

Perancangan dan Pembuatan Aplikasi Sistem Pakar Untuk Penentuan Produk dan Jenis Perawatan Tubuh di Pusat Perawatan “Epiderma”

Leo Willyanto Santoso, Gregorius Satia Budhi, Lydiawaty Mappatombong

Fakultas Teknologi Industri – Jurusan Teknik Informatika
Universitas Kristen Petra
Jl. Siwalankerto 121-131 Surabaya 60236
leow@petra.ac.id

ABSTRACT

Every human being especially women, want to have healthy skin. That is way it is very important to take care of our skin. Consider of many aspects which can cause problems of skin, therefore we need to choose the right product and treatment.

This application is based on expert system; it is the expert's way of thinking, analyzing and making conclusion or decision. At this application program, it will analyze types of skin and problem which are faced to determine the right product and treatment. Treatment which are provided consist of face treatment, body treatment, eye treatment, hair treatment, hand and foot treatment. This expert system development is using forward chaining inference method, which is a data-driven inference process for finding a conclusion.

Based on questionnaires, we can say that this expert system program is suitable to use for staff units at “Epiderma” treatment centre with percentage score 80%, we can also say that this program is accurate enough with percentage score 76%. Rate of providing information is fast enough and this expert system program is also easy to use with percentage scores 72%. The interface design needs to be more attractive with percentage score 68%.

Keyword: *Forward Chaining*, Expert System, Determine Product and Treatment

1. PENDAHULUAN

Setiap individu terutama wanita tentunya ingin memiliki penampilan yang menarik. Penampilan yang menarik tidak hanya dilihat dari pakaian dan pernak-pernik yang dipakai, melainkan ditunjang dengan adanya kulit yang sehat. Oleh karena itu, penting untuk senantiasa menjaga dan merawat kulit agar tampak bersinar.

Selama perjalanan hidup, kulit mengalami berbagai perubahan oleh karena pengaruh lingkungan luar maupun karena perubahan dari dalam tubuh sendiri. Pengaruh lingkungan luar misalnya karena pengaruh sinar matahari, iklim/cuaca, penggunaan AC, polusi, trauma pada kulit, dan penggunaan produk perawatan/tata rias yang tidak sesuai. Sedangkan pengaruh dari dalam antara lain karena pengaruh perubahan hormonal pada saat pubertas, kehamilan, menstruasi, pil KB, pengaruh zat gizi (nutrisi), dan berbagai macam penyakit. Beberapa faktor yang menyebabkan perubahan pada kulit tersebut ada yang dapat dicegah dan ada yang tidak.

Mengingat banyaknya faktor yang dapat mengakibatkan kulit menjadi bermasalah, maka untuk merawat dan mengatasi kulit bermasalah diperlukan suatu produk dan jenis perawatan yang tepat.

Pusat perawatan “Epiderma” yang berlokasi di Jalan Gunung Lokon No.10 A, Makassar hadir sebagai salah satu klinik perawatan yang membantu setiap individu untuk mendapatkan kulit sehat seperti yang mereka idamkan. Produk yang digunakan oleh pusat perawatan “Epiderma” merupakan hasil formulasi dokter dengan merk “Epiderma”. Selain menyediakan produk, pusat perawatan “Epiderma” juga menyediakan bermacam-macam pelayanan jasa dibidang perawatan tubuh antara lain *face treatment*, *body treatment*, *eye treatment*, *hand and foot care*, dan *hair treatment* yang dibawah oleh *Epiderma Paramedical Aesthetic Center* (EPAC).

Dengan adanya jenis kulit, kondisi kulit dan permasalahan kulit yang bermacam-macam, maka diharapkan adanya suatu sistem dimana dapat membantu unit pegawai “Epiderma” dalam menentukan kombinasi produk dan jenis perawatan yang tepat dan dengan waktu yang lebih cepat.

Masalah yang dihadapi dalam penelitian ini adalah:

- Seringkali dokter di pusat perawatan “Epiderma” melakukan kesalahan dalam mendiagnosa yang disebabkan adanya faktor manusiawi, seperti kurang konsentrasi atau kurang sehat.
- Dokter membutuhkan waktu yang lebih banyak dalam mendiagnosa pasien.
- Keterlambatan atau ketidakhadiran dokter yang menghambat proses kerja perusahaan

Tujuan dari penelitian ini adalah untuk membuat sistem pakar yang dapat membantu unit pegawai di pusat perawatan “Epiderma” untuk mendiagnosa jenis dan permasalahan kulit yang dihadapi oleh pasien “Epiderma” dengan lebih akurat dan cepat sehingga dapat memberikan produk dan jenis perawatan yang tepat kepada pasien.

Ruang lingkup dalam penelitian ini adalah:

- Perawatan tubuh yang disediakan oleh pusat perawatan “Epiderma” meliputi perawatan wajah, mata, badan, kaki, tangan, dan rambut.
- Masalah-masalah kulit yang sering dijumpai dibidang perawatan kecantikan pada wanita Asia, yaitu jerawat (*acne*), flek, dehidrasi dan menua (*aging*).
- Produk dan jenis perawatan yang disediakan oleh pusat perawatan “Epiderma”.
- *Software* yang digunakan adalah *Borland Delphi 6* untuk membuat *Inference Engine* dan *Microsoft Access* untuk menyimpan *database rules* serta *database* produk dan jenis perawatan.
- Metode yang digunakan yaitu *forward chaining*.
- Menggunakan *Certainty Factor* untuk menghitung tingkat keyakinan.
- User yaitu unit pegawai di pusat perawatan “Epiderma”.

2. SISTEM PAKAR

Sistem pakar adalah suatu program komputer yang didesain untuk memodelkan/mengemulasi kemampuan seorang pakar dalam memecahkan suatu masalah. [DUR 1994]

Sistem pakar dimaksudkan untuk menyelesaikan masalah-masalah nyata dimana biasanya akan membutuhkan seorang pakar (seperti dokter, teknisi, dan lainnya). Sistem pakar menghasilkan hasil yang konsisten dan mempunyai kecepatan yang konstan daripada kepakaran seseorang. Ini disebabkan oleh beberapa hal yang bersifat manusiawi dimana dapat mempengaruhi pengambilan keputusan seorang pakar seperti lingkungan kerja, kondisi kesehatan dan masalah pribadi.

Adapun langkah-langkah dalam merancang suatu sistem pakar yaitu sebagai berikut :

- Menentukan batasan-batasan dari suatu sistem pakar yang akan dirancang.
- Memilih jenis keputusan apa yang akan diambil.
- Meng-*extract* pengetahuan dari pakar, caranya yaitu dengan *dependency diagram* dan *graphical representation*.
- Merepresentasikan pengetahuan dalam sistem pakar (membuat *rules*), salah satu teknik yang dapat digunakan yaitu merepresentasikan ke dalam bentuk *IF-THEN rules*.
- Membuat *inference engine* dengan menggunakan metode yang sesuai.
- Merancang *user interface*.

Ada dua kapabilitas penting dari seorang pakar (*expert*), yang dicoba untuk dimodelkan pada sistem pakar yaitu pengetahuan (*knowledge*) dan konsep berpikir (*reasoning*) dari sang pakar. Untuk menghasilkan kedua hal tersebut, sebuah sistem pakar harus memiliki dua modul penting yaitu: *knowledge base* dan *inference engine* di luar modul-modul lain yang digunakan untuk berinteraksi dengan *user* (*user interface*).

3. FORWARD CHAINING

Forward chaining adalah strategi *inference* yang bermula dari sejumlah fakta yang diketahui, dengan menggunakan *rules* yang premisnya cocok dengan fakta yang diketahui tersebut untuk memperoleh fakta baru dan melanjutkan proses hingga *goal* dicapai atau hingga sudah tidak ada *rules* lagi yang premisnya cocok dengan fakta yang diketahui maupun fakta yang diperoleh. [DUR 1994]

Pada aplikasi *forward chaining* sederhana, *inference engine* menyalakan atau memilih *rule-rule* dimana bagian premisnya cocok dengan informasi yang ada pada *working memory*. Sistem pertama-tama memperoleh informasi masalah dari *user* dan menyimpannya dalam *working memory*. *Inference engine* lalu akan mencari *rules* pada beberapa urutan yang telah ditentukan sebelumnya, dimana premis-premisnya cocok dengan yang terdapat dalam *working memory*. Jika *rule* ditemukan, maka kesimpulan dari *rule* akan diinputkan ke dalam *working memory* (*rule fired*) lalu *cycles* dan cek lagi *rules* untuk mencari kecocokan baru. Pada *cycle* berikutnya, *rules* yang sebelumnya telah *fired* diabaikan. Proses ini akan terus berlanjut

hingga tidak ditemukan lagi adanya kecocokan. Dalam hal ini, *working memory* berisi informasi yang didapat dari *user* dan kesimpulan yang didapat dari sistem.

Karena metode *forward chaining* berangkat dari premis menuju kepada kesimpulan akhir, maka seringkali pula disebut *data driven* (yaitu, pencarian dikendalikan oleh data yang diberikan). Metode ini lebih baik digunakan apabila memiliki sedikit premis dan banyak kesimpulan.

Pada metode *forward chaining*, ada 2 cara yang dapat dilakukan untuk melakukan pencarian, yaitu [IGN 1991]:

- Pertama, dengan memasukkan semua data yang tersedia ke dalam sistem pakar pada satu kesempatan dalam sesi konsultasi. Cara ini banyak berguna pada sistem pakar yang termasuk dalam proses terautomatisasi dan menerima data langsung dari komputer yang menyimpan *database*, atau dari satu set sensor.
- Kedua, dengan hanya memberikan elemen spesifik dari data yang diperoleh selama sesi konsultasi kepada sistem pakar. Cara ini mengurangi jumlah data yang diminta, sehingga data yang diminta hanyalah data-data yang benar-benar dibutuhkan oleh sistem pakar dalam mengambil kesimpulan.

Algoritma untuk metode *forward chaining* adalah sebagai berikut [IGN 1991] :

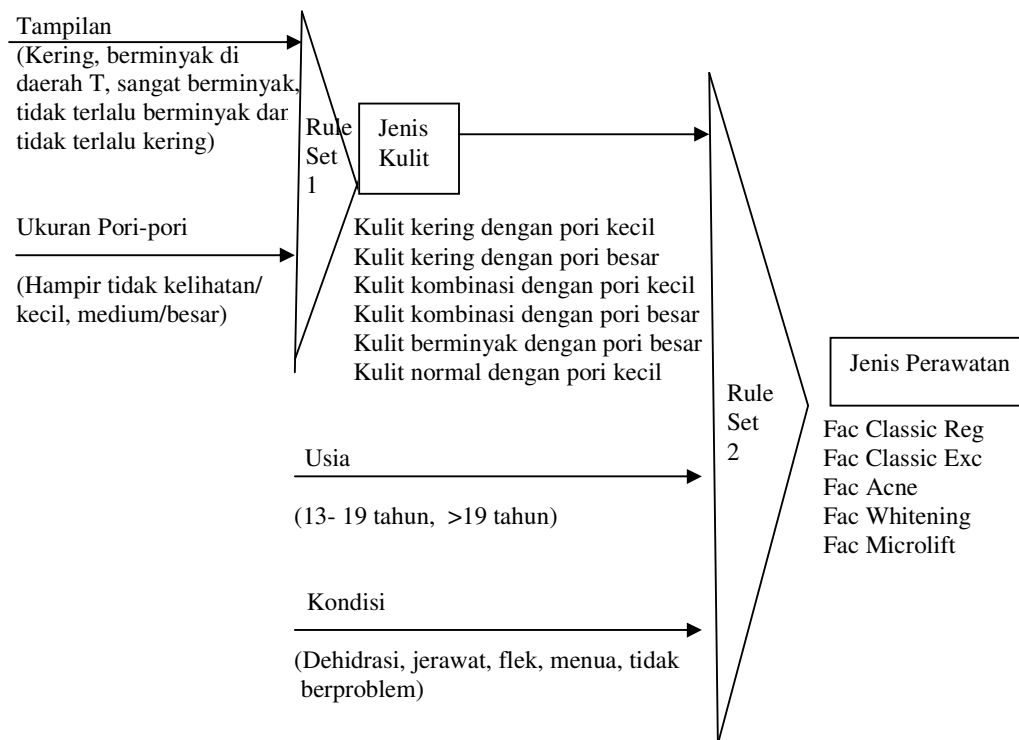
1. Inisialisasi. Dapat dibuat tiga tabel kosong, yaitu tabel *Working Memory*, tabel *Attribute Queue*, dan tabel *Rule/Premis Status*. Tabel *Working Memory* berguna untuk menyimpan setiap *input*, yaitu semua fakta yang disimpulkan selama proses konsultasi. Tabel *Attribute Queue* berguna untuk menyimpan semua atribut dari *value* yang sedang diperiksa. Atribut pada awal tabel adalah atribut yang sedang menjalani proses pemeriksaan. Tabel *Rule/Premis Status* menyimpan status dari *rule-rule* yang ada yaitu *Active*, *Marked*, *Unmarked*, *Discarded*, *Triggered*, *Fired*. Status setiap premis pada saat inisialisasi adalah bebas (*Free*). Dan status setiap *rule* pada saat inisialisasi adalah tidak bertanda (*Unmarked*) dan aktif (*Active*). Notasi yang digunakan untuk merepresentasikan keadaan suatu *rule* dan premis adalah sebagai berikut :
 - A = *Active Rule*
 - D = *Discarded Rule*
 - U = *Unmarked Rule*
 - M = *Marked Rule*
 - TD = *Triggered Rule*
 - FD = *Fired Rule*
 - FR = *Free Clause*
 - FA = *False Clause*
 - TU = *True Clause*
2. Memulai proses pengambilan keputusan. Sebuah *value* dari sebuah atribut premis diambil. Di mana atribut tersebut tidak boleh ada pada klausa kesimpulan. Atribut ini disimpan pada bagian teratas tabel *Attribute Queue*. Juga simpan atribut ini beserta *valuennya* pada bagian terbawah tabel *Working Memory*.
3. Penelitian satu persatu *rule* yang ada untuk memeriksa ada tidaknya kesamaan. Periksa tabel *Rule/Premis Status*, jika tidak ada *rule* yang statusnya '*Active*', pencarian dihentikan. Bila ada, dilakukan penelitian bagian klausa premis *rule* yang statusnya '*Active*' untuk mencocokkan klausa premis yang sesuai dengan *value* dari atribut pada bagian teratas tabel *Attribute Queue*. Simpan perubahan status klausa premis dari sekumpulan *rule* yang statusnya *Active*. Pada tabel *Rule/Premis Status* diberikan tanda FA (*False Clause*) pada status klausa premis yang bernilai salah dan tanda TU (*True Clause*) pada status klausa premis yang bernilai benar. Periksa status *Rule* pada tabel *Rule/Premis Status* :
 - a. Bila ada premis dari sebuah *rule* yang bernilai salah, maka diberi tanda D (*Discarded*) pada *rule* tersebut untuk menunjukkan bahwa *rule* tersebut bernilai salah dan tidak dipakai lagi. Langkah tersebut dilakukan pada setiap *rule* yang memiliki premis yang bernilai salah.
 - b. Bila ada sebuah *rule* yang semua premisnya dari bernilai benar, diberi tanda TD (*Triggered*) pada *rule status*. Kemudian dilanjutkan ke langkah 3c.
 - c. Bila tidak ada *rule* yang statusnya TD (*Triggered*), dilanjutkan ke langkah ke 5, bila ada satu atau lebih *rule* yang statusnya TD (*Triggered*), dilanjutkan ke langkah ke 4.
4. *Rule firing*, atau menyatakan *rule* tersebut benar dan mengambil klausa kesimpulan *rule* tersebut sebagai kesimpulan akhir. Coretlah atribut pada bagian teratas tabel *Attribute Queue*, kemudian status *rule* tersebut diganti dan ditempatkan atribut kesimpulan pada bagian terbawah tabel *Attribute Queue* dan *value*-nya pada tabel *Working Memory*.
5. Status antrian. Bagian teratas tabel *Attribute Queue* dicoret .
6. Menandai *rule*. Telitilah kumpulan *rule active* untuk mencari *rule* yang statusnya U (*Unmarked*) dan A (*Active*). Bila tidak ditemukan, pencarian dihentikan. Bila ada, *rule* pertama yang ditemui ditandai dengan M (*Marked*).

7. *Query*. Pada *rule* yang baru saja diberi tanda M (*Marked*), ditanyakan pada *user* untuk memperoleh *input*. Apabila *user* memberikan jawaban, dilanjutkan ke langkah 8. Sedangkan bila *user* tidak memberikan jawaban atau bila atribut *rule* yang ditanyakan tersebut tidak memerlukan jawaban dari *user*, dilanjutkan langkah ini pada setiap klausa premis pada *rule* yang diberi tanda M (*Marked*) tersebut. Apabila setiap klausa premis pada *rule* yang diberi tanda M (*Marked*) tersebut telah diperiksa, kembali ke langkah 6.
8. Menghilangkan tanda M (*Marked*) pada *rule*. atribut dan nomor *rule* diletakkan pada bagian teratas tabel *Attribute Queue*. Atribut ini diletakkan juga beserta *value*-nya pada bagian terbawah tabel *Working Memory*. pada *rule* yang baru saja diberi tanda M (*Marked*) diberikan tanda U (*Unmarked*), dan kembali ke langkah 3.

4. DESAIN DAN ANALISIS SISTEM

Perancangan dan pembuatan aplikasi sistem pakar untuk menentukan produk dan jenis perawatan di pusat perawatan “Epiderma” ini dibagi menjadi 8 bagian utama, yaitu:

- a. Pengumpulan pengetahuan.
Suatu sistem pakar memerlukan basis pengetahuan yang baik. Pengetahuan harus diperoleh, disusun dan direpresentasikan. Dalam sistem pakar, pengumpulan pengetahuan adalah hal yang terpenting dalam menentukan *knowledge based* aplikasi sistem pakar yang akan dibuat. Juga untuk menentukan batasan-batasan sistem pakar ini. Untuk itu diperlukan sumber bagi pembuatnya. Pengumpulan pengetahuan ini digunakan untuk menentukan ruang lingkup atau batasan-batasan yang diperlukan. Setelah mengumpulkan pengetahuan maka dapat dibuat *knowledge base* serta *rules* dari aplikasi sistem pakar ini.
- b. Pembuatan *dependency diagram* dan *graphical representation* sebagai landasan dalam membuat *knowledge based system*.
Berikut contoh *Dependency diagram* untuk aplikasi sistem pakar.



Gambar 1. Dependency Diagram Untuk Perawatan Wajah

- c. Melakukan konversi dari *dependency diagram* dan *graphical representation* menjadi *IF-THEN rules*, yang berperan penting dalam proses pengambilan keputusan.

Dari *dependency diagram*, digambarkan *Graphical Representation* untuk memperlihatkan hubungan yang lebih jelas antara *rules* dalam menyusun *rules* final. Setelah dibuat *graphical representation*, langkah berikutnya yaitu mengkonversikannya ke dalam *IF-THEN rules*. Pada aplikasi sistem pakar ini, *IF-THEN rules* disusun dengan menggunakan operator *AND*. Sehingga *rule-rule* yang seharusnya menggunakan operator *OR* atau *ELSE* akan dibuat menjadi dua buah *rule* dengan menggunakan operator *AND*.

- d. Perancangan *database* sebagai salah satu komponen penunjang *knowledge based system*.
 Pembuatan tabel-tabel pada *database* aplikasi sistem pakar ini menggunakan program Microsoft Access. *Database* digunakan untuk menyimpan *rule*, premis, dan pertanyaan. Selain itu, *inference engine* juga beberapa tabel kerja sementara yang digunakan dalam proses pencapaian kesimpulan. *Database* digunakan untuk pencapaian kesimpulan, melalui sejumlah *rule* yang diaplikasikan ke dalam *database*, menyimpan pertanyaan dan jawaban yang ada. Keuntungan yang dapat diperoleh dengan menggunakan *database* yaitu setiap *rule* dan pertanyaan, akan lebih dinamis, dimana mudah untuk diubah dan digantikan. Tetapi ada pula kekurangan dari penggunaan *database* yaitu proses pengaksesannya memakan waktu yang lebih lama.
 Tabel-tabel yang digunakan antara lain adalah:
 - a. Tabel Ask berguna untuk menyimpan pertanyaan-pertanyaan beserta jawaban-jawaban yang diperlukan dari tabel Jika. Tabel ini memiliki *field-field* sebagai berikut: Indikator, Pertanyaan, Jawaban dan Keterangan.
 - b. Tabel Jika berguna untuk menyimpan premis-premis daripada *rule* yang ada. Tabel ini memiliki *field-field* sebagai berikut: Set, No_rule, No_premis, Indikator, Value, Status dan CF_premis.
 - c. Tabel Maka berguna untuk menyimpan kesimpulan daripada *rule* yang ada. *Field-field* yang ada pada tabel ini adalah: Set, No_rule, No_Premis, Active_rule, Unmarked_rule, Fired_rule, Indikator, Value, dan CF_rule.
 - d. Tabel WM (*Working Memory*) berguna untuk menyimpan fakta-fakta yang didapat selama satu sesi. Tabel ini terdiri dari *field-field* sebagai berikut: Indikator, Value dan Status.
 - e. Tabel *Atributqueue* berguna untuk mengurut atribut-atribut dimana nilainya telah ditandai atau sedang dicari, menjadi suatu antrian. *Field-field* yang dimiliki tabel ini adalah: No_id, Indikator, Value dan Kesimpulan
 - f. Tabel *Login* berguna untuk menyimpan password dari user yang dapat mengakses form Edit untuk mengedit *rule-rule* yang ada.
- e. Perancangan *inference engine* yang menggunakan metode *forward chaining*, sebagai pusat pemrosesan aplikasi sistem pakar.
Inference engine untuk aplikasi sistem pakar ini menggunakan metode *forward chaining*. Dengan metode *forward chaining*, sistem ini akan mencari hasil kesimpulan dari hasil suatu sesi analisa tanya jawab dengan *user* secara menyeluruh. Maka dengan menggunakan metode *forward chaining* pada *inference engine*, efisiensi kerja *inference engine* akan lebih meningkat.
- f. Perhitungan *confidence factor* (CF) untuk tiap kesimpulan yang dihasilkan sistem.
 Seperti kita ketahui, CF terbagi menjadi 3, yaitu:
 - a. Faktor Keyakinan Premis, merupakan faktor keyakinan yang didapat dari jawaban user. Pada aplikasi sistem pakar penentuan produk dan jenis perawatan di pusat perawatan “Epiderma” ini, hanya dibatasi sebanyak 4 faktor keyakinan premis, yaitu:
 - Tidak tahu, dengan tingkat keyakinan 0.
 - Mungkin, dengan tingkat keyakinan 0,6.
 - Kemungkinan besar, dengan tingkat keyakinan 0,8.
 - Pasti, dengan tingkat keyakinan 1.
 - b. Faktor Keyakinan *Rule*, merupakan faktor keyakinan dari *knowledge engineer* akan *knowledgenya*.
 - c. Faktor Keyakinan Konklusi, merupakan faktor keyakinan yang didapat dari faktor keyakinan premis dikalikan dengan faktor keyakinan *rule*.

$$CCF = \min(CF_{\text{premis}}) \times CF_{\text{rule}} \quad (1)$$
- g. Perancangan fasilitas edit.
 Fasilitas edit ini dimaksudkan untuk memudahkan *user* untuk menambah, mengurangi, atau mengedit pertanyaan, jawabannya, maupun *rules*. Selain itu, dengan adanya fasilitas ini, maka aplikasi sistem pakar yang dibuat menjadi lebih fleksibel dan dinamis mengikuti perkembangan yang ada. Namun tentu saja yang dapat menggunakan fasilitas ini, haruslah *user* yang benar-benar mengerti, agar tidak terjadi hal-hal yang dapat membuat aplikasi sistem pakar ini tidak dapat berjalan sebagai mana mestinya, ataupun yang membuat hasil yang diperoleh tidak akurat.

h. Perancangan *user interface*.

User interface yang baik adalah *interface* yang *user friendly*. Yang menjadi patokan suatu *interface* dikatakan *user friendly* yaitu *interface* tersebut dapat memberikan kemudahan dan kenyamanan bagi *user* dalam mengoperasikannya. *User interface* yang dirancang harus jelas dan lazim, baik format teks, ukuran *font*, maupun komponen-komponen yang digunakan, agar tidak membingungkan *user*.

4. UJI COBA SISTEM

Berdasarkan perancangan sistem yang telah dirancang pada bab sebelumnya, berikut ini merupakan implementasi dari sistem yang telah dirancang tersebut. Implementasi program dibuat dengan menggunakan bahasa pemrograman Borland Delphi 6.0 dan menggunakan *database* Microsoft Access

Dalam aplikasi sistem pakar ini disediakan fasilitas untuk mengedit, menambah atau menghapus pertanyaan maupun *rule* yang ada pada aplikasi ini. Fasilitas ini disediakan untuk keperluan apabila sewaktu-waktu ada pertanyaan dan juga jawaban yang mungkin kurang dapat dimengerti, untuk itu pertanyaan dan jawaban yang ada dapat diubah sewaktu-waktu, juga *rule-rule* yang ada yang dirasa perlu untuk *update*. Tentu saja orang yang dapat merubahnya adalah orang yang benar-benar mengerti sistem pakar dan telah mengerti akan cara penggunaan fasilitas ini. Untuk itu diperlukan *form login* terlebih dahulu, agar hanya *user* tertentu yang dapat menggunakan fasilitas edit ini. Tampilan untuk form input rules dan input pertanyaan ini dapat dilihat pada Gambar 2 dan Gambar 3, sedangkan tampilan untuk edit rules dan edit pertanyaan dapat dilihat pada Gambar 4, dan Gambar 5.

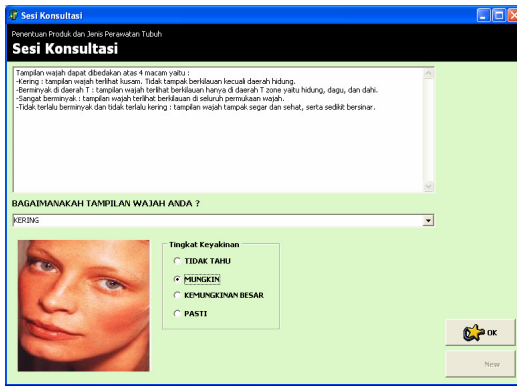
Gambar 2. Input Rules

Gambar 3. Input Pertanyaan

Gambar 4. Edit Rules

Gambar 5. Edit Pertanyaan

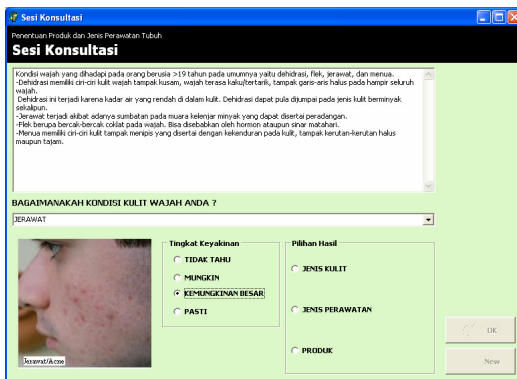
Tampilan proses konsultasi aplikasi sistem pakar ini dapat dilihat pada Gambar 6, Gambar 7 dan Gambar 8.



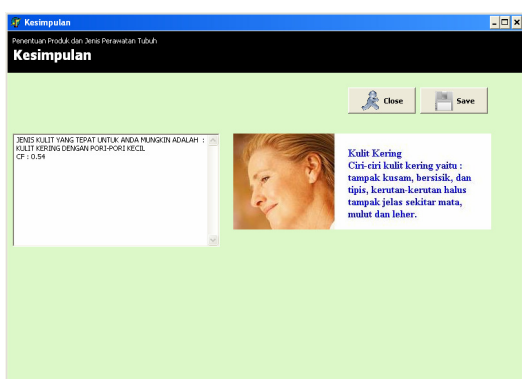
Gambar 6. Konsultasi Perawatan Wajah I



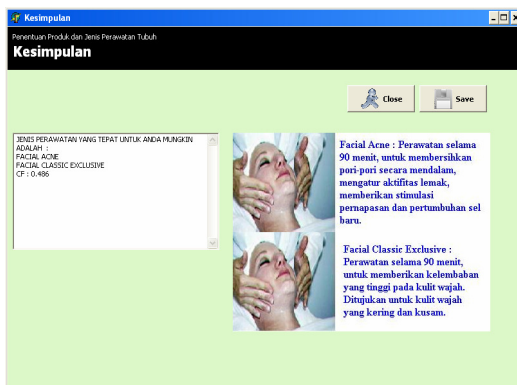
Gambar 7. Konsultasi Perawatan Wajah II



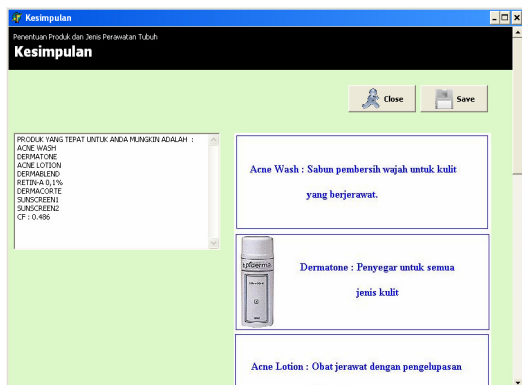
Gambar 8. Konsultasi Perawatan Wajah III



Gambar 9. Kesimpulan Jenis Kulit Wajah



Gambar 10. Hasil sesi Jenis Perawatan Wajah



Gambar 11. Hasil sesi untuk Produk Wajah

Tabel 1 menampilkan rekapitulasi hasil kuisioner responden tentang pengujian dan penilaian program sistem pakar untuk penentuan produk dan jenis perawatan tubuh di pusat perawatan “Epiderma”

Tabel 1. Rekapitulasi hasil kuisioner

Kriteria Penilaian	Prosentase Nilai
Kelayakan Software	80%
Kesimpulan yang dihasilkan	76%
Kecepatan hasil informasi	72%
Kemudahan penggunaan	72%
Tampilan aplikasi	68%

Dengan menganalisis hasil kuisioner, dapat diambil kesimpulan sebagai berikut:

- a. Program sistem pakar ini dapat dikatakan layak untuk digunakan oleh unit pegawai pusat perawatan “Epiderma”. Hal ini dapat dilihat pada hasil kuisioner yang memberikan nilai persentase 80%.
- b. Kesimpulan yang dihasilkan program sistem pakar ini cukup akurat. Hal ini dapat dilihat pada hasil kuisioner yang memberikan nilai persentase 76%.
- c. Kecepatan hasil informasi dari program ini dapat dikatakan cukup cepat. Hal ini dapat dilihat pada hasil kuisioner yang memberikan nilai persentase 72%.
- d. Penggunaan program ini cukup mudah dan tidak membingungkan. Hal ini dapat dilihat pada hasil kuisioner yang memberikan nilai persentase 72%.
- e. Para responden cenderung menilai bahwa tampilan *interface* program ini perlu dibuat lebih menarik. Hal ini dapat dilihat pada hasil kuisioner yang memberikan nilai persentase 68%.

5. KESIMPULAN

Adapun beberapa kesimpulan yang dapat diambil oleh penulis selama penelitian ini adalah sebagai berikut:

- a. *Rule-rule* yang dibuat sudah memiliki tingkat akurasi yang cukup, sehingga dapat memberikan kesimpulan yang cukup akurat kepada pengguna.
- b. Aplikasi sistem pakar telah dibuat dinamis, sehingga apabila terjadi penambahan data, program aplikasi ini tetap dapat digunakan.
- c. Penggunaan metode *forward chaining* pada *inference engine* cocok untuk permasalahan yang dibahas.
- d. Fasilitas Edit yang ada haruslah digunakan oleh orang yang mengerti Sistem Pakar, sehingga tidak terjadi kesalahan baik pada pengoperasiannya maupun terhadap kesimpulan yang dihasilkan aplikasi ini.
- e. Berdasarkan hasil kuisioner, program sistem pakar ini dapat dikatakan layak untuk digunakan oleh unit pegawai di pusat perawatan “Epiderma”, hal ini dapat dilihat dari persentase nilai yang didapat yaitu sebesar 80%. Sistem pakar ini dinilai cukup akurat, dengan persentase nilai yang didapat yaitu sebesar 76%. Kecepatan menghasilkan informasi dinilai cukup cepat dan program ini dinilai cukup mudah untuk digunakan, dengan persentase nilai yang didapat sebesar 72%. Desain tampilan dari sistem pakar ini dinilai masih perlu dibuat lebih menarik, hal tersebut dapat dilihat dari persentase nilai yang dicapai yaitu sebesar 68%.

7. DAFTAR PUSTAKA

- [1] Durkin, J., Expert Systems Design and Development, Prentice Hall International Inc, 1994.
- [2] Ignizio, J. P., Introduction to Expert System : The Development and Implementation of Rule-Based Expert System, Singapore: McGraw-Hill, 1991.
- [3] Gunawan, Kuliah Artificial Intelligence: Pengantar ke Expert System, Indonesia: Surabaya, 2000.
- [4] Bratko, I., Prolog Programming for Artificial Intelligence, Singapore: Addison-Wesley Publishers Ltd, 1990.
- [5] Martina, I., Database Menggunakan Delphi, Jakarta: PT Elex Media Komputindo, 2001.
- [6] Kelompok Penyusun Buku pada Direktorat Pendidikan Masyarakat, Direktorat Jenderal Diklusepora, Tata Kecantikan Kulit Tingkat Dasar, Depdikbud, 2001.
- [7] Kelompok Penyusun Buku pada Direktorat Pendidikan Masyarakat, Direktorat Jenderal Diklusepora, Tata Kecantikan Kulit Tingkat Terampil, Depdikbud, 2001.
- [8] Wasitaatmadja, S. M., Penuntun Ilmu Kosmetik Medik, 1997.