

**PERANCANGAN SISTEM PAKAR UNTUK DIAGNOSA
PENYAKIT ANAK**

SKRIPSI

**SAFIA DHANY
031401001**



**PROGRAM STUDI S-1 ILMU KOMPUTER
DEPARTEMEN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SUMATERA UTARA
MEDAN
2009**

PERANCANGAN SISTEM PAKAR UNTUK DIAGNOSA PENYAKIT ANAK

SKRIPSI

Diajukan untuk melengkapi tugas dan memenuhi syarat mencapai gelar
Sarjana Komputer

SAFIA DHANY
031401001



PROGRAM STUDI S-1 ILMU KOMPUTER
DEPARTEMEN ILMU KOMPUTER
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SUMATERA UTARA
MEDAN
2009

PERSETUJUAN

Judul : PERANCANGAN SISTEM PAKAR UNTUK
DIAGNOSA PENYAKIT ANAK
Kategori : SKRIPSI
Nama : SAFIA DHANY
Nomor Induk Mahasiswa : 031401001
Program Studi : SARJANA (S1) ILMU KOMPUTER
Departemen : ILMU KOMPUTER
Fakultas : MATEMATIKA DAN ILMU PENGETAHUAN
ALAM (FMIPA) UNIVERSITAS SUMATERA
UTARA

Diluluskan di
Medan, 21 Januari 2009

Komisi Pembimbing :

Pembimbing 2

Pembimbing 1

Dra. Esther Nababan, M.Sc
NIP 131757011

Prof. Dr. Muhammad Zarlis
NIP 131570434

Diketahui/Disetujui oleh
Departemen Ilmu Komputer FMIPA USU
Ketua,

Prof. Dr. Muhammad Zarlis
NIP 131570434

PERNYATAAN

PERANCANGAN SISTEM PAKAR UNTUK DIAGNOSA PENYAKIT ANAK

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, Februari 2009

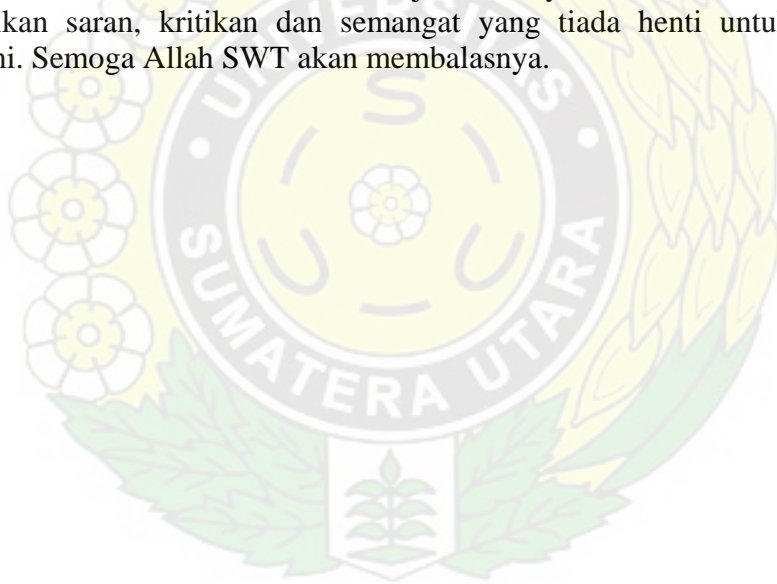
SAFIA DHANY
031401001



PENGHARGAAN

Puji dan syukur penulis panjatkan kepada Allah SWT, dengan limpahan karunia-Nya skripsi ini berhasil diselesaikan dalam waktu yang telah ditetapkan.

Ucapan terima kasih penulis sampaikan kepada Bapak Prof. Dr. Muhammad Zarlis dan Ibu Dra. Esther Nababan, M.Sc selaku pembimbing pada penyelesaian skripsi ini yang telah memberikan panduan dan penuh kepercayaan kepada penulis untuk menyempurnakan skripsi ini. Ucapan terima kasih juga ditujukan kepada Ketua dan Sekretaris Departemen Ilmu Komputer, Bapak Prof. Dr. Muhammad Zarlis dan Bapak Syahriol Sitorus, S.Si, MIT, Dekan dan Pembantu Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sumatera Utara, semua dosen pada Departemen Ilmu Komputer FMIPA USU, serta pegawai di FMIPA USU. Akhirnya, tidak terlupakan kepada keluarga, Abah dan Ibu, abang dan adik (Anda dan Sari) yang selalu memberikan dukungan dan perhatiannya, juga semua sahabat (Nurul, Ade, Novi, Tiwi, Ai, Lui, Nita, Yasri, Fajar, Ronny, Ari dan Nandar) yang selalu memberikan saran, kritikan dan semangat yang tiada henti untuk menyelesaikan skripsi ini. Semoga Allah SWT akan membalasnya.



ABSTRAK

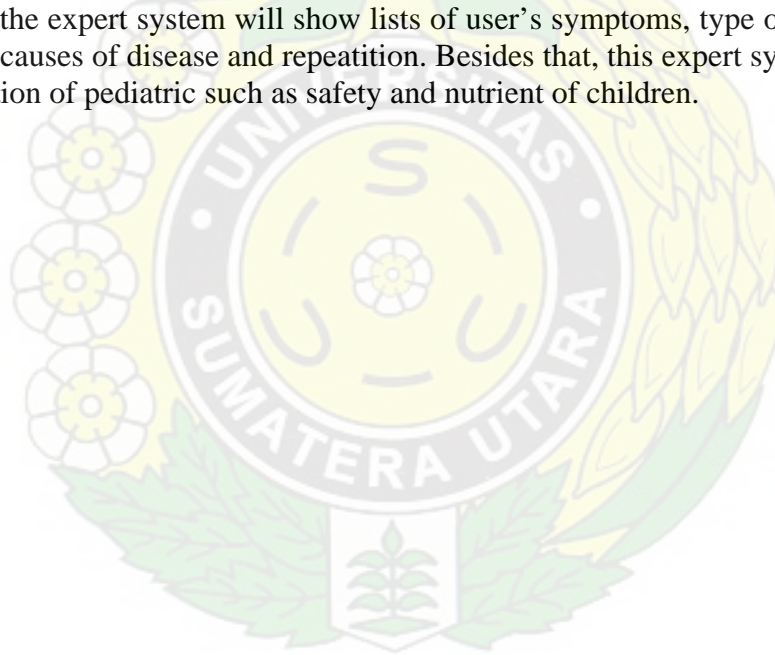
Sistem pakar untuk diagnosa penyakit anak ini merupakan suatu sistem pakar yang dirancang sebagai alat bantu untuk mendiagnosa jenis penyakit tropis khususnya pada balita dengan basis pengetahuan yang dinamis. Pengetahuan ini didapat dari berbagai sumber diantaranya penelitian dan seminar yang dilakukan pakar dalam bidangnya serta buku yang berhubungan dengan penyakit anak. Basis pengetahuan disusun sedemikian rupa ke dalam suatu database dengan beberapa tabel diantaranya tabel penyakit, tabel gejala dan tabel aturan untuk mempermudah kinerja sistem dalam penarikan kesimpulan. Penarikan kesimpulan dalam sistem pakar ini menggunakan metode inferensi *forward chaining*. Sistem pakar ini akan menampilkan pilihan gejala yang dapat dipilih oleh *user*, dimana setiap pilihan gejala akan membawa *user* kepada pilihan gejala selanjutnya sampai mendapatkan hasil akhir. Pada hasil akhir sistem pakar akan menampilkan pilihan gejala *user*, jenis penyakit yang diderita, penyebab dan penanggulangannya. Disamping itu, sistem pakar ini juga memberikan informasi anak seperti keamanan dan gizi anak.



DESIGNED EXPERT SYSTEM TO DIAGNOSE OF PEDIATRIC

ABSTRACT

The expert system to diagnose of pediatric is a system designed as a supporting tool in diagnosing type of tropical diseases, particularly on toddler under dynamic knowledge base. This knowledge is gotten from various resources such as research and seminar conducted by related experts and books about pediatric. Basic knowledge is set into a database tables including table of disease, table of symptoms and table of rule to ease up system in deciding conclusion. The conclusion deciding in this expert system is done using forward chaining of inference methods. This expert system will show choice of symptoms which can be chosen by user, where each choice of symptoms will lead the user to the next choice of symptoms until get the final results. At the final results, the expert system will show lists of user's symptoms, type of diseases suffered by user, causes of disease and repetition. Besides that, this expert system can give the information of pediatric such as safety and nutrient of children.



DAFTAR ISI

	Halaman
Pesetujuan	ii
Pernyataan	iii
Penghargaan	iv
Abstrak	v
Abstract	vi
Daftar Isi	vii
Daftar Tabel	ix
Daftar Gambar	x
 Bab 1 Pendahuluan	 1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
1.6 Metodologi Penelitian	4
1.7 Sistematika Penulisan	5
 Bab 2 Tinjauan Teori	 7
2.1 Kecerdasan Buatan	7
2.2 Sistem Pakar	9
2.2.1 Sejarah Sistem Pakar	9
2.2.2 Ciri-Ciri Sistem Pakar	10
2.2.3 Arsitektur Sistem Pakar	11
2.2.4 Representasi Pengetahuan	12
2.2.5 Metode Inferensi	15
2.3 Pediatri	16
2.4 Visual Basic 6.0	17
 Bab 3 Analisis dan Perancangan Sistem	 18
3.1 Analisis Penyakit	18
3.1.1 Diare (Gastroenteritis)	19
3.1.2 ISPA (Infeksi Saluran Pernafasan Akut)	20
3.1.2.1 Tonsilitis	21
3.1.2.2 Laringitis	21
3.1.2.3 Bronkhitis	22
3.1.2.4 Bronkiolitis	23
3.1.2.5 Pneumonia	23
3.1.3 Penyakit Kulit	23
3.1.3.1 Dermatitis Atopik	24
3.1.3.2 Mikosis Superfisialis	24
3.1.3.3 Moniliasis	24

3.1.4	Meningitis	25
3.2	Analisis Permasalahan	26
3.3	Model Analisis	31
3.3.1	Data Flow Diagram Level 1 (DFD Level 1)	32
3.3.2	Data Flow Diagram Level 2 (DFD Level 2)	32
3.4	Perancangan Basis Pengetahuan	33
3.5	Perancangan Antarmuka (Interface)	36
3.5.1	Rancangan Menu Utama	36
3.5.2	Rancangan Menu Info	37
3.5.3	Rancangan Menu Diagnosa	37
3.5.4	Rancangan Form Logon	39
3.5.5	Rancangan Menu Update	40
3.5.5.1	Rancangan Form Data Penyakit	40
3.5.5.2	Rancangan Form Data Password	41
3.5.6	Rancangan Menu Bantuan	42
Bab 4	Implementasi Sistem Pakar	43
4.1	Menu Utama	43
4.2	Menu Info	43
4.3	Menu Diagnosa	44
4.4	Menu Update	47
4.4.1	Form Data Penyakit	48
4.4.2	Form Data Password	49
4.5	Menu Bantuan	50
4.6	Teknik Pengujian Sistem	50
Bab 5	Kesimpulan dan Saran	55
5.1	Kesimpulan	55
5.2	Saran	55
	Daftar Pustaka	56
	Lampiran A: Kode Program	57

DAFTAR TABEL

	Halaman
Tabel 2.1 Representasi OAV	14
Tabel 2.2 Representasi Bingkai Pada Bingkai Penyakit	14
Tabel 3.1 Tabel Gejala	34
Tabel 3.2 Tabel Pertanyaan	34
Tabel 3.3 Tabel Penyakit	34
Tabel 3.4 Tabel Aturan	35
Tabel 3.5 Tabel Info	35
Tabel 3.6 Tabel Password	35
Tabel 4.1 Tabel Aturan Pada Aturan 24	52
Tabel 4.2 Tabel Aturan Pada Aturan 32	53



DAFTAR GAMBAR

	Halaman
Gambar 1.1 Diagram Konsep Sistem Pakar Untuk Diagnosa Penyakit Anak	2
Gambar 2.1 Representasi Jaringan Semantik	13
Gambar 2.2 Diagram Penyakit Anak	17
Gambar 3.1 Diagram Pohon Keputusan Diagnosa	27
Gambar 3.2 Diagram Pohon Keputusan I	27
Gambar 3.3 Diagram Pohon Keputusan II	28
Gambar 3.4 Diagram Pohon Keputusan III	28
Gambar 3.5 Lanjutan Diagram Pohon Keputusan III	29
Gambar 3.6 Lanjutan Diagram Pohon Keputusan III	29
Gambar 3.7 Lanjutan Diagram Pohon Keputusan III	29
Gambar 3.8 Diagram Pohon Keputusan IV	30
Gambar 3.9 Diagram Flowchart Sistem	31
Gambar 3.10 DFD Level 0	32
Gambar 3.11 DFD Level 1	32
Gambar 3.12 DFD Level 2 Untuk Proses 2 (Penelusuran)	33
Gambar 3.13 Rancangan Tampilan Menu Utama	36
Gambar 3.14 Rancangan Tampilan Menu Info	37
Gambar 3.15 Rancangan Tampilan Form Penelusuran	38
Gambar 3.16 Rancangan Tampilan Form Hasil Diagnosa	38
Gambar 3.17 Rancangan Tampilan Menu Detail Penyakit	39
Gambar 3.18 Rancangan Tampilan Form Logon	39
Gambar 3.19 Rancangan Tampilan Menu Update	40
Gambar 3.20 Rancangan Tampilan Form Data Penyakit	41
Gambar 3.21 Rancangan Tampilan Form Data Password	42
Gambar 3.22 Rancangan Tampilan Menu Bantuan	42
Gambar 4.1 Form Menu Utama	43
Gambar 4.2 Form Menu Info	44
Gambar 4.3 Form Penelusuran 1	45
Gambar 4.4 Form Penelusuran 2	45
Gambar 4.5 Form Hasil Diagnosa	46
Gambar 4.6 Form Detail Penyakit	46
Gambar 4.7 Form Menu Update	47
Gambar 4.8 Form Logon	48
Gambar 4.9 Form Data Penyakit	49
Gambar 4.10 Form Data Password	50
Gambar 4.11 Form Menu Bantuan	50
Gambar 4.12 Foto Penyakit Pada Menu Lihat Detail Setelah Proses Hasil Diagnosa	54

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Perkembangan ilmu kedokteran mengalami kemajuan pesat yang ditandai dengan ditemukannya penyakit-penyakit tropis baru yang belum teridentifikasi sebelumnya. Para dokter ahli terus mencoba menemukan solusi untuk mengatasi penemuan baru dan selalu mencoba memberikan pelayanan terbaik terhadap para pasien.

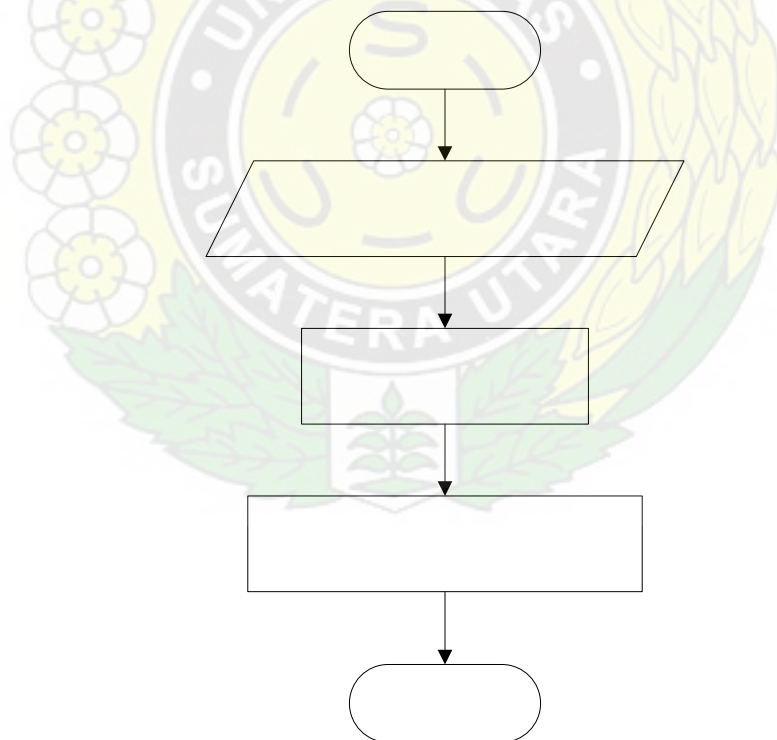
Kesehatan merupakan hal yang berharga bagi manusia, karena siapa saja dapat mengalami gangguan kesehatan. Anak sangat rentan terhadap kuman penyakit dan kurangnya kepekaan terhadap gejala suatu penyakit merupakan ketakutan tersendiri bagi orang tua. Orang tua merupakan orang awam yang kurang memahami kesehatan. Apabila terjadi gangguan kesehatan terhadap anak maka mereka lebih mempercayakannya kepada pakar atau dokter ahli yang sudah mengetahui lebih banyak tentang kesehatan, tanpa memperdulikan apakah gangguan tersebut masih dalam tingkat rendah atau kronis. Namun dengan kemudahan dengan adanya para pakar atau dokter ahli, terkadang terdapat pula kelemahannya seperti jam kerja (praktek) terbatas dan banyaknya pasien sehingga harus menunggu antrian. Dalam hal ini, orang tua selaku pemakai jasa lebih membutuhkan seorang pakar yang bisa memudahkan dalam mengdiagnosa penyakit lebih dini agar dapat melakukan pencegahan lebih awal yang sekiranya membutuhkan waktu jika berkonsultasi dengan dokter ahli. Karena hal tersebutlah maka dibutuhkan suatu alat bantu yang dapat mendiagnosa penyakit anak berupa suatu sistem pakar.

Sistem pakar merupakan salah satu cabang kecerdasan buatan yang mempelajari bagaimana “mengadopsi” cara seorang pakar berfikir dan bernalar dalam

menyelesaikan suatu permasalahan, dan membuat suatu keputusan maupun mengambil kesimpulan dari sejumlah fakta yang ada. Dasar dari sistem pakar adalah bagaimana memindahkan pengetahuan yang dimiliki oleh seorang pakar ke dalam komputer, dan bagaimana membuat keputusan atau mengambil kesimpulan berdasarkan pengetahuan itu.

1.2. Rumusan Masalah

Permasalahan yang akan dibahas dalam penulisan tugas akhir ini adalah bagaimana merancang suatu sistem pakar yang dapat digunakan untuk mendiagnosa suatu jenis penyakit berdasarkan gejala yang dirasakan *user*, sehingga *user* menemukan solusi atas permasalahan yang dihadapi. Adapun diagram konsep sistem pakar untuk diagnosa penyakit anak adalah sebagai berikut:



Gambar 1.1 Diagram Konsep Sistem Pakar Untuk Diagnosa Penyakit Anak

1.3. Batasan Masalah

Batasan-batasan masalah yang digunakan dalam tugas akhir ini adalah:

1. Sistem pakar yang akan dirancang untuk komputer PC (*stand alone*).
2. Data-data penunjang penyakit yang digunakan hanya pada usia balita (kurang dari 5 tahun) saja.
3. Menggunakan metode inferensi *forward chaining* untuk penarikan kesimpulan.
4. Interaksi antara sistem dan *user* menggunakan pertanyaan berupa daftar gejala yang sudah tampak berdasarkan kondisi fisik, kulit dan perilaku anak, dimana *user* akan diminta untuk memilih gejala pada setiap daftar gejala berdasarkan kondisi anak tersebut.
5. Jenis penyakit yang didiagnosa hanya penyakit yang umum terjadi pada anak di daerah tropis, seperti Indonesia.
6. Tidak terdapat komplikasi diantara gejala-gejala yang diderita.
7. Output yang dihasilkan dari software ini adalah jenis penyakit anak.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah membuat program sistem pakar dalam mendiagnosa penyakit anak yang dapat dikembangkan lebih lanjut dan memberikan kemudahan bagi pemakainya.

1.5. Manfaat Penelitian

Manfaat dari penulisan tugas akhir ini adalah:

1. Untuk memberikan kemudahan bagi orang awam maupun pakar sehingga dapat lebih memudahkan dalam mendapatkan penanganan lebih dini pada gangguan kesehatan anak.
2. Bagi ibu rumah tangga atau orang tua selaku *user* dapat menggunakan sistem ini untuk mengetahui jenis penyakit anak berdasarkan gejala-gejala yang ada.

1.6. Metodologi Penelitian

Penelitian ini dilakukan dengan beberapa tahapan, yaitu:

1. Pembelajaran Literatur

Metode ini dilaksanakan dengan melakukan studi kepustakaan melalui membaca buku-buku maupun artikel-artikel yang dapat mendukung penulisan Tugas Akhir.

2. Analisis

Pada tahap ini dilakukan pengumpulan fakta-fakta yang mendukung perancangan sistem dengan mengadakan konsultasi dengan seorang pakar (dokter spesialis anak) dan membandingkan hasil penelitian dengan yang ada pada buku penuntun.

3. Perancangan

Pada tahap ini akan dilakukan perancangan sistem pakar untuk diagnosa penyakit anak.

4. Pengkodean

Pada tahap ini rancangan yang akan dibuat dan diimplementasikan ke dalam bentuk kode program Visual Basic.

5. Pengujian

Setelah proses pengkodean selesai maka akan dilakukan proses pengujian terhadap program yang dihasilkan untuk mengetahui apakah program sudah berjalan dengan benar dan sesuai dengan perancangan yang dilakukan.

6. Penyusunan laporan dan kesimpulan akhir

Membuat laporan hasil analisa dan perancangan ke dalam format penulisan tugas akhir dengan disertai kesimpulan akhir.

1.7. Sistematika Penulisan

Penulisan tugas akhir ini disajikan dengan sistematika sebagai berikut:

Bab 1 : Pendahuluan

Berisi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian dan sistematika penulisan.

Bab 2 : Tinjauan Teori

Pada bab ini dijelaskan teori yang mendukung dalam perancangan sistem pakar untuk diagnosa penyakit anak.

Bab 3 : Analisis dan Perancangan Sistem

Menjabarkan tentang penyakit anak berupa nama penyakit, gejala klinis dan penyebabnya serta tahapan-tahapan dalam merancang program sistem pakar.

Bab 4 : Implementasi Sistem

Pada bab ini dijelaskan tentang implementasi dari perancangan sistem pakar yang telah dirancang pada bab sebelumnya.

Bab 5 : Kesimpulan dan Saran

Pada bab ini dijelaskan tentang kesimpulan dan saran dari penulis untuk hasil pembahasan tugas akhir.



BAB 2

TINJAUAN TEORI

2.1. Kecerdasan Buatan

Kecerdasan buatan (*Artificial Intelligence*) adalah “ide-ide untuk membuat suatu perangkat lunak komputer yang memiliki kecerdasan sehingga perangkat lunak komputer tersebut dapat melakukan suatu pekerjaan yang dilakukan oleh manusia” (Artanti, 2004), dengan kata lain membuat sebuah komputer dapat berpikir dan bernalar seperti manusia. Tujuan dari kecerdasan buatan ini adalah membuat komputer lebih cerdas, mengerti tentang kecerdasan, dan membuat mesin lebih berguna bagi manusia. Kecerdasan buatan dapat membantu meringankan beban kerja manusia misalnya dalam membuat keputusan, mencari informasi secara lebih akurat, atau membuat komputer lebih mudah digunakan dengan tampilan yang mudah dipahami. Cara kerja kecerdasan buatan adalah menerima input, untuk kemudian diproses dan kemudian mengeluarkan output yang berupa suatu keputusan.

Kecerdasan buatan memungkinkan komputer untuk berpikir dengan cara menyederhanakan program. Dengan cara ini, kecerdasan buatan dapat menirukan proses belajar manusia sehingga informasi baru dapat diserap dan digunakan sebagai acuan di masa-masa mendatang.

Kecerdasan atau kepandaian itu didapat berdasarkan pengetahuan dan pengalaman, untuk itu agar perangkat lunak yang dikembangkan dapat mempunyai kecerdasan maka perangkat lunak tersebut harus diberi suatu pengetahuan dan kemampuan untuk menalar dari pengetahuan yang telah didapat dalam menemukan solusi atau kesimpulan layaknya seorang pakar dalam bidang tertentu yang bersifat spesifik.

Kecerdasan buatan menawarkan media dan uji teori kecerdasan. Teori ini dapat dinyatakan dalam bahasa program komputer dan dibuktikan melalui eksekusinya pada komputer nyata.

Implementasi dari kecerdasan buatan saat ini dapat ditemui dalam bidang-bidang antara lain:

1. *Fuzzy logic*: suatu metode kecerdasan buatan yang banyak terdapat pada alat elektronik dan robot. Dimana alat-alat elektronik atau robot tersebut mampu berpikir dan bertindak laku sebagaimana layaknya manusia.
2. *Computer vision*: suatu metode kecerdasan buatan yang memungkinkan sebuah sistem komputer mengenali gambar sebagai inputnya. Contohnya adalah mengenali dan membaca tulisan yang ada gambarnya.
3. *Artificial Artificial intelligence dalam game*: suatu metode kecerdasan buatan yang berguna untuk meniru cara berpikir seorang manusia dalam bermain game. Contohnya adalah program *Deep Blue* yang mampu berpikir setara dengan seorang *grandmaster* catur.
4. *Speech recognition*: suatu metode kecerdasan buatan yang berguna untuk mengenali suara manusia dengan cara dicocokkan dengan acuan atau *pattern* yang telah diprogramkan sebelumnya. Contohnya adalah suara dari *user* dapat diterjemahkan menjadi sebuah perintah bagi komputer.
5. *Expert system*: suatu metode kecerdasan buatan yang berguna untuk meniru cara berpikir dan penalaran seorang ahli dalam mengambil keputusan berdasarkan situasi yang ada.

2.2. Sistem Pakar

Sistem pakar (*expert system*) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer yang dirancang untuk menyelesaikan masalah seperti layaknya seorang pakar.

Menurut Turban (2005), keahlian dipindahkan dari pakar ke suatu komputer. Pengetahuan ini kemudian disimpan di dalam komputer. Pada saat pengguna menjalankan komputer untuk mendapatkan informasi, sistem pakar menanyakan fakta-fakta dan dapat membuat penalaran (*inferensi*) dan sampai pada suatu kesimpulan. Kemudian, sistem pakar memberikan penjelasan (memberikan kesimpulan atas hasil konsultasi yang telah dilakukan sebelumnya).

Dengan sistem pakar ini, orang awam pun dapat menyelesaikan masalahnya atau hanya sekedar mencari suatu informasi berkualitas yang sebenarnya hanya dapat diperoleh dengan bantuan para ahli dibidangnya. Sistem pakar ini juga dapat membantu aktivitas para pakar sebagai asisten yang mempunyai pengetahuan yang dibutuhkan.

Pada dasarnya sistem pakar diterapkan untuk mendukung aktivitas pemecahan masalah. Beberapa aktivitas pemecahan yang dimaksud seperti pembuatan keputusan (*decision making*), pemanduan pengetahuan (*knowledge fusing*), pembuatan desain (*designing*), perencanaan (*planning*), prakiraan (*forescating*), pengaturan (*regulating*), pengendalian (*controlling*), diagnosa (*diagnosing*), perumusan (*prescribing*), penjelasan (*explaining*), pemberian nasihat (*advising*) dan pelatihan (*tutoring*) (Kusrini, 2006).

2.2.1. Sejarah Sistem Pakar

Sistem pakar mulai dikembangkan pada pertengahan tahun 1960-an oleh *Artificial Intelligence Corporation*. Periode penelitian kecerdasan buatan ini didominasi oleh suatu keyakinan bahwa nalar yang digabung dengan komputer canggih akan

menghasilkan presatasi pakar atau bahkan manusia super. Suatu usaha ke arah ini adalah *General Purpose Problem Solver* (GPS) yang dikembangkan oleh Allen Newell, John Cliff Shaw, dan Herbert Alexander Simon. GPS merupakan sebuah percobaan untuk menciptakan mesin yang cerdas.

Sistem pakar untuk melakukan diagnosa kesehatan telah dikembangkan sejak pertengahan tahun 1970 yang untuk pertama kali dibuat oleh Bruce Buchanan dan Edward Shortliffe di Standford University diberi nama MYCIN. MYCIN merupakan program interaktif yang melakukan diagnosa penyakit meningitis dan infeksi bacremia serta memberikan rekomendasi terapi antimikrobia. MYCIN mampu memberikan penjelasan atas penalarannya secara detail. Dalam uji coba, program ini mampu menunjukkan kemampuan seperti seorang spesialis.

2.2.2. Ciri-Ciri Sistem Pakar

Adapun ciri-ciri sistem pakar seperti:

1. Mudah dimodifikasi, yaitu dengan menambah atau menghapus suatu pengetahuan dari basis pengetahuannya.
2. Memiliki kemampuan untuk beradaptasi.
3. Terbatas pada bidang spesifik.
4. *Output* tergantung dialog dengan pengguna (*user*).
5. *Knowledge base* dan inferensi terpisah.

2.2.3. Arsitektur Sistem Pakar

Sistem pakar terdiri dari dua bagian utama, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). *Development Environment* dipakai oleh pembangun sistem pakar untuk membangun komponen-komponen dan mengenalkan suatu pengetahuan kepada *knowledge base*. *Consultation Environment* dipakai oleh user untuk mendapatkan suatu pengetahuan yang berhubungan dengan suatu keahlian (Setiawan, 2003).

Komponen-komponen yang biasanya terdapat dalam sebuah sistem pakar terdiri dari:

1. Antarmuka pengguna (*user interface*)

Pada komponen ini terjadi dialog antara program dan *user*, dimana sistem menerima *input* berupa informasi dan instruksi dari *user*, dan sistem memberikan *output* berupa informasi kepada *user*.

2. Basis pengetahuan (*knowledge base*)

Basis pengetahuan dapat dikatakan sebagai kumpulan informasi dan pengalaman seorang ahli pada suatu bidang tertentu.

3. Akuisisi pengetahuan (*knowledge acquisition*)

Akuisisi pengetahuan merupakan tranformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan kedalam program komputer.

4. Mesin inferensi

Mesin inferensi merupakan otak dari sistem pakar yang mengandung mekanisme fungsi berpikir dan pola-pola penalaran sistem yang digunakan oleh seorang pakar. Mesin inferensi bertindak sebagai penarik kesimpulan dan mengontrol mekanisme dari sistem pakar.

5. Memori kerja (*working memory*)

Memori kerja merupakan tempat penyimpanan fakta-fakta yang diketahui dari hasil menjawab pertanyaan.

6. Subsistem penjelasan (*explanation subsystem*)

Komponen ini merupakan komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Komponen ini menggambarkan penalaran sistem kepada pemakai dengan cara menjawab pertanyaan-pertanyaan.

7. Perbaikan pengetahuan

Pakar memiliki kemampuan untuk menganalisa dan meningkatkan kinerjanya serta kemampuan untuk belajar dari kinerjanya.

2.2.4. Representasi Pengetahuan

Representasi pengetahuan adalah suatu teknik untuk merepresntasikan basis pengetahuan yang diperoleh kedalam suatu skema/diagram tertentu sehingga dapat diketahui relasi/keterhubungan antara suatu data dengan data lainnya (Natalia, 2006).

Pengetahuan dapat direpresentasikan dalam berbagai model, beberapa model representasi pengetahuan seperti:

1. Logika

Logika merupakan suatu pengkajian ilmiah tentang serangkaian penalaran, sistem kaidah, dan prosedur yang membantu proses penalaran. Bentuk logika komputasional ada 2 macam, yaitu:

1.1. Logika Proposional

Logika proposional merupakan suatu pernyataan yang menyatakan benar (TRUE) atau salah (FALSE) yang dihubungkan dengan menggunakan operator logika seperti konjungsi (*AND*), disjungsi (*OR*), negasi (*NOT*), implikasi/kondisional (*If...Then*), equivalensi/bikondisional (*If and only If*).

Berikut adalah contoh model representasi pengetahuan logika proposional:

Jika hujan turun sekarang maka saya tidak akan ke pasar

Dapat ditulis dalam bentuk $p \rightarrow q$

1.2. Logika Predikat

Logika predikat merupakan suatu logika yang seluruhnya menggunakan konsep dan kaidah proporsional yang sama, disebut juga kalkulus predikat, yang memberi tambahan kemampuan untuk merepresentasikan pengetahuan dengan sangat cermat dan rinci.

Berikut adalah contoh model representasi pengetahuan logika predikat:

$x = \text{Rojali}$

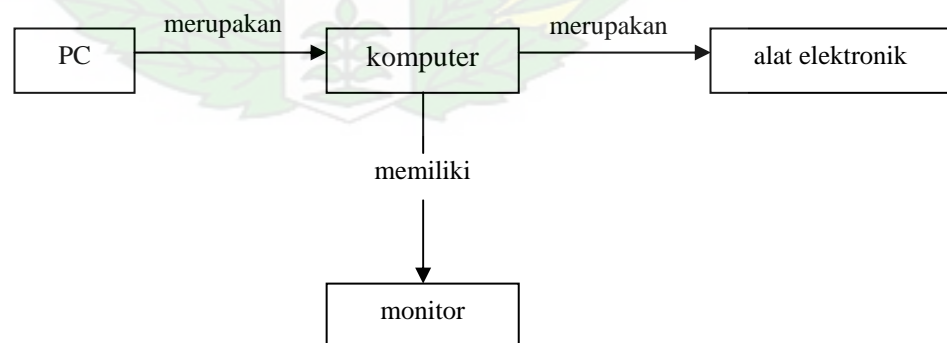
$y = \text{Juleha}$

Dapat ditulis dalam bentuk $\text{Suka}(x,y)$

2. Jaringan Semantik

Jaringan semantik merupakan representasi yang menggambarkan grafis dari pengetahuan yang memperlihatkan hubungan hirarkis dari objek-objek yang terdiri atas simpul (*node*) dan penghubung (*link*).

Berikut adalah contoh gambar model representasi pengetahuan jaringan semantik:



Gambar 2.1 Representasi Jaringan Semantik

3. *Object-Atributte-Value* (OAV)

Object dapat berupa bentuk fisik atau konsep. *Attribute* adalah karakteristik atau sifat dari object tersebut. *Value* (nilai) - besaran spesifik dari attribute tersebut yang berupa numerik, string atau boolean.

Berikut adalah contoh tabel model representasi pengetahuan *Object-Atributte-Value* (OAV):

Tabel 2.1 Representasi OAV

Object	Attribute	Value
Mangga	Warna	Hijau, Orange
Mangga	Biji	Tunggal
Mangga	Rasa	Asam, Manis
Mangga	Bentuk	Oval
Pisang	Warna	Hijau, Kuning
Pisang	Bentuk	Lonjong

4. Bingkai (*Frame*)

Bingkai berupa ruang (*slots*) yang berisi atribut untuk mendeskripsikan pengetahuan yang berupa kejadian, lokasi, situasi ataupun elemen-elemen lain.

Berikut adalah contoh tabel model representasi pengetahuan bingkai (*frame*):

Tabel 2.2 Representasi Bingkai Pada Bingkai Penyakit

Ruang (<i>slots</i>)	Isi (<i>fillers</i>)
Nama	Flu
Gejala	a. Bersin b. Pusing c. Demam
Obat	a. Ultraflu b. Mixagrib

5. Kaidah Produksi

Kaidah menyediakan menyediakan cara formal untuk merepresentasikan rekomendasi, arahan, atau strategi dalam bentuk jika-maka (*if-then*) yang menghubungkan anteseden dengan konsekuensi.

Berikut adalah contoh model representasi pengetahuan kaidah produksi:

JIKA bersin

DAN pusing

DAN demam

MAKA terserang penyakit flu

2.2.5. Metode Inferensi

Metode inferensi merupakan suatu cara penarikan kesimpulan yang dilakukan oleh mesin inferensi untuk menyelesaikan masalah. Ada dua metode inferensi yang umum dalam sistem pakar, yaitu:

1. *Forward Chaining* (Runut Maju)

Forward chaining adalah suatu strategi pengambilan keputusan yang dimulai dari bagian premis (fakta) menuju konklusi (kesimpulan akhir) (Kusrini, 2006). Metode inferensi ini yang akan digunakan dalam sistem pakar yang akan dibangun dengan contoh penalaran sebagai berikut:

IF Sulit bernafas

AND Pilek

AND Batuk kering

AND Tarikan nafas berbunyi kasar dan penghembusan nafas berbunyi mengi

THEN Laringitis

Secara sederhana dapat dijelaskan bahwa untuk kaidah diatas, agar sistem mencapai konklusi, harus diinput terlebih dahulu fakta sulit bernafas, pilek, tarikan nafas berbunyi kasar dan penghembusan nafas berbunyi mengi. Baru sistem dapat mengeluarkan konklusi bahwa penyakit yang diderita adalah laringitis.

2. *Backward Chaining* (Runut Balik)

Backward chaining adalah suatu strategi pengambilan keputusan dimulai dari pencarian solusi dari kesimpulan kemudian menelusuri fakta-fakta yang ada hingga menemukan solusi yang sesuai dengan fakta-fakta yang diberikan pengguna (Kusrini, 2006). Contoh penalaran *backward chaining* adalah:

Lampu 1 rusak,

IF Lampu 1 dinyalakan

AND Lampu 1 tidak menyala

AND Lampu 1 dihubungkan dengan sekering

AND sekering masih utuh

Secara sederhana dapat dijelaskan bahwa untuk kaidah diatas, sistem terlebih dahulu menduga bahwa lampu 1 rusak. Kebenaran praduga ini dibuktikan dengan kebenaran fakta lampu 1 tidak menyala, lampu 1 dihubungkan dengan sekering dan sekering masih utuh. Kemudian sistem mengeluarkan kesimpulan bahwa lampu 1 rusak. Namun apabila ada fakta tidak terpenuhi berarti praduga sistem salah, selanjutnya sistem akan mengecek konklusi berikutnya.

2.3. **Pediatri**

Pediatri berasal dari bahasa Yunani, yaitu *pedos* yang berarti anak dan *iatria* yang berarti pengobatan. Arti bahasa Indonesia yang sebenarnya ialah Ilmu Pengobatan Anak dan pengertian ini lebih tepat daripada Ilmu Penyakit Anak yang ternyata masih sering dipakai (Gilbert, 1986).

Pediatri telah berkembang pesat sekali terutama dalam 20 tahun terakhir ini. Di luar negeri, seperti pula dianjurkan oleh *World Health Organization* (WHO) timbul kecendrungan mengubah nama Pediatri menjadi *Child Health*. Di Indonesia sejak 1963 telah diubah menjadi Ilmu Kesehatan Anak, yaitu karena Pediatri sekarang tidak hanya mengobati anak sakit, tetapi mencakup hal-hal yang lebih luas (Staf Pengajar IKA FK UI, 1985). Ruang lingkup Pediatri yang akan dibahas disini berdasarkan pembagian ilmunya, yaitu pediatri pada penyakit tropis dan infeksi yang dapat dilihat

pada Gambar 2.2. Penyakit tropis adalah penyakit yang terjadi di negara-negara yang mengalami dua iklim, yaitu panas dan kemarau (hujan) seperti Indonesia.



Gambar 2.2 Diagram Penyakit Anak

2.4. Visual Basic 6.0

Visual Basic merupakan bahasa pemrograman yang secara cepat dan mudah dapat digunakan untuk membuat aplikasi pada Microsoft Windows. Beberapa keuntungan menggunakan Visual Basic 6.0 diantaranya adalah:

1. Visual Basic dapat menangani bermacam-macam format database seperti format database Microsoft Access, Microsoft Excel, dan FoxPro. Dengan database jumlah besar dan akses yang cepat.
2. Mudah digunakan karena dasar pembuatan visual basic adalah form, sehingga pengguna lebih mudah berkreasi dalam membuat program aplikasi.

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1. Analisis Penyakit

Untuk mendiagnosa suatu penyakit perlu diketahui terlebih dahulu gejala-gejala yang ditimbulkan. Meskipun hanya dari gejala klinis (gejala-gejala yang terlihat langsung maupun yang dirasakan oleh penderita), dokter dapat mengambil suatu kesimpulan berupa penyakit yang diderita. Tetapi ada kalanya diperlukan pemeriksaan lebih lanjut melalui pemeriksaan laboratorium untuk penyakit tertentu.

Berdasarkan penelitian dan seminar yang dilakukan oleh dr. Hj. Nurdiani, Sp.A dari RSU dr. Joelham Binjai (Purnamawati, 2007), terdapat beberapa jenis penyakit yang sekarang ini sering terjadi pada balita Indonesia, yaitu:

1. Diare (*gastroenteritis*).
2. ISPA (Infeksi Saluran Pernafasan Akut).
3. Penyakit kulit.
4. TBC.
5. Meningitis.
6. Kelainan darah.
7. HIV (*Human Immunodeficiency Virus*).

8. DBD (Demam Berdarah Dengue).

9. Flu burung.

Namun berdasarkan urutan teratas tersebut, hanya beberapa jenis penyakit beserta pembagiannya yang dapat di diagnosa secara klinis seperti:

1. Diare (*gastroenteritis*)

2. ISPA (Infeksi Saluran Pernafasan Akut)

Terbagi atas *tonsilitis*, *laringitis*, *bronkhitis*, *bronkiolitis* dan *pneumonia*.

3. Penyakit kulit

Terbagi atas *dermatitis atopik*, *mikosis superfisialis* dan *monialisis*.

4. Meningitis

3.1.1. Diare (Gastroenteritis)

Penyakit diare disebabkan oleh beberapa faktor, beberapa diantaranya yaitu:

1. Infeksi

Infeksi dengan berbagai bakteri di usus. Ini dapat terjadi karena infeksi oleh organisme disentri basiler, bakteri *salmonella*, dan berbagai virus. Infeksi hebat oleh parasit malaria tertiana.

2. Diet

Serangan diare dapat terjadi karena memakan terlalu banyak bahan makan yang sulit dicerna, seperti kacang atau cabai.

3. Gizi

Diare terjadi pada keadaan kekurangan gizi karena gangguan pencernaan dan penyerapan makan di usus.

Diare memiliki gejala klinis seperti cengeng, gelisah, demam, nafsu makan berkurang, tinja dengan ciri-ciri seperti cair dan mungkin disertai lendir dan atau darah, muntah dan dehidrasi dengan ciri-ciri seperti turgor (kelenturan) kulit berkurang, mata cekung, ubun-ubun cekung, mulut merah dan kering.

Tindakan sendiri untuk mencegah serangan lanjutan seperti menghindari makanan berbumbu dan lakukan pencegahan dehidrasi dengan pemberian larutan oralit. Pada bayi ASI, pemberian larutan oralit perlu dikurangi setelah bayi mulai membaik. Pada bayi susu formula, berikan larutan oralit hanya di hari pertama, kemudian setengah larutan oralit dan setengah porsi susu di hari kedua. Kemudian kembali ke porsi susu formula normal 24 jam kemudian. Larutan oralit harus disiapkan dengan air matang dingin dengan melarutkan gula sebanyak 2 sendok teh peres ke dalam 200 ml air matang. Bila suatu saat timbul diare, segera berikan lagi larutan oralit dan hubungi dokter anda. Segera bawa ke rumahsakit bila:

1. Mengantuk yang tidak wajar.
2. Mata cekung dan atau lidah kering.
3. Tinja menghitam atau bernoda darah.
4. Menolak disusui lebih dari 3 jam (pada bayi di bawah 3 bulan) atau lebih dari 6 jam (pada bayi usia di atas 3 bulan). Penanggulangan yang

3.1.2. ISPA (Infeksi Saluran Pernapasan Akut)

ISPA merupakan penyakit infeksi saluran nafas yang secara anatomi dibedakan atas saluran nafas atas mulai dari hidung sampai dengan faring dan saluran nafas bawah

mulai dari laring sampai dengan alveoli, akibat invasi *infecting agents* yang mengakibatkan reaksi inflamasi saluran nafas yang terlibat. Infeksi tersebut tersebut disebabkan oleh virus dan bakteri, seperti bakteri *streptococcus*.

Berdasarkan klasifikasi anatomi dibedakan atas:

1. Infeksi saluran napas atas terbagi atas *rhinitis, faringitis, tonsilitis*.
2. Infeksi saluran napas bawah terbagi atas *laringitis, trakheitis, bronkhitis, bronkiolitis, pneumonia, abses pulmonum* dan *empiema*.

Namun dari beberapa klasifikasi tersebut hanya *tonsilitis, laringitis, bronkhitis, bronkiolitis* dan *pneumonia* yang dapat di diagnosa secara klinis.

3.1.2.1.Tonsilitis

Tonsilitis memiliki gejala klinis seperti sakit tenggorokan, demam, muntah, sakit perut dan susah menelan.

Tindakan sendiri untuk mencegah serangan lanjutan seperti meredakan tenggorokan nyeri dengan memberikan minuman non-asam, misalnya susu sebanyak yang dia mau menggunakan sedotan serta memberikan ibuprofen atau parasetamol cair secara teratur. Bila anak sudah cukup besar, ajarkan dia berkumur dengan air garam hangat. Periksakan ke dokter bila tenggorokan anak anda sangat serak atau tidak membaik dalam beberapa hari.

3.1.2.2.Laringitis

Biasa ditemukan pada umur 1 – 3 tahun, dengan gejala klinis seperti pilek, batuk kuat serta kering dan suara serak, dispnea (sesak napas), inspirasi (tarikan napas) berbunyi

stridor (kasar), demam dan terkadang berkeringat, ekspirasi (penghembusan napas) berbunyi wheezing (bising mengi) dan sianosis (kebiruan kulit) bibir.

Tindakan sendiri untuk mencegah serangan lanjutan seperti meredakan batuk dengan memberikan banyak minuman hangat. Baringkan diruangan sejuk atau lembab dan periksa irama nafas si anak dengan menghitung berapa kali ia bernafas setiap menit. Letakkan tangan anda diatas dada atau punggungnya dan si anak harus dalam keadaan tenang. Tingkat pernafasan maksimum anak usia di bawah 1 tahun antara 50-60 nafas/menit dan usia diatas 1 tahun 40 nafas/menit. Bila terjadi demam, turunkan temperaturnya dengan melepaskan pakaian si anak dan usaplah kepala dan tubuhnya dengan air hangat. Bila demam tidak mereda, berikan juga ibuprofen dengan dosis sesuai anjuran. Periksakan ke dokter bila tingkat pernafasan melebihi batas maksimum dan demam tidak juga turun. Segera bawa ke rumahsakit bila:

1. Bibir atau lidah membiru.
2. Mengantuk yang tidak wajar.
3. Tidak mampu bersuara atau berbicara.

3.1.2.3.Bronkhitis

Penderita terbanyak pada umur 1 – 3 tahun, dengan gejala klinis seperti pilek, demam, batuk kering kemudian menjadi berdahak, ekspirasi (penghembusan napas) berbunyi wheezing (bising mengi), pernapasan menjadi cepat dan sianosis (kebiruan kulit) bibir dan lidah.

Tindakan untuk mencegah serangan lanjutan adalah membawanya ke dokter sekarang juga. Dokter akan memeriksa dan mungkin akan merujuk ke rumahsakit untuk menjalani beberapa tes dan dipantau.

3.1.2.4. Bronkiolitis

Sering diderita anak yang berumur 1 – 6 bulan, dengan gejala klinis seperti pilek, demam, dispnea (sesak napas), pernapasan dangkal dan cepat, batuk kering, gelisah, sianosis (kebiruan kulit) bibir dan ekspirasi (penghembusan napas) berbunyi wheezing (bising mengi).

Tindakan untuk mencegah serangan lanjutan adalah membawanya ke dokter sekarang juga. Dokter akan memeriksa dan mungkin akan merujuk ke rumah sakit untuk menjalani beberapa tes dan dipantau.

3.1.2.5. Pneumonia

Biasa ditemukan pada umur 6 bulan – 9 tahun, dengan gejala klinis seperti batuk, demam, kejang, dispnea (sesak napas), membuka hidung lebar-lebar pada saat menarik napas karena sulit bernapas dan sianosis (kebiruan kulit) hidung dan bibir.

Tindakan untuk mencegah serangan lanjutan adalah membawanya ke dokter sekarang juga. Dokter akan memeriksa dan mungkin akan merujuk ke rumah sakit untuk menjalani beberapa tes dan dipantau.

3.1.3. Penyakit Kulit

Penyakit kulit sering sekali dijumpai pada balita dengan berbagai penyebab seperti jamur, infestasi parasit, infeksi virus, alergi, kelainan bawaan, dan lain sebagainya. Sekarang ini, alergi dan jamur merupakan penyebab utama penyakit kulit balita Indonesia. Dimana alergi mengakibatkan dermatitis atopik dan jamur mengakibatkan mikosis superfisial dan moniliasis.

3.1.3.1.Dermatitis Atopik

Biasa terjadi pada pipi, tengkuk, lipatan siku dan lipatan lutut dengan gejala klinis seperti kulit gatal, kasar, bintil-bintil, eritema (warna merah pada kulit), erosi (luka pada kulit terluar yang mengeluarkan cairan darah), keropeng (cairan tubuh yang mengering pada kulit yang terluka), likenifikasi (penebalan kulit).

Tindakan sendiri untuk mencegah serangan lanjutan seperti bagi anak berkulit kering mandikan dia dengan air hangat dan minyak emolien (zat yang melembutkan kulit seperti krim larut air) yang dioleskan 2 kali sehari. Pastikan pakaiannya terbuat dari bahan katun dan jaga agar anak tidak menggaruk. Obat antihistamin yang dijual bebas dapat membantu mengontrol gatal yang parah. Jaga agar tidak terkena pilek, yang bisa menyebabkan infeksi virus yang serius. Periksakanlah ke dokter untuk penanganan lebih lanjut.

3.1.3.2.Mikosis Superfisialis

Biasa terjadi pada kepala, lipatan paha dan *nates* (bokong) dengan gejala klinis seperti kulit gatal, eritema (warna merah pada kulit), papula (penonjolan kecil) berupa sirsiner (menyerupai cincin), vesikel (lepuhan kecil berisi cairan), erosi (luka pada kulit terluar yang mengeluarkan cairan darah), keropeng (cairan tubuh yang mengering pada kulit yang terluka), likenifikasi (penebalan kulit).

Tindakan untuk mencegah serangan lanjutan adalah memeriksakan ke dokter karena tidak ada penanganan khusus untuk kondisi ini.

3.1.3.3.Moniliasis

Biasa terjadi pada lipatan-lipatan tubuh dan lekuk kulit dengan gejala klinis seperti terdapat eritema (warna merah pada kulit), vesikel (lepuhan kecil berisi cairan), memar.

Tindakan untuk mencegah serangan lanjutan adalah memeriksakan ke dokter karena tidak ada penanganan khusus untuk kondisi ini. Sambil menunggu bantuan medis, bila yang terkena infeksi adalah mulut dan terasa perih maka redakan nyerinya dengan menghindari minuman yang mengandung asam. Jaga agar si anak banyak minum untuk mencegah dehidrasi dan berikan sedotan agar cairan tidak mengenai bagian yang perih. Berhati-hatilah dalam menyikat gigi, namun tetap usahakan dalam 2 kali sehari.

3.1.4. Meningitis

Meningitis adalah suatu infeksi atau peradangan dari meninges (selaput otak), lapisan yang tipis atau encer yang mengepung otak dan jaringan saraf dalam tulang punggung, disebabkan oleh bakteri, virus, riketsia, atau protozoa, yang dapat terjadi secara akut dan kronis. Meningitis terbagi atas beberapa tipe, yaitu:

1. Meningitis kriptokokus

Meningitis kriptokokus adalah meningitis yang disebabkan oleh jamur kriptokokus. Jamur ini bisa masuk ke tubuh kita saat kita menghirup debu atau tahi burung yang kering. Kriptokokus ini dapat menginfeksi kulit, paru, dan bagian tubuh lain.

2. Meningitis virus (aseptik)

Meningitis virus adalah reaksi peradangan pada selaput otak yang dapat disebabkan berbagai virus. Antara lain virus herpes dan virus penyebab flu perut.

3. Bacterial meningitis (purulenta)

Meningitis yang bersifat akut yang disebabkan oleh bakteri tertentu, salah satu bakterinya adalah *meningococcal bacteria*.

4. Meningitis tuberkulosa (serosa)

Meningitis tuberkulosa adalah meningitis akibat komplikasi bakteri TBC.

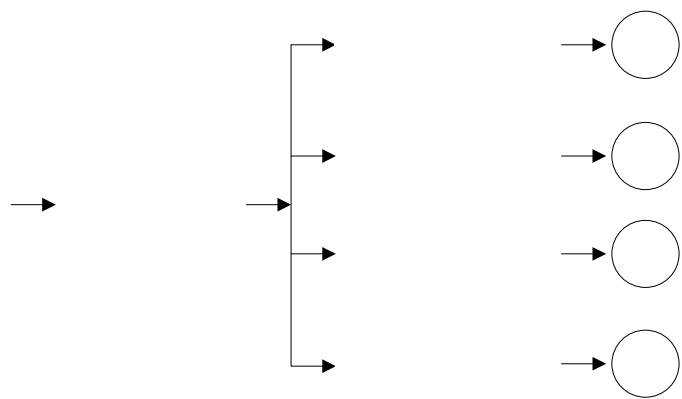
Untuk menentukan tipe meningitis dilakukan tes laboratorium terlebih dulu pada penderita. Namun pada umumnya, gejala klinis bagi balita yang terkena meningitis seperti muncul bercak berwarna biru kemerahan pada kulit, bentuk bervariasi dan jika ditekan tidak akan memudar. Tangisan lebih keras dan nadanya tinggi, demam, badan kaku, gangguan kesadaran seperti tangannya membuat gerakan tidak beraturan dan rewel.

Tindakan untuk mencegah serangan lanjutan adalah segera membawanya ke rumahsakit karena si anak akan memerlukan perawatan intensif. Sambil menunggu bantuan medis, lakukan tindakan menurunkan demam dengan melepaskan pakaian si anak dan berilah dia banyak minum air putih matang. Baringkan diruangan sejuk, dan usaplah kepala dan tubuhnya dengan air hangat.

3.2. Analisis Permasalahan

Sistem pakar merupakan sistem dengan basis pengetahuan yang dinamis. Dimana pengetahuan tersebut dapat berubah seiring 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. Perubahan hanya dilakukan pada bagian basis pengetahuan saja sehingga sistem pakar ini dapat dikembangkan lebih lanjut. Tahapan analisis terhadap suatu sistem dilakukan sebelum tahapan perancangan, hal ini agar perangkat lunak yang dirancang sesuai dengan masalah yang akan diselesaikan.

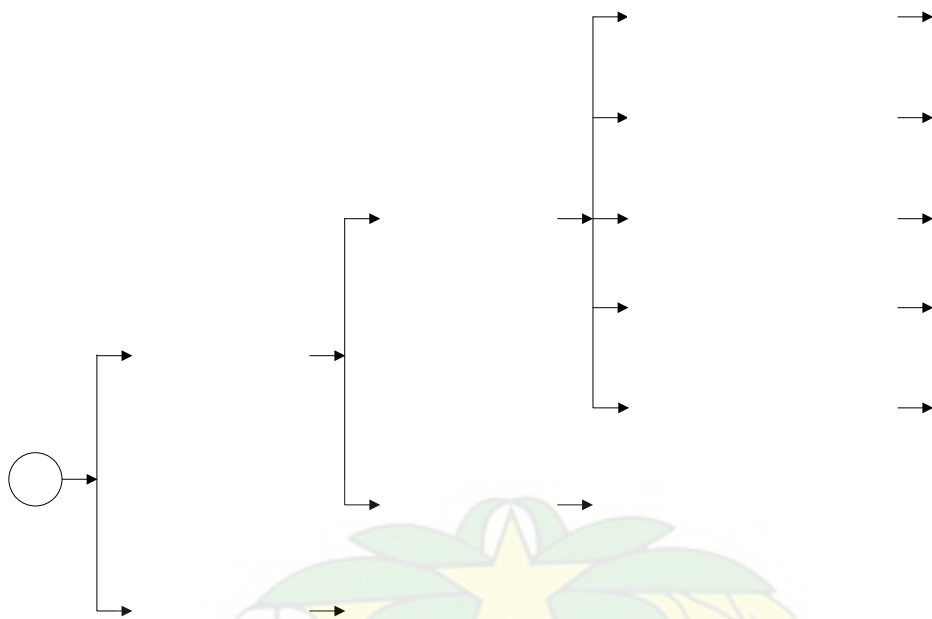
Suatu proses terhadap basis pengetahuan atau informasi yang didapat dari pakar, terlebih dahulu diubah kedalam bentuk diagram pohon keputusan, sehingga didalam penyelesaian masalah lebih mudah dilakukan penelusuran untuk mendapatkan solusi atau kesimpulan akhir yang terbaik. Adapun diagram pohon keputusan pada sistem ini adalah:



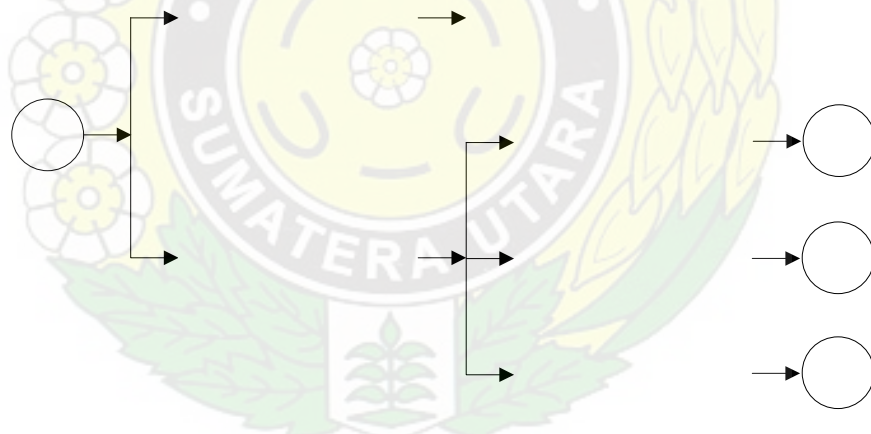
Gambar 3.1 Diagram Pohon Keputusan Diagnosa



Gambar 3.2 Diagram Pohon Keputusan I

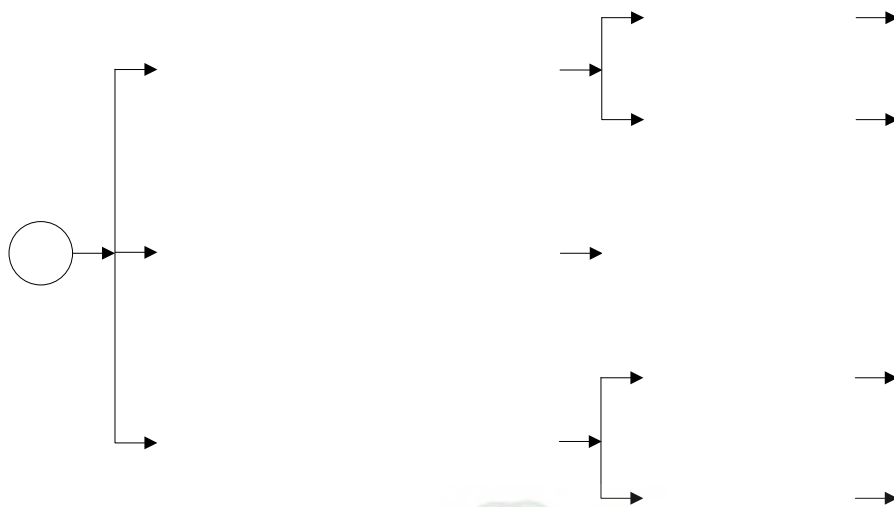


Gambar 3.3 Diagram Pohon Keputusan II

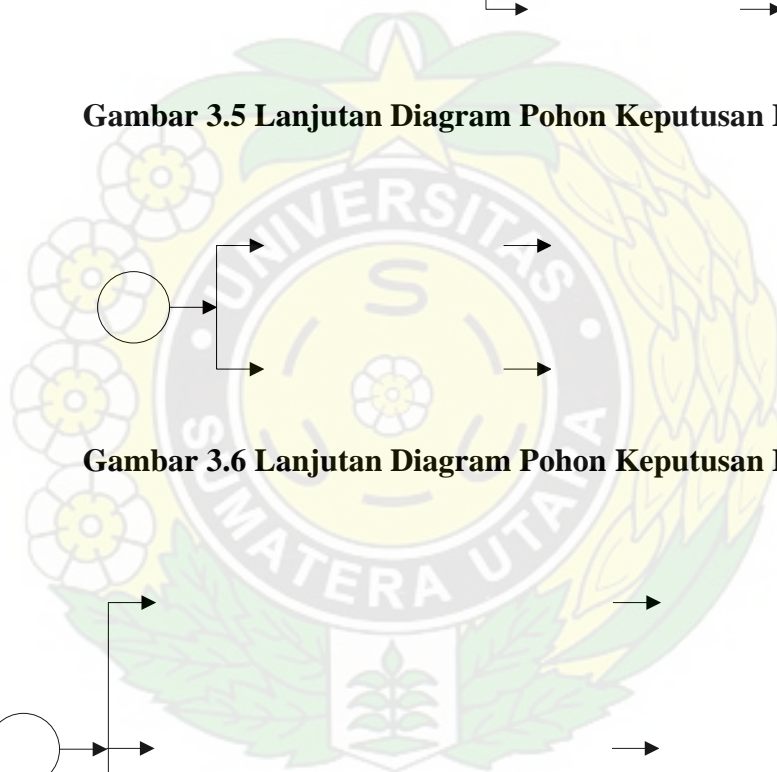


Gambar 3.4 Diagram Pohon Keputusan III

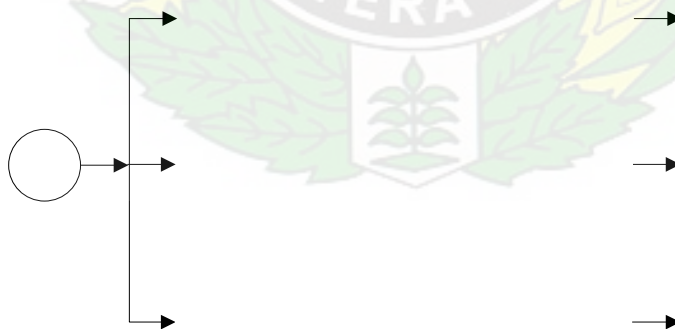
Sakit atau
tenggorok



Gambar 3.5 Lanjutan Diagram Pohon Keputusan III



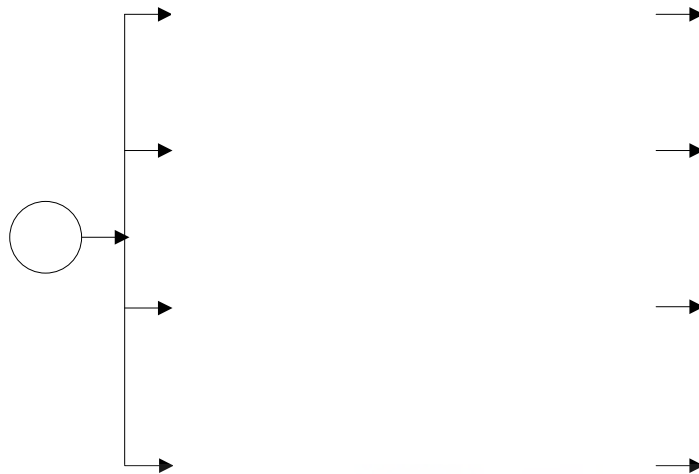
Gambar 3.6 Lanjutan Diagram Pohon Keputusan III



Gambar 3.7 Lanjutan Diagram Pohon Keputusan III

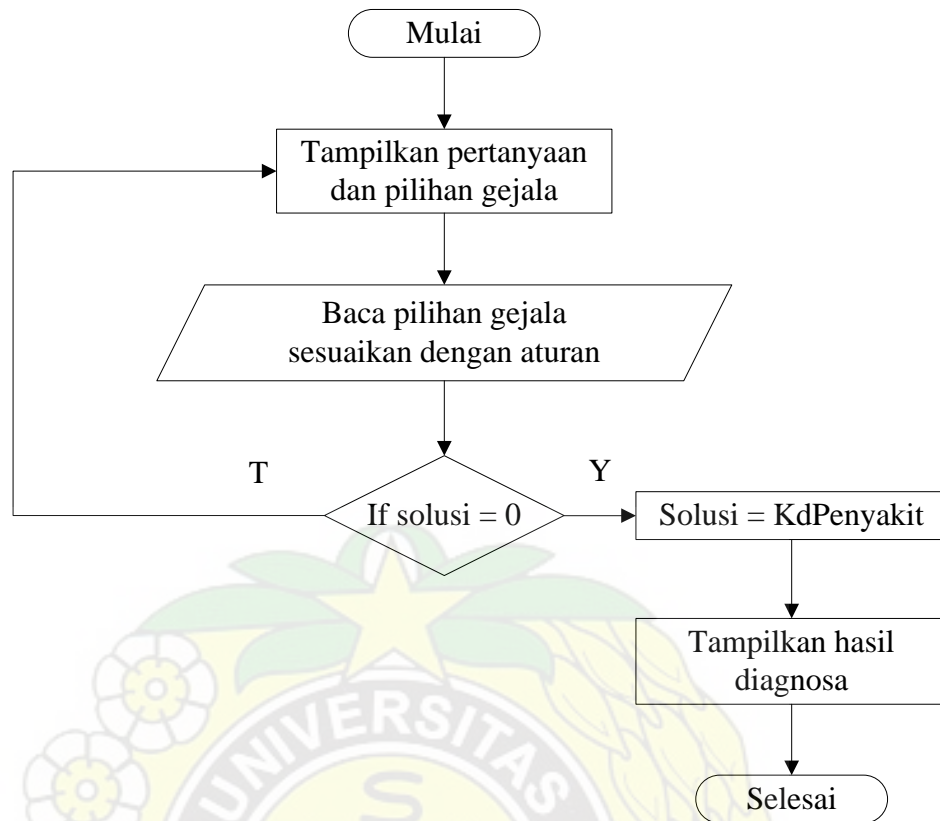
Tarikan
pengh

Pengh



Gambar 3.8 Diagram Pohon Keputusan IV

Proses yang terjadi pada sistem secara sederhana dapat dijelaskan, dimana pada proses diagnosa *user* akan dihadapkan dengan pertanyaan dan setiap pertanyaan terdapat pilihan-pilihan gejala yang ditampilkan oleh sistem. Setelah *user* memilih salah satu dari pilihan gejala yang telah disediakan, maka sistem akan membaca pilihan yang dimasukkan oleh *user* yang disesuaikan dengan aturan pada basis pengetahuan. Pada setiap aturan terdapat solusi berisi nilai 0 (solusi = 0) dan solusi berisi kode penyakit 001, 002, dan seterusnya (solusi = KdPenyakit, misal solusi = 007). Dimana solusi = 0, sistem akan terus membaca aturan selanjutnya dan menampilkan pertanyaan beserta pilihan gejala selanjutnya sesuai aturan (sistem terus melakukan proses diagnosa). Sedangkan solusi = KdPenyakit, sistem berhenti menampilkan pertanyaan beserta pilihan gejala (berhenti melakukan proses tanya jawab atau diagnosa) dan menampilkan hasil diagnosa berupa nama penyakit dan pilihan gejala *user* pada proses diagnosa. Berdasarkan solusi pada aturan tersebut maka sistem dapat memutuskan apakah sistem berhenti melakukan proses diagnosa atau lanjut melakukan proses diagnosa. Sistem berhenti melakukan proses diagnosa jika menemukan solusi = KdPenyakit. Sistem lanjut melakukan proses diagnosa jika menemukan solusi = 0 dan berhenti jika menemukan solusi = KdPenyakit pada aturan berikutnya. Proses yang terjadi pada sistem tersebut dapat dilihat pada gambar 3.9 diagram flowchart sistem:

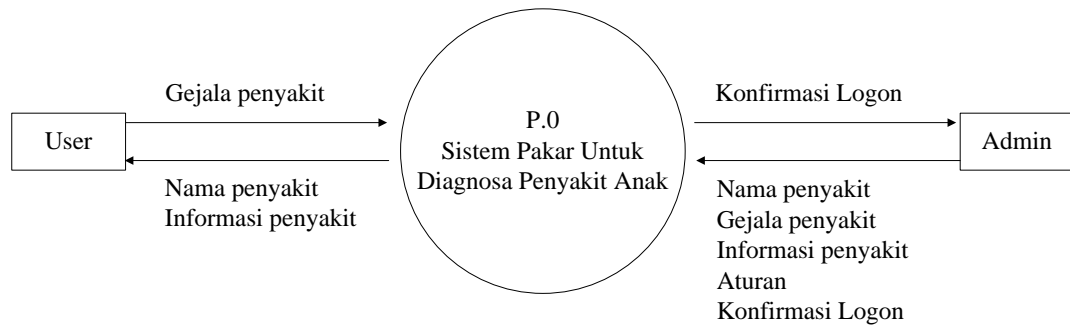


Gambar 3.9 Diagram Flowchart Sistem

3.3. Model Analisis

Model representasi aliran proses perangkat lunak yang akan dirancang akan disajikan dalam *Data Flow Diagram* (DFD). DFD digunakan untuk menggambarkan aliran informasi dan proses data yang bergerak dari *input* data hingga *output*. DFD memudahkan pemakai yang kurang menguasai bidang komputer untuk mengerti sistem yang akan dikerjakan atau dikembangkan.

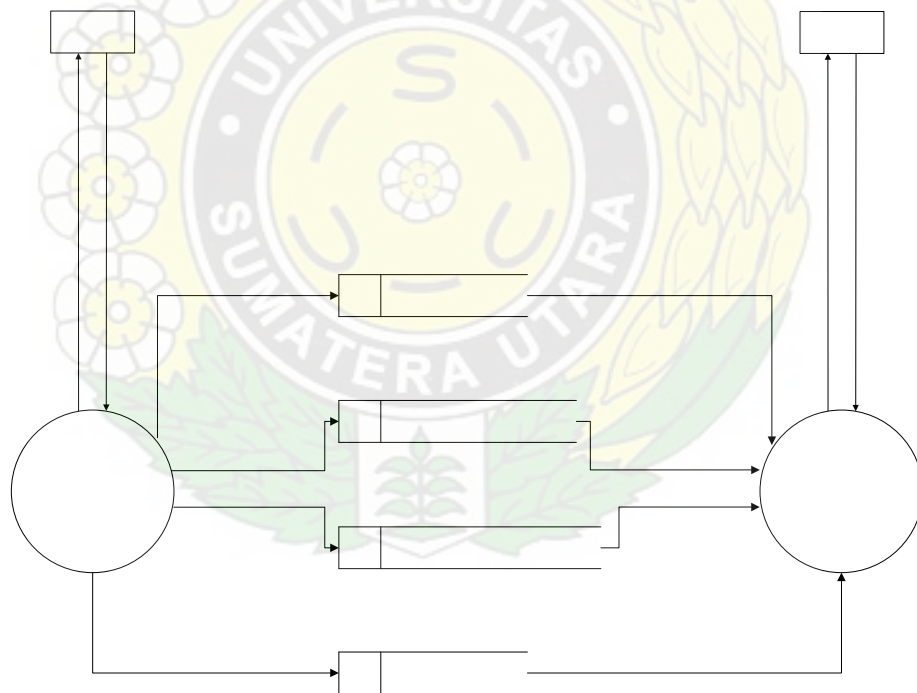
DFD Level 0 terdiri dari 2 (dua) *entity* yaitu Pakar (*Admin*) dan Pengguna (*User*). *Admin* memberikan *input* ke sistem berupa data nama penyakit, gejala penyakit serta informasi tentang penyakit. *User* dalam mendiagnosa penyakit memberikan gejala-gejala kepada sistem dan selanjutnya sistem memberikan diagnosa yang sesuai.



Gambar 3.10 DFD Level 0

3.3.1. Data Flow Diagram Level 1 (DFD Level 1)

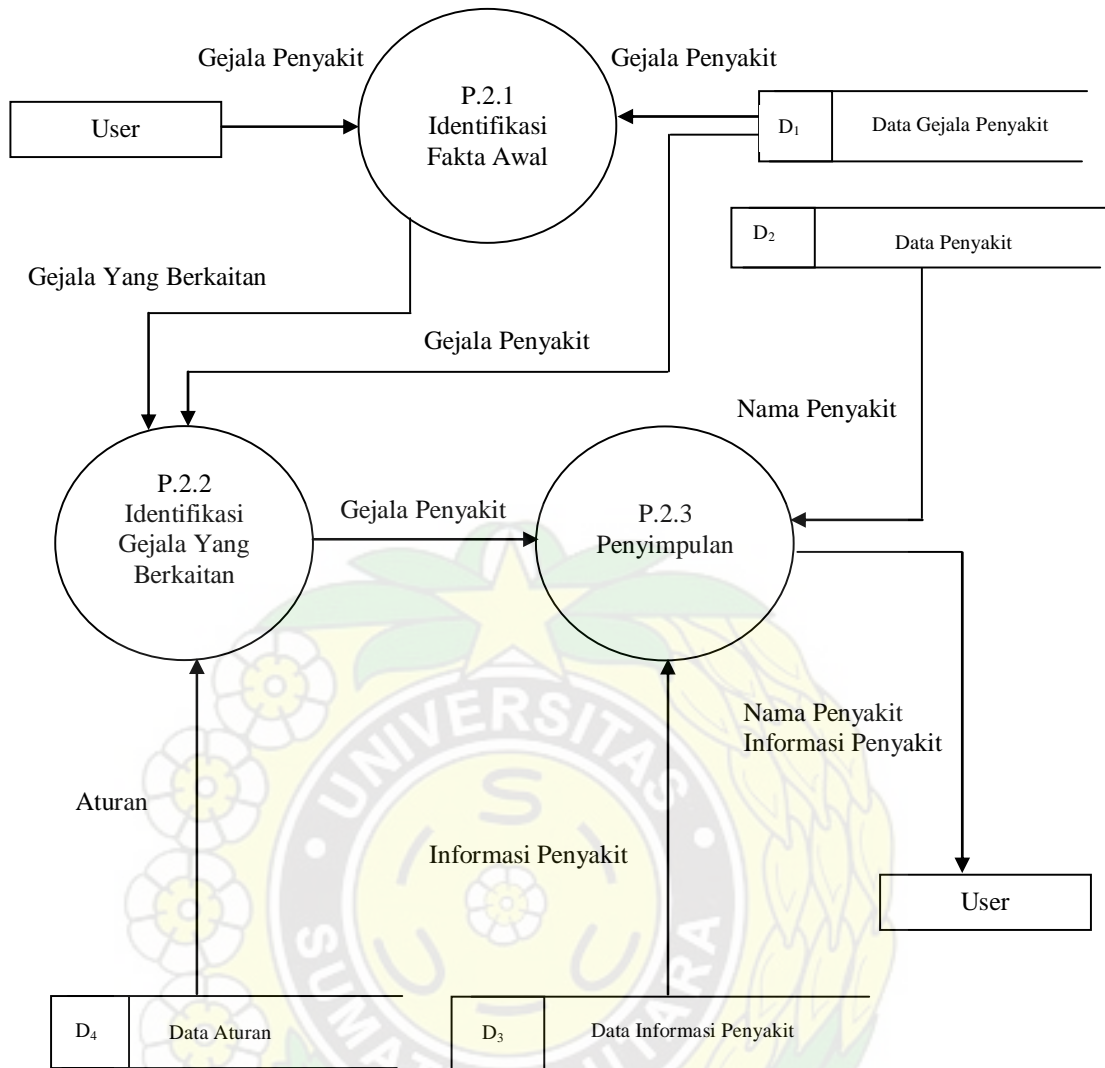
DFD Level 1 terdiri dari dua proses, yaitu proses dokumentasi basis pengetahuan dan proses penelusuran.



Gambar 3.11 DFD Level 1

3.3.2. Data Flow Diagram Level 2 (DFD Level 2)

DFD Level 2 terjadi empat proses, yaitu identifikasi fakta awal, proses identifikasi gejala yang berkaitan dan proses penyimpulan.



Gambar 3.12 DFD Level 2 Untuk Proses 2 (Penelusuran)

3.4. Perancangan Basis Pengetahuan

Pengetahuan atau data yang ada, disusun sedemikian rupa ke dalam bentuk tabel untuk mempermudah sistem dalam pengambilan keputusan. Seluruh tabel saling berhubungan satu dengan yang lainnya dan gambaran tabel basis pengetahuan yang digunakan adalah sebagai berikut:

1. Tabel Gejala

Tabel ini berisi data gejala klinis.

Table 3.1 Tabel Gejala

No	Nama Field	Tipe	Ukuran	Keterangan
1	IDGejala	Integer	4	Nomor ID Gejala
2	KdGejala	Teks	3	Kode Gejala
3	NmGejala	Teks	75	Nama Gejala

Keterangan: Primary Key = KdGejala

2. Tabel Pertanyaan

Tabel ini berisi data pertanyaan beserta pilihan gejala yang akan ditampilkan pada setiap pertanyaan.

Table 3.2 Tabel Pertanyaan

No	Nama Field	Tipe	Ukuran	Keterangan
1	KdPertanyaan	Teks	3	Kode Pertanyaan
2	NmPertanyaan	Teks	75	Nama Pertanyaan
3	IDGejala	Integer	4	Nomor ID Gejala

Keterangan: Primary Key = KdPertanyaan

3. Tabel Penyakit

Tabel ini berisi data jenis penyakit.

Table 3.3 Tabel Penyakit

No	Nama Field	Tipe	Ukuran	Keterangan
1	KdPenyakit	Teks	3	Kode Penyakit
2	NmPenyakit	Teks	75	Nama Penyakit

Keterangan: Primary Key = KdPenyakit

4. Tabel Aturan

Tabel ini berisi data aturan berupa pertanyaan dan pilihan gejala yang saling berhubungan untuk mendapatkan kesimpulan.

Table 3.4 Tabel Aturan

No	Nama Field	Tipe	Ukuran	Keterangan
1	IDAturan	Integer	4	Nomor ID Aturan
2	KdPertanyaan	Teks	3	Kode Pertanyaan
3	KdGejala	Teks	3	Kode Gejala
4	Solusi	Integer	4	Nomor Solusi

Keterangan: Primary Key = IDAturan

5. Tabel Info

Tabel ini berisi data informasi seputar penyakit anak, seperti penyebab, penaggulanggannya dan foto.

Table 3.5 Tabel Info

No	Nama Field	Tipe	Ukuran	Keterangan
1	IDInfo	Integer	4	Nomor ID Info
2	KdPenyakit	Teks	3	Kode Penyakit
3	Info1	Teks	100	Informasi 1
4	Info2	Teks	100	Informasi 2
5	Info3	Teks	100	Informasi 3

Keterangan: Primary Key = IDInfo

6. Tabel Password

Tabel ini berisi data admin selaku pengelola sistem.

Table 3.6 Tabel Password

No	Nama Field	Tipe	Ukuran	Keterangan
1	User	Char	10	Nama Admin
2	Passwd	Char	10	Password Admin

Keterangan: Primary Key = User

3.5. Perancangan Antarmuka (Interface)

Antarmuka (*interface*) merupakan bagian dari sistem pakar yang digunakan sebagai alat komunikasi antara sistem dan *user*. Perancangan antarmuka dalam sistem pakar untuk diagnosa penyakit anak dibedakan atas dua bagian, yaitu:

1. Perancangan Antarmuka Untuk Pakar (*Admin*)

Antarmuka untuk *admin* dirancang agar *admin* dapat melakukan proses pengelolaan sistem.

2. Perancangan Antarmuka Untuk Pengguna (*User*)

Antarmuka untuk *user* dirancang agar *user* dapat mencari informasi penyakit anak dengan melakukan proses diagnosa.

3.5.1. Rancangan Menu Utama

Menu Utama merupakan form utama pada saat *user* maupun *admin* mengakses sistem pakar ini. Form ini akan digunakan oleh *user* secara umum, untuk memilih apakah sebagai *user* atau *admin*. Untuk dapat melakukan pengelolaan data, maka *user* harus *logon* terlebih dahulu dengan masuk ke Menu Update. Form ini juga dimaksudkan untuk membantu *user* menuju ke menu-menu selanjutnya, seperti Menu Diagnosa.

The diagram illustrates the layout of the Main Menu Interface. It consists of a rectangular frame containing the following elements:

- A horizontal box at the top labeled "JUDUL" (Title).
- A large rectangular area on the left labeled "Gambar" (Image).
- A vertical stack of four buttons on the right side, labeled "INFO", "DIAGNOSA", "UPDATE", and "BANTUAN" (Help).

Gambar 3.13 Rancangan Tampilan Menu Utama

3.5.2. Rancangan Menu Info

Menu Info merupakan form yang berfungsi untuk menampilkan informasi anak, baik seputar keamanan maupun gizi.

Info 1	Info 1	Info 1
<div style="border: 1px solid black; padding: 10px; text-align: center;">Form ini berisi info seputar anak, sehubungan dengan pilihan judul info</div>		

Gambar 3.14 Rancangan Tampilan Menu Info

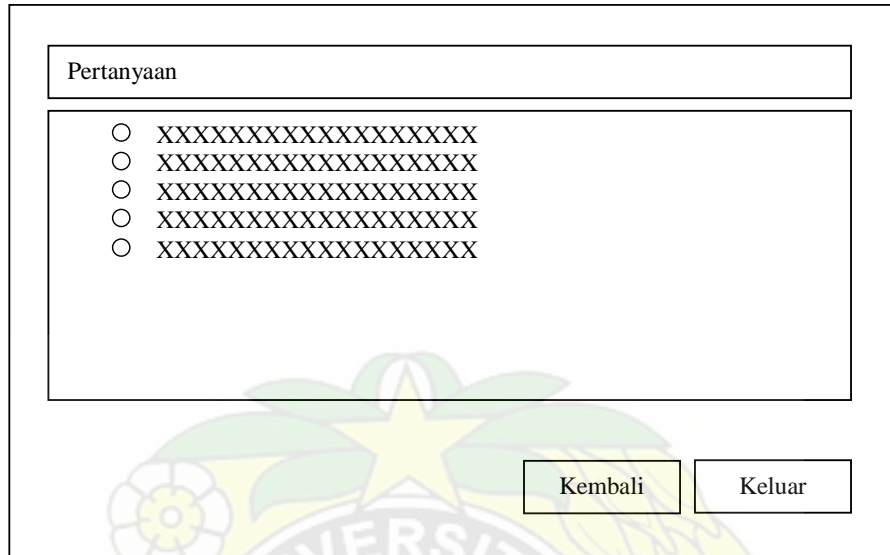
3.5.3. Rancangan Menu Diagnosa

Menu ini berfungsi untuk memulai proses diagnosa dengan cara menampilkan form penelusuran yang menampilkan pertanyaan dan pilihan gejala pada *user*, dimana pilihan *user* nantinya akan mengarah kepada pertanyaan selanjutnya atau menghasilkan kesimpulan suatu penyakit.

Setiap pilihan gejala *user* akan dibandingkan dengan data yang dimiliki oleh sistem. Berdasarkan data aturan, sistem memutuskan melakukan proses penelusuran selanjutnya atau menghasilkan hasil akhir diagnosa. Bila sistem telah selesai melakukan proses penelusuran, maka sistem akan menampilkan hasil berupa gejala input *user* dan nama penyakit yang kemungkinan diderita.

Pada Form Penelusuran terdapat dua buah *button*, yaitu *button* Kembali untuk kembali ke form sebelumnya dan melanjutkan proses diagnosa dengan klik *button* Lanjut. Sedangkan pada Form Hasil Diagnosa terdapat dua buah *button*, yaitu *button* Lihat Detail untuk masuk ke Menu Detail Penyakit dan *button* Selesai yang berfungsi

untuk keluar dari proses diagnosa dan kembali ke Menu Utama. Menu Detail Penyakit merupakan form yang berfungsi memberikan informasi kepada *user* mengenai penyakit anak yang berhubungan dengan hasil diagnosa dan sedikit lebih detail.

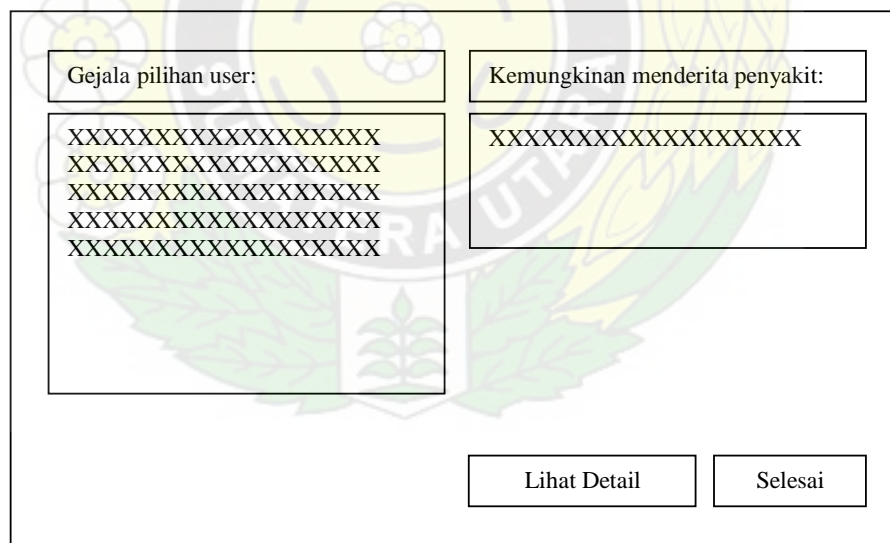


Pertanyaan

- ☐ XXXXXXXXXXXXXXXXXXXX
- ☐ XXXXXXXXXXXXXXXXXXXX
- ☐ XXXXXXXXXXXXXXXXXXXX
- ☐ XXXXXXXXXXXXXXXXXXXX
- ☐ XXXXXXXXXXXXXXXXXXXX

Kembali Keluar

Gambar 3.15 Rancangan Tampilan Form Penelusuran



Gejala pilihan user:

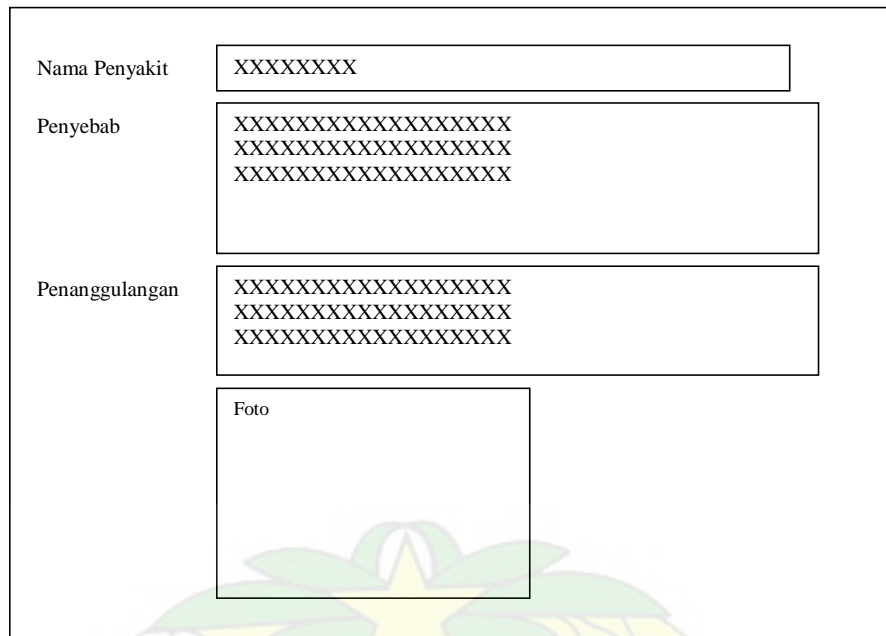
XXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXX

Kemungkinan menderita penyakit:

XXXXXXXXXXXXXXXXXXXX

Lihat Detail Selesai

Gambar 3.16 Rancangan Tampilan Form Hasil Diagnosa

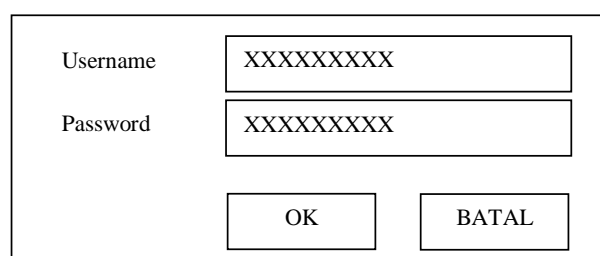


Nama Penyakit	XXXXXXXX
Penyebab	XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
Penanggulangan	XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX XXXXXXXXXXXXXXXXXXXX
Foto	

Gambar 3.17 Rancangan Tampilan Menu Detail Penyakit

3.5.4. Rancangan Form Logon

Form Logon digunakan oleh *user* yang berperan sebagai *admin*. Pada form ini, *admin* akan menginput *username* dan *password*. Sistem akan mencocokkan data yang diinput dengan data yang ada pada tabel *password*. Jika proses *logon* gagal maka akan ditampilkan pesan kesalahan (pakar) dan *admin* harus menginput ulang datanya dengan benar. Dan jika *logon* berhasil maka akan ditampilkan form utama *admin* (Menu Update). Jika ada salah satu atau semua *field* isian kosong baik *username* atau *password* maka akan ditampilkan pesan pakar, maka *admin* harus mengisi lengkap datanya. Jika *button* Batal di klik maka akan kembali ke Menu Utama dan jika *button* Ok di klik maka akan melakukan proses validasi data.

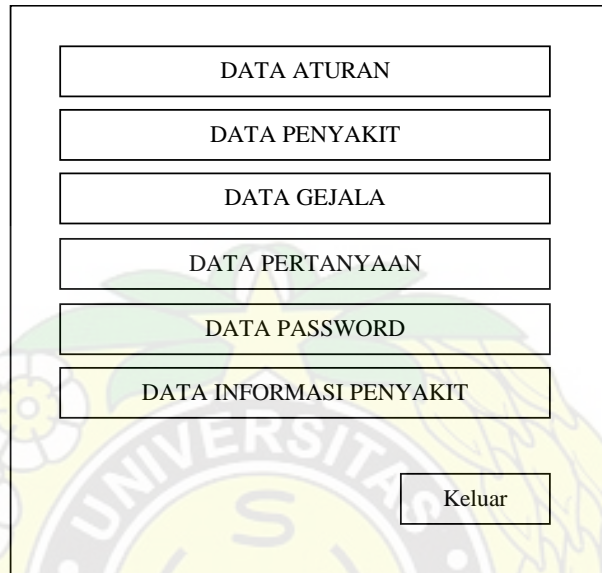


Username	XXXXXXXXXX
Password	XXXXXXXXXX
<div>OK</div> <div>BATAL</div>	

Gambar 3.18 Rancangan Tampilan Form Logon

3.5.5. Rancangan Menu Update

Menu Update merupakan form yang berfungsi untuk membantu *admin* untuk menuju ke form pengelolaan data, yaitu form data aturan, data penyakit, data gejala, data pertanyaan, data *password* dan data informasi penyakit.



The image shows a menu update form with a vertical list of buttons. The buttons are labeled: DATA ATURAN, DATA PENYAKIT, DATA GEJALA, DATA PERTANYAAN, DATA PASSWORD, and DATA INFORMASI PENYAKIT. The buttons for DATA PASSWORD and DATA INFORMASI PENYAKIT are highlighted in green. A button labeled 'Keluar' is located at the bottom right of the form.

DATA ATURAN
DATA PENYAKIT
DATA GEJALA
DATA PERTANYAAN
DATA PASSWORD
DATA INFORMASI PENYAKIT
Keluar

Gambar 3.19 Rancangan Tampilan Menu Update

3.5.5.1. Rancangan Form Data Penyakit

Pada form ini *admin* dapat menambah data penyakit baru dengan mengklik *button* Tambah, mencari data penyakit yang sudah disimpan sebelumnya dengan mengklik *button* Cari dan selanjutnya agar dapat melakukan pengubahan dan penghapusan data penyakit dengan *button* Ubah dan Hapus. Melakukan pembatalan proses yang sedang berlangsung (penambahan, pencarian, perubahan dan penghapusan) serta mengosongkan semua *field* isian yang sudah berisi data sebelumnya dengan mengklik *button* Batal. Menutup form dan kembali ke Menu Update dengan mengklik *button* Keluar.

Kode Penyakit	XXX
Nama Penyakit	XXXXXXXXX

Tombol Fungsi

Tambah
Cari
Ubah
Hapus
Batal
Keluar

Gridview

KdPenyakit	NmPenyakit
XXX	XXXXXXXXX
XXX	XXXXXXXXX

Gambar 3.20 Rancangan Tampilan Form Data Penyakit

3.5.5.2. Rancangan Form Data Password

Pada form ini *admin* dapat mengelola data *user* baru selaku *admin*. Menambah data *admin* baru dengan mengklik *button* Tambah, mencari data *admin* yang sudah disimpan sebelumnya dengan mengklik *button* Cari dan selanjutnya agar dapat melakukan pengubahan dan penghapusan data *admin* dengan *button* Ubah dan Hapus. Melakukan pembatalan proses yang sedang berlangsung (penambahan, pencarian, perubahan dan penghapusan) serta mengosongkan semua *field* isian yang sudah berisi data sebelumnya dengan mengklik *button* Batal. Menutup form dan kembali ke Menu Update dengan mengklik *button* Keluar.

Username	XXXXXXXX						
Password	XXXXXXXX						
<div>Tombol Fungsi</div> <table border="1"> <tr> <td>Tambah</td> <td>Cari</td> <td>Ubah</td> <td>Hapus</td> <td>Batal</td> <td>Keluar</td> </tr> </table>		Tambah	Cari	Ubah	Hapus	Batal	Keluar
Tambah	Cari	Ubah	Hapus	Batal	Keluar		

Gambar 3.21 Rancangan Tampilan Form Data Password

3.5.6. Rancangan Menu Bantuan

Menu ini dirancang untuk panduan bagi *user* yang ingin melakukan proses diagnosa (konsultasi) dengan sistem.

	X
<div>Form ini berisi panduan cara menggunakan sistem pakar diagnosa penyakit anak, yaitu pada saat <i>user</i> memilih Menu Diagnosa.</div>	

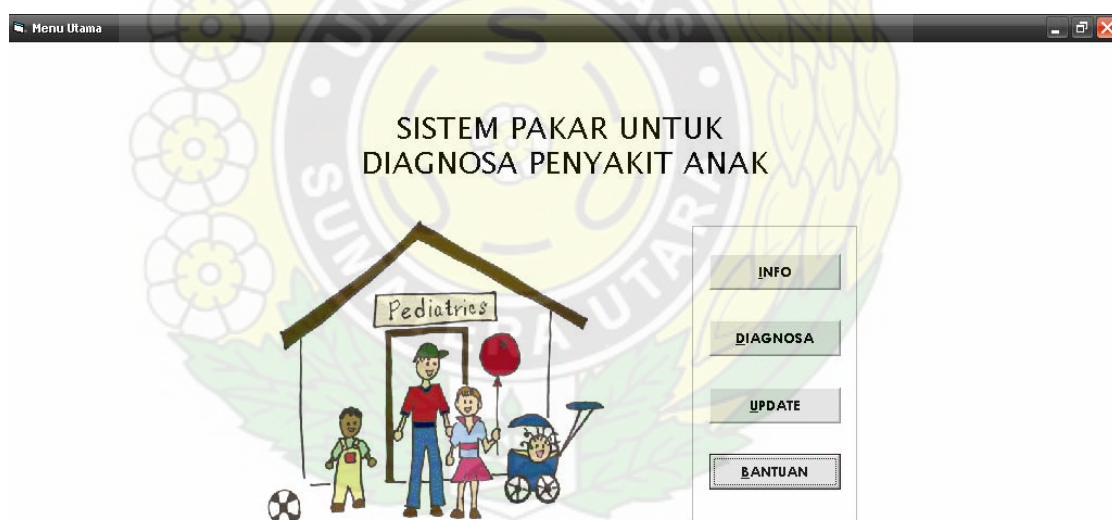
Gambar 3.22 Rancangan Tampilan Menu Bantuan

BAB 4

IMPLEMENTASI SISTEM PAKAR

4.1. Menu Utama

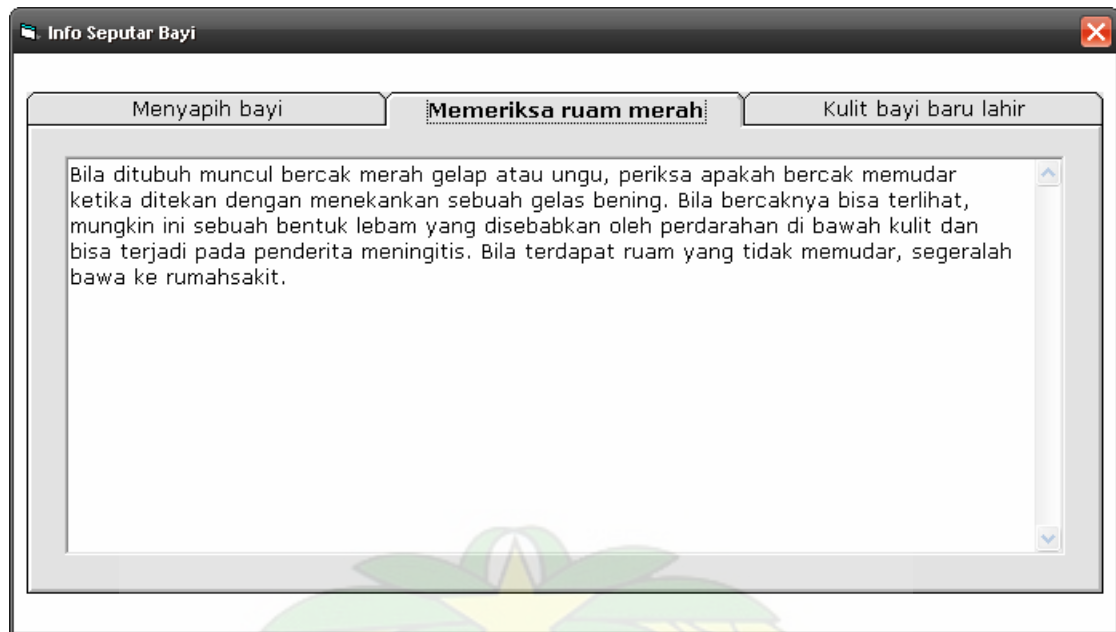
Saat pertama kali aplikasi sistem pakar dijalankan, maka akan tampil Form Menu Utama yang merupakan menu utama bagi *user* biasa maupun bagi *admin*. Form ini digunakan untuk menampilkan menu-menu program aplikasi yang telah dirancang untuk menjalankan sistem pakar.



Gambar 4.1 Form Menu Utama

4.2. Menu Info

Menu ini digunakan untuk menampilkan informasi anak, baik mengenai keamanan maupun gizi karena masa kecil ialah masa perubahan dan pencegahan.



Gambar 4.2 Form Menu Info

4.3. Menu Diagnosa

Menu ini digunakan untuk melakukan diagnosa berdasarkan gejala yang dirasakan oleh *user*. Sistem pakar akan menampilkan pertanyaan beserta pilihan gejala pada setiap penelusuran yang dapat dipilih oleh *user* dan kesimpulan akhir penelusuran (hasil diagnosa). Namun pada form hasil diagnosa, *user* dapat melihat informasi penyakit sesuai dengan hasil diagnosa lebih detail dengan klik *button* Lihat Detail. Hasil diagnosa yang diperoleh tersebut didapatkan dari hasil penelusuran terhadap jenis penyakit berdasarkan gejala yang dipilih oleh *user* serta aturan. Gejala yang dipilih *user* pada setiap penelusuran akan diambil KdGejala nya, kemudian membandingkan dengan aturan-aturan yang berlaku pada tabel aturan, sehingga didapat KdPenyakit yang memiliki kesamaan gejala dengan gejala yang dipilih *user*. Setelah itu akan dicocokkan KdPenyakit dengan NmPenyakit dari tabel penyakit, sehingga keluar sebagai hasil diagnosa.

Pertanyaan

Gejala apa yang terjadi pada balita Anda?

- ☐ Perilaku yang tidak biasa
- ☐ Susah menelan
- ☐ Sulit bernafas
- ☒ Bercak pada kulit

Lanjut

Gambar 4.3 Form Penelusuran 1

Pertanyaan

Bagaimana karakteristik kulitnya?

- ☐ Ruam merah disertai bintik-bintik yang tidak memudar bila ditekan
- ☒ Gatal, kasar disertai bintil-bintil dan kulit memerah
- ☐ Gatal dan memerah, serta terdapat lepuhan kecil berisi cairan menyerupai cincin
- ☐ Lepuhan kecil berisi cairan disertai memar

Kembali Lanjut

Gambar 4.4 Form Penelusuran 2

Hasil Diagnosa

Gejala-gejala penyakit berdasarkan pilihan user:

Bercak pada kulit
Gatal, kasar disertai bintil-bintil

Kemungkinan terbesar penyakit yang diderita:

Dermatitis Atopik

[LihatDetail](#) [Selesai](#)

Gambar 4.5 Form Hasil Diagnosa

Detail Penyakit

Nama Penyakit: Dermatitis Atopik

Penyebab: Disebabkan oleh alergi.

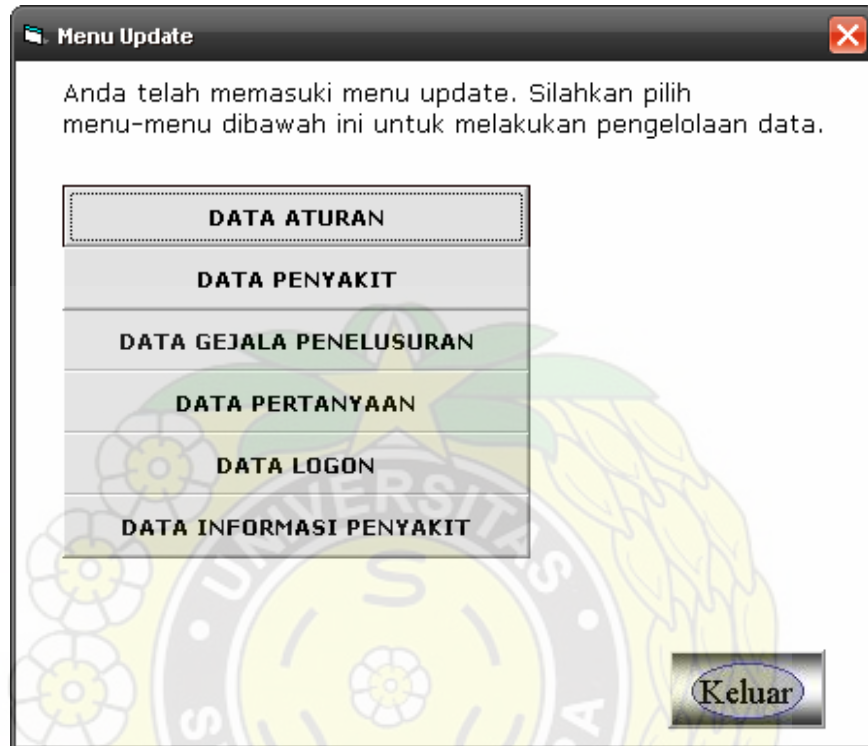
Penanggulangan: Bagi anak berkulit kering mandikan dia dengan air hangat dan minyak emolien (zat yang melembutkan kulit seperti krim larut air) yang dioleskan 2 kali sehari. Pastikan pakaiannya terbuat dari bahan katun dan jaga agar anak tidak menggaruk. Obat antihistamin yang dijual bebas dapat membantu mengontrol gatal yang parah. Jaga agar tidak terkena pilek, yang bisa menyebabkan infeksi virus yang serius. Periksakanlah ke dokter untuk penanganan lebih lanjut.

Foto:

Gambar 4.6 Form Detail Penyakit

4.4. Menu Update

Menu ini digunakan untuk menampilkan menu-menu yang dapat digunakan untuk melakukan pengelolaan data yang digunakan sebagai basis pengetahuan bagi sistem.



Gambar 4.7 Form Menu Update

Menu diatas akan ditampilkan bila *admin* berhasil melakukan LogOn, dimana LogOn memiliki fungsi . Pada saat *admin* LogOn, maka *admin* akan diminta untuk mengisi *username* dan *password*, kemudian sistem akan mencocokkan data yang diinputkan dengan data yang ada pada tabel password. Jika data yang diinputkan salah, baik data *username* maupun data *password*, maka sistem akan menampilkan pesan kesalahan (pesan pakar) “LogOn Gagal”.



Gambar 4.8 Form LogOn

4.4.1. Form Data Penyakit

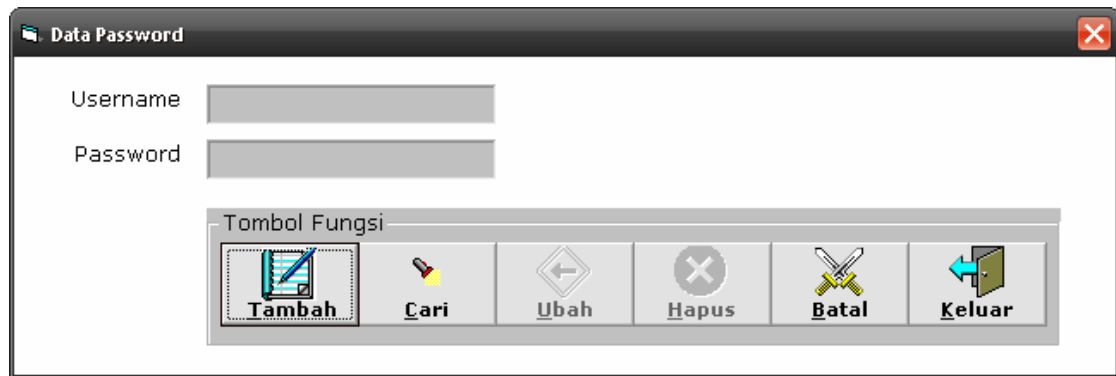
Pada form ini akan ditampilkan nama-nama penyakit yang tersimpan pada database. Dari form ini *admin* dapat melakukan proses pengelolaan data penyakit seperti penambahan data penyakit, serta melakukan perubahan dan penghapusan data penyakit yang telah disimpan sebelumnya. Untuk melakukan penambahan data penyakit, *admin* dapat mengklik *button* Tambah. Untuk melakukan perubahan data penyakit, *admin* dapat mengklik *button* Ubah. Untuk melakukan penghapusan data penyakit, *admin* dapat mengklik *button* Hapus. *Gridview* berfungsi sebagai tempat untuk menampilkan data-data yang telah diinput sebelumnya, agar mudah dalam pencarian data penyakit yang sudah disimpan sebelumnya dapat menggunakan *button* Cari. Klik *button* Batal untuk melakukan pembatalan dan klik *button* Keluar untuk menutup form Data Penyakit serta kembali ke Menu Update.

Kode	Nama Penyakit
001	DIARE
002	TONSILITIS
003	LARINGITIS
004	BRONKHITIS
005	PNEUMONIA
006	MENINGITIS
007	DERMATITIS ATOPIK
008	MIKOSIS SUPERFISIALIS
009	MONILIASIS
010	BRONKIOLITIS

Gambar 4.9 Form Data Penyakit

4.4.2. Form Data Password

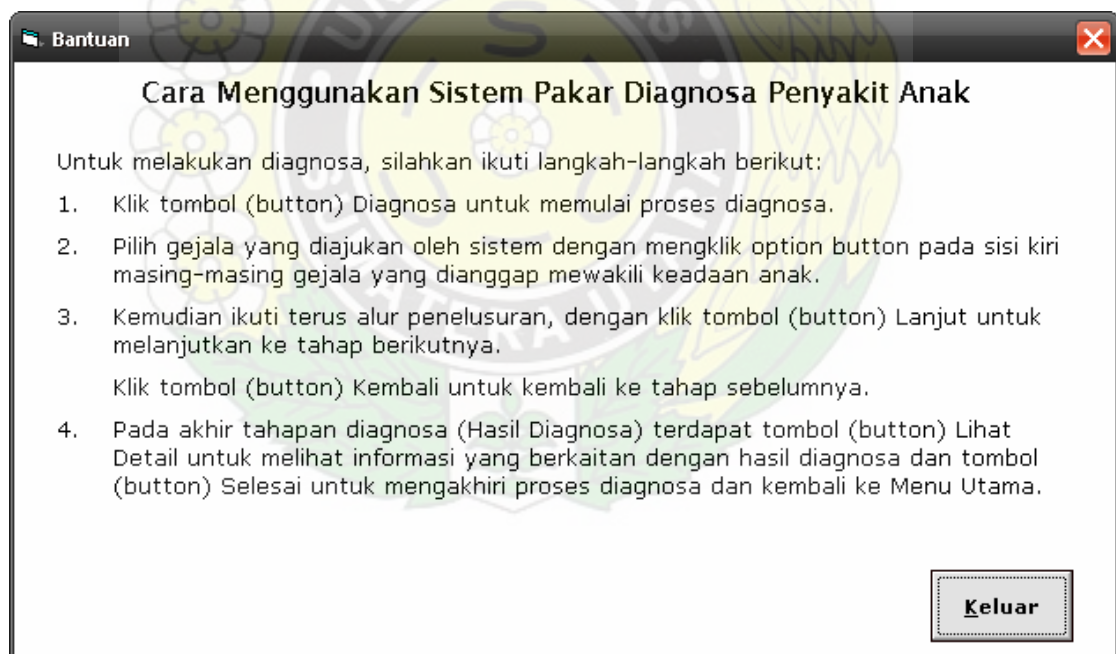
Form ini digunakan melakukan proses pengelolaan data *admin* seperti penambahan data *user*, serta melakukan perubahan dan penghapusan data *user* yang telah disimpan sebelumnya. Untuk melakukan penambahan data *user*, *admin* dapat mengklik *button* Tambah. Untuk melakukan perubahan *password* atau *username*, *admin* dapat mengklik *button* Ubah. Untuk melakukan penghapusan data *user*, *admin* dapat mengklik *button* Hapus. Agar mudah dalam pencarian data *user* yang sudah disimpan sebelumnya dapat menggunakan *button* Cari. Klik *button* Batal untuk melakukan pembatalan dan klik *button* Keluar untuk menutup form Data Password serta kembali ke Menu Update.



Gambar 4.10 Form Data Password

4.5. Menu Bantuan

Pada menu ini akan dijelaskan bagaimana menggunakan sistem, meliputi cara mencari informasi seputar anak (menu info) dan cara melakukan diagnosa (menu diagnosa).



Cara Menggunakan Sistem Pakar Diagnosa Penyakit Anak

Untuk melakukan diagnosa, silahkan ikuti langkah-langkah berikut:

1. Klik tombol (button) Diagnosa untuk memulai proses diagnosa.
2. Pilih gejala yang diajukan oleh sistem dengan mengklik option button pada sisi kiri masing-masing gejala yang dianggap mewakili keadaan anak.
3. Kemudian ikuti terus alur penelusuran, dengan klik tombol (button) Lanjut untuk melanjutkan ke tahap berikutnya.
Klik tombol (button) Kembali untuk kembali ke tahap sebelumnya.
4. Pada akhir tahapan diagnosa (Hasil Diagnosa) terdapat tombol (button) Lihat Detail untuk melihat informasi yang berkaitan dengan hasil diagnosa dan tombol (button) Selesai untuk mengakhiri proses diagnosa dan kembali ke Menu Utama.

Keluar

Gambar 4.11 Form Menu Bantuan

4.6. Teknik Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui apakah sistem telah berjalan sesuai dengan apa yang telah dirancang sebelumnya. Pengujian yang dilakukan berfokus

pada persyaratan fungsional perangkat lunak. Pengujian ini memungkinkan pakar memperoleh kumpulan kondisi input yang akan mengerjakan seluruh keperluan fungsional program. Pengujian ini bertujuan untuk menunjukkan fungsi perangkat lunak tentang cara beroperasinya, apakah pemasukan data keluaran telah berjalan sebagaimana yang diharapkan. Pengujian sistem ini disebut pengujian *black box*. Sistem akan membandingkan pilihan gejala yang telah dipilih *user* pada setiap penelusuran dengan basis pengetahuan. Tiap pilihan gejala *user* akan menentukan penelusuran yang akan ditampilkan selanjutnya oleh sistem pakar dan pada hasil akhir akan tampil hasil diagnosa. Jika hasil diagnosa diperoleh, *user* dapat melihat informasi detail penyakit sesuai penyakit hasil diagnosa berupa penyebab, penanggulangan dan foto melalui *button* Lihat Detail. Jika sistem berjalan sesuai dengan aturan pada basis pengetahuan dan semua fungsi komponen pada setiap *interface* berfungsi dengan benar, maka sistem telah berjalan dengan baik. Berikut pengujian pemasukan data keluaran dengan membandingkan hasilnya pada basis pengetahuan melalui proses tanya jawab pada sistem:

1. Pengujian pada aturan 24:

Pertanyaan: Gejala apa yang terjadi pada balita anda?

Jawaban: Perilaku yang tidak wajar.

Pertanyaan: Bagaimana perilaku balita anda?

Jawaban: Nafsu makan berkurang.

Pertanyaan: Bagaimana karakteristik fesesnya?

Jawaban: Tinja cair berlendir (mukus).

Pertanyaan: Apa yang terlihat pada wajahnya?

Jawaban: Tampak sehat.

Setiap jawaban (pilihan gejala) dibandingkan dengan basis pengetahuan. Sistem akan berhenti melakukan proses tanya jawab pada saat menemukan nilai solusi sama dengan nilai penyakit pada aturan. Berikut tabel aturan pada basis pengetahuan:

Tabel 4.1 Tabel Aturan Pada Aturan 24

IDAturan	Pertanyaan	Gejala	Solusi
24	Gejala apa yang terjadi pada balita anda?	Perilaku yang tidak wajar.	0
24	Bagaimana perilaku balita anda?	Nafsu makan berkurang.	0
24	Bagaimana karakteristik fesesnya?	Tinja cair berlendir (mukus).	0
24	Apa yang terlihat pada wajahnya?	Tampak sehat.	011

Data yang diperoleh dari proses tanya jawab dan dibandingkan dengan basis pengetahuan telah memiliki *output* sama, maka dapat disimpulkan pada pengujian ini sistem pakar sudah bekerja sebagaimana rancangan sistem yang dibuat. Berikut laporan hasil pengujian sistem:

Diagnosa

Gejala apa yang terjadi pada balita anda?

Perilaku yang tidak wajar.

Bagaimana perilaku balita anda?

Nafsu makan berkurang.

Bagaimana karakteristik fesesnya?

Tinja cair berlendir (mukus).

Apa yang terlihat pada wajahnya?

Tampak sehat.

Hasil Diagnosa

Gejala-gejala penyakit berdasarkan pilihan *user*:

Perilaku yang tidak wajar.

Nafsu makan berkurang.

Tinja cair berlendir (mukus).

Tampak sehat.

Kemungkinan terbesar penyakit yang diderita:

Gejala belum terdiagnosa.

Pilihan gejala telah dibandingkan dengan basis pengetahuan pada sistem dan tidak menghasilkan suatu penyakit dan tidak ada informasi detail penyakit yang berhubungan.

2. Pengujian pada aturan 32:

Pertanyaan: Gejala apa yang terjadi pada balita anda?

Jawaban: Bercak pada kulit.

Pertanyaan: Bagaimana karakteristik kulitnya?

Jawaban: Gatal, kasar disertai bintil-bintil dan kulit memerah.

Setiap jawaban (pilihan gejala) dibandingkan dengan basis pengetahuan. Sistem akan berhenti melakukan proses tanya jawab pada saat menemukan nilai solusi sama dengan nilai penyakit pada aturan. Berikut tabel aturan pada basis pengetahuan:

Tabel 4.2 Tabel Aturan Pada Aturan 32

IDAturan	Pertanyaan	Gejala	Solusi
32	Gejala apa yang terjadi pada balita anda?	Bercak pada kulit.	0
32	Bagaimana karakteristik kulitnya?	Gatal, kasar disertai bintil-bintil dan kulit memerah.	007

Data yang diperoleh dari proses tanya jawab dan dibandingkan dengan basis pengetahuan telah memiliki *output* sama, maka dapat disimpulkan pada pengujian ini sistem pakar sudah bekerja sebagaimana rancangan sistem yang dibuat. Berikut laporan hasil pengujian sistem:

Diagnosa

Gejala apa yang terjadi pada balita anda?

Bercak pada kulit.

Bagaimana karakteristik kulitnya?

Gatal, kasar disertai bintil-bintil dan kulit memerah.

Hasil Diagnosa

Gejala-gejala penyakit berdasarkan pilihan *user*:

Bercak pada kulit.

Gatal, kasar disertai bintil-bintil dan kulit memerah.

Kemungkinan terbesar penyakit yang diderita:

Dermatitis Atopik.

Lihat Detail

Penyebab:

Disebabkan oleh alergi.

Penaggulangannya:

Bagi anak berkulit kering mandikan dia dengan air hangat dan minyak emolien (zat yang melembutkan kulit seperti krim larut air) yang dioleskan 2 kali sehari. Pastikan pakaiannya terbuat dari bahan katun dan jaga agar anak tidak menggaruk. Obat antihistamin yang dijual bebas dapat membantu mengontrol gatal yang parah. Jaga agar tidak terkena pilek, yang bisa menyebabkan infeksi virus yang serius. Periksakanlah ke dokter untuk penanganan lebih lanjut.

Foto:



Gambar 4.12 Foto Penyakit Pada Menu Lihat Detail Setelah Proses Hasil Diagnosa

Pilihan gejala telah dibandingkan dengan basis pengetahuan pada sistem dan menghasilkan suatu penyakit dan menampilkan informasi detail penyakit yang berhubungan.

BAB 5

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari tugas akhir ini penulis dapat mengambil suatu kesimpulan yaitu, metode inferensi yang digunakan pada sistem pakar ini terbatas pada penggunaan data penyakit yang sederhana atau tidak kompleks.

5.2. Saran

Berdasarkan kesimpulan yang telah diambil, maka dapat dikemukakan saran yang akan sangat membantu untuk aplikasi sistem pakar yang telah dibangun ini agar dapat dikembangkan lebih lanjut lagi dengan mendesain metode inferensi lain seperti *backward chaining* untuk data penyakit yang lebih kompleks sehingga sistem dapat dengan mudah menjelaskan secara tepat tujuan yang akan dipenuhi.

DAFTAR PUSTAKA

- Arhami, M. 2005. *Konsep Dasar Sistem Pakar*. Jilid 1. Yogyakarta: Andi.
- Artanti, F. R. 2004. *Perancangan dan Pembuatan Sistem Pakar Hama dan Pengendaliannya untuk Tanaman Hortikultura*. Skripsi. Surabaya: Universitas Kristen Petra.
- Gilbert, P. 1986. *Penyakit yang Lazim pada Anak-Anak*. Jakarta: Arcan.
- Kendall, K. E., and Julie E. K. 2002. *Systems Analysis and Design*. Fifth Edition. New Jersey: Pearson Education, Inc.
- Kusrini. 2006. *Sistem Pakar Teori dan Aplikasi*. Yogyakarta: Penerbit Andi Yogyakarta.
- Natalia, D. A. 2006. *Pembangunan Sistem Pakar pada Perangkat Mobile dengan WML dan PHP untuk Penyakit Paru pada Anak*. Proyek Akhir. Surabaya: Politeknik Negeri Surabaya.
- Purnamawati. 2007. "Manifestasi klinis dan diagnosis penyakit tropis". Dalam Anugroho, D. *Seminar Nasional Parasitologi dan Penyakit Tropis*. Bali.
- Rosenstein, B. J. 1995. *Pediatric Pearls: The Handbook of Practical Pediatrics*. Maryland: Mosby-Year Book, Inc.
- Setiawan, S. 2003. *Perancangan dan Pembuatan Sistem Pakar pada Pengobatan Penyakit Umum dengan Tanaman Obat pada Manusia*. Skripsi. Surabaya: Universitas Kristen Petra.
- Smith, T., and Sue D. 2005. *Pertolongan Pertama Dokter Di Rumah Anda*. Edisi Baru. Jakarta: Dian Rakyat.
- Staf Pengajar Ilmu Kesehatan Anak Fakultas Kedokteran Universitas Indonesia. 1985. *Ilmu Kesehatan Anak*. Jilid 1. Jakarta: Infomedika.
- Turban, E. 2005. *Decicion Support Systems and Intelligent Systems*. Yogyakarta: Penerbit Andi Yogyakarta.
- Waterman, D. A. 1986. *A Guide to Expert Systems*. Canada: Addison-Wesley Publishing Company.

<http://kuliah.dinus.ac.id/edi-nur/sb4-5.html> Diakses tanggal 8 September 2008

<http://www.oke.or.id/tutorial/Visual%20Basic%201.pdf> Diakses tanggal 8 September 2008

LAMPIRAN A. KODE PROGRAM

```
Private Sub Form_Load()  
    Me.WindowState = 2  
End Sub  
Private Sub Form_Resize()  
    If Me.WindowState = 2 Then  
        Label1.Move Me.Width / 2 - 3000, Me.Top + 1000  
        Frame1.Move Me.Width / 2 - (-1700), Me.Top + 2500  
        Image1.Move 3500, Me.Top + 2500  
        'Command3.Move Me.Width / 2 - 500, Me.Top + 5000  
    End If  
End Sub  
Private Sub cmdBantuan_Click()  
    frmHelp.Show  
End Sub  
  
Private Sub cmdDiag_Click()  
    frmPertanyaan.Show  
End Sub  
  
Private Sub cmdInfo_Click()  
    frmInfoAnak.Show  
End Sub  
  
Private Sub cmdUpdate_Click()  
    frmLogON.Show  
End Sub  
Private Sub mnuQuit_Click()  
    bsPeng.Enabled = False  
    frmLogin.txtUid.Text = ""  
    frmLogin.txtPass.Text = ""  
End  
End Sub  
  
Private Sub Form_Unload(Cancel As Integer)  
End  
End Sub
```



```

Private Sub Form_Activate()
Text1 = UCase(Text1.Text)
Text2 = UCase(Text2.Text)
Text3 = UCase(Text3.Text)

End Sub

Private Sub SStab1_Click(PreviousTab As Integer)
If PreviousTab = 0 Then
    SStab1.Caption = UCase(SStab1.Caption)
ElseIf PreviousTab = 1 Then
    SStab1.Caption = UCase(SStab1.Caption)
End If
End Sub

```



```

Option Explicit
Dim rsAturan As Recordset
Dim rsSusuran As Recordset
Dim rsGejalal As Recordset
Dim KdPenyakit As String * 3
Dim KdGejala As String * 3
Dim rsNmGejalal As Recordset
Dim NmGejala As String * 200
Private Sub cmdKembali_Click()
Set rsHapus = New ADODB.Recordset
    rsHapus.Open "Delete from tSusuran where left(KdGejala,1)='1'",
CN, 1, 2
Me.Hide
frmMenu.Show
End Sub

Private Sub cmdBatal_Click()
On Error Resume Next
Set rsHapus = New ADODB.Recordset
    rsHapus.Open "Delete from tSusuran where left(KdGejala,1)='1'",
CN, 1, 2
    List1.Selected(0) = False
    List1.Selected(1) = False
    List1.Selected(2) = False
    List1.Selected(3) = False
    List1.Selected(4) = False
    List1.Selected(5) = False
    List1.Selected(6) = False
    List1.Selected(7) = False
    List1.Selected(8) = False
    List1.Selected(9) = False
    List1.Selected(10) = False
    List1.Selected(11) = False
    List1.Selected(12) = False
    List1.Selected(13) = False

End Sub

Private Sub cmdLanjut_Click()
Set rsGejalal = Nothing
Set rsNmGejalal = Nothing
Set rsAturan = Nothing
Me.Hide
frmSusuran2.Show
End Sub

Private Sub Form_Activate()
List1.Clear
cmdLanjut.Enabled = True
cmdLanjut.SetFocus
Set rsHapus = New ADODB.Recordset
    rsHapus.Open "Delete from tSusuran", CN, 1, 2
Set rsHapus = New ADODB.Recordset
    rsHapus.Open "Delete from tRangking", CN, 1, 2

List1.Clear
Set rsGejalal = New ADODB.Recordset
    rsGejalal.Open "Select * from tGejalal order by KdGejalal asc",
CN, adOpenDynamic, adLockOptimistic
If Not rsGejalal.EOF = True Then

```

```

        Do While Not rsGejala1.EOF = True
            List1.AddItem rsGejala1!nmGejala1
            rsGejala1.MoveNext
            If rsGejala1.EOF = True Then
                Exit Do
            End If
        Loop
    End If
End Sub

Private Sub Form_Load()
    Koneksi
End Sub

Private Sub List1_Click()
    On Error Resume Next
    Set rsNmGejala1 = New ADODB.Recordset
    rsNmGejala1.Open "Select * from tGejala1 where NmGejala1 like '"
    & "%" & List1.Text & "%" & "' ORDER BY NmGejala1 asc", CN, 1, 2
    If Not rsNmGejala1.EOF = True Then
        KdGejala = rsNmGejala1!KdGejala1
        NmGejala = rsNmGejala1!nmGejala1
    End If
    Set rsAturan = New ADODB.Recordset
    rsAturan.Open "Select * from tAturan where kdGejala='" & KdGejala
    & "'", CN, adOpenDynamic, adLockOptimistic
    If Not rsAturan.EOF = True Then
        KdPenyakit = rsAturan!KdPenyakit
        Set rsSusuran = New ADODB.Recordset
        rsSusuran.Open "tSusuran", CN, 1, 2
        rsSusuran.AddNew
        rsSusuran!KdGejala = KdGejala
        rsSusuran!KdPenyakit = KdPenyakit
        rsSusuran!NmGejala = NmGejala
        rsSusuran.Update
    End If
End Sub

Option Explicit
Dim rsGejala2 As Recordset
Dim KdPenyakit As String * 3
Dim KdGejala2 As String * 3
Dim rsAturan As Recordset
Dim rsSusuran As Recordset
Dim KdGejala As String * 3
Dim rsNmGejala2 As Recordset
Dim NmGejala As String * 200
Private Sub cmdKembali_Click()
    Set rsHapus = New ADODB.Recordset
    rsHapus.Open "Delete from tSusuran where left(KdGejala,1)='2'",
    CN, 1, 2
    List1.Clear
    Set rsGejala2 = Nothing
    Set rsNmGejala2 = Nothing
    Set rsAturan = Nothing
    Set rsHapus = Nothing
    Me.Hide
    frmSusuran1.Show
End Sub

```

```

Private Sub cmdBatal_Click()
On Error Resume Next
Set rsHapus = New ADODB.Recordset
    rsHapus.Open "Delete from tSusuran where left(KdGejala,1)='2'",
CN, 1, 2
    List1.Selected(0) = False
    List1.Selected(1) = False
    List1.Selected(2) = False
    List1.Selected(3) = False
    List1.Selected(4) = False
    List1.Selected(5) = False
    List1.Selected(6) = False
    List1.Selected(7) = False
    List1.Selected(8) = False
    List1.Selected(9) = False
    List1.Selected(10) = False
    List1.Selected(11) = False
    List1.Selected(12) = False
    List1.Selected(13) = False
End Sub

Private Sub cmdLanjut_Click()
Set rsGejala2 = Nothing
Set rsNmGejala2 = Nothing
Set rsAturan = Nothing
Set rsHapus = Nothing
Me.Hide
frmSusuran3.Show
End Sub

Private Sub Form_Activate()
Koneksi
Call Mulai
End Sub
Sub Mulai()
List1.Clear
Set rsGejala2 = New ADODB.Recordset
    rsGejala2.Open "Select * from tGejala2 order by KdGejala2 asc",
CN, adOpenDynamic, adLockOptimistic
    If Not rsGejala2.EOF = True Then
        Do While Not rsGejala2.EOF = True
            List1.AddItem rsGejala2!NmGejala2
            rsGejala2.MoveNext
            If rsGejala2.EOF = True Then
                Exit Do
            End If
        Loop
    End If
End Sub

Private Sub Form_Load()
Koneksi
Mulai
End Sub

Private Sub List1_Click()
On Error Resume Next
Set rsNmGejala2 = New ADODB.Recordset
    rsNmGejala2.Open "Select * from tGejala2 where NmGejala2 like '"
& "%" & List1.Text & "%" & "' ORDER BY NmGejala2 asc", CN, 1, 2

```

```

    If Not rsNmGejala2.EOF = True Then
        KdGejala = rsNmGejala2!KdGejala2
        NmGejala = rsNmGejala2!NmGejala2
    End If
Set rsAturan = New ADODB.Recordset
rsAturan.Open "Select * from tAturan where kdGejala='" & KdGejala &
"'", CN, adOpenDynamic, adLockOptimistic
If Not rsAturan.EOF = True Then
    Do While Not rsAturan.EOF = True
        KdPenyakit = rsAturan!KdPenyakit
        Set rsSusuran = New ADODB.Recordset
        rsSusuran.Open "tSusuran", CN, 1, 2
        rsSusuran.AddNew
        rsSusuran!KdGejala = KdGejala
        rsSusuran!KdPenyakit = KdPenyakit
        rsSusuran!NmGejala = NmGejala
        rsSusuran.Update
        rsAturan.MoveNext
        If rsAturan.EOF = True Then
            Exit Do
        End If
    Loop
End If
End Sub

```

```

Option Explicit
Dim rsGejala3 As Recordset
Dim KdPenyakit As String * 3
Dim KdGejala3 As String * 3
Dim rsAturan As Recordset
Dim rsSusuran As Recordset
Dim KdGejala As String * 3
Dim rsNmGejala3 As Recordset
Dim NmGejala As String * 200

Private Sub cmdKembali_Click()
Set rsHapus = New ADODB.Recordset
rsHapus.Open "Delete from tSusuran where left(KdGejala,1)='3'",
CN, 1, 2
'CN.Close
List1.Clear
Set rsGejala3 = Nothing
Set rsNmGejala3 = Nothing
Set rsAturan = Nothing
Set rsHapus = Nothing
Me.Hide
frmSusuran1.Show
End Sub

```

```

Private Sub cmdBatal_Click()
On Error Resume Next
Set rsHapus = New ADODB.Recordset
rsHapus.Open "Delete from tSusuran where left(KdGejala,1)='3'",
CN, 1, 2
List1.Selected(0) = False
List1.Selected(1) = False
List1.Selected(2) = False
List1.Selected(3) = False
List1.Selected(4) = False

```



```

List1.Selected(5) = False
List1.Selected(6) = False
List1.Selected(7) = False
List1.Selected(8) = False
List1.Selected(9) = False
List1.Selected(10) = False
List1.Selected(11) = False
List1.Selected(12) = False
List1.Selected(13) = False
End Sub

Private Sub cmdLanjut_Click()
Set rsGejala3 = Nothing
Set rsNmGejala3 = Nothing
Set rsAturan = Nothing
Set rsHapus = Nothing

Me.Hide
'frmHasilDiag.Show
frmHasil.Show
End Sub

Private Sub Form_Activate()
List1.Clear
Set rsGejala3 = New ADODB.Recordset
rsGejala3.Open "Select * from tGejala3 order by KdGejala3 asc",
CN, adOpenDynamic, adLockOptimistic
If Not rsGejala3.EOF = True Then
Do While Not rsGejala3.EOF = True
List1.AddItem rsGejala3!NmGejala3
rsGejala3.MoveNext
If rsGejala3.EOF = True Then
Exit Do
End If
Loop
End If
End Sub

Private Sub Form_Load()
Koneksi
End Sub

Private Sub List1_Click()
On Error GoTo salah
Set rsNmGejala3 = New ADODB.Recordset
rsNmGejala3.Open "Select * from tGejala3 where NmGejala3 like '"
& "%" & List1.Text & "%" & "' ORDER BY NmGejala3 asc", CN, 1, 2
If Not rsNmGejala3.EOF = True Then
KdGejala = rsNmGejala3!KdGejala3
NmGejala = rsNmGejala3!NmGejala3
End If
Set rsAturan = New ADODB.Recordset
rsAturan.Open "Select * from tAturan where kdGejala='" & KdGejala &
"'", CN, adOpenDynamic, adLockOptimistic
If Not rsAturan.EOF = True Then
Do While Not rsAturan.EOF = True
KdPenyakit = rsAturan!KdPenyakit
Set rsSusuran = New ADODB.Recordset
rsSusuran.Open "tSusuran", CN, 1, 2
rsSusuran.AddNew

```



```

        rsSusuran!KdGejala = KdGejala
        rsSusuran!KdPenyakit = KdPenyakit
        rsSusuran!NmGejala = NmGejala
        rsSusuran.Update
        rsAturan.MoveNext
        If rsAturan.EOF = True Then
            Exit Do
        End If
    Loop
End If
salah:
If Err.Number <> 0 Then
    'List1.Clear
    List1.SetFocus
End If
End Sub

```





```

Option Explicit
Dim rsPenyakit As Recordset
Dim rsSusuran As Recordset
Dim rsJumlah As Recordset
Dim rsGejala1 As Recordset
Dim rsGejala2 As Recordset
Dim rsGejala3 As Recordset
Dim KdGejala As String * 3
Dim KdPenyakit As String * 3
Dim KdPeny As String * 3
Dim no As Integer
Dim jum001 As Integer
Dim jum002 As Integer
Dim jum003 As Integer
Dim jum004 As Integer
Dim jum005 As Integer
Dim jum006 As Integer
Dim jum007 As Integer
Dim jum008 As Integer
Dim jum009 As Integer
Dim jum010 As Integer

Dim rsSimpan001 As Recordset
Dim rsSimpan002 As Recordset
Dim rsSimpan003 As Recordset
Dim rsSimpan004 As Recordset
Dim rsSimpan005 As Recordset
Dim rsSimpan006 As Recordset
Dim rsSimpan007 As Recordset
Dim rsSimpan008 As Recordset
Dim rsSimpan009 As Recordset
Dim rsSimpan010 As Recordset

Dim rsRangking As Recordset
Dim bolSusur1 As Boolean
Dim bolSusur2 As Boolean
Dim bolSusur3 As Boolean
Dim bolSusur4 As Boolean
Dim bolSusur5 As Boolean
Dim bolSusur6 As Boolean
Dim bolSusur7 As Boolean
Dim bolSusur8 As Boolean
Dim bolSusur9 As Boolean
Dim bolSusur10 As Boolean

Sub JumRec()
Set rsJumlah = New ADODB.Recordset
rsJumlah.Open "Select * from tSusuran", CN, adOpenDynamic,
adLockOptimistic
If Not rsJumlah.EOF = True Then
rsJumlah.MoveFirst
Do While Not rsJumlah.EOF = True
no = no + 1
rsJumlah.MoveNext
If rsJumlah.EOF = True Then
Exit Do
End If
Loop
Else
no = 0

```

```

        End If
    End Sub

    Private Sub cmdDetil_Click()
        Me.Hide
        frmDetil.Show
    End Sub

    Private Sub cmdLanjut_Click()
        Set rsSusuran = Nothing
        Set rsGejala1 = Nothing
        Set rsGejala2 = Nothing
        Set rsGejala3 = Nothing
        Set rsHapus = Nothing
        Set rsSimpan001 = Nothing
        Set rsSimpan002 = Nothing
        Set rsSimpan003 = Nothing
        Set rsSimpan004 = Nothing
        Set rsSimpan005 = Nothing
        Set rsSimpan006 = Nothing
        Set rsSimpan007 = Nothing
        Set rsSimpan008 = Nothing
        Set rsSimpan009 = Nothing
        Set rsSimpan010 = Nothing
        Set rsHapus = Nothing

        lblNmPenyakit = ""
        Me.Hide
        frmMenu.Show
    End Sub

    Sub Segar()
        Set rsJumlah = Nothing
        Set rsSusuran = Nothing
        Set rsSegar = New ADODB.Recordset
        rsSegar.Open "Select * from tSusuran order by kdGejala Asc", CN,
        1, 2
        Adodc1.Refresh
        grid.Refresh
        grid.Visible = False
        'grid.ColWidth(0) = 0
        'grid.ColWidth(1) = 500
        'grid.ColWidth(2) = 30000
        '
        'grid.TextMatrix(0, 0) = ""
        'grid.TextMatrix(0, 1) = "Kode"
        'grid.TextMatrix(0, 2) = "Gejala Penyakit"

    End Sub

    Private Sub Form_Activate()
        grid.Visible = False
        'grid.ColWidth(0) = 0
        'grid.ColWidth(1) = 500
        'grid.ColWidth(2) = 10
        'grid.ColWidth(3) = 30000
        '
        'grid.TextMatrix(0, 0) = ""
        'grid.TextMatrix(0, 1) = "Kode"
        'grid.TextMatrix(0, 2) = ""
        'grid.TextMatrix(0, 3) = "Gejala Penyakit"
    
```

```

Set rsHapus = New ADODB.Recordset
    rsHapus.Open "Delete from tRangking", CN, 1, 2
JumRec
Call Hitung

End Sub

Private Sub Form_Load()
lblNmPenyakit = ""
Koneksi
End Sub

Sub Hitung()
Set rsSusuran = New ADODB.Recordset
    rsSusuran.Open "Select * from tSusuran where KdPenyakit='001'",
CN, adOpenDynamic, adLockOptimistic
    If Not rsSusuran.EOF = True Then
        bolSusur1 = True
        Do While Not rsSusuran.EOF = True
            jum001 = jum001 + 1
            rsSusuran.MoveNext
            If rsSusuran.EOF Then
                Exit Do
            End If
        Loop
    Else
        bolSusur1 = False
        jum001 = 0
    End If

Set rsSusuran = New ADODB.Recordset
    rsSusuran.Open "Select * from tSusuran where KdPenyakit='002'",
CN, adOpenDynamic, adLockOptimistic
    If Not rsSusuran.EOF = True Then
        bolSusur2 = True
        Do While Not rsSusuran.EOF = True
            jum002 = jum002 + 1
            rsSusuran.MoveNext
            If rsSusuran.EOF Then
                Exit Do
            End If
        Loop
    Else
        bolSusur2 = False
        jum002 = 0
    End If

Set rsSusuran = New ADODB.Recordset
    rsSusuran.Open "Select * from tSusuran where KdPenyakit='003'",
CN, adOpenDynamic, adLockOptimistic
    If Not rsSusuran.EOF = True Then
        bolSusur3 = True
        Do While Not rsSusuran.EOF = True
            jum003 = jum003 + 1
            rsSusuran.MoveNext
            If rsSusuran.EOF Then
                Exit Do
            End If
        Loop
    Else

```

```

        bolSusur3 = False
        jum003 = 0
    End If
Set rsSusuran = New ADODB.Recordset
rsSusuran.Open "Select * from tSusuran where KdPenyakit='004'",
CN, adOpenDynamic, adLockOptimistic
If Not rsSusuran.EOF = True Then
    bolSusur4 = True
    Do While Not rsSusuran.EOF = True
        jum004 = jum004 + 1
        rsSusuran.MoveNext
        If rsSusuran.EOF Then
            Exit Do
        End If
    Loop
Else
    bolSusur4 = False
    jum004 = 0
End If
Set rsSusuran = New ADODB.Recordset
rsSusuran.Open "Select * from tSusuran where KdPenyakit='005'",
CN, adOpenDynamic, adLockOptimistic
If Not rsSusuran.EOF = True Then
    bolSusur5 = True
    Do While Not rsSusuran.EOF = True
        jum005 = jum005 + 1
        rsSusuran.MoveNext
        If rsSusuran.EOF Then
            Exit Do
        End If
    Loop
Else
    bolSusur5 = False
    jum005 = 0
End If
Set rsSusuran = New ADODB.Recordset
rsSusuran.Open "Select * from tSusuran where KdPenyakit='006'",
CN, adOpenDynamic, adLockOptimistic
If Not rsSusuran.EOF = True Then
    bolSusur6 = True
    Do While Not rsSusuran.EOF = True
        jum006 = jum006 + 1
        rsSusuran.MoveNext
        If rsSusuran.EOF Then
            Exit Do
        End If
    Loop
Else
    bolSusur6 = False
    jum006 = 0
End If
Set rsSusuran = New ADODB.Recordset
rsSusuran.Open "Select * from tSusuran where KdPenyakit='007'",
CN, adOpenDynamic, adLockOptimistic
If Not rsSusuran.EOF = True Then
    bolSusur7 = True
    Do While Not rsSusuran.EOF = True
        jum007 = jum007 + 1
        rsSusuran.MoveNext
        If rsSusuran.EOF Then

```

```

        Exit Do
    End If
Loop
Else
    bolSusur7 = False
    jum007 = 0
End If
Set rsSusuran = New ADODB.Recordset
rsSusuran.Open "Select * from tSusuran where KdPenyakit='008'",
CN, adOpenDynamic, adLockOptimistic
If Not rsSusuran.EOF = True Then
    bolSusur8 = True
    Do While Not rsSusuran.EOF = True
        jum008 = jum008 + 1
        rsSusuran.MoveNext
    If rsSusuran.EOF Then
        Exit Do
    End If
    Loop
Else
    bolSusur8 = False
    jum008 = 0
End If
Set rsSusuran = New ADODB.Recordset
rsSusuran.Open "Select * from tSusuran where KdPenyakit='009'",
CN, adOpenDynamic, adLockOptimistic
If Not rsSusuran.EOF = True Then
    bolSusur9 = True
    Do While Not rsSusuran.EOF = True
        jum009 = jum009 + 1
        rsSusuran.MoveNext
    If rsSusuran.EOF Then
        Exit Do
    End If
    Loop
Else
    bolSusur9 = False
    jum009 = 0
End If

Set rsSusuran = New ADODB.Recordset
rsSusuran.Open "Select * from tSusuran where KdPenyakit='010'",
CN, adOpenDynamic, adLockOptimistic
If Not rsSusuran.EOF = True Then
    bolSusur10 = True
    Do While Not rsSusuran.EOF = True
        jum010 = jum010 + 1
        rsSusuran.MoveNext
    If rsSusuran.EOF Then
        Exit Do
    End If
    Loop
Else
    bolSusur10 = False
    jum010 = 0
End If

Set rsSimpan001 = New ADODB.Recordset
Set rsSimpan002 = New ADODB.Recordset
Set rsSimpan003 = New ADODB.Recordset

```



```

Set rsSimpan004 = New ADODB.Recordset
Set rsSimpan005 = New ADODB.Recordset
Set rsSimpan006 = New ADODB.Recordset
Set rsSimpan007 = New ADODB.Recordset
Set rsSimpan008 = New ADODB.Recordset
Set rsSimpan009 = New ADODB.Recordset
If bolSusur1 = True Then
    rsSimpan001.Open "tRangking", CN, 1, 2
    rsSimpan001.AddNew
    rsSimpan001!KdPenyakit = "001"
    rsSimpan001!jum = jum001
    rsSimpan001.Update
End If

If bolSusur2 = True Then
    rsSimpan002.Open "tRangking", CN, 1, 2
    rsSimpan002.AddNew
    rsSimpan002!KdPenyakit = "002"
    rsSimpan002!jum = jum002
    rsSimpan002.Update
End If

If bolSusur3 = True Then
    rsSimpan003.Open "tRangking", CN, 1, 2
    rsSimpan003.AddNew
    rsSimpan003!KdPenyakit = "003"
    rsSimpan003!jum = jum003
    rsSimpan003.Update
End If

If bolSusur4 = True Then
    rsSimpan004.Open "tRangking", CN, 1, 2
    rsSimpan004.AddNew
    rsSimpan004!KdPenyakit = "004"
    rsSimpan004!jum = jum004
    rsSimpan004.Update
End If

If bolSusur5 = True Then
    rsSimpan005.Open "tRangking", CN, 1, 2
    rsSimpan005.AddNew
    rsSimpan005!KdPenyakit = "005"
    rsSimpan005!jum = jum005
    rsSimpan005.Update
End If

If bolSusur6 = True Then
    rsSimpan006.Open "tRangking", CN, 1, 2
    rsSimpan006.AddNew
    rsSimpan006!KdPenyakit = "006"
    rsSimpan006!jum = jum006
    rsSimpan006.Update
End If

If bolSusur7 = True Then
    rsSimpan007.Open "tRangking", CN, 1, 2
    rsSimpan007.AddNew
    rsSimpan007!KdPenyakit = "007"
    rsSimpan007!jum = jum007

```

```

        rsSimpan007.Update
    End If

    If bolSusur8 = True Then
        rsSimpan008.Open "tRangking", CN, 1, 2
        rsSimpan008.AddNew
        rsSimpan008!KdPenyakit = "008"
        rsSimpan008!jum = jum008
        rsSimpan008.Update
    End If

    If bolSusur9 = True Then
        rsSimpan009.Open "tRangking", CN, 1, 2
        rsSimpan009.AddNew
        rsSimpan009!KdPenyakit = "009"
        rsSimpan009!jum = jum009
        rsSimpan009.Update
    End If

    If bolSusur10 = True Then
        rsSimpan010.Open "tRangking", CN, 1, 2
        rsSimpan010.AddNew
        rsSimpan010!KdPenyakit = "010"
        rsSimpan010!jum = jum010
        rsSimpan010.Update
    End If

    Call Segar
    grid.Visible = True
    Set rsRangking = New ADODB.Recordset
    rsRangking.Open "Select * from tRangking order by Jum desc", CN,
1, 2
    If Not rsRangking.EOF = True Then
        KdPeny = rsRangking!KdPenyakit
        lblKdPenyakit.Caption = KdPeny
        Set rsPenyakit = New ADODB.Recordset
        rsPenyakit.Open "Select * from tPenyakit where KdPenyakit='"
& KdPeny & "'", CN, adOpenDynamic, adLockOptimistic
        If Not rsPenyakit.EOF = True Then
            lblNmPenyakit.Caption = rsPenyakit!NmPenyakit '& " " &
Format((rsRangking!jum / 9) * 100, "##.##")
        End If
    Else
        lblNmPenyakit.Caption = "Penyakit tidak ditemukan"
    End If
End Sub

Sub bersih()
    grid.Visible = False
    lblNmPenyakit = ""
    no = 0
    jum001 = 0
    jum002 = 0
    jum003 = 0
    jum004 = 0
    jum005 = 0
    jum006 = 0
    jum007 = 0
    jum008 = 0
    jum009 = 0
    jum010 = 0
End Sub

```

```

Option Explicit
Dim rsPenyakit As Recordset
Dim rsInfo As Recordset
Dim KdPenyakit As String * 3
Sub CariPenyakit()
List1.Clear
List2.Clear
Set rsPenyakit = New ADODB.Recordset
rsPenyakit.Open "Select * from tPenyakit Where KdPenyakit =" &
KdPenyakit & "'", CN, adOpenDynamic, adLockOptimistic
If Not rsPenyakit.EOF = True Then
lblNmPenyakit.Caption = rsPenyakit!NmPenyakit
Else
lblNmPenyakit.Caption = ""
End If

If KdPenyakit = "007" Then
Picture1.Picture = LoadPicture(App.Path &
"\Foto\FotoDermatitisAtropik.bmp")
ElseIf KdPenyakit = "008" Then
Picture1.Picture = LoadPicture(App.Path &
"\Foto\FotoMikosisSuperfisialis.bmp")
Else
Picture1.Picture = Nothing
End If

Set rsInfo = New ADODB.Recordset
rsInfo.Open "Select * from tInfo where kdpenyakit='" & KdPenyakit
& "'" order by KdPenyakit asc", CN, adOpenDynamic, adLockOptimistic
If Not rsInfo.EOF = True Then
Do While Not rsInfo.EOF = True
List1.AddItem rsInfo!info1
rsInfo.MoveNext
If rsInfo.EOF = True Then
Exit Do
End If
Loop
End If

Set rsInfo = New ADODB.Recordset
rsInfo.Open "Select * from tInfo where kdpenyakit='" & KdPenyakit
& "'" order by KdPenyakit asc", CN, adOpenDynamic, adLockOptimistic
If Not rsInfo.EOF = True Then
List2.AddItem rsInfo!info2
End If

End Sub

Private Sub cmdLanjut_Click()
Me.Hide
Unload Me
End Sub

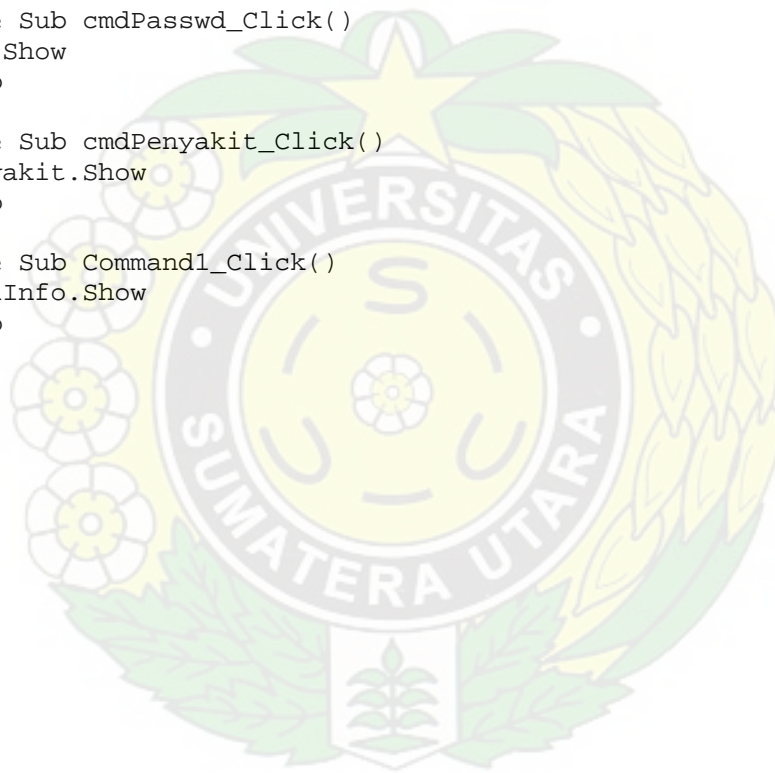
Private Sub Form_Activate()
KdPenyakit = frmHasil.lblKdPenyakit.Caption
List1.Clear
List2.Clear
CariPenyakit
End Sub

```

```
Private Sub Form_Load()  
Koneksi  
End Sub
```



```
Private Sub cmdAturan_Click()  
frmDataAturan.Show  
End Sub  
  
Private Sub and Pertanyaan_(Click)  
frmPertanyaan.Show  
End Sub  
  
Private Sub cmdGejala_Click()  
frmGejala.Show  
End Sub  
  
Private Sub cmdKeluar_Click()  
Unload Me  
frmMenu.Show  
End Sub  
  
Private Sub cmdPasswd_Click()  
frmPwd.Show  
End Sub  
  
Private Sub cmdPenyakit_Click()  
frmPenyakit.Show  
End Sub  
  
Private Sub Command1_Click()  
frmDataInfo.Show  
End Sub
```



```

Option Explicit
Private Sub cmdCancel_Click()
Unload Me
End Sub

Private Sub cmdOK_Click()
On Error Resume Next
Set rsCari = New ADODB.Recordset
    rsCari.Open "Select * from tPassword where User='" & txtUid.Text
& "'" and Passwd='" & txtPass.Text & "'", CN, adOpenDynamic,
adLockOptimistic
    If rsCari.EOF = True Then
        MsgBox "Logon Gagal"
        txtPass = ""
        txtUid = ""
        txtUid.SetFocus
    Else
        Me.Hide
        frmMenuUpdate.Show
    End If
End Sub

Private Sub Form_Activate()
txtPass = ""
txtUid = ""
txtUid.SetFocus
End Sub
Private Sub Form_Load()
Koneksi
End Sub

Private Sub txtPass_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    cmdOk.Enabled = True
    cmdOk.SetFocus
End If
End Sub

Private Sub txtUid_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    txtPass.Locked = False
    txtPass.SetFocus
End If
End Sub

```



```

Option Explicit
Dim rsPenyakit As Recordset
Private Sub cmdBatal_Click()
Cari = False
Tambah = False
Ubah = False
mati
bersih
cmdRubah.Enabled = False
cmdHapus.Enabled = False
cmdcari.Enabled = True
cmdtambah.Visible = True
cmdtambah.Enabled = True
cmdSimpan.Visible = False
cmdtambah.SetFocus
End Sub

Private Sub cmdcari_Click()
Cari = True
Tambah = False
cmdtambah.Enabled = False
cmdcari.Enabled = False
txtNama.Locked = False
txtNama.SetFocus
End Sub

Private Sub CmdHapus_Click()
cmdHapus.Enabled = False
cmdRubah.Enabled = False
Set rsHapus = New ADODB.Recordset
rsHapus.Open "Delete from tPenyakit where KdPenyakit ='" &
txtKdPenyakit.Text & "'", CN, adOpenDynamic, adLockOptimistic
bersih
mati
Segar
Adodcl.Refresh
grid.Refresh
MsgBox "Data sudah diHapus"
cmdtambah.Enabled = True
cmdcari.Enabled = True
cmdtambah.Visible = True
cmdSimpan.Visible = False
End Sub

Private Sub cmdKeluar_Click()
Unload Me
End Sub
Private Sub CmdRubah_Click()
cmdHapus.Enabled = False
cmdRubah.Enabled = False
cmdUpdate.Visible = True
cmdRubah.Visible = False
Ubah = True
txtNama.Locked = False
txtNama.SetFocus
End Sub

Private Sub cmdSimpan_Click()
If Tambah = True Then
Set rsSimpan = New ADODB.Recordset

```

```

        rsSimpan.Open "Insert into tPenyakit " & "
(KdPenyakit,NmPenyakit) " & " values ('" & txtKdPenyakit.Text & "','"
& txtNama.Text & "')" , CN, adOpenDynamic, adLockOptimistic
        bersih
        MsgBox "Data sudah disimpan"
        mati
        bersih
        Tambah = False
        Segar
        Adodc1.Refresh
        grid.Refresh
        cmdtambah.Enabled = True
        cmdcari.Enabled = True
        cmdtambah.Visible = True
        cmdSimpan.Visible = False
        cmdtambah.SetFocus
End If
End Sub
Sub Segar()
Set rsSegar = New ADODB.Recordset
        rsSegar.Open "Select * from tPenyakit order by KdPenyakit asc",
CN, 1, 3
grid.ColWidth(0) = 0
grid.ColWidth(1) = 800
grid.ColWidth(2) = 8000

grid.TextMatrix(0, 0) = ""
grid.TextMatrix(0, 1) = "Kode"
grid.TextMatrix(0, 2) = "Nama Penyakit"

End Sub
Private Sub cmdTambah_Click()
Tambah = True
Cari = False
cmdtambah.Enabled = False
cmdcari.Enabled = False
txtKdPenyakit.Locked = False
txtKdPenyakit.SetFocus
End Sub

Private Sub cmdUpdate_Click()
cmdUpdate.Visible = False
cmdRubah.Visible = True
cmdRubah.Enabled = False
If Ubah = True Then
        Set rsRubah = New ADODB.Recordset
        rsRubah.Open "Update tPenyakit set NmPenyakit = '" &
txtNama.Text & "' where KdPenyakit='" & txtKdPenyakit.Text & "'", CN,
adOpenDynamic, adLockOptimistic
        bersih
        MsgBox "Data sudah diUbah"
        mati
        Ubah = False
        Adodc1.Refresh
        grid.Refresh
        Segar
        cmdtambah.Enabled = False
        cmdSimpan.Visible = False
        cmdcari.Enabled = True
        cmdtambah.Visible = True

```

```

End If
End Sub

Private Sub Form_Activate()
grid.ColWidth(0) = 0
grid.ColWidth(1) = 800
grid.ColWidth(2) = 8000

grid.TextMatrix(0, 0) = ""
grid.TextMatrix(0, 1) = "Kode"
grid.TextMatrix(0, 2) = "Nama Penyakit"

cmdSimpan.Visible = False
cmdtambah.Visible = True
cmdtambah.Enabled = False
cmdcari.Enabled = True
cmdRubah.Visible = True
cmdUpdate.Visible = False
cmdRubah.Enabled = False
cmdHapus.Enabled = False
mati
bersih
cmdtambah.Enabled = True
cmdcari.Enabled = True
cmdtambah.SetFocus
End Sub
Private Sub Form_Load()
Koneksi
End Sub
Sub bersih()
txtKdPenyakit.Text = ""
txtNama.Text = ""
End Sub
Sub mati()
txtKdPenyakit.Locked = True
txtNama.Locked = True
End Sub

Private Sub txtKdPenyakit_Change()
If Len(txtKdPenyakit) < 3 Then
Exit Sub
Else
Set rsPenyakit = New ADODB.Recordset
rsPenyakit.Open "Select * from tPenyakit where KdPenyakit='"
& txtKdPenyakit.Text & "'", CN, adOpenDynamic, adLockOptimistic
If Cari = True Then
If rsPenyakit.EOF = True Then
MsgBox "Data tidak ada"
txtKdPenyakit.Text = ""
txtKdPenyakit.SetFocus
Else
txtNama.Text = UCase(rsPenyakit!NmPenyakit)
Cari = False
cmdRubah.Enabled = True
cmdHapus.Enabled = True
cmdSimpan.Visible = False
cmdtambah.Visible = True
cmdtambah.Enabled = False
cmdRubah.SetFocus
End If
End If

```

```

        ElseIf Tambah = True Then
            If rsPenyakit.EOF = True Then
                txtNama.Locked = False
                txtNama.SetFocus
            Else
                MsgBox "Data sudah ada"
                txtKdPenyakit.Text = ""
                txtKdPenyakit.SetFocus
            End If
        End If
    End If
End Sub

Private Sub txtNama_KeyPress(KeyAscii As Integer)
    If KeyAscii = 13 Then
        txtNama = UCase(txtNama)
        If Tambah = True Then
            txtNama = UCase(txtNama)
            cmdtambah.Visible = False
            cmdSimpan.Visible = True
            cmdSimpan.Enabled = True
            cmdSimpan.SetFocus
        ElseIf Ubah = True Then
            cmdRubah.Visible = False
            cmdUpdate.Visible = True
            cmdUpdate.SetFocus
        ElseIf Cari = True Then
            Set rsPenyakit = New ADODB.Recordset
            rsPenyakit.Open "Select * from tPenyakit where NmPenyakit='"
            & txtNama.Text & "'", CN, adOpenDynamic, adLockOptimistic
            If rsPenyakit.EOF = True Then
                MsgBox "Penyakit tidak ada"
                txtNama.Text = ""
                txtNama.SetFocus
            Else
                txtKdPenyakit.Text = UCase(rsPenyakit!KdPenyakit)
                Cari = False
                cmdRubah.Enabled = True
                cmdHapus.Enabled = True
                cmdSimpan.Visible = False
                cmdtambah.Visible = True
                cmdtambah.Enabled = False
                cmdRubah.SetFocus
            End If
        End If
    End If
    KeyAscii = Asc(UCase(Chr(KeyAscii)))
End Sub

```

```

Dim rsPasswd As Recordset
Dim rsSimpan As Recordset
Dim rsCari As Recordset
Dim rsUbah As Recordset
Dim rsHapus As Recordset
Dim Tambah As Boolean
Dim Cari As Boolean
Dim Ubah As Boolean
Sub Segar()
Set rsSegar = New ADODB.Recordset
    rsSegar.Open "Select * from tPassword order by User asc", CN, 1,
2

End Sub
Private Sub cmdBatal_Click()
txtUser.Text = ""
txtPass.Text = ""
Ubah = False
Tambah = False
Cari = False
cmdRubah.Enabled = False
cmdHapus.Enabled = False
cmdcari.Enabled = True
cmdtambah.Enabled = True
cmdtambah.SetFocus
End Sub

Private Sub cmdcari_Click()
Cari = True
cmdtambah.Enabled = False
cmdcari.Enabled = False
txtUser.Locked = False
txtUser.SetFocus
End Sub

Private Sub CmdHapus_Click()
Set rsHapus = New ADODB.Recordset
    rsHapus.Open "Delete from tPassword where user='" & txtUser.Text
& "'" and passwd='" & txtPass.Text & "'", CN, adOpenDynamic,
adLockOptimistic
    txtUser.Text = ""
    txtPass.Text = ""
    MsgBox "User sudah di hapus ..."
    cmdRubah.Enabled = False
    cmdHapus.Enabled = False
    cmdcari.Enabled = True
    cmdtambah.Enabled = True
    cmdtambah.SetFocus

End Sub

Private Sub cmdKeluar_Click()
Unload Me
End Sub

Private Sub CmdRubah_Click()
Ubah = True
cmdRubah.Enabled = False
cmdHapus.Enabled = False
txtPass.Locked = False

```



```

txtPass.SetFocus
End Sub

Private Sub cmdTambah_Click()
Tambah = True
cmdtambah.Enabled = False
cmdcari.Enabled = False
txtUser.Locked = False
txtUser.SetFocus
End Sub

Private Sub Form_Activate()
txtPass.Locked = True
txtUser.Locked = True
txtUser.Text = ""
txtPass.Text = ""
Ubah = False
Tambah = False
Cari = False
cmdRubah.Enabled = False
cmdHapus.Enabled = False
cmdcari.Enabled = True
cmdtambah.Enabled = True
cmdtambah.SetFocus
End Sub
Sub mati()
txtPass.Locked = True
txtUser.Locked = True
cmdRubah.Enabled = False
cmdHapus.Enabled = False
End Sub
Sub bersih()
txtUser.Text = ""
txtPass.Text = ""
End Sub
Private Sub Form_Load()
Koneksi
End Sub
Private Sub txtPass_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
Set rsPasswd = New ADODB.Recordset
rsPasswd.Open "Select * from tPassword where user='" &
txtUser.Text & "' and passwd='" & txtPass.Text & "'", CN,
adOpenDynamic, adLockOptimistic
If Tambah = True Then
If rsPasswd.EOF = True Then
Set rsSimpan = New ADODB.Recordset
rsSimpan.Open "select * from tPassword", CN,
adOpenDynamic, adLockOptimistic
rsSimpan.AddNew
rsSimpan.Fields("user") = txtUser.Text
rsSimpan.Fields("Passwd") = txtPass.Text
rsSimpan.Update
txtUser.Text = ""
txtPass.Text = ""
bersih
MsgBox "Data sudah tersimpan"
Tambah = False
mati
cmdtambah.Enabled = True

```

```

        cmdcari.Enabled = True
        cmdtambah.SetFocus
    Else
        MsgBox "User sudah ada ..."
        txtPass.Text = ""
        txtUser.Text = ""
        txtUser.SetFocus
    End If
    ElseIf Ubah = True Then
        Set rsUbah = New ADODB.Recordset
        rsUbah.Open "Update tPassword Set passwd='" &
txtPass.Text & "' where user='" & txtUser.Text & "'", CN,
adOpenDynamic, adLockOptimistic
        txtUser.Text = ""
        txtPass.Text = ""
        MsgBox "Data sudah Ubah"
        Ubah = False
        mati
        bersih
        cmdtambah.Enabled = True
        cmdcari.Enabled = True
        cmdtambah.SetFocus
    End If
End If
End Sub

Private Sub txtUser_KeyPress(KeyAscii As Integer)
If KeyAscii = 13 Then
    If Tambah = True Then
        Set rsCari = New ADODB.Recordset
        rsCari.Open "Select * from tPassword where user='" &
txtUser.Text & "'", CN, adOpenDynamic, adLockOptimistic
        If rsCari.EOF = True Then
            txtPass.Locked = False
            txtPass.SetFocus
        Else
            MsgBox "User sudah ada ..."
            txtUser.Text = ""
            txtPass.Text = ""
            txtUser.SetFocus
        End If
    ElseIf Cari = True Then
        Set rsCari = New ADODB.Recordset
        rsCari.Open "Select * from tPassword where user='" &
txtUser.Text & "'", CN, adOpenDynamic, adLockOptimistic
        If rsCari.EOF = True Then
            MsgBox "User belum ada ..."
            txtUser.Text = ""
            txtUser.SetFocus
        Else
            txtPass.Text = rsCari.Fields("passwd")
            cmdRubah.Enabled = True
            cmdHapus.Enabled = True
            cmdRubah.SetFocus
        End If
    End If
End If
End Sub

```

```
Private Sub Command1_Click()  
Me.Hide  
Unload Me  
End Sub
```

