

PENGEMBANGAN SISTEM PAKAR DALAM MEMBANGUN SUATU APLIKASI

Muhammad Dahria

Program Studi Sistem Informasi, STMIK Triguna Dharma

m.dahria@gmail.com

ABSTRACT: Expert system is one branch of AI (Artificial Intelligence) which makes extensive use of knowledge especially to problem resolution of expert human level. An expert is a person who has expertise in a particular field, that is, an expert who has knowledge or special skill that others do not know or are not able in the field he has. Applications or expert systems can be written in any computer language, either in C, Pascal, Basic or language with Development Tool Visual Basic. But in the later development which is more popular today, it is developed with Development Tool Visual Basic 8.0 (Vb. Net).

Key words: *expert system, knowledge, problem solving.*

A. PENDAHULUAN

Sistem pakar (expert system) adalah sistem yang berusaha mengapdosikan pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para ahli. Dengan sistem pakar, orang awam pun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli. Bagi para ahli, sistem pakar juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman.

Sistem pakar adalah salah satu cabang dari AI yang membuat penggunaan secara luas *knowledge* yang khusus untuk penyelesaian masalah tingkat manusia yang pakar. Seorang pakar adalah orang yang mempunyai keahlian dalam bidang tertentu, yaitu pakar yang mempunyai *knowledge* atau kemampuan khusus yang orang lain tidak mengetahui atau mampu dalam bidang yang dimilikinya. Ketika sistem

pakar dikembangkan pertama kali sekitar tahun 70-an, sistem pakar hanya berisi *knowledge* yang eksklusif. Namun demikian sekarang ini istilah sistem pakar sudah digunakan untuk berbagai macam sistem yang menggunakan teknologi sistem pakar itu. Teknologi sistem pakar ini meliputi bahasa sistem pakar, program dan perangkat keras yang dirancang untuk membantu pengembangan dan pembuatan sistem pakar.

Walaupun tujuan umum penyelesaian masalah masih jauh dari apa yang diharapkan, namun sistem pakar berfungsi sangat baik dalam batasan domainnya. Hal ini dapat dibuktikan bahwa sistem pakar telah banyak diaplikasikan dalam berbagai bidang yang trend saat ini, seperti bisnis, kedokteran, ilmu pengetahuan, dan teknik.

B. KEUNTUNGAN SISTEM PAKAR

Sistem pakar mampu memecahkan masalah tanpa dipengaruhi oleh faktor dari luar seperti

intimidasi, paksaan kejiwaan, faktor ekonomi ataupun perasaan. Berikut ini beberapa keuntungan dari syitem pakar:

1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli.
2. Bisa melakukan proses secara berulang secara otomatis.
3. Menyimpan pengetahuan dan keahlian para pakar.
4. Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka).
5. Mampu beroperasi dalam lingkungan yang berbahaya.
6. Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian.
7. Tidak memerlukan biaya saat tidak digunakan, sedangkan pada pakar manusia memerlukan biaya sehari-hari.
8. Dapat digandakan (diperbanyak) sesuai kebutuhan dengan waktu yang minimal dan sedikit biaya.
9. Dapat memecahkan masalah lebih cepat daripada kemampuan manusia dengan catatan menggunakan data yang sama.
10. Menghemat waktu dalam pengambilan keputusan.
11. Meningkatkan kualitas dan produktivitas.

C. KONSEP DASAR SISTEM PAKAR

Konsep dasar sistem pakar mengandung keahlian, ahli/pakar, pengalihan keahlian, mengambil keputusan, aturan, kemampuan menjelaskan.

Keahlian

Keahlian adalah suatu kelebihan penguasaan pengetahuan di bidang tertentu yang diperoleh dari pelatihan, membaca atau dari pengalaman.

Bentuk pengetahuan yang termasuk keahlian:

1. Fakta-fakta pada lingkup permasalahan tertentu.

2. Teori-teori pada lingkup permasalahan tertentu.
3. Aturan-aturan berkenaan dengan lingkup permasalahan tertentu.
4. Meta –knowledge (pengetahuan tentang pengetahuan)

Ahli / Pakar

Seorang ahli adalah seseorang yang mampu menjelaskan suatu tanggapan, mempelajari hal-hal baru seputar topik permasalahan, menyusun kembali pengetahuan jika dipandang perlu, memecahkan masalah dengan cepat dan tepat.

Pengalihan Keahlian

Tujuan dari sistem pakar adalah untuk mentransfer keahlian dari seorang pakar ke dalam komputer kemudian ke masyarakat. Proses ini meliputi 4 kegiatan, yaitu perolehan pengetahuan (dari para ahli atau sumber-sumber lainnya), representasi pengetahuan ke komputer, kesimpulan dari pengetahuan dan pengalihan pengetahuan ke pengguna.

Mengambil Keputusan

Hal yang unik dari sistem pakar adalah kemampuan untuk menjelaskan dimana keahlian tersimpan dalam basis pengetahuan. Kemampuan komputer untuk mengambil kesimpulan dilakukan oleh komponen yang dikenal dengan mesin inferensi yaitu meliputi prosedur tentang pemecahan masalah.

Aturan (Rule)

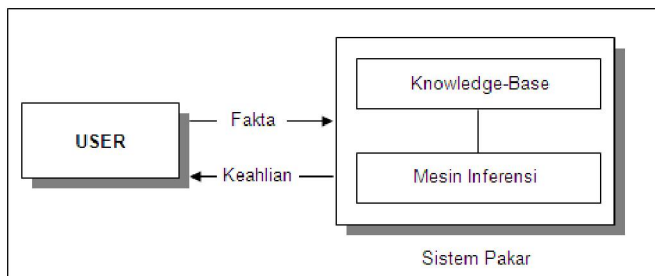
Sistem pakar yang dibuat merupakan sistem yang berdasarkan pada aturan – aturan dimana program disimpan dalam bentuk aturan-aturan sebagai prosedur pemecahan masalah. Aturan tersebut biasanya berbentuk IF – THEN.

Kemampuan Menjelaskan

Keunikan lain dari sistem pakar adalah kemampuan dalam menjelaskan atau memberi saran/rekomendasi serta juga menjelaskan mengapa beberapa tindakan/saran tidak direkomendasikan.

Gambar 1 menggambarkan konsep dasar suatu sistem pakar knowledge-base. Pengguna menyampaikan fakta atau informasi untuk

sistem pakar dan kemudian menerima saran dari pakar atau jawaban ahlinya. Bagian dalam sistem pakar terdiri dari dua komponen utama, yaitu knowledge base yang berisi knowledge dan mesin inferensi yang menggambarkan kesimpulan. Kesimpulan tersebut merupakan respons dari sistem pakar atas permintaan pengguna.



Gambar 1. Konsep Dasar Fungsi Sistem Pakar

D. TUJUAN SISTEM PAKAR

Pengalihan keahlian dari para ahli ke komputer untuk kemudian dialihkan ke orang lain yang bukan ahli. Seorang ahli adalah seseorang yang mampu menjelaskan suatu tanggapan, mempelajari hal-hal baru seputar topik permasalahan (domain), menyusun kembali pengetahuan jika dipandang perlu, memecahkan aturan-aturan jika dibutuhkan dan menentukan relevan tidaknya keahlian mereka.

Proses pengalihan pengetahuan membutuhkan 4 aktivitas, yaitu :

1. Tambahkan pengetahuan (dari para ahli atau sumber-sumber lainnya).
2. Representasikan pengetahuan (ke komputer)
3. Lakukan inferensi pengetahuan.
4. Pengalihan pengetahuan ke user.

Pengetahuan yang disimpan ke dalam komputer disebut basis pengetahuan. Ada 2 tipe pengetahuan, yaitu: fakta dan prosedur (biasanya berupa aturan).

E. PERMASALAHAN YANG DAPAT DISELESAIKAN DENGAN SISTEM PAKAR

Sistem pakar dapat diaplikasikan untuk memecahkan berbagai permasalahan. Umumnya kecepatan dalam memecahkan masalah pada suatu sistem pakar relatif lebih cepat dibandingkan dengan seorang pakar manusia. Hal ini sudah dibuktikan pada beberapa sistem pakar yang terkenal di dunia.

Berikut ini contoh berbagai masalah yang dapat diselesaikan dengan menggunakan sistem pakar:

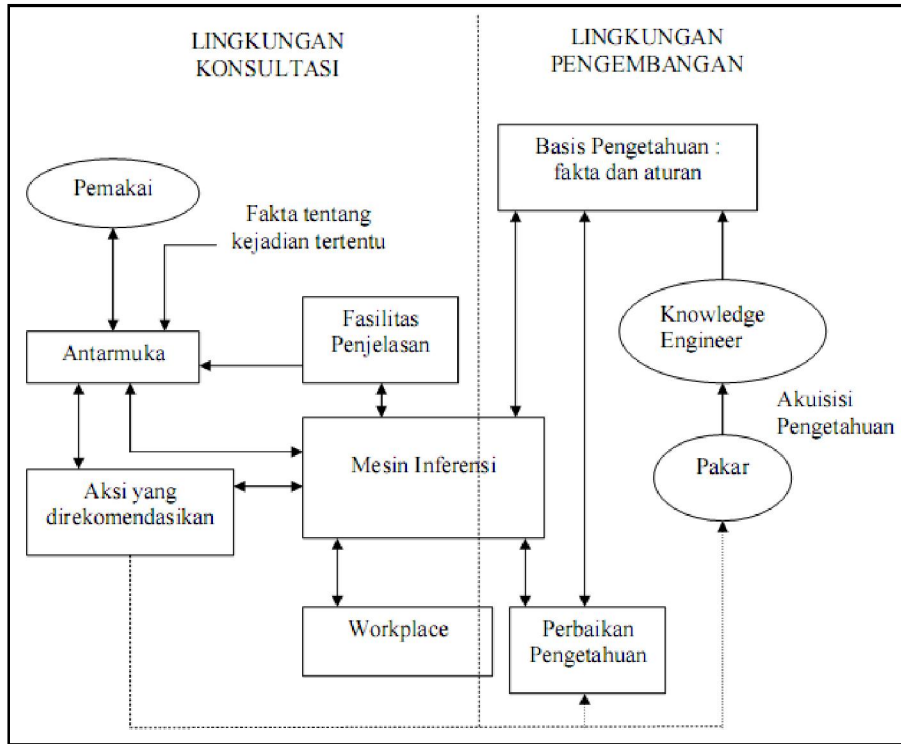
1. Interpretasi, pengambilan keputusan dari hasil observasi, termasuk: pengawasan, pengenalan ucapan, analisis citra, interpretasi sinyal dan beberapa analisis kecerdasan lainnya.
2. Prediksi, diantaranya: peramalan, prediksi demografis, peramalan ekonomi, prediksi lalu lintas, estimasi hasil, militer, pemasaran dan keuangan.
3. Diagnosis, diantaranya: medis, elektronik, mekanis dan diagnosis perangkat lunak.
4. Perancangan: perancangan sirkuit dan bangunan.
5. Perencanaan, seperti: perencanaan keuangan, komunikasi, produk dan manajemen proyek.
6. Monitoring: Computer-Aided Monitoring Sistem.
7. Debugging: memberikan resep obat terhadap suatu kegagalan.
8. Instruksi, untuk diagnosis, debugging dan perbaikan kinerja.
Kontrol, terhadap interpretasi, prediksi, perbaikan dan monitoring kelakuan sistem.

F. STRUKTUR SISTEM PAKAR

Ada dua bagian utama sistem pakar:

1. Lingkungan pengembangan (development environment): digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar.
2. Lingkungan konsultasi (consultation environment): digunakan oleh pengguna yang bukan pakar untuk memperoleh pengetahuan pakar.

Struktur dari sistem pakar diberikan pada gambar 1 berikut.



Gambar 2. Struktur Sistem Pakar

G. BASIS PENGETAHUAN PAKAR

Basis pengetahuan berisi pengetahuan-pengetahuan dalam penyelesaian masalah. Ada 2 bentuk pendekatan basis pengetahuan:

1. Penalaran berbasis aturan (*rule-based reasoning*)

Pada penalaran berbasis aturan, pengetahuan direpresentasikan dengan menggunakan aturan berbentuk IF-THEN. Bentuk ini digunakan apabila kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu, dan si pakar dapat menyelesaikan masalah tersebut secara berurutan. Disamping itu, bentuk ini juga digunakan apabila dibutuhkan penjelasan tentang jejak (langkah-langkah) pencapaian solusi.

Contoh : aturan identifikasi hewan

Rule 1 : IF hewan berambut dan menyusui THEN hewan mamalia

Rule 2 : IF hewan mempunyai sayap dan bertelur THEN hewan jenis burung

Rule 3 : IF hewan mamalia dan memakan daging THEN hewan karnivora
Dan seterusnya.....

2. Penalaran berbasis kasus (*case-based reasoning*)

Pada penalaran berbasis kasus, basis pengetahuan akan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang terjadi sekarang (fakta yang ada). Bentuk ini digunakan apabila user menginginkan untuk tahu lebih banyak lagi pada kasus-kasus yang hampir sama (mirip). Selain itu bentuk ini juga digunakan bila kita telah memiliki sejumlah situasi atau kasus tertentu dalam basis pengetahuan.

H. MOTOR INFERENSI (INFERENCE ENGINE)

Mesin inferensi (*inference engine*) merupakan bagian yang bertindak sebagai pencari solusi dari suatu permasalahan berdasar pada kaidah-kaidah yang ada dalam basis

pengetahuan sistem pakar. Selama proses inferensi, mesin inferensi memeriksa status dari basis pengetahuan dan memori kerja (working memory) untuk menentukan fakta apa saja yang diketahui dan untuk menambahkan fakta baru yang dihasilkan ke dalam memori kerja tersebut. Fakta-fakta yang merupakan hasil dari proses inferensi disimpan dalam memori kerja.

Ada dua strategi pencarian dasar yang bisa digunakan oleh mesin inferensi dalam mencari kesimpulan untuk mendapatkan solusi bagi permasalahan yang dihadapi sistem pakar, yaitu runut maju (*forward chaining*) dan runut balik (*backward chaining*).

1. Forward Chaining (Runut Maju)

Runut maju merupakan strategi pencarian yang memulai proses pencarian dari sekumpulan data atau fakta, dari data-data tersebut dicari suatu kesimpulan yang menjadi solusi dari permasalahan yang dihadapi. Mesin inferensi mencari kaidah-kaidah dalam basis pengetahuan yang premisnya sesuai dengan data-data tersebut, kemudian dari kaidah-kaidah tersebut diperoleh suatu kesimpulan. Runut maju memulai proses pencarian dengan data sehingga strategi ini disebut juga data-driven.

2. Backward Chaining (Runut Balik)

Runut balik merupakan strategi pencarian yang arahnya kebalikan dari runut maju. Proses pencarian dimulai dari tujuan, yaitu kesimpulan yang menjadi solusi permasalahan yang dihadapi. Mesin inferensi mencari kaidah-kaidah dalam basis pengetahuan yang kesimpulannya merupakan solusi yang ingin dicapai, kemudian dari kaidah-kaidah yang diperoleh, masing-masing kesimpulan dirunut balik jalur yang mengarah ke kesimpulan tersebut. Jika informasi-informasi atau nilai dari atribut-atribut yang mengarah ke kesimpulan tersebut sesuai dengan data yang diberikan maka kesimpulan tersebut merupakan solusi yang dicari, jika tidak sesuai maka kesimpulan tersebut bukan merupakan solusi yang dicari. Runut balik memulai proses pencarian dengan suatu tujuan sehingga strategi ini disebut juga goal-driven.

I. PEMILIHAN MODEL PELACAKAN (INFERENCE ENGINE)

Dalam pengembangan Sistem Pakar manakah metode yang harus dipilih, apakah Forward Chaining atau Backward Chaining ?

Sebagai panduan adalah sebagai berikut :

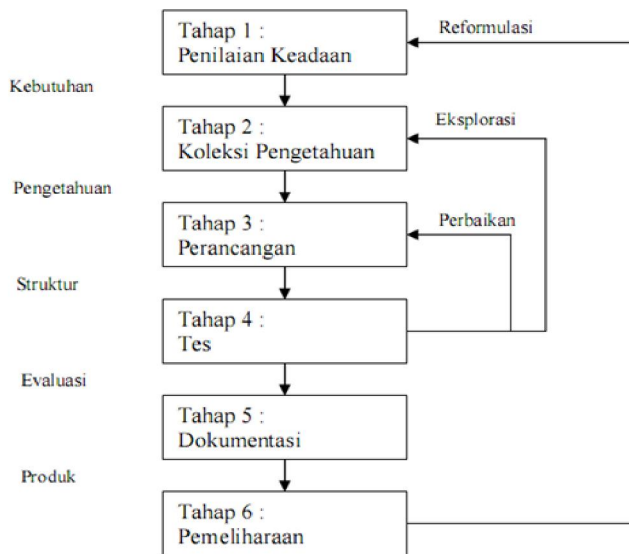
1. Jika masalah yang dihadapi lebih dekat ke fan out (sekumpulan fakta yang bisa menuju ke banyak konklusi), maka pilihlah backward chaining.
2. Jika masalah yang dihadapi lebih dekat ke fan in (sekumpulan hipotesis yang bisa menuju ke banyak pertanyaan), maka pilihlah forward chaining.
3. Banyak cara untuk mendapatkan sedikit konklusi, maka pilih forward chaining.
4. Sedikit cara untuk mendapatkan banyak konklusi, maka pilih backward chaining.
5. Jika kita belum mendapatkan berbagai fakta, dan kita tertarik hanya pada satu konklusi yang mungkin, maka gunakan backward chaining.
6. Jika kita benar-benar sudah mendapatkan berbagai fakta, dan kita ingin untuk mendapatkan konklusi dari fakta-fakta itu, maka gunakan forward chaining.

J. LANGKAH-LANGKAH PENGEMBANGAN SISTEM PAKAR

Berikut ini tahapan-tahapan pembuatan/ pengembangan Sistem pakar:

1. Mengidentifikasi masalah dan kebutuhan
Mengkaji situasi dan memutuskan dengan pasti tentang masalah yang akan dikomputerisasi dan apakah dengan sistem pakar bisa lebih membantu atau tidak.
2. Menentukan problema yang cocok
Ada beberapa syarat yang harus dipenuhi agar sistem pakar dapat bekerja dengan baik, yaitu:
 - Domain masalah tidak terlalu luas.
 - Kompleksitasnya menengah, jika masalah terlalu mudah atau masalah yang terlalu kompleks tidak perlu menggunakan sistem pakar.
 - Tersedianya ahli.
 - Menghasilkan solusi mental bukan fisik.

- Sistem pakar hanya memberikan anjuran tidak bisa melakukan aktifitas fisik, seperti membau atau merasakan.
3. Mempertimbangkan alternatif
Kaji alternatif lain yang lebih mudah, cepat dan sesuai dengan masalah yang ingin diselesaikan, menggunakan sistem pakar atau komputer tradisional.
 4. Menghitung pengembalian investasi
Termasuk diantaranya biaya pembuatan sistem pakar, biaya pemeliharaan, biaya training.
 5. Memilih alat pengembangan
Bisa menggunakan software pembuat sistem pakar (seperti SHELL) atau dirancang dengan bahasa pemrograman sendiri (misal dengan PROLOG, Bahasa C++ atau Visual Basic).
 6. Merekayasa pengetahuan
Memperoleh pengetahuan dan menyempurnakan banyak kaidah yang paling sesuai.
 7. Merancang system
Pembuatan prototype dan menterjemahkan pengetahuan menjadi aturan-aturan.
 8. Melengkapi pengembangan.
Perluasan prototype ke dalam sistem yang final yaitu dengan meluaskan bagian demi bagian dan setiap bagian diuji apakah sudah berjalan sesuai. Langkah-langkah pengembangan sistem pakar diberikan pada gambar 3 berikut.



Gambar 3. Langkah-langkah Pengembangan Sistem Pakar

9. Menguji dan mencari kesalahan system
Lakukan percobaan dengan user yang menginginkannya, user akan menunjukkan bagian mana yang harus dirobah/dikoreksi/dikurangi sesuai dengan keinginannya.
 10. Memelihara system
Memperbaharui pengetahuan, mengganti pengetahuan yang sudah ketinggalan, meluweskan sistem agar bisa lebih baik lagi dalam menyelesaikan masalah.
- Berikut ini contoh sederhana tahapan pengembangan sistem pakar untuk membuat aplikasi diagnose penyakit THT dengan menggunakan *Development Tool Visual Basic*. Model pelacakan (*inference engine*) menggunakan *Forward Chaining*.

1. Membuat Tabel Gejala Penyakit THT

No	Gejala	Kanker			
		Leher	Nafasoring	Tonsil	Laring
1	Nyeri_Bicara_Menelan	*			*
2	Benjolan di Leher	*		*	
3	Pendarahan Hidung	*	*		
4	Hidung Tersumbat		*		
5	Nyeri Tenggorokan			*	*
6	Batuk				*
7	Suara Serak				*
8				

Gambar 4. Tabel gejala penyakit THT

2. Tabel Fakta Diagnosa Penyakit THT (gambar 5)

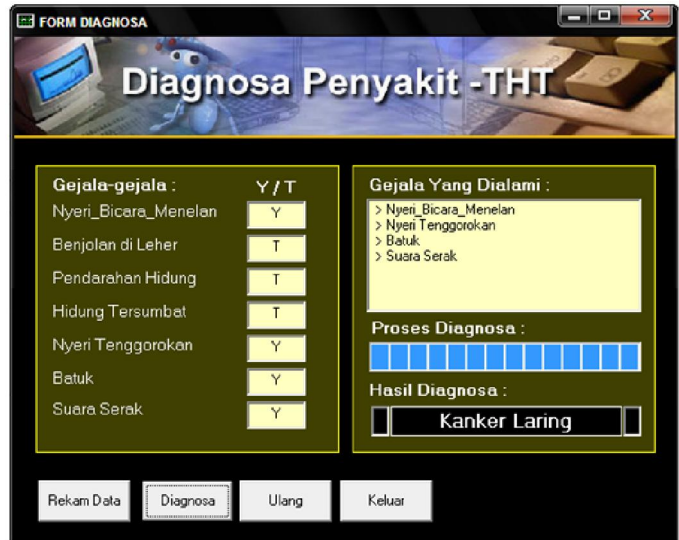
Initial	Fact
A	Nyeri_Bicara_Menelan
B	Benjolan di Leher
C	Pendarahan Hidung
D	Hidung Tersumbat
E	Nyeri Tenggorokan
F	Batuk
G	Suara Serak
H	Kanker Leher
I	Kanker Nafasoring
J	Kanker Tonsil
K	Kanker Laring
.....

Gambar 5. Tabel fakta penyakit THT

3. Penyusunan Rule (Aturan-aturan) (gambar 6)

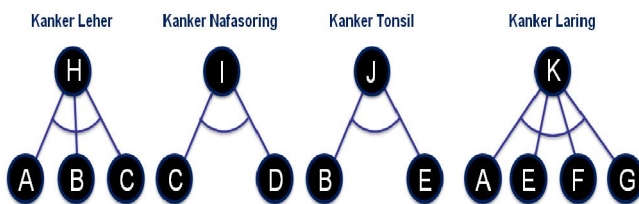
Rule 1	: IF	Nyeri_Bicara_Menelan <i>is True</i>
	AND	Benjolan di Leher <i>is True</i>
	AND	Pendarahan Hidung <i>is True</i>
	THEN	Kanker Leher
Rule 2	: IF	Hidung Tersumbat <i>is True</i>
	AND	Pendarahan Hidung <i>is True</i>
	THEN	Kanker Nafasoring
Rule 3	: IF	Nyeri Tenggorokan <i>is True</i>
	AND	Benjolan di Leher <i>is True</i>
	THEN	Kanker Tonsil
Rule 4	: AND	Nyeri_Bicara_Menelan <i>is True</i>
	AND	Batuk <i>is True</i>
	AND	Suara Serak <i>is True</i>
	AND	Nyeri Tenggorokan <i>is True</i>
	THEN	Kanker Laring

Gambar 6. Rule yang digunakan untuk mendiagnosa



Gambar 8. Tampilan interface aplikasi sistem pakar

4. Membuat Alur Inferensi Diagnosa Penyakit THT (gambar 7)



Gambar 7. Alur inferensi dalam bentuk struktur pohon Keterangan :

A : Nyeri_Bicara_Menelan
 B : Benjolan di Leher
 C : Pendarahan Hidung
 D : Hidung Tersumbat
 E : Nyeri Tenggorokan
 F : Batuk
 G : Suara Serak

5. Merancang Interface dengan Development Tools Visual Basic (gambar 8)

K. DAFTAR PUSTAKA

Arhami, Muhammad . 2005. *Konsep Dasar Sistem Pakar*. Yogyakarta : Penerbit ANDI.
 Hermawan, Arief. 2006. *Jaringan Saraf Tiruan (Teori dan Aplikasinya)*. Yogyakarta: Penerbit ANDI.
 Jong Jek Siang, Drs, M.Sc. 2005. *Jaringan Saraf Tiruan & Pemrogramannya Menggunakan Matlab*. Yogyakarta: Penebit ANDI.
 Kusumadewi, Sri. 2003. *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.
 Kristanto, Andri. 2004. *Kecerdasan Buatan*. Yogyakarta: Graha Ilmu.
 Kuswadi, Son. 2007. *Kendali Cerdas (Teori dan Aplikasi Praktisnya)*. Yogyakarta: Penerbit ANDI.
 Puspitaningrum, Diyah. 2006. *Pengantar Jaringan Saraf Tiruan*. Yogyakarta: Penerbit ANDI.