

# REPRESENTASI PENGETAHUAN (*KNOWLEDGE*) BERBASIS RULE (*RULE-BASED*) DALAM MENGANALISA KEKURANGAN VITAMIN PADA TUBUH MANUSIA

Ruri Hartika Zain, S. Kom, M. Kom\*)  
Dosen Tetap Universitas Putra Indonesia YPTK Padang  
Padang – Sumatera Barat – Indonesia - 2012

## Abstrak

Seiring perkembangan teknologi, dikembangkan pula suatu teknologi yang mampu mengadopsi proses dan cara berpikir manusia yaitu teknologi Artificial Intelligence atau Kecerdasan Buatan. Sistem Pakar adalah salah satu bagian dari Kecerdasan Buatan yang mengandung pengetahuan dan pengalaman yang dimasukkan oleh satu atau banyak pakar ke dalam satu area pengetahuan tertentu sehingga setiap orang dapat menggunakannya untuk memecahkan berbagai masalah yang bersifat spesifik, dalam hal ini adalah masalah dalam menganalisa kekurangan vitamin pada tubuh manusia.

Keyword: Sistem Pakar, knowledge dan rule based

## I. PENDAHULUAN

Vitamin adalah suatu zat senyawa kompleks yang sangat dibutuhkan oleh tubuh kita yang berfungsi untuk membantu pengaturan atau proses kegiatan tubuh. Tanpa vitamin manusia, hewan dan makhluk hidup lainnya tidak akan dapat melakukan aktifitas hidup dan kekurangan vitamin dapat menyebabkan memperbesar peluang terkena penyakit pada tubuh kita. Dengan adanya sistem pakar maka akan dapat menganalisa gejala-gejala kekurangan vitamin pada tubuh manusia sehingga segera tertangani dengan efektif dan efisien.

## II. LANDASAN TEORI

### 2.1 Rekayasa Perangkat Lunak

Rekayasa perangkat lunak telah berkembang sejak pertama kali diciptakan pada tahun 1940-an hingga kini

#### 2.1.1 Definisi Rekayasa Perangkat Lunak

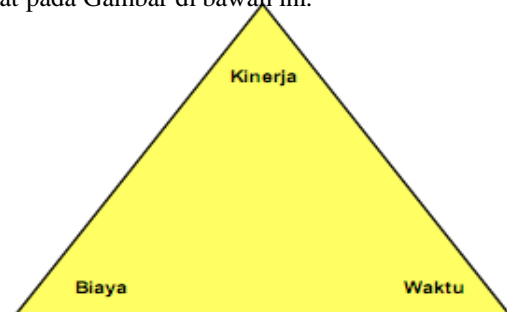
Istilah Rekayasa Perangkat Lunak (RPL) secara umum disepakati sebagai terjemahan dari istilah *Software engineering*. Istilah *Software Engineering* mulai dipopulerkan pada tahun 1968 pada *Software Engineering Conference* yang diselenggarakan oleh NATO. Sebagian orang mengartikan RPL hanya sebatas pada bagaimana membuat program komputer. Padahal ada perbedaan yang mendasar antara perangkat lunak (*software*) dan program komputer.

Perangkat lunak adalah seluruh perintah yang digunakan untuk memproses informasi. Perangkat lunak dapat berupa program atau prosedur. Program adalah kumpulan perintah yang dimengerti oleh komputer sedangkan prosedur adalah perintah yang dibutuhkan oleh pengguna dalam memproses informasi (O'Brien, 1999).

Pengertian RPL sendiri adalah suatu disiplin ilmu yang membahas semua aspek produksi perangkat lunak, mulai dari tahap awal yaitu analisa kebutuhan pengguna, menentukan spesifikasi dari kebutuhan pengguna, disain, pengkodean, pengujian sampai pemeliharaan sistem setelah digunakan.

#### 2.1.2 Tujuan Rekayasa Perangkat Lunak

Secara umum tujuan RPL tidak berbeda dengan bidang rekayasa yang lain. Hal ini dapat kita lihat pada Gambar di bawah ini.



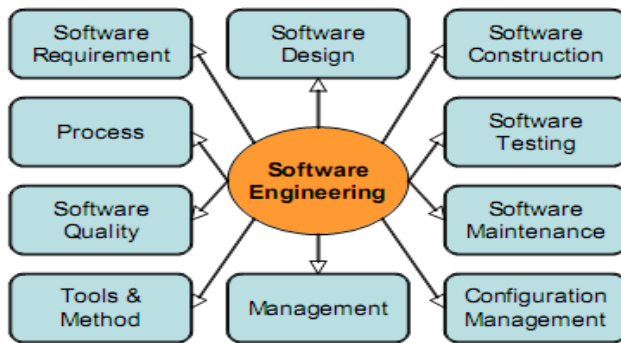
Gambar 2.1 Tujuan RPL

Dari Gambar di atas dapat diartikan bahwa bidang rekayasa akan selalu berusaha menghasilkan *output* yang kinerjanya tinggi, biaya rendah dan waktu penyelesaian yang tepat. Secara lebih khusus kita dapat menyatakan tujuan RPL adalah:

- Memperoleh biaya produksi perangkat lunak yang rendah.
- Menghasilkan perangkat lunak yang kinerjanya tinggi, andal dan tepat waktu.
- Menghasilkan perangkat lunak yang dapat bekerja pada berbagai jenis platform.
- Menghasilkan perangkat lunak yang biaya perawatannya rendah.

#### 2.1.3 Ruang Lingkup Rekayasa Perangkat Lunak

Sesuai dengan definisi yang telah disampaikan sebelumnya, maka ruang lingkup RPL dapat digambarkan sebagai berikut:



**Gambar 2.2 Ruang Lingkup RPL**

- *Software Requirements* berhubungan dengan spesifikasi kebutuhan dan persyaratan perangkat lunak.
- *Software design* mencakup proses penampilan arsitektur, komponen, antar muka, dan karakteristik lain dari perangkat lunak.
- *Software construction* berhubungan dengan detail pengembangan perangkat lunak, termasuk algoritma, pengkodean, pengujian dan pencarian kesalahan.
- *Software testing* meliputi pengujian pada keseluruhan perilaku perangkat lunak.
- *Software maintenance* mencakup upaya-upaya perawatan ketika perangkat lunak telah dioperasikan.
- *Software configuration management* berhubungan dengan usaha perubahan konfigurasi perangkat lunak untuk memenuhi kebutuhan tertentu.
- *Software engineering management* berkaitan dengan pengelolaan dan pengukuran RPL, termasuk perencanaan proyek perangkat lunak.
- *Software engineering tools and methods* mencakup kajian teoritis tentang alat bantu dan metode RPL.
- *Software engineering process* berhubungan dengan definisi, implementasi pengukuran, pengelolaan, perubahan dan perbaikan proses RPL.
- *Software quality* menitik beratkan pada kualitas dan daur hidup perangkat lunak.

### III. ANALISA DAN PERANCANGAN

#### 3.1 Analisa Vitamin

1. Vitamin A, yang juga dikenal dengan nama *retinol*, merupakan vitamin yang berperan dalam pembentukan Indra penglihatan yang baik, terutama di malam hari, dan sebagai salah satu komponen penyusun *pigmen* mata

di retina. Selain itu, vitamin ini juga berperan penting dalam menjaga kesehatan kulit dan imunitas tubuh. Vitamin ini bersifat mudah rusak oleh paparan panas, cahaya matahari, dan udara.

2. Vitamin B. Secara umum, golongan vitamin B berperan penting dalam *metabolisme* di dalam tubuh, terutama dalam hal pelepasan energi saat beraktivitas. Hal ini terkait dengan peranannya di dalam tubuh, yaitu sebagai senyawa *koenzim* yang dapat meningkatkan laju reaksi *metabolisme* tubuh terhadap berbagai jenis sumber energi.
3. Vitamin C. Vitamin C (*asam askorbat*) banyak memberikan manfaat bagi kesehatan tubuh kita. Di dalam tubuh, vitamin C juga berperan sebagai senyawa pembentuk *kolagen* yang merupakan protein penting penyusun jaringan kulit, sendi, tulang, dan jaringan penyokong lainnya. Vitamin C merupakan senyawa *antioksidan* alami yang dapat menangkal berbagai *radikal* bebas dari polusi di sekitar lingkungan kita.
4. Vitamin D. Vitamin D juga merupakan salah satu jenis vitamin yang banyak ditemukan pada makanan hewani, antara lain ikan, telur, susu, serta produk olahannya, seperti keju. Bagian tubuh yang paling banyak dipengaruhi oleh vitamin ini adalah tulang. Vitamin D ini dapat membantu *metabolisme* kalsium dan mineralisasi tulang.
5. Vitamin E. Vitamin E berperan dalam menjaga kesehatan berbagai jaringan di dalam tubuh, mulai dari jaringan kulit, mata, sel darah merah hingga hati. Selain itu, vitamin ini juga dapat melindungi paru-paru manusia dari polusi udara. Nilai kesehatan ini terkait dengan kerja vitamin E di dalam tubuh sebagai senyawa *antioksidan* alami.
6. Vitamin K. Vitamin K banyak berperan dalam pembentukan sistem peredaran darah yang baik dan penutupan luka. *Defisiensi* vitamin ini akan berakibat pada pendarahan di dalam tubuh dan kesulitan pembekuan darah saat terjadi luka atau pendarahan.

#### 3.2 Analisa Sistem

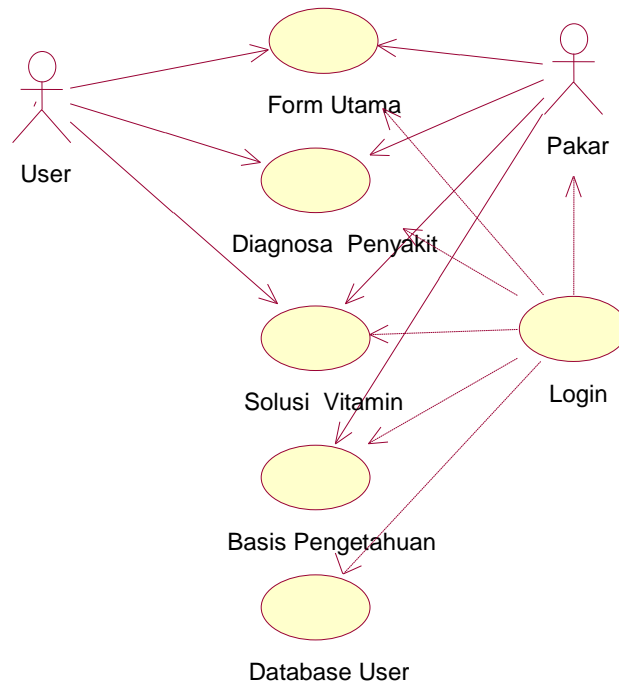
*Artificial Intelligence* merupakan sistem dengan basis pengetahuan yang dinamis, dimana pengetahuan itu dapat berubah seiring dengan berjalannya waktu, sehingga harus dapat dilakukan pembaharuan seperti penambahan, penghapusan maupun perubahan terhadap data yang sudah disimpan sebelumnya tanpa harus mengubah isi dari program secara keseluruhan.

##### 3.2.1 Use Case Diagram

Diagram ini menjelaskan interaksi yang terjadi antara *user* dengan tingkatan tertentu dengan

kegiatan apa saja yang dapat dilakukannya pada aplikasi sistem pakar ini. Apabila kita masuk sebagai *user*, kita tidak bisa melihat basis pengetahuannya.

Sedangkan kalau kita masuk sebagai pakar, kita dapat menambah dan mengubah basis pengetahuannya.



**Gambar 3.1 Use Case Diagram**

**Tabel 3.1 Definisi Use Case**

No	Use Case	Deskripsi
1	<i>Login</i>	Untuk masuk kedalam sistem
2	<i>Form Utama</i>	Merupakan bentuk tampilan menu awal dari sistem yang di rancang yang berisi tentang informasi tentang sistem
3	<i>Diagnosa Penyakit</i>	Menampilkan daftar-daftar pertanyaan, disini akan dilakukan pendiagnosaan berupa pemberian pertanyaan terhadap <i>user</i> yang berhubungan dengan gejala penyakit kekurangan vitamin pada tubuh
4	<i>Solusi Vitamin</i>	Berisi tentang hasil dari pendiagnosaan dari masalah kekurangan vitamin, disini juga di cantumkan cara penanggulangan kekurangan vitamin dan cara pencegahan agar masalah vitamin yang sama tidak terjadi kembali
5	<i>Basis Pengetahuan</i>	Berisi tentang masalah kekurangan vitamin berupa pertanyaan-pertanyaan dan solusinya
6	<i>Database User</i>	Tempat menyimpan data <i>user</i> berupa <i>user name</i> dan <i>password</i>

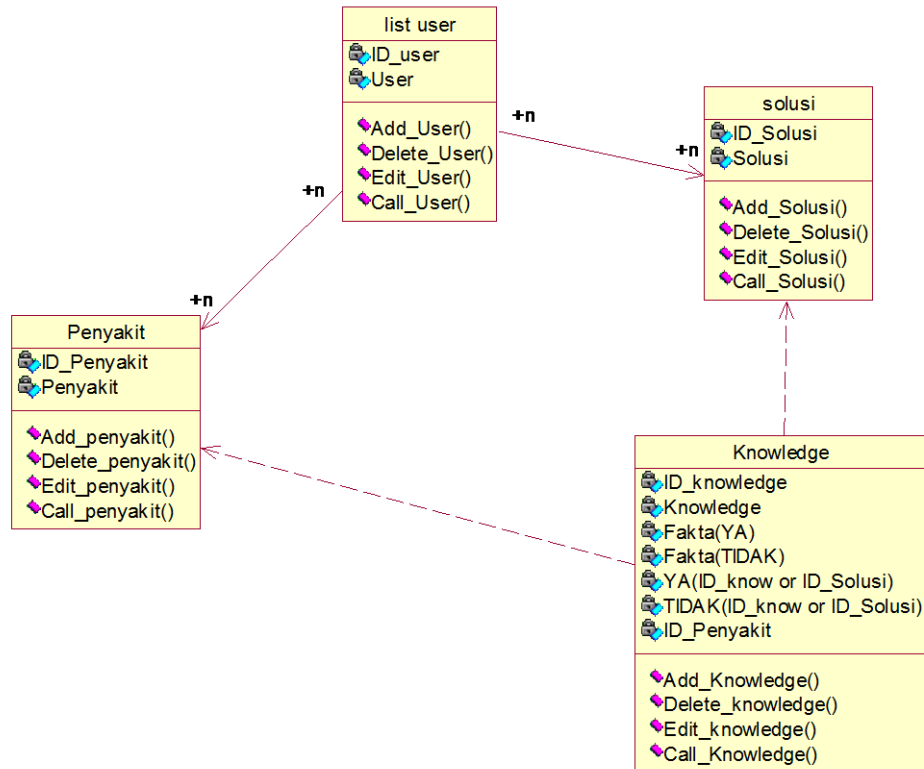
### 3.2.2 Definisi Aktor

Pada bagian ini akan menjabarkan aktor yang terlibat dalam Perancangan Sistem Pakar Menganalisa Kekurangan Vitamin pada Tubuh Manusia.

**Tabel 3.2 Defenisi Aktor**

No	Aktor	Peran
1	User	Sebagai pengguna dari sistem pakar, yang mengharuskan mereka untuk <i>login</i> terlebih dahulu. Setelah itu mereka dapat menggunakan sumberdaya yang terdapat didalam sistem tersebut
2	Pakar	Sebagai pakar dalam sistem ini, yang dapat menambah dan merubah basis pengetahuan

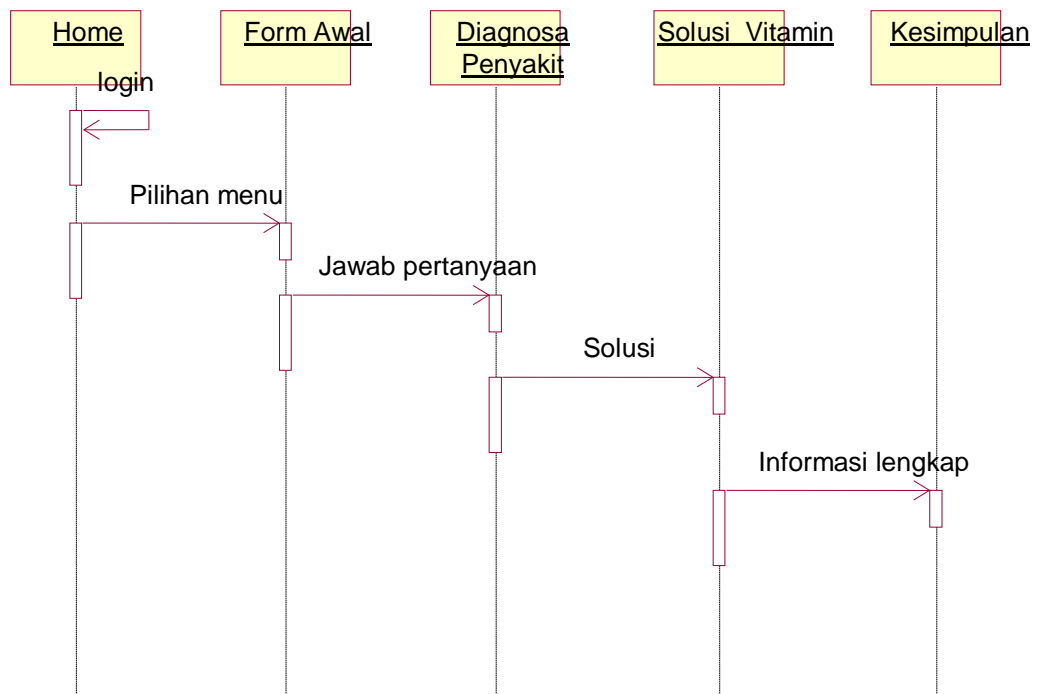
### 3.2.3 Class Diagram



**Gambar 3.2 Class Diagram**

Class Diagram tersebut menjelaskan hubungan antara tabel pada database yang digunakan oleh sistem pakar ini, seperti yang dilihat diatas bahwa antar tabel penyakit, list user dan solusi mempunyai hubungan many to many. Pada tabel knowledge terlihat bahwa tabel ini memiliki ketergantungan dengan tabel penyakit dan solusi.

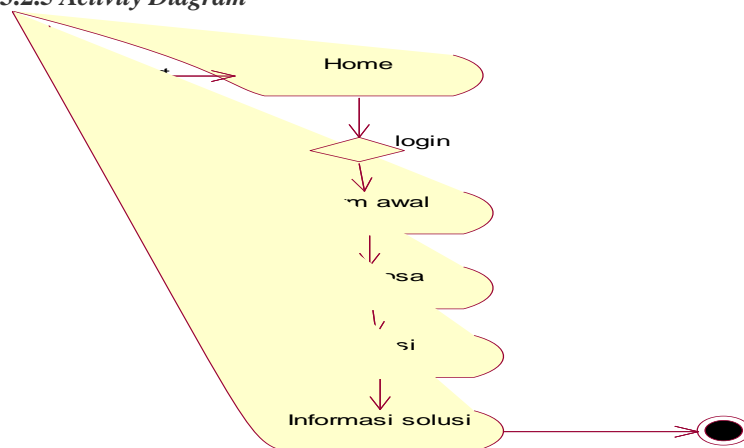
### 3.2.4 Sequence Diagram



**Gambar 3.3 Sequence Diagram**

Diagram ini menjelaskan hubungan antara *form*, dari *form home* anda akan dibawa ke *form awal*, pada *form* ini akan ada beberapa pilihan tergantung dari *level user* yang dipilih oleh *user*, selanjutnya ada *form diagnosa penyakit*, berisikan pertanyaan pendiagnosaan penyakit, setelah *user* menjawab semua pertanyaan ini *user* akan mendapatkan solusi yang diinginkan, jika *user* ingin mengetahui informasi lengkap mengenai penyakitnya *user* tinggal menekan tombol alasan.

### 3.2.5 Activity Diagram



**Gambar 3.4 Activity Diagram**

Gambar ini menjelaskan bagaimana alur kerja dan aktivitas aplikasi jika dijalankan, Aplikasi akan dimulai dengan tampilan *home* dilanjutkan dengan tampilan *form* utama, diagnosa, solusi dan terakhir tampilan informasi lengkap.

### 3.3 Perancangan Database

Perancangan *database* dilakukan setelah perancangan sistem dibuat. Dengan menggunakan *Access* sebagai *database* dilakukan perancangan terhadap *field-field* yang akan digunakan pada setiap *database*.

1. Tabel *List User*. Tabel ini digunakan untuk menyimpan informasi dari *user* yang berperan sebagai pakar.

**Tabel 4.3 Tabel List User**

No	Field	Keterangan

1	<i>User id</i>	<i>Primary key</i>
2	<i>Password</i>	Merupakan hak akses untuk masuk kedalam sistem
3	<i>Level</i>	Merupakan tingkatan pengguna

2. Tabel *Knowledge*. Tabel ini berisi tentang pertanyaan yang bersangkutan dengan penganalisaan masalah kekurangan vitamin yang terdapat pada sistem ini.

**Tabel 4.4 Tabel Knowledge**

No	Field	Keterangan
1	<u>Id</u>	<i>Primary key</i>
2	Pertanyaan	Berisi pertanyaan penganalisaan masalah kekurangan vitamin
3	Fakta Ya	Berisi suatu keadaan apabila <i>user</i> menjawab dengan jawaban ya maka akan di lanjutkan ke mana pertanyaan tersebut
4	Fakta Tidak	Berisi suatu keadaan apabila <i>user</i> menjawab dengan jawaban tidak maka akan di lanjutkan ke mana pertanyaan tersebut
5	Id_penyakit	Berisi tentang jenis penyakit

3. Tabel Penyakit. Tabel ini digunakan untuk memasukan jenis penyakit baru atau penyakit lain.

**Tabel 3.5 Tabel Penyakit**

No	Field	Keterangan
1	<u>Id_penyakit</u>	<i>Primary key</i>
2	Jenis_penyakit	Merupakan suatu masalah penyakit

4. Tabel Solusi. Tabel ini digunakan untuk memasukan solusi dari masalah kekurangan vitamin atau solusi penyakit lainnya.

**Table 3.6 Tabel Solusi**

No	Field	Keterangan
1	<u>Id_solusi</u>	<i>Primary key</i>
2	Solusi	Merupakan jawaban dari hasil penganalisaan.

#### 4.4 Perancangan Tampilan Program

Setelah semua data yang diperlukan dalam merancang sistem pakar menganalisa kekurangan vitamin pada tubuh didapatkan dan *diinputkan* dalam database maka selanjutnya akan dibuatkan desain *interface* yang berguna bagi *user* untuk mengakses program sistem pakar yang telah kita bangun.

### IV. PENGUJIAN DAN IMPLEMENTASI

#### 4.1 Implementasi Sistem

Implementasi sistem dilakukan setelah melakukan perancangan sistem pakar. Perancangan *interface* dilakukan untuk interaksi *user* dengan sistem yang telah dibuat. Untuk melakukan sebuah implementasi sistem diperlukan program dan penulisan program sesuai sistem pakar yang dirancang.

#### 4.2 Pengujian Sistem

##### 4.2.1 Tampilan Awal

Tampilan ini muncul ketika *user* telah berhasil *login* kedalam program, baik sebagai pengguna ataupun sebagai pakar, bedanya pada *level* pakar akan ada tambahan tombol *edit* pengetahuan untuk menambah, menghapus atau bahkan mengubah data yang ada pada *database*. Pengguna hanya dapat memilih tombol diagnosa penyakit atau tombol *help*.



Gambar 4.1 Tampilan Awal

#### 4.2.2 Tampilan Diagnosa

*Form* ini akan tampil jika *user* menekan tombol diagnosa penyakit, baik sebagai pakar ataupun pengguna. Pada *form* ini *user* akan diberi pilihan jenis penyakit apa yang akan diperiksa oleh aplikasi ini, *form* ini ada karena banyaknya jenis penyakit dalam yang ada pada tubuh manusia, sehingga peneliti membuat *form* ini agar penyakit-penyakit tersebut dapat dikelompokkan pada satu bidang penyakit. *User* memilih jenis penyakit yang dirasa cocok dengan gejalanya, aplikasi akan memberikan pertanyaan-pertanyaan untuk mengklasifikasikan penyakit apa yang diderita *user*.



Gambar 4.2 Tampilan Diagnosa

#### 4.2.3 Tampilan Pertanyaan

Tampilan ini keluar setelah *user* memilih perkiraan jenis penyakit yang dideritanya. Sesuai dengan metode yang dipakai sistem ini maka *user* akan diberi pertanyaan dengan klasifikasi bertingkat sampai aplikasi mendapatkan pertanyaan dengan jawaban kriteria khusus dari penyakit tersebut.



**Gambar 4.3 Tampilan Pertanyaan 1**

Pada pertanyaan satu diatas, *user* diberi pertanyaan, tetapi pertanyaan ini masih sangat luas, terdapat beberapa penyakit yang memiliki kriteria seperti jawaban pertanyaan diatas.

#### 4.2.4 Tampilan Solusi

Tampilan ini keluar setelah *user* menyelesaikan sesi pertanyaan pada aplikasi ini, *user* akan mendapatkan informasi singkat pada tampilan ini, pada tampilan ini terdapat 2 tombol yang dapat dipilih *user* untuk langkah selanjutnya, yaitu tombol ke *form* Alasan atau tombol Selesai.



**Gambar 4.4 Tampilan Solusi**

#### 4.2.5 Tampilan Informasi

Tampilan terakhir dari sesi diagnosa pada aplikasi ini, *form* ini akan keluar jika *user* memilih tombol Alasan pada tampilan sebelumnya, pada *form* ini *user* akan mendapatkan informasi akurat mengenai masalah kekurangan vitamin yang ia derita.

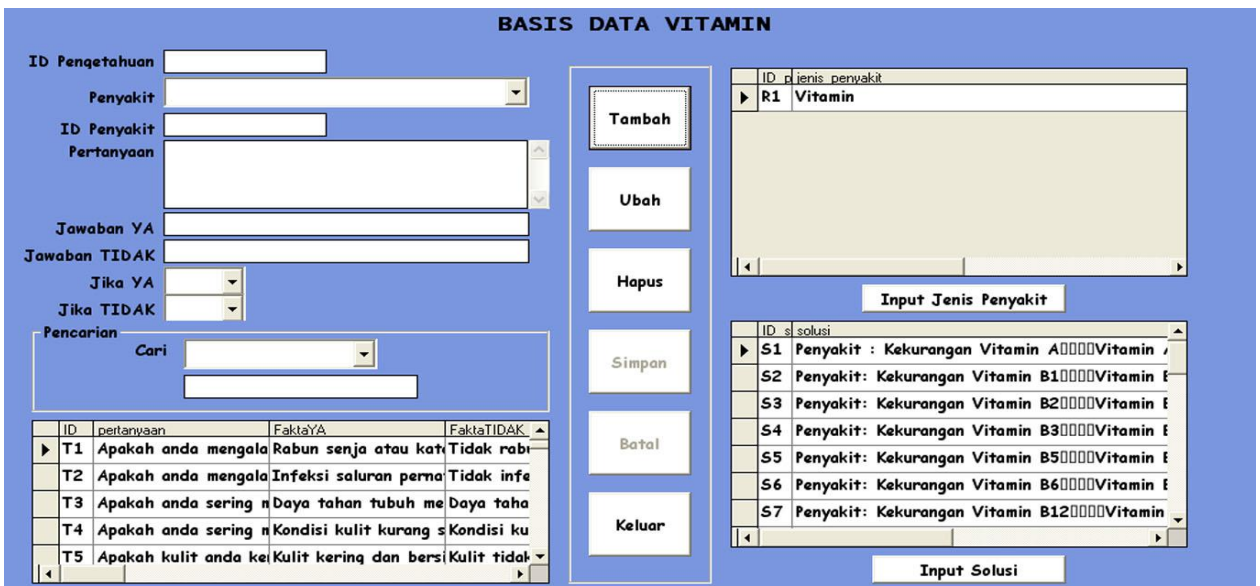


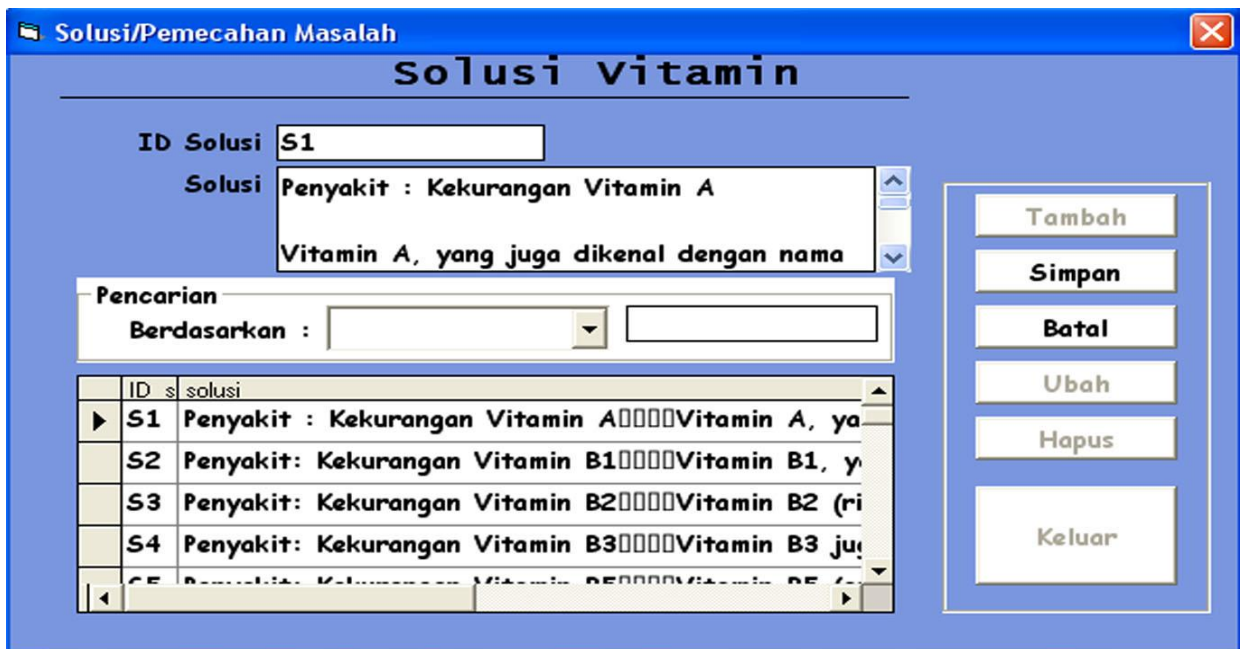


Gambar 4.5 Tampilan Informasi

#### 4.2.6 Tampilan Basis Pengetahuan

Teori *certainty factor* merupakan teori yang menyatakan bahwa setiap aplikasi sistem pakar harus siap untuk menghadapi semua faktor-faktor ketidakpastian yang mungkin terjadi di masa yang akan datang, kemungkinan penyakit itu menghilang, atau adanya penyakit baru harus dapat dihadapi oleh aplikasi ini, maka peneliti menambahkan *form* basis pengetahuan pada aplikasi ini, *form* ini hanya dapat diakses oleh level pakar. Pada tampilan *edit* pengetahuan ini *user* dengan level pakar dapat melakukan *entry* data baru, menghapus data yang tidak sesuai lagi ataupun mengubah data yang tersedia. Tampilan ini memberikan semua akses pada setiap tabel *database*, sehingga sangat memudahkan pakar untuk melakukan pengeditan secara otomatis.





Gambar 4.7 Tampilan Edit Solusi

## V PENUTUP

Berdasarkan hasil perancangan sistem dengan judul Representasi Pengetahuan (*Knowledge*) Berbasis Rule (*Rule-Based*) Dalam Menganalisa Kekurangan Vitamin Pada Tubuh Manusia dapat diambil kesimpulan yaitu :

1. Perancangan sistem ini bertujuan untuk memberikan kemudahan kepada *user* untuk dapat mengetahui masalah kekurangan vitamin yang dideritanya dan solusi vitamin tersebut.
2. *Visual Basic* yang digunakan dalam penerapan sistem ini bertujuan agar *user* selaku pengguna yang membutuhkan banyak pengetahuan tentang masalah kekurangan vitamin dapat secara leluasa menggunakannya tanpa di persulit sedikit pun.
3. Tampilan sistem yang dirancang ini merupakan tampilan yang sederhana dan

mudah dipahami penggunaannya sehingga pengguna dapat melakukan pengaksesan dengan mudah dan tidak merasa bingung untuk menggunakannya.

4. Aplikasi ini juga dapat meringankan kerja para ahli dalam menganalisa kekurangan vitamin pada tubuh manusia, dengan kata lain aplikasi ini dapat dijadikan sebagai asisten pakar dalam melakukan pekerjaannya.

## DAFTAR PUSTAKA

2009, Pengembangan sistem pakar menggunakan Visual basic, Yogyakarta : ANDI OFFSET.Kusrini, 2006, *Sistem Pakar Teori dan Aplikasi*. Yogyakarta : Andi Offset.Kusumadewi, S. , 2003, *Artificial Intelligence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.

