tipe dari tree atau lattice yang digunakan dalam masalah representasi runut balik ini adalah AND-OR tree. Gambar II.2 menunjukkan contoh sederhana AND-OR lattice untuk mendapatkan tujuan mendapat gelar sarjana. Untuk menyelesaikan tujuan ini, Anda dapat mendaftar di universitas dengan dating sendiri atau melalui bagian korespondensi. Dengan bagian korespondensi dapat dilakukan dalam bentuk surat menyurat atau secara elektronik menggunakan komputer dan modem.

Urutan untuk pemenuhan persyaratan tingkat ada tiga subtujuan yang harus diselesaikan, yaitu: melamar untuk mendapat hak masuk, mengambil mata kuliah, mengajukan untuk mendapatkan gelar sarjana. Busur yang melalui *edge* dari syarat yang harus dipenuhi untuk ketiga subtujuan ini. Busur yang melalui *edge* menunjukkan syarat yang harus dipenuhi bila ketiga-tiga subtujuan tersebut dipenuhi. Tujuan tanpa busur seperti surat, komputer dan modem dan dating sendiri adalah node OR dimana penyelesaian dari beberapa tujuan mendapat gelar kesarjanaan.

Diagram ini merupakan suatu lattice, karena syarat dipenuhi dari subtujuan mempunyai tiga node orangtua, yaitu mail, komputer dan modem melalui orang. Sebagai catatan bahwa hal ini memungkinkan untuk menggambarkan diagram ini sebagai suatu tree yang sederhana yang menyerupai subtujuan kecukupan persyaratan dan subtree dari tujuan untuk mail, komputer dan modem dan melalui orang. (Sumber: Muhammad Arhami 2005 hal:91-92)



Gambar II.2 Pohon AND-OR Lattice Menunjukkan Bagaimana Memperoleh Gelar Keserjanaan

(Sumber :Muhammad Arhami 2005 hal:92)

II.3 Sistem Pakar

Sistem pakar (*expert system*) merupakan paket perangkat lunak atau paket program komputer yang ditunjukkan sebagai penyedia nasihat dan sarana bantu dalam memecahkan masalah di bidang-bidang spesialisasi tertentu seperti sains, perekayasaan, matematika, kedokteran, pendidikan dan sebagainya. (Arhami, Muhammad : 2005).

1 Ciri – ciri Sistem Pakar

- a. Terbatas pada bidang yang spesifik.
- b. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti.

- c. Dapat mengemukakan rangkaian alasan yang diberikannya dengan cara yang dapat dipahami.
- d. Berdasarkan pada *rule* atau kaidah tertentu.
- e. Dirancang untuk dikembangkan secara bertahap.
- f. *Outputnya* bersifat nasihat atau anjuran.
- g. Output tergantung dari dialog dengan user.
- h. *Knowledge base* dan *Inference engine* terpisah.

(Kusrini, S.Kom : 2006 : 14).

2. Keuntungan Sistem Pakar

- a. Menjadikan pengetahuan dan nasihat lebih mudah didapat.
- b. Meningkatkan output dan produktivitas.
- c. Menyimpan kemampuan dan keahlian pakar.
- d. Meningkatkan penyelesaian masalah-masalah paduan pakar, penerangan, system pakar khas.
- e. Meningkatkan reliabilitas.
- f. Memberikan respons (jawaban) yang cepat.
- g. Merupakan panduan yang *intelligence* (cerdas).
- h. Dapat bekerja dengan informasi yang kurang lengkap dan mengandung ketidakpastian.
- i. *Intelligence database* (basis data cerdas), bahwa sistem pakar dapat digunakan untuk mengakses basis data dengan cara cerdas (Kerschberg:86, Schur:88)

(Arhami, Muhammad : 2005 : 10).

3. Kelemahan Sistem Pakar

Selain keuntungan-keuntungan di atas, sistem pakar seperti halnya sistem lainnya, juga memiliki kelemahan, diantaranya adalah:

- a. Masalah dalam mendapatkan pengetahuan di mana pengetahuan tidak selalu bisa didapatkan dengan mudah, karena kadangkala pakar dari masalah yang kita buat tidak ada, dan walaupun ada kadang-kadang pendekatan yang dimiliki oleh pakar berbeda-beda.
- b. Untuk membuat suatu sistem pakar yang benar-benar berkualitas tinggi sangatlah sulit dan memerlukan biaya yang sangat besar untuk pengembangan dan pemeliharaannya.
- c. Boleh jadi sistem tak dapat membuat keputusan.
- d. Sistem pakar tidaklah 100% menguntungkan, walaupun seorang tetap tidak sempurna atau tidak selalu benar. Oleh karena itu, perlu diuji ulang secara teliti sebelum digunakan. Dalam hal ini peran manusia tetap merupakan faktor dominan.

(Arhami, Muhammad : 2005 : 10-11).

4. Komponen Sistem Pakar

Komponen-komponen yang terdapat dalam sistem pakar adalah sebagai berikut:

- a. *User Interface* (Antar muka pengguna).
- b. Basis Pengetahuan.
- c. Akuisisi pengetahuan.
- d. Mesin Inferensi.

- e. Workplace.
- f. Fasilitas penjelasan.
- g. Perbaikan pengetahuan.

5. Basis Pengetahuan

Basis Pengetahuan mengandung pengetahuan untuk pemahaman, formulasi, dan penyelesaian masalah. Komponen sistem pakar ini disusun atas dua elemen dasar yaitu, fakta dan aturan. Fakta merupakan informasi tentang obyek dalam area permasalahan tertentu, sedangkan aturan merupakan informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui.

Dalam studi kasus pada sistem berbasis pengetahuan terdapat beberapa karakteristik yang dibangun untuk membantu kita di dalam membentuk serangkaian prinsip-prinsip arsitekturnya. Prinsip tersebut meliputi:

- a. Pengetahuan merupakan kunci kekuatan sistem pakar.
- b. Pengetahuan sering tidak pasti dan tidak lengkap.
- c. Pengetahuan sering miskin spesifikasi.
- d. Amatir menjadi ahli secara bertahap.
- e. Sistem pakar harus fleksibel.
- f. Sistem pakar harus transparan.

(Arhami, Muhammad : 2005 : 14-15)

II.3.1 Mesin Inferensi

Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin inferensi adalah

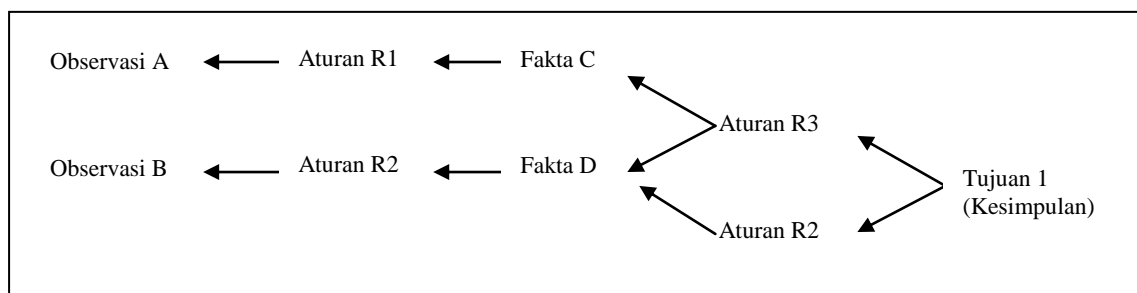
program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam *workplace*, dan untuk memformalisasikan kesimpulan (Turban,1995).

Kebanyakan sistem pakar berbasis aturan menggunakan strategi inferensi yang dinamakan modus ponens. Berdasarkan strategi ini, jika terdapat aturan “IF A THEN B”, dan jika diketahui bahwa A benar, maka dapat disimpulkan bahwa B juga benar, Strategi inferensi modus ponens dinyatakan dalam bentuk:

$$[A \text{ AND } (A \rightarrow B)] \rightarrow B$$

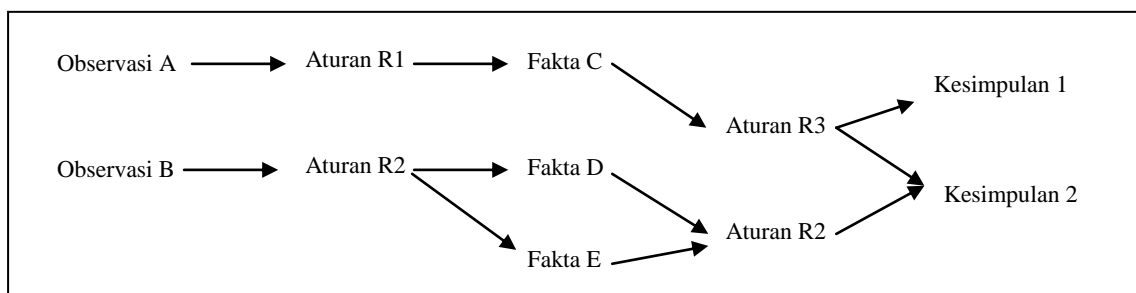
Dengan A dan $A \rightarrow B$ adalah proposisi-proposisi dalam basis pengetahuan.

Terdapat dua pendekatan untuk mengontrol inferensi dalam sistem pakar berbasis aturan, yaitu runut balik (*backward chaining*) dan runut maju (*forward chaining*). Runut balik adalah pendekatan yang dimotori tujuan (*goal-driven*). Dalam pendekatan ini pelacakan dimulai dari tujuan, selanjutnya dicari aturan yang memiliki tujuan tersebut untuk kesimpulannya. Selanjutnya proses pelacakan menggunakan premis untuk aturan tersebut sebagai tujuan baru sebagai kesimpulan. Proses berlanjut sampai semua kemungkinan ditemukan. Gambar II.2 menunjukkan proses *backward chaining*.



Gambar II.3 Proses Backward Chaining
(Sumber : Arhami : 2005)

Runut maju adalah pendekatan yang dimotori data (data-driven). Dalam pendekatan ini pelacakan dimulai dari informasi masukan, dan selanjutnya mencoba menggambarkan kesimpulan. Runut maju mencari fakta yang sesuai dengan bagian IF dari aturan IF-THEN. Gambar II.3 menunjukkan proses *forward chaining*.



Gambar II.4 Proses Forward Chaining

(Sumber : Arhami : 2005)

II.3.2 Kategori Representasi Pengetahuan

Representasi pengetahuan merupakan kombinasi sistem berdasarkan dua elemen, yaitu struktur data dan penafsiran prosedur untuk digunakan pengetahuan dalam menyimpan struktur data. Hal ini penting untuk merealisasikan kedua elemen tersebut dan dalam sistem representasi pengetahuan adalah suatu hal yang perlu. Struktur data tanpa penafsiran prosedur adalah seperti menggunakan kamus tanpa program pengecekannya.

Menurut Turban (2001), ada beberapa tipe pengetahuan yang bisa dikategorikan dalam bentuk keahlian, yaitu:

- a. Teori-teori yang mendasari suatu permasalahan.

- b. Aturan-aturan baku dan prosedur-prosedur yang berkaitan dengan permasalahan tertentu.
- c. Aturan-aturan (heuristik) tentang apa yang harus dikerjakan dalam suatu permasalahan yang diberikan.
- d. Strategi-strategi global untuk pemecahan dari tipe-tipe ini.
- e. *Meta knowledge* (pegetahuan dari pengetahuan).
- f. Fakta atau bukti tentang suatu permasalahan.

Selanjutnya, Feigenbaum dkk (1981) menyarankan untuk mengikuti kategori-kategori berikut ini untuk pengetahuan, yaitu:

- a. Objek-objek

Setiap sistem harus mempunyai kemampuan untuk mengkodekan informasi mengenai sifat-sifat fisik dari objek dan konsep.

- b. *Events* (Kejadian)

Kategori ini termasuk aksi dan kejadian dalam dunia. Kejadian secara umum menentukan suatu elemen waktu dan mengidentifikasi hubungan sebab dan akibat.

- c. *Performance* (Tampilan)

Kategori ini meliputi informasi bagaimana menjalankan tugas tertentu. Sebagai contoh, bagaimana mengendarai sepeda, bagaimana menyusun kalimat, bagaimana membuktikan teorema.

- d. *Meta-Knowledge*

Meta knowledge merupakan pengetahuan tentang merepresentasikan pengetahuan atau pengetahuan yang dimiliki sistem tentang pengetahuan

internalnya (Rolston, 1988). Pengetahuan dalam kategori ini adalah pengetahuan sistem tentang bagaimana sistem berpikir. (Arhami, Muhammad : 2005 : 29-30)

II.3.3 Kategori Masalah Sistem Pakar

Masalah-masalah yang dapat diselesaikan dengan sistem pakar, diantaranya:

1. Interpretasi : membuat kesimpulan atau deskripsi dari sekumpulan data mentah.
2. Prediksi : memproyeksikan akibat-akibat yang dimungkinkan dari situasi-situasi tertentu.
3. Diagnosis : menentukan sebab malfungsi dalam situasi kompleks yang didasarkan pada gejala-gejala yang teramati.
4. Desain ; menentukan konfigurasi komponen-komponen sistem yang cocok dengan tujuan-tujuan kinerja tertentu yang memenuhi kendala-kendala tertentu.
5. Perencanaan : merencanakan serangkaian tindakan yang akan dapat mencapai sejumlah tujuan dengan kondisi awal tertentu.
6. *Debugging* dan *Repair* : menentukan dan menginterpretasikan cara-cara untuk mengatasi malfungsi.
7. *Instruksi* : mendeteksi dan mengoreksi defisiensi dalam pemahaman domain subyek.
8. Pengendalian : mengatur tingkah laku suatu *environment* yang kompleks.

9. *Selection* : mengidentifikasi pilihan terbaik dari sekumpulan (*list*) kemungkinan.
10. *Simulation* : pemodelan interaksi antara komponen-komponen sistem.
11. *Monitoring* : membandingkan hasil pengamatan dengan kondisi yang diharapkan.

(sumber Kusrini 2006 hal 21 – 22)

II.3.4 Prinsip Kerja Motor

Motor bensin dan motor diesel bekerja dengan gerakan torak bolak balik (bergerak naik turun pada motor tegak), sedangkan motor wankel bekerja dengan torak atau rotor yang berputar. Motor bensin diesel bekerja menurut prinsip kerja motor 4 tak atau 2 tak. Langkah (S) adalah perjalanan torak dari satu titik mati ke titik mati yang lain. Dalam motor 4 tak satu lingkaran kerja memerlukan 4 langkah (dua putaran poros engkol) dan langkah usaha diperoleh dalam empat langkah torak. (Sumber Daryanto : hal 19)

1. Prinsip Kerja Motor Diesel

a. Motor Diesel Empat Langkah

1. Langkah Pemasukan

Udara bersih mengalir ke dalam silinder melalui katup masuk yang terbuka.

2. Langkah Kompresi

Torak mengompresikan udara yang pada akhir langkah kompresi udara tersebut dikompresikan menjadi seperdua-puluh bagian isi sebelumnya,

suhu udara kompresi mencapai 700-900 °C. Pengabutan bahan bakar terjadi pada akhir langkah kompresi. Sebuah injektor mengabutkan bahan bakar dengan tekanan yang tinggi. Bahan bakar terbakar oleh udara panas kemudian tekanan di dalam ruang bakar mencapai 70-90 kg/cm².

3. Langkah Usaha

Tekanan yang tinggi menekan torak ke bawah dan gaya torak dipindahkan kepada poros engkol.

4. Langkah pembuangan

Gas bekas dikeluarkan melalui lubang katup buang yang terbuka. (Sumber Daryanto : hal 21-22)

b. Motor Diesel Dua Langkah

Jenis motor ini jarang digunakan dalam motor untuk kendaraan bermotor, disini dilengkapi dengan katup gas bekas dan kompresor untuk udara pembilas dan bekerja menurut cara berikut.

1. Langkah kompresi (Langkah Pertama)

Udara bersih di dalam silinder dikompresikan oleh torak, sebagai akibat dari kenaikan tekanan maka suhu udara mencapai 700-900 °C.

Injeksi Bahan Bakar

Bahan bakar disemprotkan atau diinjeksikan ke dalam udara panas dan terbakar dalam cara yang sama seperti dalam motor diesel empat langkah.

2. Langkah Usaha (Langkah Kedua)

Torak ditekan ke bawah oleh tekanan yang tinggi disebabkan oleh adanya pembakaran.

Pembilasan

Motor dilengkapi dengan sebuah kompresor yang menekan udara bersih ke dalam ruang bilas, setiap saat torak melalui titik mati bawah torak membuka lubang udara bilas sehingga udara mengalir ke dalam silinder. Udara bilas menekan gas bekas melalui katup buang yang terbukakan keluar melalui lubang buang. (Sumber Daryanto : hal 23-24)

2. Mesin Diesel

Mesin diesel dan mesin bensin tidak banyak berbeda dalam hal layoutnya, keduanya mempunyai engkol penggerak, mekanisme katup, rangka pendingin, sistem pelumasan dan lain sebagainya. Perencanaan mesin diesel dibagi dalam dua model, dilengkapi dengan peralatan injeksi bahan bakar dan perencanaan komponen yang besar untuk dapat menahan muatan besar yang diakibatkan tekanan pembakaran yang besar, mesin diesel sering lebih berat dalam hubungan ke tenaga mesin, sekitar 7 kg membangun per kW kira-kira setengah untuk mesin bensin, menjaga ukuran dan pengurangan berat komponen tersendiri mesin diesel dibuat bahan yang kuat. (Sumber Daryanto : hal 138)

a. Ruang Puser

Volume ruang puser lebih besar daripada ruang pembakaran pendahuluan dan lobang laluan diantara rongga puser dan silinder lebih besar daripada ruang pembakaran pendahuluan diesel. Proses pembakaran lebih kurang sama dengan ruang pembakaran pendahuluan diesel. Pada waktu menghidupkan dingin (mula), panas dihantarkan dengan sebuah busi pijar.

b. Perlengkapan Injeksi

Jika mesin diesel bekerja tanpa gangguan dan menimbulkan tenaga yang penuh, dua kondisi pemberian bahan bakar yang benar diperlukan.

1. Jumlah bahan bakar yang sesuai diberikan
2. Bahan bakar diberikan pada waktu yang tepat

Bahan bakar dihisap oleh pompa bahan bakar yang dilengkapi dengan sebuah saringan pendahuluan. Bahan bakar didesak melalui saringan pada sebuah pemberi tekanan dari 0,6 – 0,8 kg/cm² ke pompa penekan bahan bakar. Sebuah katup pengatur pipa pengembali menghantarkan kelebihan bahan bakar kembali ke tangki. Dari pompa penekan bahan bakar dalam tekanan tinggi ke injektor. Tekanan tinggi dihubungkan dengan lubang kecil di ujung injektor (moncong pengabut), akibatnya bahan bakar diinjeksikan memperoleh bagian yang sangat kecil, regulator dan pembatas asap mengatur dan menjamin jumlah bahan bakar yang tepat dalam hubungannya pada udara yang dimasukkan untuk semua kondisi kerja.

Pompa pemberi bahan bakar dibuat dari pompa piston/torak, pompa diafragma atau kombinasi pompa diafragma. Pompa piston adalah yang umum digunakan, hasil pompa pemberi dimasukkan seperlunya pada pompa penekan, akibatnya katup pengatur aliran kelebihan dan mengalir dengan sebuah pompa tangan yang dipergunakan untuk sistem pemompaan apabila tangki telah kosong atau sebelum bekerja dengan peralatan injeksi bahan bakar yang telah dilakukan.

c. Saringan

Saringan perlu melindungi pompa bahan bakar dan kotoran yang datang dari injektor, bagian kotoran akan mengakibatkan kerusakan seperti barrel pompa dan kelonggaran plunyer sampai mencapai 0,001-0,002 mm. Saringan pertama disisipkan diantara tangki dan pompa pemberi yang ditempatkan dalam tangki atau pompa pemberi/pensuplai, bahan bakar disaring melalui sebuah saringan kain logam, perpindahan kotoran kasar yang dapat merusak ke pompa pensuplai. Saringan utama disisipkan diantara pompa pemberi dan pompa penekan. Bahan bakar mengalir melalui elemen saringan di dalam rumah saringan. Elemen saringan dapat terdiri dari kain, saringan pelat atau kertas, elemen saringan terbuat dari lipatan kertas. Katup pengatur peluapan ditempatkan pada saringan utama.

d. Pompa Segaris

Pompa bubungan diputar oleh mekanik gigi timing dan mempunyai sebuah bubungan untuk tiap elemen pompa dan sebuah eksentrik menggerakkan pompa pemberi (beberapa pompa penekan bahan bakar tidak mempunyai poros bubungan tersendiri dan diputar dari sebuah poros bubungan mesin). Bubungan menjalankan bubungan penekan yang memindahkan gerakan pada plunyer pompa, selama plunyer bergerak ke atas (TMA) bahan bakar didesak keluar melalui katup penyalur dan pipa ke injektor.

Pengembalian gerakan plunyer dilengkapi dengan sebuah pegas plunyer, duduk diantara piringan pegas, plunyer pompa dapat berputar sebab penarik penggeser plunyer didalam alur dari pengatur geseran dapat

berputar dengan menggunakan kontrol rack. Penginjeksian ke dalam ruangan silinder mengambil tempat dalam urutan yang tepat, urutan penginjeksian 4 silinder mesin segaris 1-3-4-2 atau 1-2-4-3 dan untuk mesin segaris 6 silinder 1-5-3-6-2-4. Urutan penginjeksian mesin 8 silinder segaris 1-6-2-5-8-3-7-4, mesin 5 silinder segaris 1-2-4-5-3. Poros bubungan penekan, saringan pompa pemberi dan kontrol rack sebaiknya dilumasi dibagian bawah pohon pompa, bagian pompa dilumasi bahan bakar.

e. Regulator

Regulator vacuum adalah digerakkan kevakuman dalam saluran pemasukan, pengaturan dimaksud pada rumah *throttle* dan regulator, rumah *throttle* pada pipa pemasukan dan berisikan *throttle* yang prinsipnya sama dengan karburator, juga sebuah pipa venturi dihubungkan dengan ruangan vakum regulator, regulator mempunyai sebuah diafragma yang ditempatkan diantara dua ruangan, ruangan dalam dihubungkan dengan udara bebas (atmosfer). Diafragma lokasi pegas digerakkan oleh kontrol rack, apabila mesin bekerja dengan pembukaan kecil *throttle* kevakuman dalam pipa venturi dan ruangan vakum semakin tinggi mengatasi tegangan pegas kemudian diafragma berpindah/bergerak. Kontrol rack menerima posisi yang member sejumlah injeksi yang cocok untuk kecepatan lambat. Pada kecepatan tinggi kevakuman menurun dan kontrol rack member untuk sejumlah penginjeksian yang lebih besar, jika muatan bertambah pembukaan *throttle* konstan, penunjuk kecepatan turun dan kecepatan

udara pada pipa venturi menjadi rendah, penurunan vakum dan kontrol jika muatan menurun, kecepatan mengarah naik dan kontrol rack bergerak maju memperkecil jumlah penginjeksian.

f. Regulator Sentrifugal dengan Tipe Dua Kecepatan

Bobot regulator dipasang pada hub (leher) regulator dan berputar pada kecepatan poros bubungan, pegas regulator menekan bobot regulator maju ke titik tengah, pegas lemah *idle* bekerja selama *idle* yang berpegas kuat regulator bekerja pada kecepatan tinggi.

Bila mesin distater pedal gas ditekan ke bawah dan lewat lever/tuas regulator aktif dengan eksentrik, tuas regulator bergerak maju pada jumlah penginjeksian maksimum. Dengan bebasnya pedal gas lengan regulator berada lagi pada *idle stop*, pompa penekan memberi jumlah bahan bakar yang cocok untuk *idle*, jika kecepatan menurun bobot tertekan maju ke titik tengah dan jari pemindah dan tuas menggerakkan tuas regulator maju memperbesar jumlah penginjeksian, jika kecepatan bertambah hal ini mengakibatkan menyesuaikan penurunan.

Apabila pedal ditekan ke bawah (kandas) kecepatan bertambah sampai gaya sentrifugal menekan bobot mengembang keluar, tuas regulator bergerak dan gerakan memajukan pengecilan jumlah penginjeksian dan kecepatan menurun. Diantara kecepatan maksimum dan minimum jumlah regulator ditentukan oleh posisi pedal gas.

g. Pompa Distributor

Pompa ini berbeda dengan pompa segaris dalam pembuatannya dan baik bekerjanya. Rotor pembagi mencakup dua plunyer pemompa baik saluran dari pemasukan dan pengeluaran.

Distributor pembagi berputar sehingga saluran distributor berhubungan dengan pengeluaran yang dihubungkan ke injektor melalui pipa pembagi. Apabila *roller* (bantalan) saatnya berhubungan dengan bubungan ring plunyer menekan lagi dan bahan bakar tertekan keluar di atas tekanan tinggi melalui lobang pengeluaran ke injektor.

Sumbu pemutar diputar dari rangkaian gigi timing, gerakan dipindahkan oleh poros tengah ke regulator atau pengatur, rotor pembagi dan pompa penekan. Tekanan pemberian pompa tekanannya cukup untuk menekan plunyer keluar. Regulator mengatur jumlah penginjeksian dengan menggerakkan katup pengatur yang disisipkan diantara pompa tekan dan pemasukan rotor pembagi. Penyetel *hidraulik* mengatur saat penginjeksian sesuai dengan kecepatan. Rumah pompa diisi dengan bahan bakar yang mengalir melalui pompa selama bekerjanya dan membantu melumasi.

h. Injektor

Fungsi injektor menyemprotkan bahan bakar yang telah menjadi kabut ke dalam ruang pembakaran. Pengabutan dilakukan dengan penginjeksian tekanan tinggi. Bagian bawah injektor, nosel dapat dibuat dalam berbagai ragam tergantung pada antara lain jenis mesin dan bentuk ruang bahan

bakar. Nosel terdiri dari bodi nosel yang "*lapping nozzle*"nya bekerja dengan baik. Mesin penginjeksian langsung menggunakan lobang nosel majemuk dimana ruang bakar pendahuluan dan kamar pusat mesin memakai *nosel pintle*.

Rumah nosel dilengkapi nosel dan member hubungan ke pipa pembagi dari pompa penekan bahan bakar. Rumah nosel meliputi sebuah pegas yang gayanya dipindahkan batang penekan katup nose ke katup nosel. Buat penyetel dipergunakan untuk menyetel tekanan pembukaan injektor dengan menurunkan dan menaikkan tegangan pegas. Bahan bakar dari pompa penekan bahan bakar ditekan melalui pipa pembagi sambungan injektor dan pada saluran tekanan ke ruang bakar. Apabila tekanan telah mencapai harga yang cukup katup nosel terangkat dari kedudukannya dan kabutan bahan bakar disemprotkan ke mesin. Tekanan pembukaan beragam untuk tiap jenis mesin tetapi tekanan itu sekitar 100 kg/cm^2 untuk ruang pembakaran pendahuluan mesin dan 150-175 bar untuk mesin dengan penyemprotan langsung.

Oleh karena pembuatan khusus dari katup pembagi pada pompa injeksi, penurunan tekanan sangat cepat apabila langkah plunyer berakhir dan katup nosel tertekan lagi pada dudukannya oleh karena tegangan pegas. Kebocoran diantara katup nosel dan bodi nosel melayani pelumasan dan pendinginan. Bahan bakar yang bocor kembali lagi ke tangki melalui pipa pengembali.

i. Supercharging

Supercharging (memperbanyak jumlah pengisian) maksudnya dari mesin itu sendiri kompresor memberi jumlah yang lebih banyak udara biasanya berlangsung selama langkah pemasukan. Maksud itu menunjukkan tekanan efektif dan tenaga mesin bertambah kuat dimana berat mesin hanya bertambah tetapi tidak berarti. *Supercharging* dilengkapi dengan beberapa bentuk untuk mesin diesel mobil dipergunakan sentrifugal kompresor salura pemasukan. Mesin untuk motor diesel pada mobil dipergunakan sentrifugal kompresor diputar oleh turbin gas buang.

Turbin gas buang mengubah panas gas buang dan tenaga kinetic ke gerakan putar dan tidak ada tenaga diambil dari mesin untuk memutar kompresor sesuai dengan desakan pada tutup mekanik pemutar kompresor. Bagian *supercharging* (turbo kompresor) terdiri dari turbin gas buang, rumah bantalan dan sentrifugal kompresor. Gas buang bergerak melalui pipa buang langsung ke bagian turbin dan member gerakan putar pada roda turbin. Pemasukan udara mengalir melalui saringan udara dan pipa ke bagian kompresor dimana menimbulkan tekanan $0,2-1 \text{ kg/cm}^2$ yang dihasilkan, pembangkitan tekanan tergantung pada kecepatan mesin.

Udara didesak melalui pipa ke saluran pemasukan dan lewat katup masuk yang terbuka ke dalam ruangan silinder. Porosnya dari roda turbin dan roda kompresor didukung dalam bantalan pada rumah bantalan dan pelumasan dari sistem pelumasan mesin. Bagian *supercharging* dibuat dengan ketelitian yang tinggi. Bekerja diatas putaran sekitar 72000 rpm

dan memerlukan keseimbangan yang baik dan tugas bantalan harus sangat menyakinkan.

(Sumber : Daryanto hal : 138-153).

II.3.5 Aplikasi Web

Kepopuleran Internet di seluruh penjuru dunia mendorong aplikasi *web* semakin diminati. Dengan menggunakan aplikasi *web*, Anda hanya perlu menempatkan aplikasi dalam sebuah *server* dan dengan sendirinya aplikasi tersebut dapat diakses dari manapun, sepanjang pemakai dapat mengakses *web server* nya. Aplikasi *web* yang paling dasar ditulis dengan menggunakan HTML. Sebagaimana diketahui, HTML (*HyperText Markup Language*) adalah bahasa standar untuk membuat halaman-halaman *web*. (Abdul Kadir, 2009:2)

II.3.6 Database

Database sering didefinisikan sebagai kumpulan data yang terkait. Secara teknis, yang berada dalam sebuah *database* adalah sekumpulan tabel atau objek lain (*indeks, view, dan lain-lain*). Tujuan utama pembuatan *database* adalah untuk memudahkan dalam mengakses data. Data dapat ditambahkan, diubah, dihapus, atau dibaca dengan relatif mudah dan cepat.

Sebuah tabel (atau kadang disebut relasi) berisi sejumlah baris dan kolom menyatakan sebuah data. Saat ini tersedia banyak perangkat lunak yang ditujukan untuk mengelola *database*. Perangkat lunak seperti itu biasa dinamakan DBMS (*database management system*). *Access, Ms SQL Server, dan MySQL* merupakan

kelas *database server*, yaitu jenis yang secara aktif memantau permintaan akses terhadap data. Dalam hal ini, *database server* akan segera menanggapi permintaan data. Adapun yang bukan termasuk *database server* adalah *Access*.

(Abdul Kadir, 2009:14-15)

Data dalam sebuah database disusun berdasarkan system hierarki yang unik, yaitu:

1. Database, merupakan kumpulan file yang saling terkait satu sama lain, misalnya file data induk karyawan, file jabatan, file penggajian dan lain sebagainya. Kumpulan file yang tidak saling terkait satu sama lain tidak dapat disebut database, misalnya file data induk karyawan, file tamu undangan perkawinan, file barang retail pasar swalayan.
2. File, yaitu kumpulan dari record yang saling terkait dan memiliki format field yang sama dan sejenis.
3. Record, yaitu kumpulan field yang menggambarkan suatu unit data individu tertentu.
4. Field, yaitu atribut dari record yang menunjukkan suatu item dari data seperti nama, alamat, dan lain sebagainya.
5. Byte, yaitu atribut dari field yang berupa huruf yang membentuk nilai dari sebuah field. Huruf tersebut dapat berupa numerik maupun abjad atau karakter khusus.
6. Bit, yaitu bagian terkecil dari data secara keseluruhan, yaitu berupa karakter ASCII nol atau satu yang merupakan komponen pembentuk byte.

(Budi Sutedjo Dharma Oetomo, S.Kom; 2006:102)

II.3.7. Perangkat Lunak Pendukung

Perangkat lunak pendukung digunakan sebagai alat untuk membantu penulis dalam proses pembuatan program sistem pakar untuk mengidentifikasi kerusakan mesin mobil panther. Perangkat lunak yang digunakan oleh penulis diantaranya yaitu :

1. PHP

Menurut dokumen resmi PHP, PHP merupakan singkatan dari *PHP Hypertext Preprocessor*. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam *server*. Hasilnyalah yang dikirimkan ke klien, tempat pemakai menggunakan *browser*.

Secara khusus, PHP dirancang untuk membentuk aplikasi *web* dinamis. Artinya, ia dapat membentuk suatu tampilan berdasarkan permintaan terkini. Misalnya, Anda bisa menampilkan isi *database* ke halaman *web*. Pada prinsipnya PHP mempunyai fungsi yang sama dengan skrip-skrip seperti ASP (*Active Server Page*), *Cold Fusion*, ataupun *Perl*. Namun, perlu diketahui bahwa PHP sebenarnya bisa dipakai secara *command line*. Artinya, skrip PHP dapat dijalankan tanpa melibatkan *web server* maupun *browser*.

Pada saat ini cukup populer sebagai peranti pemrograman *Web*, terutama di lingkungan Linux. Walaupun demikian, PHP sebenarnya juga dapat berfungsi pada *server-server* yang berbasis UNIX, Windows, dan Macintosh.

Pada awalnya, PHP dirancang untuk diintegrasikan dengan *web server Apache*. Namun, belakangan PHP juga dapat bekerja dengan *web server* seperti PWS (*Personal Web Server*), IIS (*Internet Information Server*), dan Xitami.

(Abdul Kadir,2008: 2)

2. MySQL

MySQL adalah salah satu jenis *database server* yang sangat terkenal. Kepopulerannya disebabkan MYSQL menggunakan SQL sebagai bahasa dasar untuk mengakses *databasenya*. Selain itu, ia bersifat *Open Source* (Anda tidak perlu membayar untuk menggunakannya) pada berbagai *platform* (kecuali untuk jenis *Enterprise*, yang bersifat komersial).

MYSQL termasuk jenis RDBMS (*Relational Database Management System*). Itulah sebabnya, istilah seperti tabel, baris, dan kolom digunakan pada MYSQL. Pada MYSQL, sebuah *database* mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau beberapa kolom. (Abdul Kadir, 2008:348)

MySQL (baca: mai-se-kyu-el) merupakan software yang tergolong *database server* dan bersifat *open source*. *Open source* menyatakan bahwa software ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat MySQL), selain itu tentu saja bentuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa diperoleh dengan cara mengunduh di internet secara gratis. Hal menarik lainnya adalah MySQL juga bersifat *multiplatform*. MySQL dapat dijalankan pada berbagai sistem operasi.

Pengaksesan data dalam database dapat dilakukan dengan mudah melalui SQL (*Structured Query Language*). Data dalam *database* bisa diakses melalui aplikasi non-*web* (misalnya dengan *Visual Basic*) maupun aplikasi *web* (misalnya dengan PHP). (Abdul Kadir, 2009:15)

3. Adobe Dreamweaver CS3

Dreamweaver adalah sebuah HTML editor profesional untuk mendesain web secara visual dan mengelola situs *web* maupun halaman *web*. Saat ini terdapat software dari kelompok adobe yang belakangan banyak digunakan untuk mendesain suatu situs *Web*. Versi terbaru dari Adobe Dreamweaver saat ini adalah Dreamweaver CS3. Pada Dreamweaver CS3 terdapat beberapa kemampuan bukan hanya sebagai software untuk mendesain web saja tetapi juga untuk menyunting kode serta pembuatan aplikasi *Web* dengan menggunakan berbagai bahasa pemrograman *Web*, antara lain: JPS, PHP, ASP dan *ColdFusion*.

Dreamweaver merupakan *software* utama yang digunakan oleh *Web Designer* maupun *Web Programmer* dalam mengembangkan suatu situs *Web*. Hal ini disebabkan ruang kerja, fasilitas dan kemampuan Dreamweaver yang mampu meningkatkan produktivitas dan efektivitas dalam desain maupun membangun suatu situs *Web*.

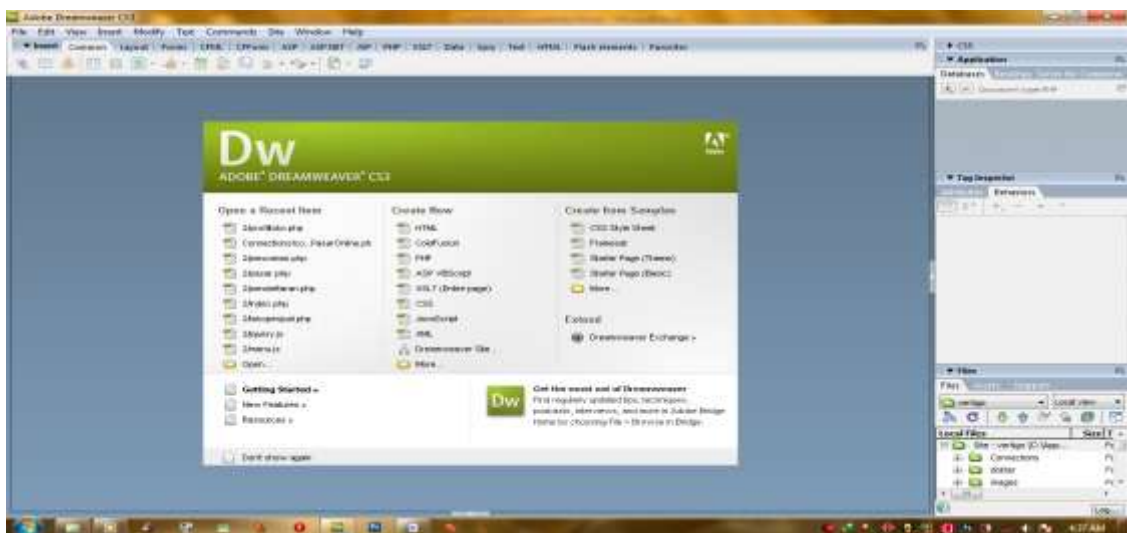
1. Fasilitas Dalam Dreamweaver CS3

Dreamweaver CS3 memiliki peningkatan kemampuan *toolbar*, dimana Dreamweaver CS3 dapat digunakan untuk memodifikasi tampilan *toolbar* atau menambahkan fungsi baru. Selain *user interface* baru, Dreamweaver CS3

memiliki kemampuan untuk menyunting kode dengan lebih baik. Dreamweaver CS3 juga dapat melakukan print kode pada jendela *Code View*, selain itu juga memiliki fasilitas *Code Hints* yang membantu dalam urusan *tag-tag*, serta *Tag Inspector* yang sangat berguna dalam menangani *tag-tag* HTML.

2. Menjalankan Dreamweaver CS3

Langkah untuk memulai Adobe Dreamweaver CS3 adalah : klik tombol *Start*, pilih *All Programs*, pilih **Dreamweaver CS3**. Setelah itu tampilan awal dari Dreamweaver CS3 akan terbuka seperti gambar II.4 berikut ini :



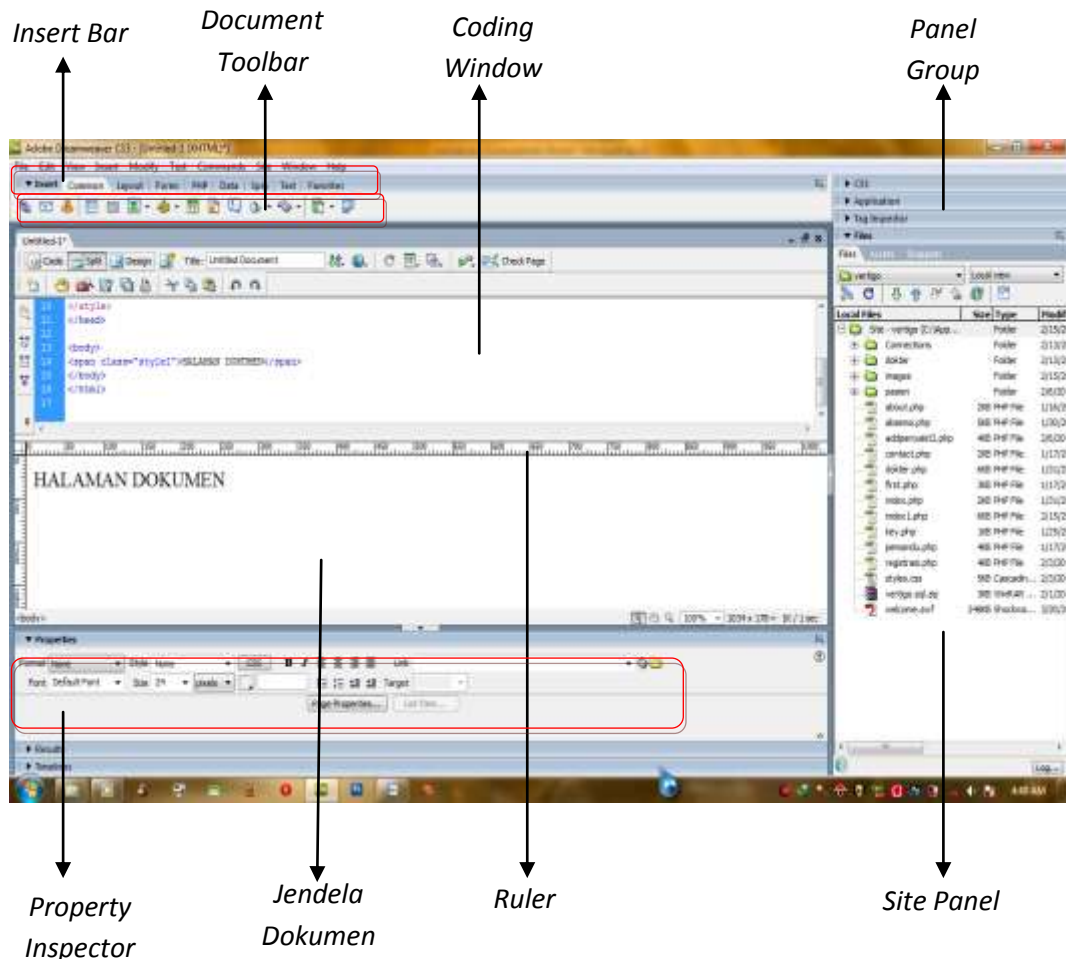
Gambar II.5 Tampilan awal halaman Adobe Dreamweaver CS3

(Sumber : Andi : 2008)

3. Ruang Kerja Dreamweaver CS3

Ruang kerja Dreamweaver CS3 memiliki komponen-komponen yang memberikan fasilitas dan ruang untuk menuangkan kreasi anda saat bekerja, seperti yang terlihat pada gambar II.5. Komponen-komponen yang disediakan oleh ruang kerja Dreamweaver CS3 antara lain *Insert Bar*, *Document*

Toolbar, Jendela Dokument Panel Group, Tag Selector, Property Inspector, dan Site Panel.



Gambar II.6 Tampilan ruang kerja tipe Coder

(Sumber : Andi : 2008)

Penjelasan dari komponen-komponen yang terdapat di dalam ruang kerja Dreamweaver CS3 adalah :

- Insert Bar**, berisi tombol-tombol untuk meyisipkan berbagai macam objek seperti: *image*, tabel dan *layer* ke dalam dokumen.

- b. **Document Toolbar**, berisi tombol-tombol dan menu *pop-up* yang menyediakan tampilan berbeda dari jendela dokumen.
- c. **Coding Window**, berisi kode-kode HTML dan tempat untuk menuliskan kode-kode pemrograman, misalnya PHP atau ASP.
- d. **Panel Group**, adalah kumpulan panel yang saling berkaitan satu sama lainnya yang dikelompokkan dibawah satu judul.
- e. **Property Inspector**, digunakan untuk melihat dan mengubah berbagai properti objek atau teks.
- f. **Jendela Dokumen**, berfungsi untuk menampilkan dokumen dimana anda sekarang bekerja.
- g. **Ruler**, mempermudah ukuran dalam mendesain halaman *Web*.
- h. **Site Panel**, digunakan untuk mengatur *file-file* dan *folder-folder* yang membentuk situs *Web* anda. (Madcoms, 2008:3-6)

II.3.8. Data Flow Diagram (DFD)

Pendekatan analisis terstruktur diperkenalkan oleh DeMarco (1978) dan Gane Sarson (1979) melalui buku metodologi struktur analisis dan disain sistem informasi. Mereka menyarankan untuk menggunakan data flow diagram (DFD) dalam menggambarkan atau membuat model sistem.

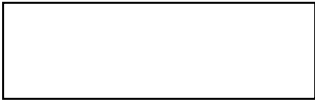
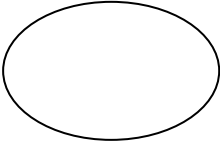
Data Flow Diagram (DFD) adalah suatu network yang menggambarkan suatu sistem otomatis/komputerisasi, manualisasi atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya. (Tata Sutabri, S.Kom; 2004:163).

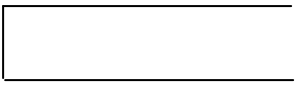
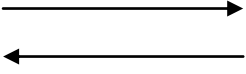
Ada beberapa tahapan dalam pembuatan DFD yaitu:

- a. Diagram Konteks; digunakan untuk menggambarkan sistem secara umum/global dari keseluruhan sistem yang ada.
- b. Diagram Level nol; digunakan untuk menggambarkan tahapan proses yang ada di dalam diagram konteks, yang penjabarannya lebih terperinci.
- c. Diagram Detil (level satu); digunakan untuk menggambarkan arus data secara lebih mendetail lagi dari tahapan proses yang ada di dalam diagram nol. (Tata Sutabri, S.Kom; 2004:166).

Simbol-simbol yang digunakan dalam DFD adalah :

Tabel II.1: Simbol-Simbol DFD

No	Simbol	Fungsi
1		External Entity, Simbol ini digunakan untuk menggambarkan asal atau tujuan data.
2		Atribut/ Proses, Simbol ini digunakan untuk proses pengolahan atau transformasi data.

3		Data Storage (penyimpanan data), Simbol ini digunakan untuk menggambarkan data flow yang sudah disimpan atau diarsipkan.
4		Data Flow(arus data), Simbol ini digunakan untuk menggambarkan aliran data yang berjalan.

(Tata Sutabri, S.Kom; 2004:163)

II.3.9. *Flowchart*

Flowchart merupakan metode untuk menggambarkan tahap-tahap pemecahan masalah dengan mempresentasikan simbol-simbol tertentu yang mudah dimengerti, mudah digunakan, dan standar.

Tujuan utama penggunaan *flowchart* adalah untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi, dan jelas dengan menggunakan simbol-simbol yang standar. Tahap penyelesaian masalah yang disajikan harus jelas, sederhana, efektif dan tepat. Dalam penulisan *flowchart* dikenal dua model, yaitu *sistem flowchart* dan *program flowchart*.

Sistem *flowchart* merupakan diagram alir yang menggambarkan suatu sistem peralatan komputer yang digunakan dalam proses pengolahan data serta hubungan antar peralatan tersebut. Sistem *flowchart* tidak digunakan untuk menggambarkan urutan langkah untuk memecahkan masalah, tetapi hanya untuk menggambarkan prosedur dalam sistem yang dibentuk.






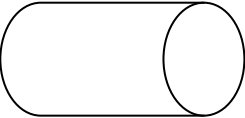
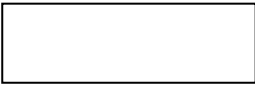
Dalam menggambar *flowchart* biasanya digunakan simbol-simbol yang standar, tetapi pemrogram juga dapat membuat simbol-simbol sendiri apabila simbol-simbol yang telah tersedia dirasa masih kurang. Dalam kasus ini,

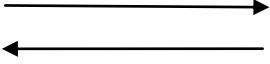
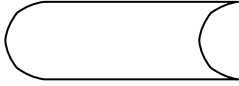
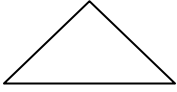
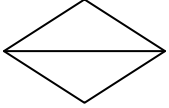
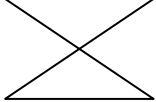
pemrogram harus melengkapi gambar *flowchart* tersebut dengan kamus simbol untuk menjelaskan arti dari masing-masing simbol yang digunakannya, agar pemrogram lain dapat mengetahui maksud simbol-simbol tersebut.

Berikut ini gambar dari simbol-simbol standar yang telah banyak digunakan pada penggambaran sistem *flowchart* serta contoh penggunaannya.

(Budi Sutedjo Dharma Oetomo, S.Kom; 2006: 127-128).

Tabel II.2: Simbol-Simbol *Flowchart*

No	Simbol	Fungsi
1		Pita magnetik
2		Kartu Plong/Keyboard
3		Punched Paper Tape
4		On Line Storage/VDU
5		Input/Output
6		Magnetic Drum
7		Proses

8		Arus
9		Magnetic Disc
10		Off Line Storage
11		Proses Sortir
12		Proses Merge

(Budi Sutedjo Dharma Oetomo, S.Kom; 2006:128)