

## **BAB II**

### **LANDASAN TEORI**

Pada bab ini akan dibahas dasar teori yang berhubungan dengan judul pada tugas akhir ini yaitu Pembangunan Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan dan Perawatan Mobil. Hal-hal yang akan dibahas meliputi teori dasar tentang *artificial intelligence*, teori tentang sistem pakar, representasi pengetahuan, dan masih banyak teori lainnya.

#### **2.1 *Artificial Intelligence***

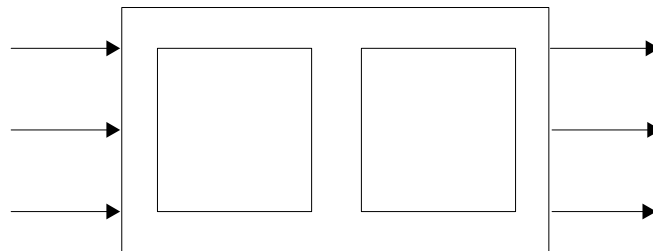
*Intelligence* atau intelegensia adalah seorang yang pandai melaksanakan pengetahuan yang dimilikinya. Dengan pengertian tersebut diatas maka bisa ditarik suatu kesimpulan, walaupun seseorang banyak memiliki pengetahuan, tetapi bila ia tidak bisa melaksanakannya dalam praktek, maka ia tidak dapat digolongkan ke dalam intelegensia. Dengan kata lain, intelegensia adalah kemampuan manusia untuk memperoleh pengetahuan dan pandai melaksanakannya dalam praktek. Hal ini berarti kemampuan dalam berfikir dan menalar. Pada batas-batas tertentu *artificial intelligence* (AI) memungkinkan komputer bisa menerima pengetahuan melalui input manusia dan menggunakan pengetahuannya itu melalui simulasi proses penalaran dan berfikir manusia untuk memecahkan berbagai masalah.

Bagian utama aplikasi *artificial intelligence* adalah pengetahuan (*knowledge*), suatu pengertian tentang beberapa wilayah subjek yang diperoleh melalui pendidikan dan pengalaman. Walaupun komputer tidak mungkin mendapat pengalaman atau belajar dan meneliti seperti manusia, tetapi ia dapat memperoleh pengetahuan yang dibutuhkannya itu melalui upaya yang diberikan oleh seorang pakar manusia.

Hampir semua pangkalan pengetahuan (*knowledge base*) sangat terbatas dalam arti terfokuskan kepada suatu masalah khusus. Pada saat pangkalan pengetahuan itu sudah terbentuk, maka teknik *artificial intelligence* bisa digunakan untuk memberi kemampuan baru kepada komputer agar bisa berfikir,

menalar dan membuat inferensi (mengambil keputusan berdasarkan pengalaman) dan membuat pertimbangan-pertimbangan yang didasarkan kepada fakta dan hubungan-hubungannya yang terkandung dalam pangkalan pengetahuan itu.

Dengan pangkalan pengetahuan dan kemampuan untuk menarik kesimpulan melalui pengalaman (inferensi), komputer dapat disejajarkan sebagai alat bantu yang bisa digunakan secara praktis dalam memecahkan masalah dan pengambilan keputusan. (Gambar 2.1, menggambarkan konsep komputer yang menggunakan teknik *artificial intelligence (AI)* dalam suatu aplikasi).



**Gambar 2.1 Penerapan konsep AI dalam Komputer**

Dengan teknik pelacakan pangkalan pengetahuan untuk mencari fakta dan hubungannya yang relevan, komputer bisa mencapai satu atau lebih solusi alternatif pada masalah yang diberikan. Pangkalan pengetahuan komputer dan kemampuan inferensi telah meningkatkan daya guna komputer bagi manusia.

*Artificial intelligence* telah memberikan suatu kemampuan baru kepada komputer untuk memecahkan masalah yang lebih besar dan lebih luas, tidak hanya terbatas kepada soal-soal perhitungan, penyimpanan dan pengambilan data atau pengendalian yang sederhana saja. Berdasarkan Gambar 2.1 di atas menyimpulkan beberapa cara komputer tradisional dalam mengolah data.

Tujuan kedua *artificial intelligence (AI)* adalah untuk bisa memahami intelegensia manusia. Dengan menerapkan pola berfikir manusia pada komputer, maka kita dilatih untuk belajar bagaimana cara menyimpan pengetahuan dalam otak kita dan bagaimana cara mengaplikasikannya. Terlebih dahulu kita harus mengerti betul tentang pola berfikir kita sendiri, bagaimana teknik penalarannya, dan bagaimana teknik pendekatannya dalam memecahkan suatu masalah.

Inputs

(Questions

Problems, etc)

Knowledge  
Base

Dengan kata lain kita harus belajar dari diri sendiri, bagaimana cara kita belajar, dan sampai sejauh mana kelemahan dan kekuatan kita. Singkatnya kita harus mengetahui bagaimana cara kita belajar. Sehingga dengan demikian, kita bisa memperoleh pengertian yang lebih baik atas pikiran kita dan akan bisa mengarahkan kepada metoda belajar yang lebih baik dan mampu menerapkan intelegensia kita ke dalam masalah dunia nyata. Bidang AI yang khusus ditujukan kepada penelitian seperti tersebut diatas disebut sebagai *sain kognitif*.

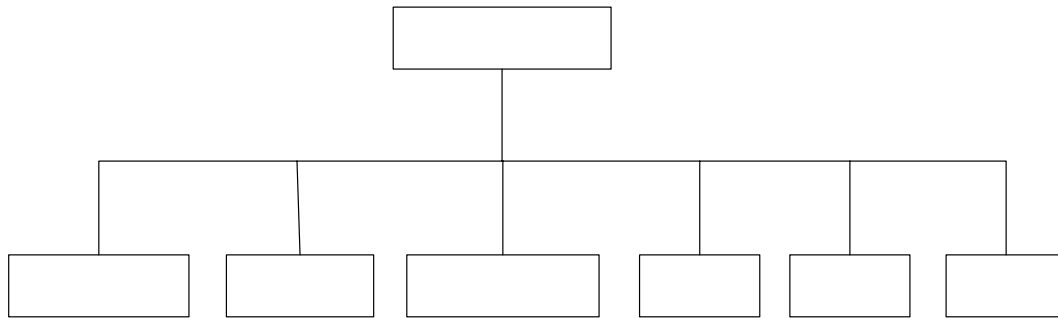
## 2.2 Aplikasi Artificial Intelligence

Karena kelenturan proses AI yang sangat luar biasa, maka perangkat lunak (*software*) AI dapat dirancang untuk berbagai masalah yang memerlukan kekhususan. Banyak macam masalah yang tidak bisa menggunakan solusi algoritma. Masalah non-bilangan misalnya sering tidak berhasil dikerjakan dengan algoritma. Tetapi dengan teknik AI, baik masalah yang tidak menentu atau yang mempunyai arti ganda (*ambiguous*), bisa dipecahkan dengan mudah. Dengan software algoritma, suatu masalah prosedural pasti bisa terpecahkan. Sementara dengan teknik AI masih diragukan, apakah bisa atau tidak, atau mungkin hanya bisa dipecahkan sebagian. Oleh karena itu, AI lebih tepat digunakan untuk masalah-masalah yang tidak terorganisasi, dunia nyata yang tidak sempurna. Dalam hal ini teknik AI lebih baik daripada software konvensional lainnya.

Aplikasi AI pertama kali digunakan dalam memecahkan permasalahan permainan (*games*) dan pemecahan masalah teka-teki. Para peneliti merasa, kemampuan komputer untuk menjalankan games merupakan demonstrasi yang luar biasa tentang intelegensia manusia. Tentu saja dalam hal yang terkendali dan terbatas, program AI juga ditulis untuk memecahkan teka-teki dan pertanyaan teka-teki. Sampai sekarang permainan atau *games* selalu digunakan sebagai contoh dalam mengajar AI.

Walaupun permainan sangat menarik, tetapi tidak terlalu berguna bagi kepentingan masyarakat luas. Oleh karena itu para pemakai sekarang mulai mencari program yang bisa lebih meningkatkan produktifitas kerjanya secara optimal. Dalam hal ini sudah terjawab oleh AI dengan diterapkannya aplikasi

pemecahan masalah (*problem solving*), sistem pakar (*expert system*), pengolahan bahasa alami (*natural language processing*), *computer vision*, robot dan pendidikan. Semua itu dapat dilihat dalam Gambar 2.2. Dalam memecahkan masalah atau membantu membuat keputusan dalam suatu dominan tertentu. Semua aplikasi tersebut diatas sudah menggunakan pangkalan pengetahuan dan teknik inferensi.



**Gambar 2.2 Daerah utama aplikasi Artificial Intelligence**

### 2.3 Sistem Pakar

Pada beberapa tahun sebelumnya sistem pakar telah muncul dan mengalami perkembangan yang sangat pesat sebagai cabang dari artificial intelligence. Sasaran dari sistem ini adalah mengambil pengetahuan yang disimpan dalam domain tertentu, menggambarkannya dalam suatu modul, memperluas strukturnya dan mentransfer ke user lain yang memiliki domain yang sama. Seperti yang telah dikemukakan oleh Mishkoff (1985:6) mengenai pengertian dari sistem pakar yaitu

*“Sistem Pakar adalah program komputer yang meniru kemampuan seseorang atau beberapa pakar dalam bidang-bidang pengetahuan tertentu memecahkan masalah seperti pakar-pakar tersebut memecahkan masalah dalam bidangnya”.*

Penggunaan teknik AI ini pada umumnya dilakukan untuk membuat *software* sistem pakar, yaitu suatu program yang bertindak sebagai penasehat atau konsultan pintar. Dengan mengambil pengetahuan yang disimpan dalam domain suatu masalah yang bagaimanapun rumitnya dan bisa mengambil keputusan yang tepat dan akurat yang selalu dilakukan seorang pakar.

Artificial

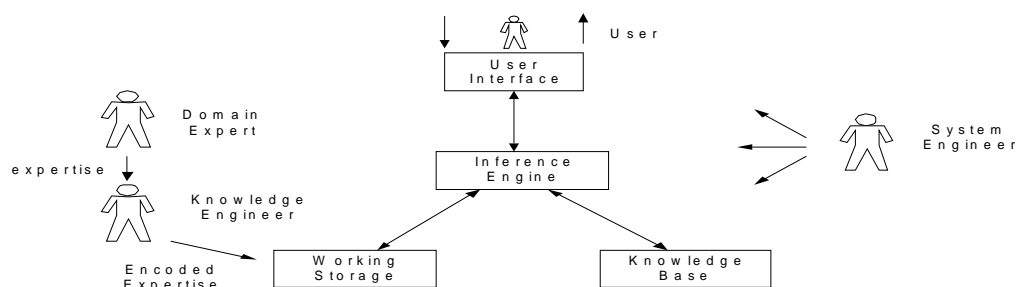
Natural  
Proc

General Problem  
Solving

Expert System

Pengetahuan yang sangat bernilai merupakan sumber utama yang sangat penting. Tetapi sayangnya, hanya dimiliki segelintir orang pakar saja. Oleh karena itu penting sekali untuk memperoleh kepakaran, agar masyarakat luas bisa menggunakannya. Seorang pakar bisa meninggal, bisa pensiun, atau sakit, atau bisa juga pindah ke tempat lain, sehingga menimbulkan kekosongan seorang pakar yang sangat dibutuhkan. Buku-buku bisa menyajikan pengetahuan yang berharga, tetapi buku masih meninggalkan masalah pemahaman dan penerapannya bagi para pembaca, tetapi sistem pakar merupakan media langsung untuk melakukan pekerjaan seorang pakar.

Sistem pakar terdiri dari empat komponen utama yaitu : *Knowledge Base* (Pangkalan Pengetahuan), *Working Storage*, *Motor Inferensi* dan *User Interface*. Seperti Gambar 2.3 yang tercantum di bawah ini :



**Gambar 2.3 Hubungan komponen-komponen utama Sistem Pakar**

Berdasarkan hal tersebut diatas maka menurut Dennis Merrit (1989:4) mengemukakan bahwa :

*“Knowledge base berisi semua fakta, ide, hubungan, dan interaksi suatu domain kecil. Motor inferensi bertugas untuk analisis pengetahuan dan menarik kesimpulan berdasarkan pangkalan pengetahuan. Software user interface berfungsi sebagai media pemasukan pengetahuan ke dalam pangkalan pengetahuan dan melakukan komunikasi dengan user. User juga bisa meminta sistem pakar untuk menerangkan proses jalannya pengambilan keputusan. Hal ini bisa dilakukan, karena biasanya user tidak atau kurang percaya kepada kesimpulan atau keputusan yang diambil sistem pakar karenanya user tidak mau melaksanakan nasehat-nasehat atau rekomendasinya”.*

Beberapa sistem pakar bisa menjelaskan langkah-langkah yang ditempuhnya sampai pada suatu kesimpulan atau keputusan, sehingga dengan demikian user bisa mengerti logika yang diambil oleh sistem pakar, karena user akan melaksanakan segala keputusannya itu dengan perasaan senang dan puas. Sistem pakar sangat ideal bagi seseorang yang harus memilih serangkaian alternatif terbaik dari daftar pilihan panjang. Berdasarkan kriteria yang diberikan kepadanya, sistem pakar dapat menentukan pilihan yang terbaik.

Banyak sekali keuntungan yang diperoleh dari pemakaian suatu sistem pakar. Keuntungan-keuntungan itu di antaranya sebagai berikut.

1. Sistem pakar dapat dipakai kapan saja dan dengan waktu pemakaian yang relatif tidak terbatas. Hal ini berbeda dengan seorang pakar, yang perlu makan, minum, santai, tidur dan istirahat. Sistem pakar tidak memerlukan semua itu, karena sistem pakar tidak merasa lelah dan tidak butuh istirahat sehingga dapat dipakai selama 24 jam, tentu saja bila tidak terjadi kerusakan pada sistem yang digunakan.
2. Banyak bidang yang dapat dibuat sistem pakarnya, sehingga kemampuan seorang pakar yang mungkin sangat langka dapat diperoleh dan diterapkan oleh orang yang membutuhkannya.
3. Sistem pakar merupakan program komputer yang selalu berada pada kondisi terbaiknya, hal ini berbeda dengan seorang pakar ketika merasa lelah maka kemampuan analisis masalahnya akan berkurang. Pada saat demikian seorang pakar akan sering membuat banyak kekeliruan dan kesalahan analisis. Sebaliknya suatu sistem pakar akan selalu memberikan analisis yang terbaik, walaupun sebatas pada pengetahuan yang dimilikinya.

4. Suatu sistem pakar tidak melibatkan hubungan pribadi, hal ini karena pada kenyataannya tidak setiap orang merasa cocok dengan orang lain. Karena seseorang mungkin saja menolak memakai hasil analisis masalah dari seorang pakar tertentu, sebaliknya, seorang pakar juga mungkin tidak akan memberikan hasil analisis terbaiknya pada seseorang yang tidak disukainya. Sistem pakar komputer tidak akan terlibat dengan hubungan antar pribadi yang melibatkan emosi antar manusia sehingga hasil analisisnya menjadi sangat akurat atau objektif.
5. Sistem pakar komputer merupakan suatu perangkat lunak yang mudah untuk diperbaharui, diperbanyak, dipercanggih sesuai dengan perkembangan zaman. Sistem pakar juga dapat dimasukkan basis pengetahuan dengan cepat tanpa memerlukan banyak waktu berbeda dengan seorang pakar.

Walaupun banyak sekali keuntungan yang dapat ditemukan pada suatu sistem pakar, tetapi tetap saja suatu komputer yang menjalankan suatu sistem pakar tersebut hanya dapat menampung sedikit pengetahuan jika dibandingkan dengan otak manusia yang begitu kompleks. Daya tampung sistem pakar ini sangat dipengaruhi oleh alat simpan misalnya harddisk, disket, dan memori pada komputer tersebut. Akibatnya, maka suatu sistem pakar diusahakan dibuat untuk menganalisis dan memecahkan suatu masalah tertentu atau biasa disebut dengan domain. Contoh sistem pakar :

- a) Diagnosis medis kedokteran.
- b) Diagnosis kerusakan peralatan elektronik misalnya komputer, radio, tape, vcd, televisi.
- c) Perancangan sistem komunikasi dan radio.
- d) Perancangan tata ruang kota dan sistem informasi geografis.
- e) Sistem pemantauan dan peramalan cuaca.
- f) Diagnosis kerusakan kendaraan bermotor maupun mobil

Terdapat dua jenis sistem pakar, sesuai dengan cara pencarian penyelesaian masalahnya, yaitu :

### 2.3.1 Sistem Pakar Produksi (*production system*)

Sistem pakar produksi adalah jenis sistem pakar yang menyimpan pengetahuan sebagai fakta-fakta dan kaidah-kaidah, dimana masing-masing kaidah tersebut disebut kaidah produksi.

Sistem pakar produksi memiliki dua komponen dasar yaitu basis pengetahuan (*Knowledge Base*) dan mesin inferensi (*Inference Engine*). Basis pengetahuan adalah data atau pengetahuan yang dipergunakan untuk membuat suatu keputusan. Sedang mesin inferensi adalah bagian dari sistem pakar yang mereduksi fakta-fakta baru dari fakta-fakta yang telah ada dengan mempergunakan kaidah-kaidah yang ada.

Proses produksi ini menyangkut penjadwalan dan unifikasi. Disamping itu mesin inferensi juga mengontrol aliran dan tahapan inferensi. Mesin inferensi akan mengambil fakta yang ada dari basis kaidah atau basis data statis dan memori yang bekerja atau basis data dinamis kemudian menggunakannya untuk menguji kaidah-kaidah tersebut selama proses unifikasi dan ketika sukses, kaidah tersebut ditambahkan ke memori yang bekerja.

### 2.3.2 Sistem Pakar Kerangka (*frame-based system*)

Sistem pakar jenis ini dalam melakukan diagnosis dan memecahkan masalah memerlukan suatu kerangka (*frame*), Dimana semua masalah dan diagnosis kemungkinan pemecahan masalahnya telah didefinisikan terlebih dahulu.

Cara pendefinisiannya dengan cara membuat pohon-pohon masalah dimana puncaknya berupa gejala awal diteruskan dengan pertanyaan-pertanyaan yang akan menanyakan gejala lanjutan dari gejala awal tadi, selanjutnya dengan melakukan pendeteksian maka akan diketahui pemecahan masalahnya. Pendefinisian yang lain yaitu dengan cara perancangan basis aturan yang akan dijelaskan pada bab berikutnya. Cara perancangan basis aturan inilah yang digunakan penulis sebagai basis dari sistem pakar untuk memecahkan masalah pada mobil. Selain cara diatas juga dengan menggunakan metode rangkaian maju (*Forward Chaining*) dan metode rangkaian mundur (*Backward Chaining*).

## 2.4 Mesin Inferensi

Mesin inferensi selain mengontrol aliran dan tahapan inferensi juga merupakan bagian dari sistem pakar yang mendeduksi fakta-fakta baru dari fakta-fakta yang telah ada dengan mempergunakan kaidah-kaidah yang ada. Dimana proses ini menyangkut penjadwalan dan unifikasi. Pada pengontrolan ini mesin inferensi menentukan kaidah mana yang akan diuji terlebih dahulu dan apa yang dilakukan seandainya suatu kaidah sukses atau gagal. Mesin inferensi mengambil fakta yang ada dari basis kaidah statis dan memori yang bekerja atau basis data dinamis kemudian menggunakannya untuk menguji kaidah-kaidah selama proses unifikasi. Ketika kaidah sukses, kaidah tersebut ditambahkan ke memori yang bekerja. Pada beberapa kasus, mungkin terdapat lebih dari satu fakta yang dapat mengunifikasi kaidah yang ada. Oleh karenanya mesin inferensi harus menentukan kaidah mana yang harus dipenuhi.

Adapun mesin inferensi terdiri atas dua komponen yaitu strategi inferensi dan juga strategi kontrol yang memiliki penjelasan seperti dibawah ini:

### 2.4.1 Strategi Inferensi

Strategi inferensi adalah berkaitan dengan cara yang digunakan untuk menarik suatu kesimpulan atau suatu keputusan. Dalam strategi ini yang sering digunakan adalah aturan logika yang menyatakan bahwa jika diketahui B bernilai benar maka akan terdapat aturan yang berbunyi, IF B THEN A, dan dapat disimpulkan bahwa A bernilai benar.

Penggunaan aturan logika terdapat beberapa hal yang dapat kita pelajari. Pertama bahwa aturan logika sangat sederhana penalarannya sehingga dapat dengan mudah dimengerti. Kedua, terdapatnya implikasi yang juga sah namun tidak dapat kita lakukan. Contoh suatu logika menyatakan bahwa jika suatu nilai A bernilai salah maka dapat disimpulkan bahwa nilai B juga bernilai salah.

Dengan demikian dapat diambil suatu kesimpulan bahwa aturan logika adalah pembentukan fakta baru dari suatu aturan yang telah diketahui. Adapun penjelasan mengenai metoda-metoda tersebut adalah sebagai berikut :

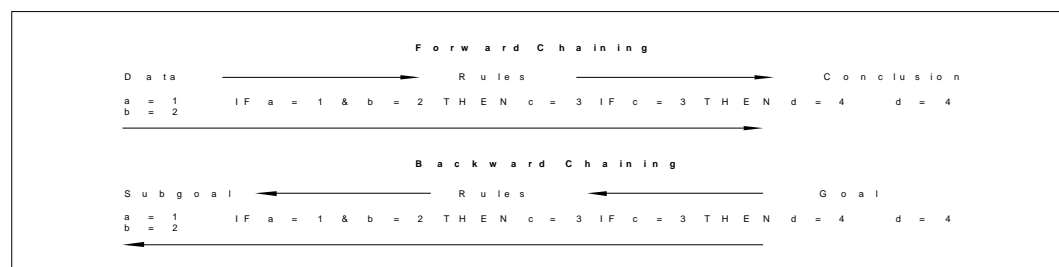
### 2.4.1.1 Metoda Rangkaian Mundur (*Backward Chaining*)

Pada metoda ini penelusuran dimulai dari hipotesa kemudian barulah mencari informasi untuk memenuhi hipotesa tersebut.

### 2.4.1.2 Metoda Rangkaian Maju (*Forward Chaining*)

Penelusuran dimulai dengan mencari informasi-informasi kemudian barulah untuk menyimpulkan atau mencari hipotesa berdasarkan informasi yang ada, dalam hal ini fakta-fakta atau aturan-aturan haruslah bernilai benar.

Adapun perbedaan untuk masing-masing metoda tersebut dapat ditunjukkan pada gambar dibawah ini.



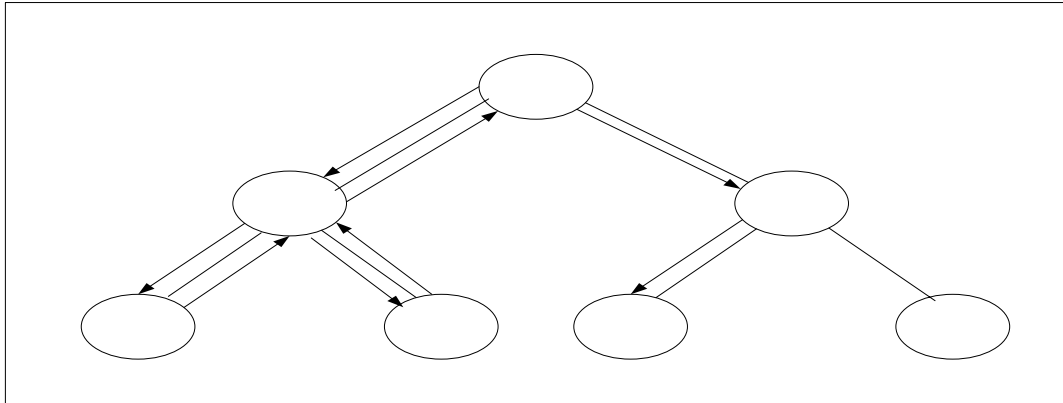
Gambar 2.4 Perbedaan diantara *Forward* dan *Backward Chaining*

## 2.4.2 Strategi Kontrol

Strategi kontrol adalah berkaitan dengan teknik penelusuran yang digunakan untuk mencapai keadaan tujuan. Teknik yang biasa digunakan untuk memecahkan masalah dalam sistem pakar adalah telusur mundur dan telusur maju namun kadang juga menggunakan pencarian dengan sistem BFS (*Breadth First Search*) dan sistem DFS (*Depth First Search*). Penggunaan keempat metoda tersebut dapat digunakan secara bersama dan mengatasi suatu masalah.

### 2.4.2.1 Metoda DFS (*Depth First Search*)

Metoda DFS merupakan teknik penyelesaian problema dengan menelusuri setiap lintasan yang mungkin sampai kedalaman maksimal untuk mencapai suatu konklusi atau tujuan (*goal*) sebelum mencoba lintasan yang lain atau dengan kata lain metoda ini mencari informasi dengan detail terlebih dahulu baru mencari konklusi. Sebagai gambaran dari metoda tersebut berikut ini :

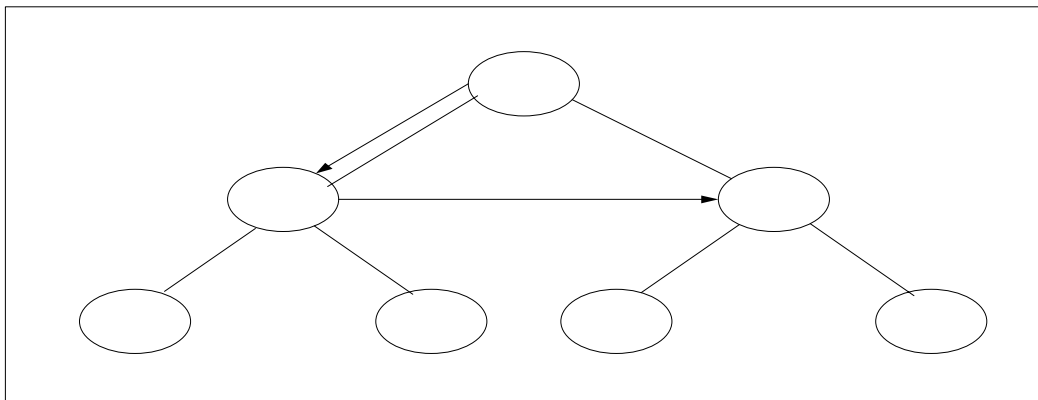


**Gambar 2.5 Metoda DFS**

Metoda ini akan menelusuri dengan urutan sebagai berikut : A - B - D - B - E - B - A - C - F (*goal*), dalam pencarian ini kita telusuri dari cabang kiri sampai kedalaman tertentu atau sampai goal. **B**

**2.4.2.2 Metoda BFS (*Breath First Search*)**

Metoda BFS adalah melakukan pencarian dengan menelusuri setiap simpul pada level yang sama terlebih dahulu, kemudian turun ke level yang berikutnya sampai ditemukannya suatu goal. Contoh pencarian dari BFS adalah sebagai berikut **E**



**Gambar 2.6 Metoda BFS**

Metoda akan menelusuri dengan urutan sebagai berikut : A - B - C - (*goal*). Tetapi kebanyakan dalam sistem pakar menggunakan metoda pencarian *Depth First* yaitu dengan menggali sedalam mungkin dan dengan mengikuti rantai aturan yang mengarahkan pertanyaan-pertanyaan yang diberikan oleh sistem.

### 2.4.2.3 Metode Hill-Climbing

Teknik pencarian dengan menggunakan metode *Hill-Climbing* merupakan teknik pencarian yang sedikit lebih pintar dari DFS dan BFS, karena dalam metoda *Hill-climbing* telah ditambahkan suatu teknik Heuristic. Heuristic adalah teknik atau aturan yang menjamin bahwa pencarian yang dilakukan dalam arah yang benar. Dalam metoda *Hill-Climbing* secara formal mencari simpul berikutnya yang notabene jaraknya terdekat ke simpul asal dan pada akhirnya merupakan jarak yang terdekat ke tujuan.

Ketiga metoda tersebut diatas mengalami kelelahan (*exhaustive*) karena sistem mengevaluasi seluruh aturan yang ada sehingga pencarian memakan waktu lama juga memakan memori yang banyak. Tetapi dengan adanya strategi kontrol masalah tersebut dapat di atasi yaitu dengan mengkombinasikan antara telusur maju, BFS.

## 2.5 Representasi Pengetahuan

Hampir semua *artificial intelligence* terbuat dari dua bagian pokok, yaitu pangkalan pengetahuan dan mekanisme inferensi. Pangkalan pengetahuan berisi fakta-fakta tentang objek dalam domain yang ditentukan dan saling berhubungan satu sama lain. Seluruh program AI difokuskan pada domain tertentu. Pangkalan pengetahuan juga bisa berisi pikiran, teori, prosedur praktis dan saling hubungan. Pangkalan pengetahuan membentuk sumber sistem intelegensia dan digunakan oleh mekanisme inferensi untuk melakukan penalaran dan menarik kesimpulan.

Mekanisme inferensi adalah suatu rangkaian prosedur yang digunakan untuk menguji pangkalan pengetahuan dengan cara yang sistematik pada saat menjawab pertanyaan, memecahkan persoalan atau membuat keputusan dalam suatu domain yang telah ditentukan. Cara komputer “berpikir” tentang subjek domain, mekanisme inferensi melakukan pelacakan melalui pangkalan pengetahuan untuk mencari jawaban dan solusinya.

### 2.5.1 Skema Representasi Pengetahuan

Langkah pertama dalam membuat program artificial intelligence adalah membangun pangkalan pengetahuan. Agar komputer bisa bertingkah laku seperti seorang intelegensia, maka ia harus mempunyai pengetahuan terlebih dahulu tentang suatu domain tertentu. Sebenarnya pengetahuan itu sudah ada. Apakah ia masih berada dalam otak seorang atau beberapa orang pakar, atau mungkin masih dalam bentuk buku, artikel, memo, dan prosedur, yang penting semuanya harus sudah terkodifikasi.

Sebenarnya berbagai representasi skema pengetahuan itu sudah dikembangkan sejak bertahun-tahun yang lalu. Representasi skema ini mempunyai dua karakteristik umum. Pertama, bisa diprogram dengan bahasa komputer yang ada dan disimpan dalam memori. Kedua, dirancang agar fakta-fakta dan pengetahuan lainnya yang terkandung didalamnya bisa digunakan untuk penalaran. Penalaran pengetahuan berisi data struktur yang bisa dimanipulasi oleh sistem inferensi, yang merupakan bagian utama program AI. Sistem inferensi menggunakan teknik pelacakan dan penyocokan pola (*pattern matching*) dalam pangkalan pengetahuan agar bisa menjawab pertanyaan, menarik kesimpulan atau tugas-tugas lainnya yang bersifat intelegensi. Skema representasi pengetahuan secara umum telah dikategorikan sebagai suatu prosedural. Skema representasi prosedural berhubungan dengan aksi dan prosedur.

Kunci keberhasilan setiap pembuatan program AI terletak pada pemilihan skema representasi pengetahuan yang paling baik dan paling tepat, dan yang paling penting lagi yang sesuai dengan domain pengetahuan dan masalah yang akan dipecahkan. Pemilihan ini akan tergantung kepada rekayasa pengetahuan yang dihasilkan oleh pengalaman yang luas dalam merancang software AI.

### 2.5.2 Logika

Bentuk representasi pengetahuan yang paling tua adalah logika, yaitu suatu pengkajian ilmiah tentang serangkaian penalaran, sistem kaidah dan prosedur yang membantu proses penalaran. Logika dianggap sebagai bagian filsafat. Pengembangan dan perbaikan proses tersebut umumnya terjadi pada zaman Yunani kuno.



**Gambar 2.7 Menggunakan logika menuju penalaran**

Gambar 2.7 menjelaskan bentuk umum proses logika. Pertama, memberikan informasi, membuat statement atau mencatat observasi, bentuk ini merupakan input proses logika dan disebut **Input Premises** or **Facts** (suatu hal-ihwal yang dianggap benar). Premis ini digunakan oleh proses logika untuk membuat output yang terdiri dari konklusi dan inferensi. Dengan proses tersebut, fakta yang diketahui benar akan digunakan untuk memperoleh fakta baru yang juga harus benar.

Dalam hal ini ada dua bentuk dasar penalaran : Deduktif dan Induktif. Kedua bentuk ini digunakan dalam logika untuk membuat inferensi yang diambil dari premis.

### 2.5.3 Penalaran Deduktif

Jika untuk memperoleh inferensi tertentu kita menggunakan premis umum (kebenaran umum yang dianggap benar), maka proses ini disebut penalaran deduktif atau deduksi. Penalaran ini bergerak dari prinsip umum menuju khusus.

Proses deduksi umumnya dimulai dari suatu pernyataan premis dan inferensi. Umumnya terdiri dari tiga bagian : **Premis Mayor**, **Premis Minor**, dan **Konklusi**. Untuk maksud penalaran deduktif hampir setiap masalah dan argument dapat diletakkan pada bentuk ini, Misalnya :

**Premis mayor** : Jika hujan turun saya tidak akan lari pagi.

**Premis minor** : Pagi ini hujan turun.

**Konklusi** : Oleh karena itu pagi ini saya tidak akan lari pagi

Agar bisa menggunakan penalaran deduktif, pada umumnya kita harus memformat masalahnya terlebih dahulu. Jika format itu sudah dibuat, maka konklusi harus benar bila premisnya benar. Sekali lagi, seluruh ide dalam soal ini adalah tidak lain terkecuali untuk mengembangkan pengetahuan baru dari

pengetahuan yang sudah ada dan diberikan sebelumnya.

#### 2.5.4 Penalaran Induktif

Penalaran induktif adalah kebalikan dari deduktif, yaitu bila deduktif dimulai dari masalah umum menuju ke masalah khusus sedangkan induktif mulai dari masalah khusus ke masalah umum. Atau dengan kata lain, penalaran induktif menggunakan sejumlah fakta atau premis yang mantap untuk menarik kesimpulan umum. Sebagai contoh dibawah ini akan mencoba menjelaskan proses induktif, pernyataan hanya digunakan untuk mengekspresikan masalah.

**Premis** : Sepuyer yang salah menyebabkan peralatan mobil .

**Premis** : karburator rusak menyebabkan mobil tidak berjalan.

**Premis** : Dinamo stater rusak menyebabkan peralatan peralatan mobil tidak berfungsi.

**Konklusi** : Peralatan stater otomatis rusak, merupakan penyebab utama rusaknya peralatan mobil.

Satu hal yang sangat menarik tentang penalaran induktif, ialah bahwa konklusi ini tidak pernah final atau mutlak. Konklusi dapat berubah jika fakta baru sudah ditemukan. Jika semua fakta yang mungkin, tidak dimasukkan ke dalam salah satu premis, maka selalu akan timbul ketidak pastian di dalam konklusi. Akibatnya, hasil proses penalaran induktif akan selalu mengandung beberapa ukuran ketidak pastian. Walaupun demikian, jika lebih banyak fakta dan premis yang digunakan dalam proses penalarannya, maka ketidakpastian konklusi itu akan berkurang. Semakin banyak pengetahuan yang di miliki, semakin banyak inferensi konklusinya.

#### 2.6 Modul Antarmuka (*Interface*)

Antarmuka (*Interface*) merupakan komponen dalam sistem pakar yang dapat berkomunikasi dengan pengguna. Komunikasi yang dilakukan oleh suatu antarmuka adalah komunikasi yang bersifat timbal balik atau dua arah. Pada sistem pakar yang sederhana pemakai harus dapat menerangkan gambaran dari permasalahannya, oleh sistem pakar kemudian akan memberikan jawaban atau

tanggapan yang berupa saran-saran atau kesimpulan.

Dalam antarmuka dari sistem harus ada atau mempunyai suatu fungsi tambahan. Dimungkinkan pemakai ingin sistem dapat menerangkan proses pengambilan keputusan, bagaimana cara yang dilakukan oleh sistem untuk sampai pada keputusan tersebut.

## 2.7 Modul Akuisisi Pengetahuan

Suatu sistem akuisisi data pada umumnya dibentuk sedemikian rupa sehingga sistem tersebut berfungsi untuk mengambil, mengumpulkan dan menyimpan data dalam bentuk siap untuk diproses lebih lanjut. Seperti yang dikemukakan oleh Abu Hasan (1999:1) mengenai sistem akuisisi pengetahuan bahwa :

*“Sistem akuisisi pengetahuan dapat didefinisikan sebagai suatu sistem yang berfungsi untuk mengambil, mengumpulkan dan menyiapkan pengetahuan, sehingga memprosesnya untuk menghasilkan pengetahuan yang dikehendaki. Jenis serta metode yang dipilih pada umumnya bertujuan untuk menyederhanakan setiap langkah yang dilaksanakan pada keseluruhan proses.”*

Modul akuisisi pengetahuan adalah bagian dari sistem pakar yang bertugas untuk memperoleh pengetahuan dari seorang pakar. Adapun memperoleh pengetahuan seorang pakar ada beberapa cara, yaitu :

- *Full Manual* : Pencarian pengetahuan tidak menggunakan alat tetapi *knowledge enginer* yang berperan aktif, misalnya dengan melakukan interview (menanyakan pada seorang pakar baik secara terencana maupun tidak terencana dan juga dilakukan dengan cara observasi).
- *Semi Automatic* : Pencarian pengetahuan dengan menggunakan *step-by-step* yaitu dengan menggunakan alat disamping dengan melakukan interview.
- *Full Automatic* : Pencarian pengetahuan dilakukan sepenuhnya oleh

alat yaitu komputer.

Karena sistem berbasis pengetahuan bergantung pada jumlah dan kualitas pengetahuan maka untuk keberhasilan dari sistem pakar, metoda akuisisi yang baik mutlak untuk diperlukan.

## **2.8 Keuntungan dan Kelemahan *Artificial Intelligence***

Bila pada suatu saat nanti AI sudah betul-betul mapan, bagi para pemakai, teknik AI ini benar-benar akan merupakan berita besar. Tetapi walaupun demikian, tidak berarti bahwa AI tidak mempunyai kelemahan. Seperti penemuan-penemuan manusia lainnya, AI pun tidak luput dari segala kelemahan kelemahan yang ada.

Sebagai karya perdana yang besar, software AI akan mendorong harga komputer menjadi semakin mahal, tetapi sebaliknya komputer yang akan datang akan memberikan kenikmatan, kenyamanan, dan kesenangan yang lebih. Para pemakai akan menjadi lebih akrab dengan komputer, karena mereka bisa berkomunikasi dengan menggunakan bahasa alami seperti bahasa Inggris atau bahasa lainnya. Mereka akan terbebas dari segala kewajiban dan keharusan belajar bahasa pemrograman dan sistem operasi. Dengan karya besar ini para pemakai yang tidak terlatih sama sekali pun dapat menggunakan dan memanfaatkan komputer itu dan akan menghasilkan karya yang sangat berguna bagi kepentingannya. Keuntungan utama lainnya adalah komputer akan menjadi semakin lebih berguna. Seperti sudah dijelaskan diatas, komputer merupakan suatu solusi untuk mencari dan memecahkan masalah.

AI juga memiliki kelemahan dan kekurangan, dalam pengembangan aplikasi AI menunjukkan kemajuan yang sangat mengagumkan. Tapi kemajuan dan keuntungan itu tidak cuma-cuma dan harus dibayar dengan mahal. Memang komputer yang dilengkapi dengan teknik AI segala-galanya akan lebih hebat, tapi dari segi financialnya pun akan lebih mahal, dan barangkali tentang harga inilah yang akan menjadi kendala kekurangan dan kelemahannya yang utama. Dewasa ini teknik AI masih belum bisa dipraktekkan pada komputer pribadi yang kecil. Hampir seluruh penelitian AI dan banyak aplikasinya, masih dilaksanakan dalam

main frame atau komputer besar lainnya seperti *Digital Equipment Corporation*.

Kekurangan dan kelemahan lainnya, pengembangan perangkat lunak AI akan tetap merupakan hal yang sulit. Program AI masih tetap akan merupakan barang mewah karena rumitnya pembuatan. Untuk membuat software AI diperlukan waktu yang sangat lama dan karenanya biayanya akan menjadi lebih tinggi. Alat pengembangan software seperti bahasa pemrograman AI sudah lebih baik dan generator sistem pakar akan bisa membantu mempercepat dan memudahkan pembuatan software AI. Tetapi semua ini masih tetap merupakan barang yang langka dan mahal dan bahkan dalam penggunaan software-software pemrograman AI dan generator itu pun nampaknya masih diperlukan adanya bakat. Akhirnya hanya ada beberapa produk AI komersial yang praktis sudah dipasarkan. Ada beberapa interface bahasa alami, beberapa sistem pakar, dan banyak bahasa dan alat pembuatan program lainnya.

## **2.9 Alat Pembangun Sistem Pakar**

Ada dua jenis alat pembangun sistem pakar yaitu bahasa pemrograman dan shell.

### **2.9.1 Bahasa Pemrograman**

Secara umum pemrograman sistem pakar ditujukan terutama untuk mendukung pengolahan pengetahuan, pencarian secara heuristic serta kemampuan pengolahan atau manipulasi data secara simbolik. Bahasa yang umumnya digunakan adalah bahasa pemrograman LISP dan Prolog yang merupakan sebagian dari bahasa yang banyak digunakan untuk proses pengembangan sistem pakar, meskipun bahasa pemrograman yang lainnya yang lebih konvensional masih dapat digunakan.

### **2.9.2 Shell**

*Shell expert system* atau juga disebut generator adalah paket *software*, yang khusus dibuat untuk membantu pembuatan *expert system*. *Shell* menyediakan kerangka kerja dasar di dalam data atau pengetahuan dapat

dimasukkan dan dimanipulasi dengan cara yang sudah ditentukan terlebih dahulu.

## 2.10 Rekayasa Perangkat Lunak dengan Metode Waterfall

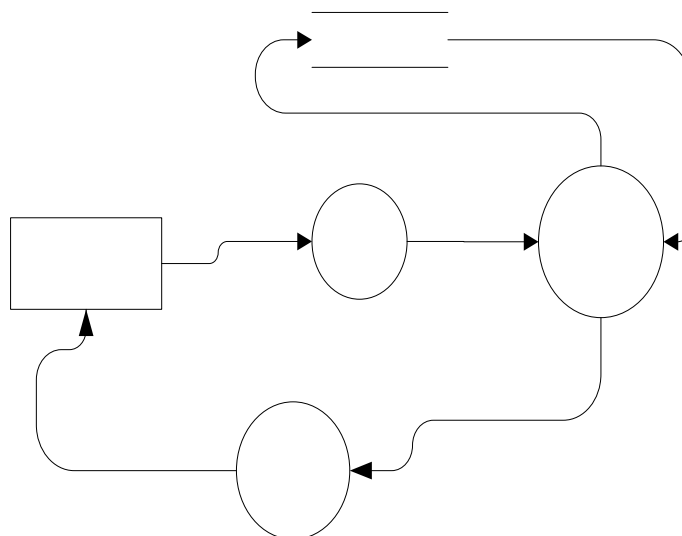
Waterfall adalah sebuah model pengembangan perangkat lunak dilakukan secara sekuensial, dimana satu tahap dilakukan setelah tahap sebelumnya selesai dilaksanakan.

### 2.10.1 Rekayasa Sistem

Merupakan tahap pertama kali untuk merumuskan sistem yang akan dibuat, dimana bertujuan agar pihak pengembang benar-benar memahami sistem yang akan dibuat. Adapun rekayasa yang dipakai adalah :

#### 2.10.1.1 Data Flow Oriented dengan tool Data Flow Diagram (DFD).

DFD merupakan suatu teknik penggambaran / pemodelan menggunakan notasi-notasi grafis yang menunjukkan aliran informasi dan perubahannya yang diterapkan sebagai perubahan data masukan menjadi keluaran.



**Gambar 2.8 Contoh DFD**

#### 2.10.1.2 Kamus Data

Adalah daftar terorganisir dari semua elemen data yang ada pada suatu sistem dengan definisi yang jelas dan tepat, sehingga user dan analisis sistem bisa mendapat kesepahaman dari input, output yang ada. Adapun hasil dari proses

analisa ini adalah SRS (*Software Requirement Specification*).

**Tabel 2.1 Contoh Kamus Data**

No	Data	Kamus Data
1	User_ID	[UserID+Password]
2	Hasil_Inputan	/*UserID+Password*/
3	Data_User_ID	[UserID+NamaUser+Password]
4	Hasil_Olah_User_ID	/*Data hasil pengolahan data user*/
5	Info_Data_User	/*Data hasil pengolahan data user*/

### 2.10.1.3 Analisis

Merupakan proses atau tahapan penggalan *requirement*, yaitu kemampuan-kemampuan apa saja yang harus dimiliki oleh sistem yang akan dikembangkan, data apa yang akan diolah atau yang akan dihasilkan oleh sistem. Kemampuan pada proses analisa mencakup :

1. Mampu menganalisa konsep yang abstrak serta mengatur kembali ke dalam pembagian logika dan mensintesis pemecahan masalah berdasarkan pembagiannya.
2. Mampu menyerap fakta atau informasi.
3. Mampu mengerti lingkungan pemakai.
4. Mampu menerapkan sistem dari software maupun perangkat keras.
5. Mampu berkomunikasi baik dalam bentuk tulisan maupun lisan.

### 2.10.1.4 Implementasi

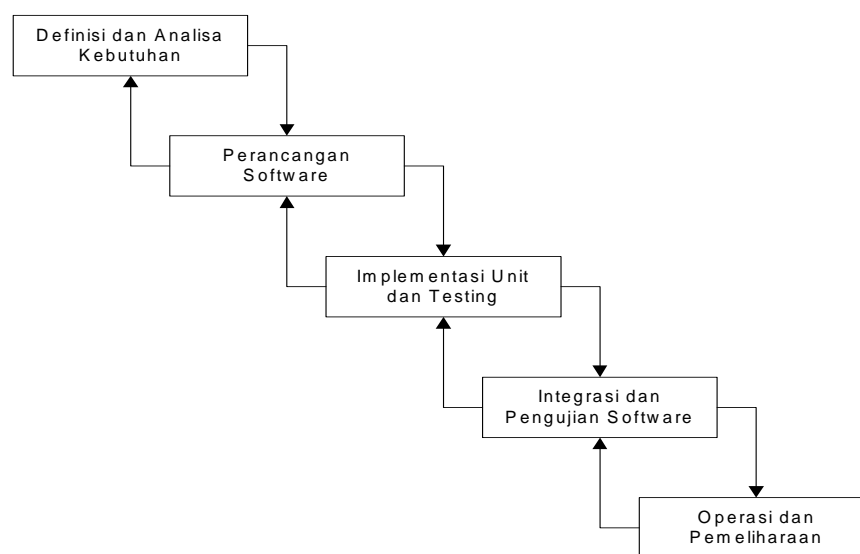
Merupakan tahap pengimplementasian hasil perancangan ke dalam bahasa pemrograman, yang selanjutnya dibangun buku manual atau pedoman sebagai acuan informasi untuk user yang akan menggunakan aplikasi sistem ini.

### 2.10.1.5 Pengujian

Setelah implementasi sistem, selanjutnya dilakukan pengujian untuk mendapatkan perangkat lunak yang benar-benar valid dan sesuai dengan kebutuhan yang sudah dideskripsikan.

### 2.10.1.6 Pemeliharaan

Setelah dilakukan pengujian dan sistem diyakini benar-benar memenuhi syarat dan valid, selanjutnya sistem tersebut didistribusikan ke *user*. Akan tetapi, hal ini tidak berarti pekerjaan pengembangan telah selesai, masih terdapat tahapan pemeliharaan sistem yang harus dilakukan. Selain itu tahap ini juga dilakukan evaluasi terhadap sistem yang baru untuk melihat apakah sistem telah memenuhi tujuan yang diinginkan. Dari hasil tersebut akan dilakukan perubahan yang perlu terhadap sistem.



**Gambar 2.9 Metode RPL dengan Metode Waterfall**

### 2.11 Sekilas Tentang Visual Basic 6.0

Microsoft Visual Basic adalah bahasa pemrograman yang bekerja dalam lingkup Microsoft Windows. Microsoft Visual Basic bekerja dengan menggunakan objek-objek sebagai komponen pemrogramannya. Setiap objek digambarkan pada layar dan melakukan pengaturan property terhadap objek yang digambarkan. Beberapa kemampuan atau manfaat dari Visual Basic di antaranya seperti :

- a. Untuk membuat program aplikasi berbasis Windows.
- b. Untuk membuat objek-objek pembantu program seperti kontrol

ActiveX, file Help, aplikasi Internet, dan pembuatan sistem pakar.

- c. Menguji program (debugging) dan menghasilkan program akhir berakhiran EXE yang bersifat *executable*, atau dapat digunakan.

Beberapa dukungan yang diberikan oleh Microsoft Visual Basic adalah

1. Visual Basic memungkinkan programnya untuk mengembangkan antar muka pemakai secara mudah dan tepat.
2. Visual Basic merupakan produk pengembangan dari bahasa basic, sebuah bahasa yang sangat populer terutama dalam lingkungan akademis. Meskipun lingkungan pengembangan yang sepenuhnya grafis dan bahasa pemrograman dari Visual Basic dengan interpreter basic awal, keluwesan dan kesederhanaan basic yang semula tetap, sampai batas-batas tertentu.
3. Visual Basic mampu menghasilkan file EXE yang berdiri sendiri, kecuali jika aplikasi yang dibuat memerlukan pengaksesan terhadap basis data. Yang dimaksud dengan berdiri sendiri adalah file yang sudah dikompilasi tersebut tidak lagi memerlukan *dynamic link libraries* (DLL).
4. Visual Basic menyediakan banyak objek kontrol yang berhubungan dengan pemrograman basis data, sehingga dapat mempermudah dan mempersingkat pemrograman itu sendiri.
5. Visual Basic menyediakan database desktop.

Berikut ini merupakan langkah-langkah pembuatan program dengan Microsoft Visual Basic for windows adalah :

1. Membuat tampilan antarmuka program. Pada tahap awal, tampilan tidak perlu dibuat sempurna, karena tampilan ini dapat diubah-ubah pada tahap akhir pembuatan program.
2. Membuat kode program. Kode Visual Basic akan diletakkan pada kontrol atau *form* yang akan menggunakan kode tersebut. Kode menjadi milik sebuah kontrol atau *form* akan dijalankan jika terdapat kejadian terhadap kontrol atau *form* tersebut.
3. Mengkompilasikan program. Tahap akhir dari pembuatan program adalah mengkompilasikan program sehingga menjadi program yang berdiri sendiri dan dapat dijalankan dalam lingkungan windows.