

Sistem Pakar untuk Pertolongan Pertama pada Penyakit Umum menggunakan Metode *Forward Chaining*

Joshua Ricky Hartono¹, Gregorius Satia Budhi², Lily Puspa Dewi³

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121-131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-mail: swissarmys2007@yahoo.co.id¹, greg@petra.ac.id², lily@petra.ac.id³

ABSTRAK

Aplikasi sistem pakar saat ini sudah banyak dimanfaatkan oleh banyak orang, serta perkembangan aplikasi ini cukup cepat dan baik. Sehingga mudah digunakan dan banyak manfaat yang didapat dengan adanya aplikasi ini. Dengan adanya aplikasi ini, orang awam dapat mempelajari ilmu seorang pakar agar dapat diterapkan kedalam kehidupan sehari-hari.

Dalam penelitian ini diimplemmentasikan sistem pakar untuk mendiagnosis penyakit sebagai pertolongan pertama sebelum konsultasi kepada dokter. Dalam pembuatan aplikasi ini, programmer melakukan konsultasi kepada pakar (dokter) untuk membuat *knowledge based* (basis pengetahuan).

Berdasarkan hasil kuisioner didapat bahwa program aplikasi sistem pakar sebagai pertolongan pertama untuk penyakit umum ini dapat berjalan dengan baik dan mudah dalam penggunaannya.

Kata Kunci: *Forward Chaining*, Sistem Pakar, Diagnosa Penyakit Anak

ABSTRACT

Application of expert systems are now widely used by many people, as well as application development is quite fast and good. So easy to use and many of the benefits with this application. With this application, common people can learn the science of an expert to be applied into daily life.

In this study expert system is applied to diagnose the disease as first aid before consulting a doctor. In making this application, programmers consult the experts (doctors) to create a knowledge-based.

Based on the questionnaire results obtained that the expert system application program as a first aid for common ailments can run well and well-presented in user interface.

Keywords: *Forward Chaining*, *Expert System*, *the general Disease Diagnosis*

1. PENDAHULUAN

Semakin berkembangnya zaman, sekarang ini faktor penyakit tidak hanya dipengaruhi oleh faktor lingkungan saja, namun juga faktor lain, seperti gaya hidup. Gaya hidup yang tidak baik dapat menyebabkan manusia menjadi rawan untuk terkena suatu penyakit. Apabila kita mengalami suatu penyakit paling tidak kita harus tahu bagaimana cara penanganannya sebelum kita

konsultasi ke dokter. Bahkan, korban dari rokok bukan hanya diderita oleh orang tersebut, namun orang disekitarnya juga dapat terkena penyakit akibat dekat dengan orang yang merokok.

Dengan penerapan teknologi informasi, masyarakat dapat mengetahui penanganan pertama terlebih dulu berdasarkan gejala apa yang dirasakan, sehingga masyarakat dapat mengetahui jenis penyakit yang diderita dan obat apa saja yang dapat digunakan untuk penanganan penyakit tersebut sebelum konsultasi ke dokter.

Dari permasalahan tersebut, dirancang suatu program aplikasi sistem pakar yang mampu membantu masyarakat dalam mempelajari beberapa gejala penyakit pertolongan pertama, penanganan yang harus dilakukan dan penggunaan obat apa saja yang dapat digunakan dalam penyembuhan suatu penyakit. Metode yang akan digunakan dalam pembuatan aplikasi ini adalah metode *Forward Chaining*, dimana metode tersebut digunakan untuk menguji faktor – faktor sehingga mendapatkan suatu kesimpulan.[9]

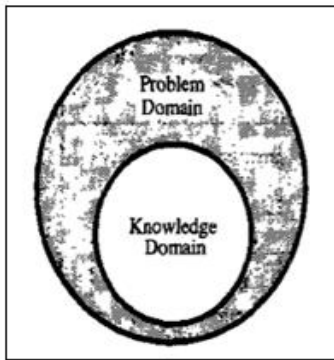
2. DASAR TEORI

2.1 Sistem Pakar

Definisi sistem pakar, sebagai berikut :

1. Menurut Durkin : Sistem pakar adalah suatu program computer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan seorang pakar.[8]
2. Menurut Ignizio : Sistem pakar adalah suatu model dan prosedur yang berkaitan, dalam suatu domain tertentu, yang mana tingkat keahliannya dapat dibandingkan dengan keahlian seorang pakar.[8]
3. Menurut Giarratano dan Riley : Sistem pakar adalah suatu sistem computer yang bias menyamai atau meniru kemampuan seorang pakar.[8]
4. Menurut Turban : Sistem pakar adalah program komputer yang menirukan penalaran seorang pakar dengan keahlian pada suatu wilayah keahlian tertentu.[7]

Pada Gambar 1 dijelaskan mengenai hubungan antara *problem* dan *knowledge domain*. Lihat pada *knowledge domain* yang seutuhnya berada di dalam area *problem domain*. Area di luar *knowledge domain* mengartikan ketiadaan *knowledge* mengenai masalah yang ada di dalam *problem domain*. [3]



Gambar 1. Hubungan Antara *Problem Domain* dan *Knowledge Domain*

Beberapa keuntungan menggunakan sistem pakar sebagai berikut [6] :

1. Membuat seorang yang awam dapat memahami ilmu layaknya seorang pakar.
2. Meningkatkan produktivitas akibat meningkatnya kualitas pekerjaan karena dapat meningkatkan efisiensi kerja.
3. Menghemat waktu kerja dan memudahkan pekerjaan.
4. Merupakan aplikasi yang terpercaya dari sebuah keahlian, sehingga bagi pemakai sistem pakar akan seolah-olah berkonsultasi langsung dengan seorang pakar, meskipun pakar tersebut tidak ada.
5. Memperluas jangkauan dari keahlian seorang pakar, dimana sebuah sistem pakar yang disahkan akan sama saja artinya dengan seorang pakar dalam jumlah besar (dapat diperbanyak dengan kemampuan yang persis sama dengan keahlian seorang pakar), dapat diperoleh dan dipakai dimana saja.[4]

Sistem pakar yang baik harus memenuhi ciri – ciri sebagai berikut [6] :

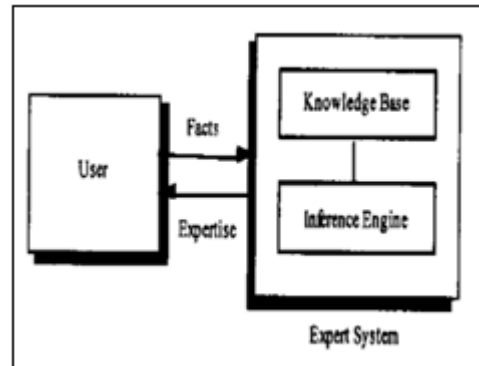
1. Memiliki informasi yang handal
2. Mudah menambahkan *Knowledge based*
3. Dapat digunakan dalam berbagai jenis komputer
4. Memiliki kemampuan untuk beradaptasi [5]

2.2 *Knowledge Based System*

Knowledge base system adalah sebuah sistem komputer yang mencoba untuk mereplikasi fungsi yang dilakukan oleh manusia. *knowledge base system* digunakan untuk menangkap, meneliti, dan mendistribusikan akses untuk pengambilan keputusan. Pada dasarnya, KBS membuat pengguna dapat melakukan konsultasi dengan sistem komputer seakan-akan berkonsultasi dengan konsultan untuk mendiagnosa masalah atau penentuan pengambilan keputusan untuk suatu masalah.[3]

Gambar 2 menjelaskan mengenai konsep dasar dari *knowledge-based* dari sistem pakar. *User* memberikan inputan berupa fakta atau informasi lainnya kepada sistem pakar. Kemudian *user* akan menerima respon dari sistem pakar berupa saran oleh pakar. Di dalam sistem pakar terdapat dua komponen, yaitu *knowledge-base*, yang di dalamnya terdapat berbagai macam *knowledge* dan *inference engine*, yang dapat menghasilkan kesimpulan. Kesimpulan yang telah dibuat oleh sistem pakar tersebut adalah

respon terhadap pertanyaan pengguna mengenai suatu keahlian.[3]



Gambar 2. Fungsi Dasar Sistem Pakar

2.3 *Certainty Factor*

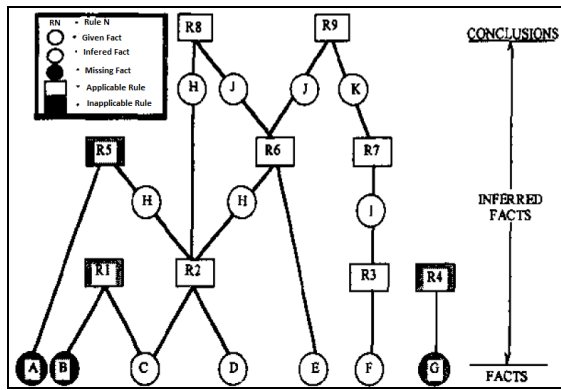
Expert sering membuat perkiraan saat memecahkan masalah. Informasi yang didapatkan sering hanya merupakan perkiraan dan tidak lengkap, sehingga dibutuhkan suatu cara untuk menyatakan suatu informasi yang tidak pasti. Misalnya ketika muncul pertanyaan “Apakah pasien merasa pusing?” Jawaban dari pasien bisa ya, tidak, mungkin, atau tidak tahu. Oleh karena itu, untuk keadaan yang tidak pasti digunakan *Certainty Factor (CF)* yang menyatakan tingkat keyakinan pakar dalam suatu pernyataan. Rentang nilai CF -1 (yakini negatif) sampai 1 (yakini positif).[1]

2.4 Metode *Forward Chaining*

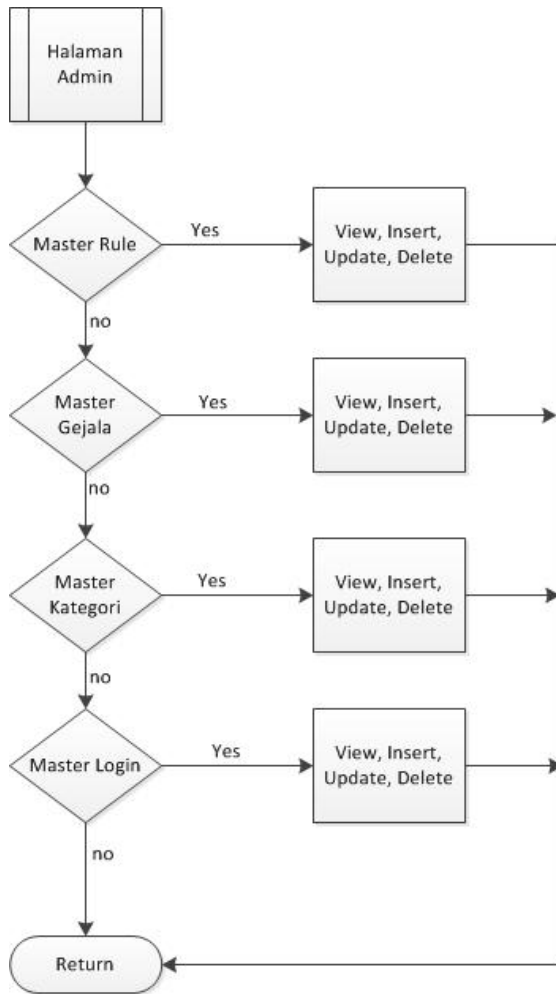
Forward chaining adalah salah satu metode dari sistem pakar yang mencari atau menelusuri solusi melalui masalah. Dengan kata lain metode ini melakukan pertimbangan dari fakta-fakta yang kemudian berujung pada sebuah kesimpulan yang berdasarkan pada fakta-fakta.[3]

Forward chaining disebut juga *bottom-up reasoning* atau pertimbangan dari bawah ke atas, karena metode ini mempertimbangkan dari bukti-bukti pada level bawah, fakta-fakta, menuju ke kesimpulan pada level atas yang berdasarkan pada fakta-fakta.[3]

Gambar 3 menjelaskan konsep dasar dari *forward chaining* dalam *rule-based system*. *Rule* digerakkan oleh fakta-fakta yang memenuhi sebelumnya atau disebut dengan *left-hand-sides (LHS)*. Sebagai contoh, *rule R₁* harus memenuhi fakta B dan C agar bisa diaktifkan. Namun, hanya fakta C yang tersedia, sehingga *R₁* tidak bisa diaktifkan. *Rule R₂* diaktifkan oleh fakta C dan D, di mana keduanya tersedia, sehingga *R₂* menghasilkan fakta menengah, yaitu fakta H. *Rule* lain yang terpenuhi adalah *R₃*, *R₆*, *R₇*, *R₈*, dan *R₉*. Eksekusi dari *rule R₈* dan *R₉* menghasilkan kesimpulan dari proses *forward-chaining*. Kesimpulan-kesimpulan itu bisa berupa fakta, *output* dan sebagainya.[3]



Gambar 3. Forward Chaining



Gambar 4. Flowchart Halaman Admin

3. ANALISIS SISTEM

3.1 Analisis Permasalahan

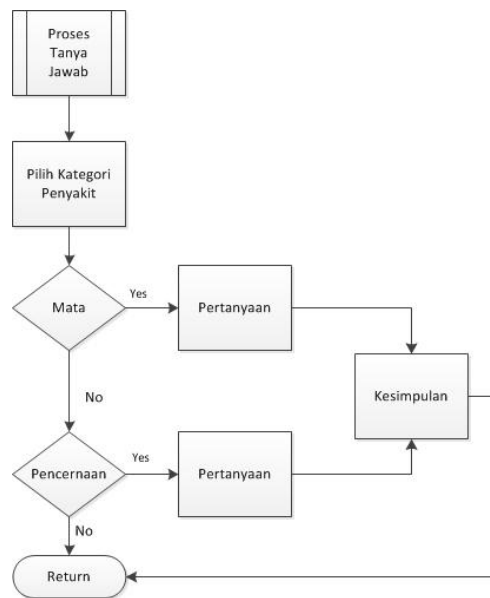
Dalam pembuatan suatu program, umumnya akan terjadi masalah – masalah yang harus diselesaikan oleh seorang programmer. Oleh karena itu diperlukan suatu metode untuk menyelesaikan persoalan tersebut dengan cara melakukan suatu analisa. Analisa yang dilakukan berupa analisa untuk mencari sumber

permasalahan dari program tersebut, serta bagaimana mengaplikasikan dalam suatu sistem pakar.

3.2 Flowchart

Pada Gambar 4 merupakan flowchart program pada admin, didalam halaman admin, admin memiliki hak untuk melihat, memasukan, menghapus dan mengganti data master. User tidak memiliki hak untuk melakukan hal ini. Flowchart halaman admin dapat dilihat pada Gambar 4.

Pada Gambar 5 merupakan halaman user untuk melakukan proses diagnosa, awalnya user memilih kategori penyakit yang ingin dicari, setelah itu program memberikan pertanyaan menurut kategori yang dipilih oleh user. Hal ini dibuat, agar pertanyaan yang diberikan kepada user langsung menuju penyakit yang dicari. Karena apabila tidak diberi kategori, maka program akan memberikan semua pertanyaan yang telah diinputkan admin. Flowchart program tanya jawab dapat dilihat pada Gambar 5.



Gambar 5. Flowchart Tanya Jawab

4. IMPLEMENTASI SISTEM

4.1 Dependency Diagram

Pada Gambar 6 merupakan dependency diagram dari penyakit yang digunakan dalam pembuatan rule pada program sistem pakar.

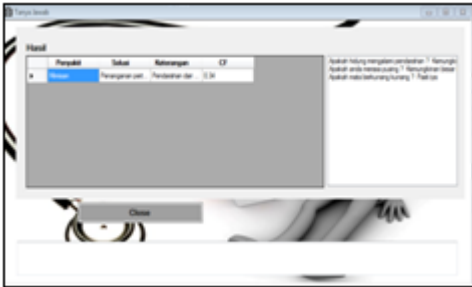


Gambar 6. Dependency Diagram

4.2 Form Hasil Tanya Jawab

Form hasil tanya jawab, merupakan form hasil diagnosa program, yang outputnya berupa penyakit yang dialami oleh user. Dalam form hasil tanya jawab terdapat hasil diagnosa, solusi dan keterangan tentang penyakit yang ditampilkan. Serta, dalam form

tanya jawab terdapat histori jawaban dari *user*. Contoh *form* hasil tanya jawab dapat dilihat pada Gambar 7.



Gambar 7. Form Hasil Tanya Jawab

Perhitungan manual :

Diketahui :

CF mimisan : 0,68

CF Jawaban 1 (Kemungkinan Besar) : 0,5

CF Jawaban 2 (Kemungkinan Besar) : 0,5

CF Jawaban 3 (Kemungkinan Besar) : 1

Jadi perhitungan CF :

$$CF = 0,68 \times (\min(0,5,0,5,1))$$

$$CF = 0,68 \times (0,5)$$

$$CF = 0,34$$

Jadi, dari hal tersebut pengujian program dengan perhitungan CF sudah benar dan terbukti sama dengan perhitungan manualnya. Contoh *form* ensiklopedia dapat dilihat pada Gambar 7.

5. KESIMPULAN DAN SARAN

Dari hasil pembuatan program aplikasi sistem pakar dengan visual basic 2010 dapat diambil beberapa kesimpulan, antara lain :

- Proses tanya jawab lebih singkat karena ada proses pemilihan kategori penyakit.
- Proses input ke *form master* mudah dilakukan.
- Tampilan program sudah cukup *user friendly*.
- Proses perhitungan CF juga dapat dihitung dengan baik dan benar.

Saran yang diberikan dalam pengembangan aplikasi ini adalah :

- Kembangkan *knowledge based* agar program dapat bekerja lebih baik.
- Pada halaman hasil diagnosa juga baik, karena terdapat nama penyakit, solusi, dan keterangan atas penyakit tersebut.
- *Knowledge Based* masih terbatas.

6. DAFTAR PUSTAKA

- [1] Aziz, F. M. 1994. Belajar Sendiri Pemrograman Sistem Pakar. Jakarta: PT Elex Media Komputindo.
- [2] Durkin, J. 1994. *Expert Systems: Design and Development*. Amerika : Macmillan.
- [3] Giarratano, J. C. dan Riley, G.D. 2005. *Expert Systems Principles and Programming Fourth Edition*. New York: Course Technology.
- [4] Halim, N. R. M. 2011, September. Implementasi Sistem Pakar untuk Mendeteksi Kerusakan Peralatan Elektronik dengan Bahasa Pemrograman Visual Basic 6.0, 1(3), , 282-296. *Jurnal Teknologi dan Informasi (Teknomatik)*, 1(3). Retrieved Januari 21, 2015. <http://news.palcomtech.com/wp-content/uploads/2012/01/NASRUL->
- [5] Budiman, I. 2010. Pembuatan Aplikasi Tes Kepribadian Berbasis Sistem Pakar Menggunakan Visual Studio .Net 2008. Unpublished undergraduate thesis, Universitas Gunadarma, Depok. http://www.gunadarma.ac.id/library/articles/graduate/computer-science/2009/Artikel_10104875.pdf.
- [6] Kusumadewi, S. 2003. *Artificial Intelligence Teknik dan Aplikasinya*. Yogyakarta : Graha Ilmu.
- [7] Kusriani. 2010. *Sistem Pakar Teori dan Aplikasi*. Yogyakarta: Andi Publisher.
- [8] Merlina, N. dan Hidayat, R. 2012. *Perancangan Sistem Pakar*. Bogor: Ghalia Indonesia.
- [9] Sutojo, T., Mulyanto, E., dan Suhartono, V. 2011. *Kecerdasan Buatan*. Yogyakarta: Andi Offset.