

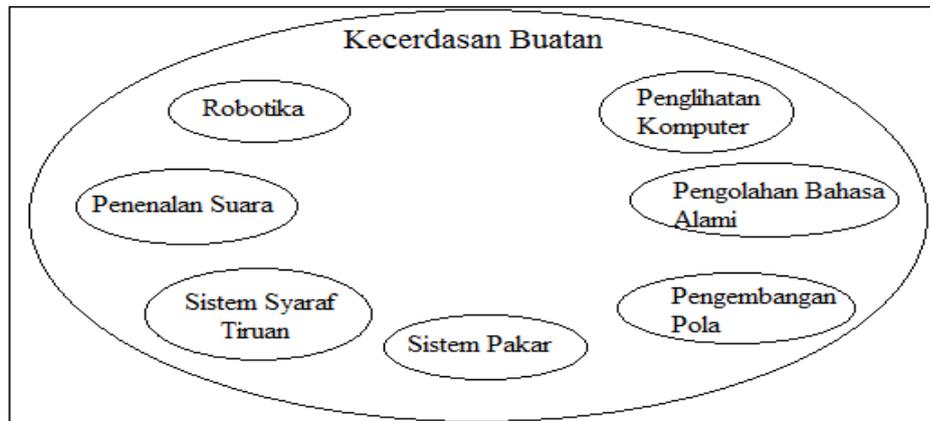
BAB II KAJIAN PUSTAKA

2.1. Teori Dasar

2.1.1 Kecerdasan Buatan

Kecerdasan buatan (*Artificial Intelligent*) merupakan suatu bidang ilmu komputer yang bertujuan untuk membuat kinerja komputer dapat berfikir dan bernalar seperti pikiran atau otak manusia. Sehingga pada saatnya nanti, dapat membantu dan mempermudah manusia dalam melakukan aktifitasnya masing-masing.

(Hartati & Iswanti, 2008) Kecerdasan buatan merupakan salah satu bidang ilmu komputer yang dapat mengoperasikan komputer sehingga berkepribadian cerdas seperti layaknya manusia. Ilmu komputer tersebut mengembangkan perangkat lunak dan perangkat keras untuk mengikuti tindakan manusia. Aktifitas manusia yang diikuti seperti penglihatan, penalaran, pembelajaran, solusi masalah, pemahaman bahasa alami dan sebagainya. Sesuai dengan definisi tersebut, maka teknologi kecerdasan buatan dipelajari dalam bidang-bidang seperti: robotika, penglihatan komputer, pengolahan bahasa, sistem syaraf buatan, pengenalan suara, dan sistem pakar. Bidang-bidang yang termasuk dalam kecerdasan buatan terlihat pada gambar dibawah ini:



Gambar 2.1, Beberapa Bidang Kecerdasan Buatan
 Sumber: Data Olahan

(Suyanto, 2014) Para ahli memberikan pendapat yang berbeda-beda, tergantung pada pemikiran masing-masing. Ada yang fokus pada logika berfikir manusia saja, ada juga yang menyatakan AI secara luas pada tingkah laku manusia. Stuart Russel dan Petter Norvig mengelompokkan definisi AI, yang diperoleh dari beberapa *textbook* berbeda, kedalam 4 kategori, yaitu:

1. *Thinking humanly: the cognitive modeling approach*

Pendekatan ini dilakukan dengan dua cara sebagai berikut:

- a. Melalui introspeksi: mencoba menangkap pemikiran-pemikiran kita sendiri pada saat kita berfikir.
 - b. Melalui eksperimen psikologi.
2. *Acting humanly: the turing test approach*

Pada tahun 1995, Alan Turing merancang suatu ujian bagi komputer *berintelijensia* untuk menguji apakah komputer tersebut mampu mengelabui seorang manusia yang mengintrogasinya melalui *teletype* (komunikasi berbasis teks jarak jauh). Jika *interrogator* tidak dapat membedakan yang diinterogasi

adalah manusia dan komputer, maka komputer *berintelejensia* tersebut lolos dari *Turing test*.

3. *Thinking rationally: the laws of thought approach*

Terdapat dua madalah dalam pendekatan ini, yaitu;

- a. Tidak mudah untuk membuat pengetahuan informal dan menyatakan pengetahuan tersebut ke dalam *formal term* yang diperlukan oleh notasi logika, khususnya ketika pengetahuan tersebut memiliki kepastian kurang dari 100%.
- b. Terdapat perbedaan besar antara dapat memecahkan masalah “dalam prinsip” dan memecahkan “dalam dunia nyata”.

4. *Acting rationally: the rational agent approach*

Membuat inferensi yang logis merupakan bagian dari suatu *rational agent*. Hal ini disebabkan satu-satunya cara untuk melakukan aksi secara rasional adalah menalar dengan secara logis.

2.1.2 Sejarah Kecerdasan Buatan (AI)

Istilah AI dikemukakan pada tahun 1956 di konferensi Dartmouth, sejak saat itu AI terus dikembangkan sebab berbagai penelitian mengenai teori-teori dan prinsip-prinsipnya juga terus berkembang. Meskipun istilah AI baru muncul tahun 1956, tetapi teori-teori yang mengarah ke AI sudah muncul sejak tahun 1941 (Suyanto, 2014).

2.1.3 Konsep Kecerdasan Buatan

Menurut (Kusrini, 2006) ada beberapa konsep yang harus dipahami dalam kecerdasan buatan, di antaranya yaitu:

1. *Turing Test* Metode Pengujian Kecerdasan

Turing Test merupakan sebuah metode pengujian kecerdasan yang dibuat oleh Alan Turing. Proses pengujian ini melibatkan seorang penanya (Manusia) dan dua objek yang ditanya. Yang satu adalah seorang manusia dan satunya adalah sebuah mesin yang akan diuji. Penanya tidak melihat langsung kepada obyek yang ditanya. Penanya diminta untuk membedakan mana jawaban manusia dan mana jawaban komputer berdasarkan jawaban kedua objek tersebut. Jika penanya tidak dapat membedakan mana jawaban mesin yang diuji tersebut dapat diasumsikan cerdas.

2. Pemrosesan Simbolik

Komputer semula didesain untuk memproses bilangan/ angka (pemrosesan numerik). Sementara manusia dalam berfikir dan menyelesaikan masalah lebih bersifat simbolik, tidak didasarkan pada jumlah rumus atau melakukan komputasi matematis. Sifat penting dari AI adalah bahwa AI merupakan bagian dari ilmu komputer yang dilakukan proses secara simbolik dan non algoritma dalam menyelesaikan masalah.

3. *Heuristic*

Istilah *Heuristic* di ambil dari bahasa Yunani yang berarti menemukan. *Heuristic* merupakan suatu strategi untuk menemukan proses pencarian (*search*)

ruang *problem* secara selektif, yang memandu proses pencarian yang kita lakukan di sepanjang jalur yang memiliki kemungkinan sukses paling besar.

4. Penarikan Kesimpulan (*Inferencing*)

AI mencoba membuat mesin yang memiliki kemampuan berfikir atau mempertimbangkan (*reasoing*). Kemampuan berfikir (*reasoing*) termasuk di dalamnya proses penarikan kesimpulan (*inferencing*) berdasarkan fakta dan aturan dengan menggunakan metode heuristik atau metode pencarian lainnya.

5. Pencocokan Pola (*Pattern Matching*)

AI bekerja dengan metode pencocokan pola (*Pattern Matching*) yang berusaha untuk menjelaskan obyek, kejadian (*events*) atau proses, dalam hubungan logic atau konfutasional.

2.1.4 Perbandingan Kecerdasan Buatan dengan Alamiah

Keuntungan kecerdasan buatan dibandingkan kecerdasan alamiah:

1. Lebih permanen.
2. Memberikan kemudahan dalam duplukasi dan penyebaran.
3. Relatif murah dari kecerdasan alamiah.
4. Konsisten dan teliti.
5. Dapat didokumentasi.
6. Dapat mengerjakan beberapa task dengan lebih cepat dan lebih baik dibandingkan manusia..

Sedangkan keuntungan kecerdasan alamiah dibandingkan kecerdasan buatan:

1. Bersifat lebih kreatif.

2. Dapat melakukan proses pembelajaran secara langsung, sementara AI harus mendapatkan masukan berupa simbol representasi.
3. Menggunakan fokus yang luas sebagai referensi untuk pengambilan keputusan. Sebaliknya, AI menggunakan fokus yang sempit.

Komputer dapat digunakan untuk mengumpulkan informasi tentang obyek, kegiatan (*events*), proses, dan dapat memproses sejumlah besar informasi dengan lebih efisien daripada yang dapat dikerjakan manusia. Namun di sisi lain, dengan menggunakan *insting*, manusia dapat melakukan hal yang sulit untuk diprogram pada komputer. Manusia dapat mengenali (*recognize*) hubungan antara beberapa hal, menilai kualitas dan menemukan pola yang menjelaskan hubungan tersebut.

Secara konvensional komputer memproses data dengan cara berikut:

1. Proses

Yang dikerjakan kalkulasi, mengerjakan operasi matematis: tambah, kurang, bagi, kali, atau mencari akar. Menyelesaikan rumus atau permasalahan.

2. Logika

Mengerjakan operasi logika *and*, *or* atau *invert*.

3. Penyimpanan

Penyimpanan data dan gambar pada *file*.

4. *Retrieve*

Mengakses data yang tersimpan pada *file*.

5. *Translate*

Mengkonversi data dari satu bentuk ke bentuk yang lain.

6. *Sort*

Memeriksa data dan menamilkannya dalam urutan yang diinginkan.

7. *Edit*

Melakukan perubahan, penambahan, penghapusan pada data.

8. *Monitor*

Mengamati *event* dan internal dalam melakukan tindakan jika kondisi tertentu tercapai. Kontrol, memberikan perintah atau mengendalikan peralatan di luar.

Berikut ini adalah perbandingan AI dengan Pemrograman Konvensional:

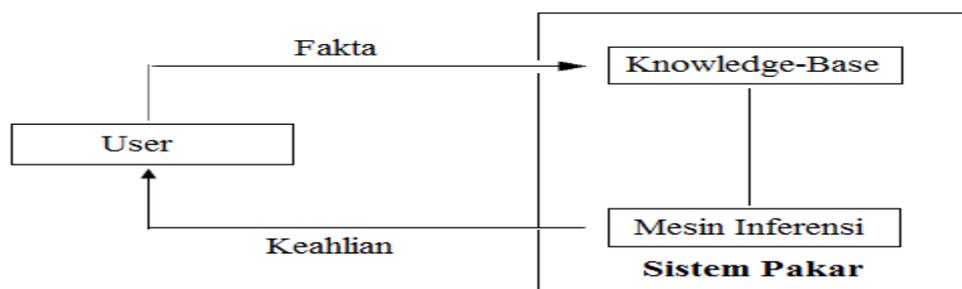
Tabel 2.1, Perbandingan AI dengan Programan Konvensional

Dimensi	Kecerdasan Buatan	Pemrograman Konvensional
<i>Processing</i>	<i>Simbolik</i>	<i>Algoritmik</i>
Input	Tidak harus lengkap	Harus lengkap
<i>Search</i>	<i>Heuristic</i>	Algoritmik
<i>Major Interest</i>	<i>Knowledge data</i>	Informasi
Struktur	Terpisah antara kontrol dan <i>knowledge</i>	Kontrol terintegrasi dengan data
<i>Output</i>	Tidak harus lengkap	Harus tetap
<i>Maintenance</i> dan update	Mudah karena menggunakan modul - modul	Umumnya susah dilakukan
Kemampuan pemikiran	Terbatas tetapi dapat ditingkatkan	Tidak ada

2.1.5 Sistem Pakar

(Andriani, 2016) Sistem pakar adalah sebuah sistem yang kinerjanya mengadopsi keahlian yang dimiliki seorang pakar dalam bidang tertentu ke dalam

sistem atau program komputer yang disajikan dengan tampilan yang dapat digunakan oleh pengguna yang bukan seorang pakar sehingga dengan sistem tersebut penggunaan dapat membuat sebuah keputusan atau menentukan kebijakan layaknya seorang pakar. Konsep dasar sistem pakar dapat dilihat pada gambar dibawah ini:



Gambar 2.2, *Konsep Dasar Sistem Pakar*
 Sumber: *Data Olahan*

Gambar 2.2, menggambarkan konsep dasar dari sistem pakar. Dalam konsep sistem pakar tersebut, *user* atau pengguna menyampaikan fakta atau informasi kedalam sistem pakar, yang selanjutnya fakta dan informasi tersebut akan tersimpan ke *knowledge-base* dan diolah oleh mesin inferensi, sehingga sistem dapat memberikan timbal balik kepada *user* berupa keahlian atau jawaban berdasarkan pengetahuan yang disampaikan sebelumnya.

2.1.6 Ciri-ciri Sistem Pakar

Ciri-ciri sistem pakar yang membedakan dengan sistem informasi biasanya adalah sebagai berikut:

1. Memiliki dan memberikan informasi yang andal.
2. Mudah untuk dimodifikasi.

3. Terbatas pada domain keahlian tertentu.
4. Dapat memberikan penalaran untuk data-data yang difatnya tidak pasti.
5. Sistem berdasarkan pada kaidah/ rule tertentu.
6. Memiliki kemampuan untuk beradaptasi.
7. Keluarannya bersifat anjuran.

2.1.7 Keuntungan dan Kelemahan Sistem Pakar

Penggunaan sistem pakar secara umum memberikan keuntungan yang dapat dimanfaatkan langsung oleh pengguna. Adapun keuntungan dari pengguna sistem pakar yaitu:

Penggunaan sistem pakar secara umum memberikan keuntungan yang dapat dimanfaatkan langsung oleh pengguna. Adapun keuntungan dari pengguna sistem pakar yaitu:

1. Memungkinkan pengguna yang bukan seorang pakar pada bidang tertentu dapat mengerjakan tugas seorang pakar.
2. Bisa melakukan proses yang sama secara berulang.
3. Sistem pakar dapat menyimpan pengetahuan dan keahlian dari pakar.
4. Dengan adanya sistem pakar produktivitas dan *output* sistem dapat ditingkatkan.
5. Meningkatkan kualitas.
6. Mampu mengambil dan melestarikan keahlian para pakar.
7. Mampu melestarikan keahlian para pakar.
8. Memiliki kemampuan untuk mengakses pengetahuan.
9. Memiliki reabilitas.

10. Meningkatkan kapabilitas sistem komputer.
11. Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dalam penelitian.
12. Sebagai media pelengkap dalam pelatihan.
13. Meningkatkan kapabilitas dalam penyelesaian masalah.
14. Menghemat waktu dalam pengambilan keputusan.

Sistem pakar juga memiliki kelemahan selain banyaknya keuntungan yang diberikan, antara lain:

1. Biaya untuk membuat, memelihara, dan mengembangkan sistem pakar sangat mahal.
2. Sulit dikembangkan, karena ketersediaan pakar dibidangnya dan kepakaran sulit diekstrak dari manusia karena terkadang sulit bagi seorang pakar untuk menjelaskan langkah mereka dalam menangani masalah.
3. Sistem pakar tidak 100% benar karena seseorang yang terlibat dalam pembuatan sistem pakar tidak selalu benar.
4. Pendekatan oleh sistem pakar untuk suatu situasi atau masalah bisa berbeda-beda, meskipun sama-sama benar.
5. Transfer pengetahuan dapat bersifat subjektif atau bias.
6. Kurangnya rasa percaya diri perngguna dapat menghalangi pemakaian sistem pakar.

2.1.8 Perbedaan Sistem Pakar dan Sistem Konvensional

Perbedaan utama dari sistem pakar dengan sistem konvensional adalah dasar pengetahuan (*knowledge base*) yang menjadi dasar dari pembuatan sistem pakar

tersebut. Perbedaan secara lengkap antara sistem pakar dengan sistem konvensional dapat dilihat pada tabel berikut:

Tabel 2.2, Perbandingan Sistem Pakar dengan Programan Konvensional

Sistem Konvensional	Sistem Pakar
Informasi dan pemrosesan pada sistem konvensional biasanya jadi satu dalam program	Basis pengetahuan merupakan bagian terpisah dari mekanisme inferensi
Program tidak pernah salah (kecuali pemrogramannya salah) dalam memberikan hasil	Program bisa saja melakukan kesalahan dalam memberikan hasil atau membuat kesimpulan
Biasanya tidak bisa menjelaskan mengapa suatu input data itu dibutuhkan atau bagaimana output itu diperoleh	Penjelasan adalah bagian terpenting dari sistem pakar
Pengubahan program cukup sulit dan merepotkan	Pengubahan pada aturan/ kaidah dapat dilakukan dengan mudah
Sistem hanya akan berkerja jika sistem tersebut sudah lengkap	Sistem dapat bekerja hanya dengan aturan saja
Eksekusi dilakukan langkah demi langkah secara algoritma	Eksekusi dilakukan pada keseluruhan basis pengetahuan secara heuristik dan logis
Menggunakan data	Menggunakan pengetahuan
Tujuan utama adalah efisiensi	Tujuan utama adalah efektifitas

2.1.9 Komponen Sistem Pakar

(Hartati & Iswanti, 2008) Sistem pakar adalah sebuah program yang digunakan untuk menirukan layaknya seorang pakar manusia sehingga dapat melakukan hal-hal yang dikerjakan oleh seorang pakar. Komponen-komponen yang harus dilakukan untuk membangun sebuah sistem tersebut adalah sebagai berikut:

1. Basis Pengetahuan.
2. Antar Muka Pengguna.
3. Memori Kerja.

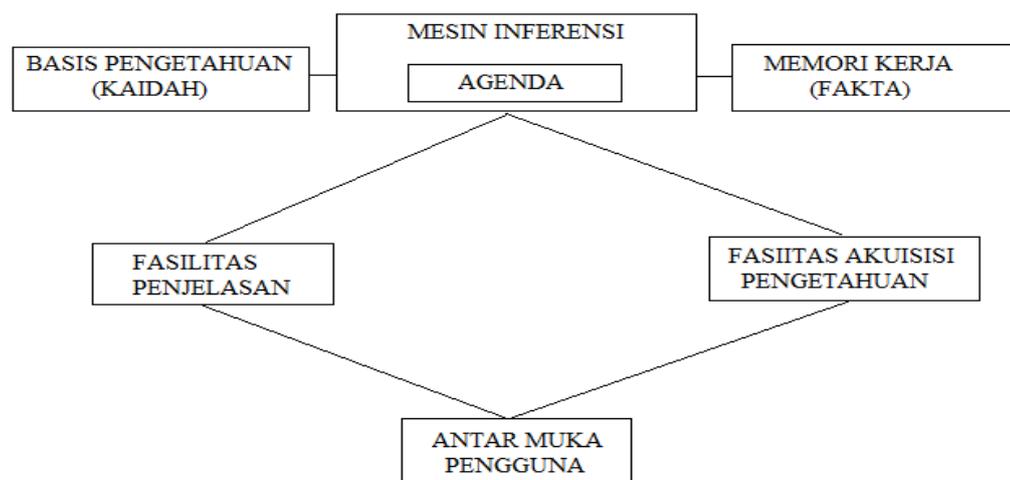
4. Mekanisme Inferensi.

Sedangkan untuk menciptakan sistem pakar menjadi lebih menyamai seorang pakar yang berhubungan dengan pemakai maka harus dilengkapi dengan fasilitas dibawah ini:

1. Fasilitas Penjelasan (*Explanation Facility*)
2. Fasilitas Akuisisi Pengetahuan (*Knowledge Acquisition facility*)

2.1.10 Struktur Sistem Pakar

(Hartati & Iswanti, 2008) dibawah ini adalah macam-macam struktur sistem pakar dan penjelasannya:



Gambar 2.3, *Struktur Dasar Sistem Pakar*
 Sumber: Data Olahan

1. Antarmuka Pengguna

Sistem pakar dapat mengambil alih seorang pakar dalam suatu keadaan tertentu, untuk dapat melakukan hal tersebut maka sistem pakar harus mempersiapkan pendukung yang diperlukan oleh pengguna yang tidak memahami masalah teknik. Sistem pakar juga harus menyediakan komunikasi antara sistem

dan pemakainya, yang disebut dengan antar muka. Antar muka yang baik, efisien dan *user-friendly*, terutama bagi yang tidak kompeten dalam bidang sistem pakar.

2. Basis Pengetahuan

Merupakan kumpulan ilmu dalam aspek tertentu pada tingkatan pakar dengan struktur tertentu. Pengetahuan ini diperoleh dari akumulasi pengetahuan seorang pakar dan sumber pengetahuan lainnya yang telah dinyatakan sebelumnya.

3. Mesin Inferensi

Merupakan suatu perangkat lunak yang bertugas untuk melakukan penalaran pada sistem pakar, atau biasa disebut sebagai mesin berfikir (*Thinking Machine*)

4. Memori Kerja

Bagian dari sistem pakar yang menyimpan fakta-fakta yang diperoleh saat dilakukan proses konsultasi. Fakta-fakta inilah yang nantinya akan diolah mesin inferensi berdasarkan pengetahuan yang disimpan dalam basis pengetahuan untuk menentukan suatu keputusan pemecahan masalah. Konklusinya bisa berupa hasil diagnosa, tindakan, dan akibat.

5. Fasilitas Penjelasan

Proses menentukan keputusan yang dilakukan oleh mesin inferensi selama sesi konsultasi mencerminkan proses penalaran seorang pakar. Karena pemakai kadangkala ahli dalam bidang tersebut, maka dibuatlah fasilitas penjelasan.

6. Fasilitas Akuisisi Pengetahuan

Proses pengumpulan, pemindahan, dan transformasi dari keahlian/kepakaran pemecahan masalah yang berasal dari beberapa sumber pengetahuan ke dalam bentuk yang dimengerti oleh komputer.

2.1.11 Inferensi

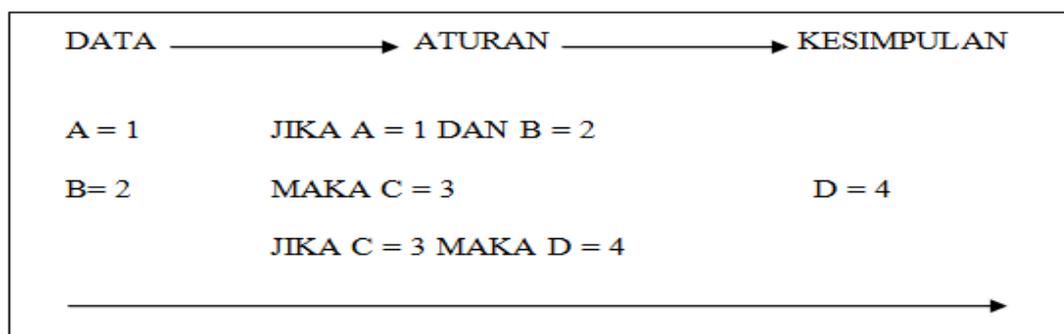
(Kusrini, 2006) teknik untuk menciptakan sebuah informasi dari kebenaran yang diketahui. Inferensi merupakan kesimpulan logis berdasarkan informasi yang ada. Dalam sistem pakar teknik inferensi dilakukan dalam modul yang disebut *Inference Engine*.

Berikut ini adalah mesin inferensi yang penting dalam sistem pakar:

1. Runut Maju (*Forward Chaining*)

Runut maju berarti menggunakan himpunan aturan kondisi-aksi. Dalam metode ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut diajalankan. Mungkin proses menambah data ke memori kerja, proses diulang sampai menemukan hasil.

Gambar 2.4, berikut ini menunjukkan bagaimana cara kerja metode inferensi runut maju.

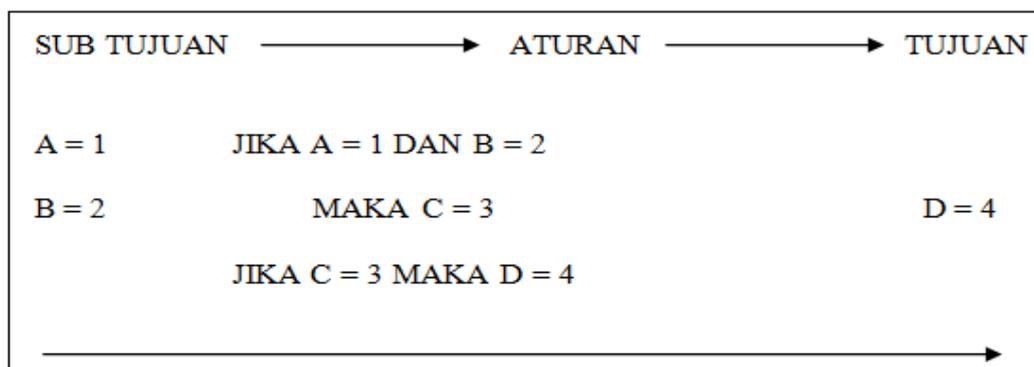


Gambar 2.4, Runut Maju
Sumber: Data Olahan

2. Runut Balik (*Backward Chaining*)

Metode penalaran kebalikan dari runut maju, dalam runut balik penalaran dimulai dengan tujuan kemudian merunut balik ke jalur yang akan mengarah ke tujuan tersebut. Runut balik disebut juga sebagai *goal-driven reasoning*, merupakan cara yang efisien untuk memecahkan masalah yang dimodelkan sebagai masalah pemilihan terstruktur.

Gambar 2.5, berikut ini menunjukkan proses penalaran menggunakan runut balik.



Gambar 2.5, Runut Balik
Sumber: Data Olahan

2.2. Variabel

2.2.1 Mesin

(Yulirianto, 2014) Mesin adalah suatu media atau alat untuk mengubah energi seperti air, panas, angin, listrik, atom menjadi energi gerak (*mechanical energy*). Mesin sepeda motor rata-rata menggunakan energi panas atau listrik. Mesin sepeda motor terbagi menjadi 2, yaitu mesin 2 tak dan mesin 4 tak. Mesin 4 tak atau mesin 4 langkah artinya untuk menghasilkan satu kali tenaga, mesin tersebut membutuhkan 4 kali langkah gerak balak-balik piston dari TMA (Titik

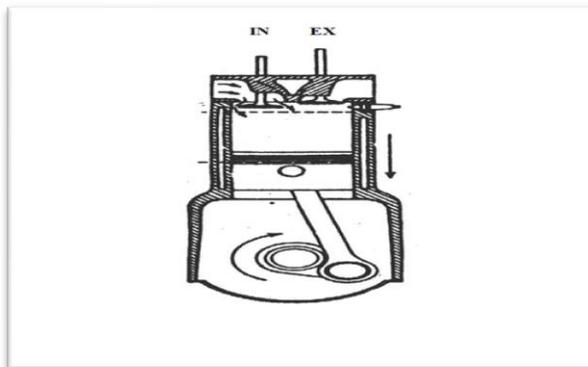
Mati Atas) ke TMB (Titik Mati Bawah) dan sebaliknya dari TMB ke TMA, yaitu dua kali putaran poros engkol atau *crankshaft/kruk as*, dan 1 kali putaran *noken as* atau mekanik biasa menyebutkan *camshaft*.

Berikut adalah siklus kerja motor 4 tak/ 4 langkah :

1. Langkah Isap

Pada langkah ini, terjadi pemasukan campuran bahan bakar dengan udara yang telah diubah menjadi gas yang dihasilkan oleh daya isapan atau kevakuman diruang bakar. Lalu campuran bahan bakar dari karburator terisap masuk keruang bakar melalui *intake manifold*.

- a. *Klep in* (masuk) terbuka: membuka saluran campuran bahan bakar dari karburator ke ruagn bakar melalui *intake manifold* (istilah bengkel: leher angsa).
- b. *Klep ex* (buang) tertutup: menutup saluran buang supaya isapan piston tidak bocor.
- c. Piston bergerak dari TMA menuju TMB dan menimbulkan kevakuman diruang bakar.
- d. *Klep in* dibuka sebelum TMA dan ditutup setelah TMB (lebih lama) untuk memaksimalkan pemasukan campuran bahan bakar.



*Gambar 2.6, Siklus Kerja Langkah Isap
Sumber: Data Olahan*

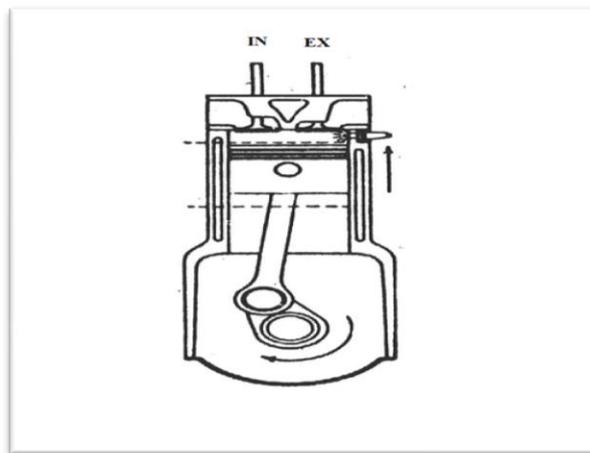
2. Langkah Kompresi

Pada langkah ini, terjadi pemadatan campuran bahan bakar di atas piston yang bertujuan untuk meningkatkan tekanan dan suhu. Pembakaran campuran ini (beberapa derajat sebelum TMA) akan menghasilkan pembakaran yang sempurna dan ledakan yang besar, yang nantinya dapat mendorong piston ke bawah.

- a. Kedua klep in dan ex dalam keadaan tertutup rapat supaya pemadatan campuran bahan bakar tidak terjadi kebocoran
- b. Piston bergerak dari TMB menuju ke TMA, supaya bisa mendorong dan memadatkan campuran.

Pada umumnya, besarnya tekanan kompresi untuk motor 4 tak berkisar antara 8-10 psi/kg sedangkan motor 2 tak di antara 6-8 psi/kg. Untuk mengukur tekanan kompresi dengan cara manual, yaitu dengan cara melepaskan terlebih dahulu busi pada kedudukannya di bagian kop mesin atau kepala silinder. Setelah tidak panas tekan lalu engkol. Ketika diengkol, tekanan yang besar akan dihasilkan pada lubang busi. Itu menandakan tekanan kompresi baik. Apabila

tidak ada tekanan berarti kompresi bocor yang disebabkan keausan pada bagian dinding silinder atau *liner boring*.

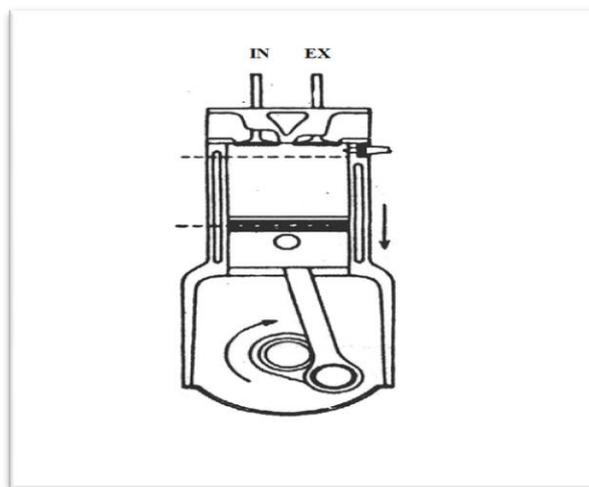


Gambar 2.7, Siklus Kerja Langkah Kompresi
Sumber: Data Olahan

3. Langkah Usaha/ Tenaga

Ledakan dari proses hasil pembakaran mendorong piston ke bawah menuju TMB, dan di sinilah tenaga dari sebuah mesin bakar 4 tak dihasilkan.

- a. Kedua klep *in* dan *ex* dalam keadaan tertutup rapat sehingga ledakan dari hasil pembakaran dapat dengan mudah penuh mendorong piston tanpa terjadi kebocoran
- b. Piston bergerak dari TMA menuju ke TMB, karena terdorong oleh hasil pembakaran yang berupa ledakan, dan nantinya tenaga disalurkan ke *Kruk As* atau poros engkol melalui stang piston.



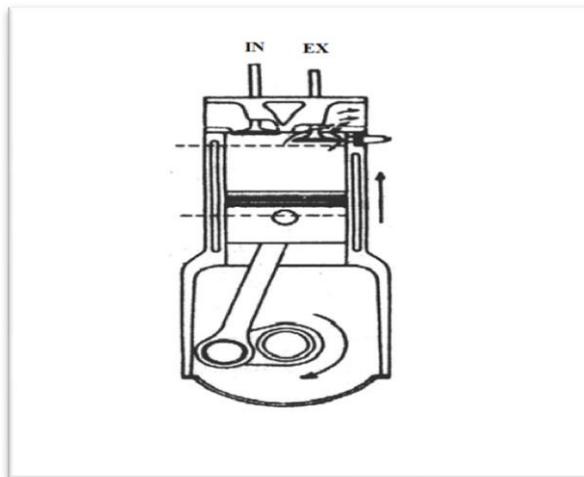
Gambar 2.8, Siklus Langkah Usaha
Sumber: Data Olahan

4. Langkah Buang

Sisa-sisa hasil proses pembakaran didorong keluar ruang bakar oleh piston menuju knalpot, supaya menghasilkan pembakaran yang sempurna pada langkah berikutnya.

- a. Klep *ex* (buang) terbuka untuk membuka saluran buang.
- b. Klep *in* (tutup) tertutup agar tidak bocor ke saluran masuk.
- c. Piston bergerak dari TMB menuju TMA, mendorong sisa-sisa pembakaran ke luar ruang bakar melalui saluran *exhaust manifold* lalu ke knalpot.

Piston hanya mendorong sisa-sisa pembakaran sampai di TMA sedangkan ruangan di atas piston yang disebut ruang bakar perlu dibersihkan. Maka dari itu, dilakukan pembilasan (pendorongan segar), yaitu pada saat akhir langkah buang dan awal langkah isap kedua klep *in* dan *ex* dalam keadaan bersamaan terbuka (*over lapping klep*).



Gambar 2.9, Siklus Kerja Langkah Buang
Sumber: Data Olahan

2.2.2 Motor Bakar

(Yulirianto, 2014) alat untuk menghasilkan tenaga dari proses pembakaran yang terjadi di dalam mesin. Dalam bagian ini, terjadi proses mengubah suatu energi (energi air, angin, kalor, listrik, panas, uap, dan lain-lain) menjadi gerak, atau disebut juga dengan *mechanical energy*. Contohnya pada sepeda motor, terjadi proses perubahan bahan bakar cair berupa bensin yang dicampur dengan udara di dalam karburator. Campuran terlebih dahulu diubah menjadi molekul yang lebih kecil atau berupa gas (pengkabutan). Kemudian, molekul ini masuk ke dalam ruang bakar suatu mesin, lalu diubah menjadi tenaga gerak melalui transmisi dan roda. Sepeda motor kemudian dapat melaju di jalan.

Tipe motor terbagi menjadi 2, yaitu:

1. Motor bakar luar (*External Combustion Engine/ ECE*), yaitu mekanisme pembangkit panas yang dihasilkan di luar mesin itu sendiri. Contohnya uap kereta api batu bara atau lokomotif dan turbin uap.

2. Motor bakar dalam (*Internal Combustion Engine/ICE*), yaitu mekanisme pengangkit panas yang dihasilkan didalam mesin itu sendiri. Contohnya, mesin sepeda motor, mobil, mesin diesel, mesin *rotary*, turbin gas, dan mesin jet.

Berdasarkan sistem penyalaaan, proses pembakaran terbagi 2:

1. SIE (*Spark Ignition Engine*), yaitu penyalaaan proses pembakaran di dalam mesin yang menggunakan percikan buanga api listrik dari busi (*spark plug*)di dalam ruang bakar. Contohnya mesin motor dan mobil yang menggunakan bahan bakar bensin.
2. CIE (*Compression Ignition Engine*), yaitu penyalaaan proses pembakaran di dalam mesin yang menggunakan suhu panas kompresi didalam ruang bakar. Suhu temperatur kira – kira mencapai 600°C dan tekanan kompresi di antara 15 – 22 psi/bar. Contoh, mesin mobil diesel yang menggunakan bahan bakar solar.

2.3. Software Pendukung

2.3.1 PHP (*Hipertext Preprocessor*)

(Rawung, 2017) PHP merupakan bahasa pemrograman yang banyak digunakan untuk membuat web dinamis. PHP sendiri singkatan dari “PHP *Hipertext Preprocessor*”. PHP adalah *software* yang diperoleh secara gratis karena bersifat *open source* dan dapat digunakan berbagai jenis platform sistem operasi. PHP merupakan bahasa pemrograman yang dapat disisipkan dalam skrip

HTML untuk membuat web dinamis dengan cepat. Untuk memerlukan bahasa pemrograman PHP, kita memerlukan web server untuk dapat menjalankannya.

2.3.2 Keunggulan PHP

Banyak sekali pemrograman yang berjalan pada server, masing-masing memiliki kelebihan dan kekurangan. Pada saat ini banyak website yang menggunakan program PHP sebagai dasar pengolahan data. Beberapa keunggulan yang dimiliki program PHP:

1. Tingkat akses PHP yang lebih cepat dan memiliki tingkat keamanan yang tinggi.
2. PHP memiliki tingkat lifecycle yang cepat, sehingga selalu mengikuti perkembangan teknologi internet.
3. PHP mampu berjalan di beberapa server yang ada, misalnya Apache, Microsoft IIS, PWS, AOLserver, Phttpd, dan Xitama.
4. PHP mampu berjalan di Linux sebagai platform sistem operasi utama bagi PHP, tetapi juga dapat berjalan di FreeBSD, Unix, Solaris, Windows, dan yang lain.

2.3.3 MySQL

(Rawung, 2017) MySQL adalah sebuah perangkat lunak manajemen database yang open source untuk digunakan sebagai menambah, mengupdate, menghapus, dan menampilkan data. MySQL merupakan tergolong bahasa Sql (*Structure Query Language*) yang mempunyai beberapa perintah yang pada umumnya digunakan yaitu *select*, *insert*, *update* dan *delete*.

(Harison & Alexyusandera, 2014) MySQL merupakan suatu manajemen database, suatu database adalah koleksi data terstruktur. Data tersebut dapat berupa apa saja, dari *list* sederhana sampai sebuah galeri gambar. Untuk menambah, mengakses, dan memproses data yang tersimpan dalam sebuah database, dibutuhkan suatu sistem manajemen database. Sejak komputer menjadi suatu alat yang digunakan untuk menanggulangi data dalam ukuran besar, manajemen database memegang peranan utama dalam perhitungan, sebagai utilitas tunggal maupun sebagian dari aplikasi lain.

2.3.4 UML (*United Modeling Language*)

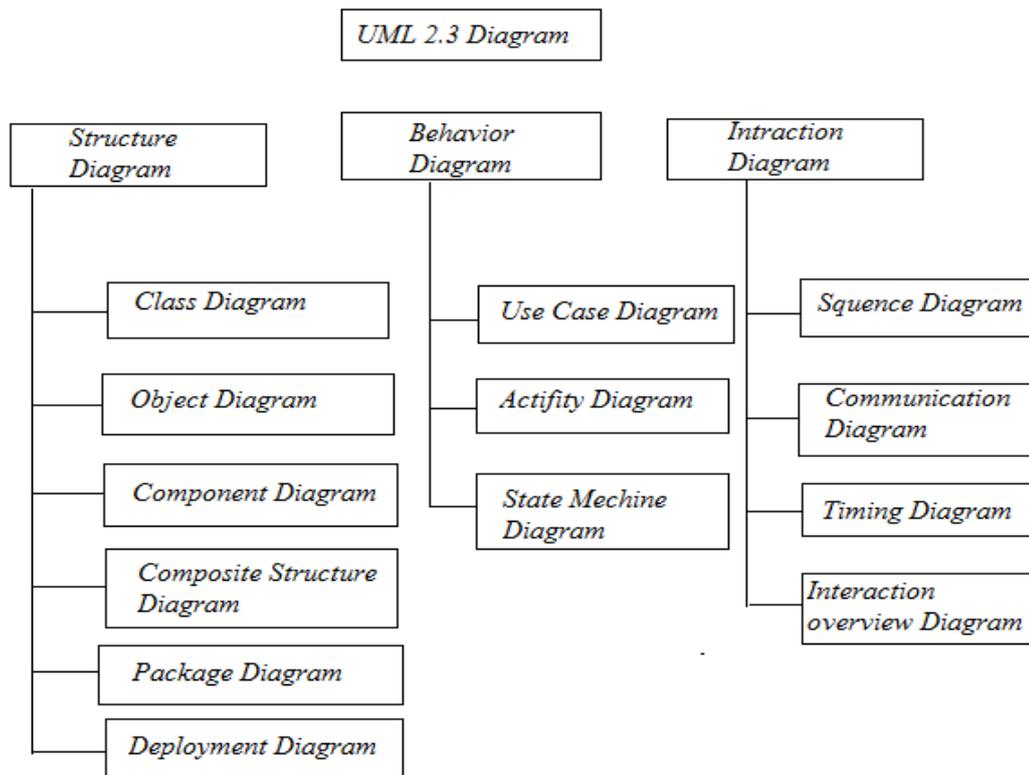
(A.S & Shalahuddin, 2013) Pada perkembangan teknik pemrograman berbasis objek, muncullah sebuah standarisasi bahasa pemodelan untuk pembangunan perangkat lunak yang dibangun dengan menggunakan teknik pemrograman berorientasi objek, yaitu *United Modeling Language* (UML). UML muncul karena adanya kebutuhan pemodelan visual untuk menspesifikasikan, menggambarkan, membangun, dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

2.3.5 Sejarah UML

Bahasa pemrograman berorientasi objek yang pertama dikembangkan dikenal dengan nama Simula-67 yang dikembangkan pada tahun 1967. Bahasa pemrograman ini kurang berkembang dan dikembangkan lebih lanjut, namun dengan kemunculannya telah memberikan sumbangan yang besar pada developer pengembangan bahasa pemrograman berorientasi objek selanjutnya.

2.3.6 Diagram UML

Dibawah ini adalah gambar UML 2.3 yang terdiri dari berbagai macam diagram dan di kelompokkan dalam 3 golongan.



Gambar 2.10, Diagram UML
Sumber: Data Olahan

Berikut adalah fungsi dari tiap-tiap golongan tersebut:

1. *Structure diagram* yaitu untuk mengilustrasikan suatu struktur statis dari sistem yang dimodelkan.
2. *Behavior diagram* yaitu untuk mengilustrasikan rangkaian perubahan yang terjadi pada sistem.
3. *Interaction diagram* yaitu kumpulan diagram yang digunakan untuk mengilustrasikan hubungan sistem dengan sistem yang lain.

2.3.7 Class Diagram

Untuk mengilustrasikan struktur sistem dari aspek penjabaran kelas yang akan dilakukan untuk membangun sistem. Ada dua macam yang terdapat didalam kelas diagram yaitu sebagai berikut:

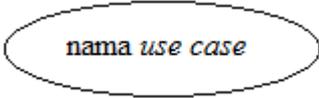
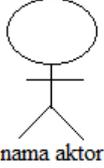
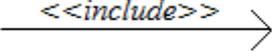
1. Atribut merupakan variabel yang dimiliki oleh suatu kelas.
2. Operasi atau metode adalah fungsi yang dimiliki kelas.

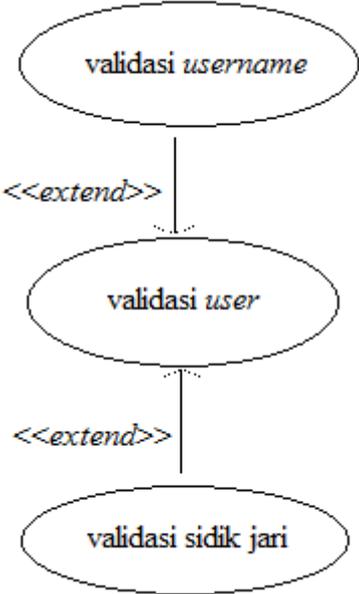
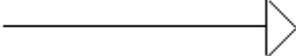
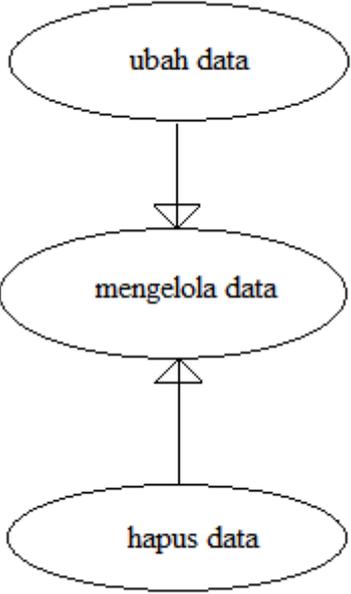
Diagram kelas dibuat agar pembuat program atau programmer membuat kelas-kelas sesuai rancangan di dalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sinkron.

2.3.8 Use Case Diagram

Sebuah gambaran hubungan antara satu atau lebih aktor dengan sistem yang akan dibuat. Dan dibawah ini adalah simbol yang digunakan pada bagian use case diagram:

Tabel 2.3, *Use Case Diagram*

Simbol	Deskripsi
<p><i>Use case</i></p> 	<p>Fungsionalitas yang disediakan sebagai unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja awal di awal <i>frase</i> nama <i>use case</i></p>
<p>Aktor/ <i>Actor</i></p> 	<p>Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat sendiri, jadi walaupun simbol dari aktor adalah gambar atau orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata di awal <i>frase</i> nama aktor</p>
<p>Asosiasi/ <i>Association</i></p> 	<p>Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor</p>
<p>Ekstensi/ <i>Extend</i></p> 	<p>Relasi <i>use case</i> tambahan sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip <i>inheritance</i> pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan, misal</p>

	 <p>Arah panah mengarah pada use case yang ditambah; biasanya use case yang menjadi <i>extend</i>-nya merupakan jenis yang sama dengan <i>use case</i> yang menjadi induknya.</p>
<p>Generalisasi/ <i>Generalization</i></p> 	<p>Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya, misal:</p>  <p>Arah panah mengarah pada <i>use case</i> menjadi generalisasinya (umum)</p>