Syamsul Islam. 2008. Sistem Pakar Untuk Mendiagnosis Gangguan dan Penyakit Kulit Pada Manusia Menggunakan Faktor Kepastian. Sekripsi Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya. Pembimbing: Arief Andy Soebroto, ST., MKom dan dr. Aunur Rofig, SpKK

Kulit mempunyai fungsi penting pada tubuh manusia, yaitu sebagai tempat pembentukan vitamin D, pengatur suhu badan, perlindungan dan pertahanan tubuh. Banyak faktor yang dapat mempengaruhi kondisi kulit, seperti ras, iklim, polusi, genetik, hormonal, usia dan gaya hidup. Penanganan awal pada penderita yang terindikasi penyakit kulit secara tepat dan cepat mutlak dilakukan. Penderita harus memiliki pengetahuan tentang gejala - gejala yang timbul akibat suatu penyakit, pengetahuan ini dapat diperoleh dari buku-buku atau situs di Internet. Namun untuk mempelajarinya perlu waktu yang lama dan tidak mudah. Untuk itu diperlukan suatu sistem yang lebih praktis dan memiliki kemampuan layaknya seorang dokter dalam mendiagnosis gejala yang timbul.

Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar. Pengetahuan dan kemampuan seorang dokter dapat dialihkan ke dalam sistem pakar, sehingga sistem pakar tersebut memiliki keahlian menyerupai seorang dokter dalam melakukan diagnosis. Sistem pakar yang memiliki pengetahuan seorang dokter ini, dapat membantu mereka untuk mengetahui penyakit yang diderita dari gejala yang timbul, serta tindakan apa yang harus dilakukan. Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia Menggunakan Faktor Kepastian dibuat dengan menggunakan Borland Delphi 7 dan Ms. Access 2003 sebagai sarana penyimpanan database.

Aturan-aturan yang terdapat dalam sistem pakar ini dapat dengan muda diperbaharui karena pakar dengan mudah memasukkan pengetahuan dengan mengisikan tabel yang terdapat dalam data file. Dari hasil pengujian test turing, sistem pakar ini memiliki tingkat akurasi sebesar 87,5%.

Kata kunci: Kecerdasan Buatan, Sistem Pakar, Borland Delphi 7, Ms Access 2003, **Turing Test** 

### BRAWIJAY

### BAB I PENDAHULUAN

### 1.1 Latar Belakang

Kulit mempunyai fungsi penting pada tubuh manusia, yaitu sebagai tempat pembentukan vitamin D, pengatur suhu badan, perlindungan dan pertahanan tubuh. Banyak faktor yang dapat mempengaruhi kondisi kulit, seperti ras, iklim, polusi, genetik, hormonal, usia dan gaya hidup. Menjaga kondisi kulit yang baik memerlukan perhatian khusus, agar kulit terhindar dari segala factor yang mengakibatkan kesehatan kulit menurun.

Penanganan awal pada penderita yang terindikasi penyakit kulit secara tepat dan cepat mutlak dilakukan. Penderita harus memiliki pengetahuan tentang gejala - gejala yang timbul akibat suatu penyakit, pengetahuan ini dapat diperoleh dari buku-buku atau situs di Internet. Namun untuk mempelajarinya perlu waktu yang lama dan tidak mudah. Untuk itu diperlukan suatu sistem yang lebih praktis dan memiliki kemampuan layaknya seorang dokter dalam mendiagnosis gejala yang timbul.

Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidangnya. Sistem pakar berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh pakar. Seorang dokter dapat dikatakan seorang pakar karena ia merupakan orang yang ahli di bidangnya dan dia mempunyai kemampuan seorang yang orang lain tidak memilikinya. Pengetahuan dan kemampuan seorang dokter dapat dialihkan ke dalam sistem pakar, sehingga sistem pakar tersebut memiliki keahlian menyerupai seorang dokter dalam melakukan diagnosis. Sistem pakar yang memiliki pengetahuan seorang dokter ini, dapat membantu mereka untuk mengetahui penyakit yang diderita. Cukup dengan menjalankan aplikasi sistem pakar di rumah, mereka dapat mengetahui penyakit apa yang diderita dari gejala yang timbul, serta tindakan apa yang harus dilakukan [KUS-06:11];

Kepastian data dalam mendiagnosis suatu gejala penyakit harus dimiliki oleh sebuah sistem pakar. Untuk itu diperlukan suatu metode yang mampu meminimalisir ketidak pastian data yaitu dengan menggunakan faktor kepastian. Hasil yang tidak pasti

disebabkan oleh dua faktor, yaitu aturan yang tidak pasti dan tidak pastinya jawaban pengguna atas pertanyaan yang diajukan oleh sistem [KUS-06:39];

Penggunaan bahasa pemrograman Borland Delphi dan basis data seperti Ms Acces menjadi aplikasi visual bersifat dinamis dengan menyediakan fasilitas objek yang kuat dan lengkap, hal ini memudahkan programer dalam membuat program.

Borland Delphi merupakan statu bahasa pemrograman yang memberikan berbagai fasilitas pembuatan aplikasi visual. Keunggulan bahasa pemrograman ini terletak pada productivitas, kualitas, pengembangan perangkat lunak, kecepatan kompilasi, pola desain yang menarik, serta diperkuat dengan pemrogramanya yang terstruktur. Keunggulan lain dari delphi adalah dapat digunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis Windows.

Ms. Access merupakan salah satu contoh produk RDBMS yang sangat popular dilingkungan Windows. RDBMS (*Relational Database Management System*) merupakan salah satu macam DBMS yang menggunakan model basis data relasional atau dalam bentuk tabel-tabel yang saling terhubungkan. Pada Ms.Access, sebuah database disimpan dalam sebuah berkas dengan ekstensi "MDB" Di dalam berkas inilah semua objek yang terkait dengan database, termasuk semua tabel, disimpan. Pada Microsoft Access, istilah kolom yang biasa dipakai pada basis data relasional disebut *field* dan baris biasa disebut *record*:

### 1.2 Rumusan Masalah

Mengacu pada permasalahan yang telah diuraikan pada latar belakang, maka rumusan masalah dapat disusun sebagai berikut:

- Merancang dan membuat suatu sistem pakar yang dapat mendiagnosis gangguan dan penyakit kulit pada manusia
- 2. Merancang dan membuat antarmuka sistem pakar berbasis windows untuk mendiagnosis gangguan dan penyakit kulit dengan menggunakan faktor kepastian (*certainty factor*).
- 3. Merancang dan membangun basis pengetahuan dengan menggunakan *Rule-Base Reasioning* pada sistem pakar untuk mendiagnosis gangguan dan penyakit kulit pada manusia.
- 4. Merancang Motor Inferensi yang digunakan untuk melakukan penalaran dengan menggunakan metode *Forward Chaning*.

5. Melakukan pengujian terhadap *system* yang telah dibangun, penujian dilakukan dengan metode *white box dan black box* (menggunakan basic path).

### 1.3 Batasan Masalah

Supaya yang dihasilkan tepat sasaran maka pembuatan aplikasi ini dibatasi sebagai berikut:

- a. Diagnosis dilakukan berdasarkan pada pemeriksaan gejala fisik.
- b. Hanya mendiagnosis gangguan dan penyakit pada kulit manusia
- c. Data yang digunakan sesuai dengan hasil wawancara dengan dokter kulit.
- d. Jenis gangguan dan penyakit hanya delapan macam
- e. Menggunakan Borland Delphi 7 untuk mengimplementasikan rancangan sistem pakar ke dalam suatu bentuk aplikasi.
- f. Menggunakan Ms. Access sebagai basis data

### 1.4 Tujuan Penulisan

Tujuan penulisan skripsi ini adalah untuk merancang dan membangun suatu sistem pakar untuk mendiagnosis gangguan dan penyakit pada kulit manusia berdasarkan basis pengetahuan yang tersimpan di dalamnya dengan menggunakan faktor kepastian untuk mengatasi masalah ketidakpastian data.

### 1.5 Manfaat

Manfaat yang dapat diambil dari sistem pakar untuk mendiagnosis gangguan dan penyakit kulit pada manusia dengan menggunakan faktor kepastian yang ini antara lain:

- Membantu mereka yang mengalami gangguan dan penyakit kulit, mereka dapat mengetahui gangguan dan penyakit kulit yang dideritanya serta dapat mengetahui tindakan apa yang harus dilakukan untuk mengatasi gangguan kesehatan tersebut.
- 2. Memungkinkan paramedis non dokter kulit berlaku sebagai dokter kulit sehingga pada suatu saat dibutuhkan dapat mengambil tindakan tepat sesuai dengan pengetahuan dan prosedur inferensi yang dilakukan oleh seorang dokter kulit.
- 3. Meningkatkan kualitas keputusan yang akan diambil dalam tindakan medis yang akan dilaksanakan.

### 1.6 Sistematika Penulisan

Pembahasan Tugas Akhir ini terdiri dari 7 bab yang disusun dengan sistematika penulisan sebagai berikut :

### Bab I :Pendahuluan

Pada bab ini akan dibahas latar belakang masalah yang melandasi perlunya sistem pakar untuk mendiagnosis gangguan dan penyakit kulit pada manusia, kemudian perumusan masalah yang akan menjelaskan masalah yang dibahas, dilanjutkan dengan batasan masalah yang memberikan batasan dan arah penulisan, tujuan penulisan, manfaat, metode penelitian serta sistematika penulisan.

### Bab II: Dasar Teori

Dalam bab ini akan dijelaskan tentang teori-teori yang menunjang penulisan laporan tugas akhir ini, seperti Kecerdasan Buatan, Sistem Pakar, macam-macam Gangguan dan penyakit kulit pada manusia, Basis Data serta bahasa pemrograman yang akan dipakai nanti.

### Bab III : Metodologi Penelitian

Bab ini menjelaskan secara singkat mengenai metode atau langkah-langkah yang digunakan dalam penelitian tugas akhir.

### Bab IV: Perancangan

Pada bab ini dijelaskan tentang proses perancangan sistem. Metode yang digunakan adalah analisis terstruktur yaitu dengan membuat :

- Perancangan Rule
- Inference Tree
- ER diagram
- Data Flow Diagram (DFD)
- State Transition Diagram (STD)

### Bab V: Implementasi Sistem

Pada bab ini akan dibahas mengenai algoritma sistem dan hasil dari program yang dibuat. Dalam tugas akhir ini program dibuat dengan menggunakan bahasa pemrograman Borland Delphi 7 serta Ms access 2003 sebagai sarana untuk penyimpanan basis datanya.

Bab ini membahas tentang pengujian yang akan dilakukan. Pengujian meliputi pengujian algoritma menggunakan teknik *white-box testing dan black box*, dan pengujian tingkat akurasi sistem yang dilakukan oleh dokter kulit dengan menggunakan *Turing Test*.

### **Bab VII: Penutup**

Pada bab ini akan dikemukakan kesimpulan-kesimpulan yang dapat diambil oleh penulis dari hasil perancangan, implementasi, dan pengujian aplikasi Sistem Pakar untuk Mendiagnosis Gangguan dan penyakit kulit pada manusia dengan Menggunakan Faktor Kepastian serta saran yang menurut penulis sangat diperlukan untuk pengembangan aplikasi ini pada masa yang akan datang.



BAB II TINJAUAN PUSTAKA Pada bab ini membahas kajian pustaka dan dasar teori yang digunakan untuk penulisan skripsi dan pengembangan aplikasi sistem pakar. Kajian pustaka diperlukan untuk melakukan kajian terhadap karya ilmiah yang berkaitan dengan skripsi ini. Kajian pustaka yang dipaparkan adalah mengenai penerapan Sistem Pakar dalam mendiagnosis penyakit kulit pada manusia.

Dasar teori yang diperlukan berdasar kajian pustaka untuk penyusunan skripsi ini diantaranya adalah teori tentang Kecerdasan Buatan, Sistem Pakar, Borland Delphi, Ms Acces.

### 2.1 Penerapan Sistem Pakar Dalam Mendiagnosis Penyakit

GPS (*General Purpose Problem-Solver*) merupakan sebuah predecessor menuju Sistem Pakar, dikembangkan oleh Newel dan Simon yang bertujuan untuk menciptakan mesin yang cerdas. GPS ini mengalami kegagalan dikarenakan cakupanya yang terlalu luas sehingga terkadang justru meninggalkan pengetahuan-pengetauan penting yang seharusnya disediakan [DWI-03:110].

Sistem pakar yang dikembangkan setelah kegagalan GPS (*General Purpose Problem-Solver*) dibuat pada bidang pengetahuan tertentu untuk suatu kepakaran tertentu yang mendekati kemampuan manusia disalah satu bidang. Sebuah sistem pakar memiliki dua komponen utama yaitu basis pengetahuan dan mesin inferensi. Basis pengetahuan merupakan tempat penyimpanan pengetahuan dalam memori komputer, dimana pengetahuan ini diambil dari pengetahuan pakar [SRI-08:3].

Mesin inferensi adalah suatu program atau olgaritma berisi metodologi yang digunakan untuk melakukan penalaran dan memformulasikan konklusi terhadap informasi-informasi dalam basis pengetahuan. Kemampuan untuk menalar dan merekomendasi inilah yang membedakan sistem pakar dengan sistem konvensional [DWI-03:112]:

- Pada sistem konvensional informasi dan pemrosesanya jadi satu dengan program, sedangkan pada sistem pakar terpisah.
- Pegubahan program pada sistem konvensional cukup sulit, sedangkan pada sistem pakar mudah.
- Sistem hanya akan beroprasi jika sistem sudah lengkap, sedangkan pada sistem pakar dapat beroperasi hanya dengan beberapa aturan.

- Eksekusi dilakukan langkah demi langkah, pada sistem pakar eksekusi dilakukan pada keseluruhan basis pengetahuan.
- Menggunakan data, sistem pakar menggunakan pengetahuan.
- Tujuan utama sistem konvensional adalah efisiensi, tujuan utam sistem pakar adalah efectivitas.

Secara garis besar, banyak manfaat yang dapat diambil dengan adanya sistem pakar, antara lain [KUS-06:15]:

- Membuat seorang yang awam dapat bekerja seperti layaknya seorang pakar.
- Dapat bekerja dengan informasi yang tidak lengkap atau tidak pasti.
- Meningkatkan output dan produktivitas. Dapat bekerja lebih cepat dari manusia. Keuntungan ini berarti mengurangi jumlah pekerja yang dibutuhkan, dan akhirnya mereduksi biaya.
- Meningkatkan kualitas
- Menyediakan nasihat yang konsisten dan dapat mengurangi tingkat kesalahan.
- Membuat peralatan yang kompleks lebih mudah dioprasikan karena dapat melatih pekerja yang tidak berpengalaman.
- Handal (realibility).
- Tidak dapat lelah atau bosan. Juga konsisten dalam memberi jawaban dan selalu memberikan perhatian penuh.
- Memiliki kemampuan untuk memecahkan masalah yang kompleks.
- Memungkinkan pemindahan pengetahuan ke lokasi yang jauh serta memperluas jangkauan seorang pakar, dapat diperoleh dan dipakai dimana saja. Merupakan arsip yang terpercaya dari sebuah keahlian sehingga user seolah-olah berkonsultasi langsung dengan sang pakar meskipun mungkin sang pakar sudah pensiun.

Dari kajian tersebut maka judul sekripsi yang dikemukakan adalah "Sistem Pakar Untuk Mendiagnosis Gangguan dan Penyakit kulit Pada Manusia Menggunakan Faktor Kepastian".

### 2.2 Kecerdasan Buatan

Kecerdasan buatan merupakan salah satu bagian ilmu komputer yang mempelajari bagaimana cara membuat mesin (komputer) agar dapat melakukan pekerjaan seperti yang dilakukan oleh manusia. Kecerdasan Buatan (*Artificial Intelligence* atau *AI*) didefinisikan sebagai kecerdasan yang ditunjukkan oleh suatu entitas buatan. Sistem seperti ini umumnya dianggap komputer. Kecerdasan diciptakan dan dimasukkan ke dalam suatu mesin (komputer) agar dapat melakukan pekerjaan seperti yang dapat dilakukan manusia. Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar, permainan komputer, logika fuzzy, jaringan syaraf tiruan dan robotika [AOM-06].

Rich and Knight [1991] mendefinisikan Kecerdasan Buatan (AI) merupakan sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia [REK-91].

Tujuan dari kecerdasan buatan yaitu [KUS-06:4]:

- Tujuan utama: membuat mesin menjadi lebih pintar.
- Tujuan ilmiah : memahami apa itu kecerdasan.
- Tujuan entrepreneurial: membuat mesin lebih bermanfaat

Banyak hal yang kelihatannya sulit untuk kecerdasan manusia, tetapi untuk Informatika relatif tidak bermasalah. Seperti mentransformasikan persamaan, menyelesaikan persamaan integral, membuat permainan catur atau *Backgammon*. Di sisi lain, hal yang bagi manusia kelihatannya menuntut sedikit kecerdasan, sampai sekarang masih sulit untuk direalisasikan dalam informatika. Seperti pengenalan obyek/muka dan bermain sepakbola [AOM-06].

AI membentuk cabang yang sangat penting pada ilmu computer yaitu berhubungan dengan perilaku, pembelajaran dan adaptasi yang cerdas dalam sebuah mesin. Penelitian dalam AI menyangkut pembuatan mesin untuk mengotomatisasikan tugas-tugas yang membutuhkan perilaku cerdas. Termasuk contohnya adalah pengendalian, perencanaan dan penjadwalan, kemampuan untuk menjawab diagnosa dan pertanyaan pelanggan, serta pengenalan tulisan tangan, suara dan wajah. Hal-hal seperti itu telah menjadi disiplin ilmu tersendiri, yang memusatkan perhatian pada penyediaan solusi masalah kehidupan yang nyata. Sistem AI sekarang ini sering digunakan dalam bidang ekonomi, obat-obatan, teknik dan militer, seperti yang telah

dibangun dalam beberapa aplikasi perangkat lunak komputer rumah dan video game [AOM-06].

AI dapat dipandang dalam berbagai perspektif. Dari perspektif Kecerdasan AI adalah bagaimana membuat mesin yang "cerdas" dan dapat melakukan hal-hal yang sebelumnya dapat dilakukan oleh manusia. Dari perspektif bisnis, AI adalah sekelompok alat bantu yang berdaya guna, dan metodologi yang menggunakan alat bantu tersebut guna menyelesaikan masalah-masalah bisnis. Sedangkan dari perspektif pemrograman, AI termasuk didalamnya adalah studi tentang pemrograman simbolik, pemecahan masalah, proses pencarian.

### 2.2.1 Domain Penelitian dalam Kecerdasan Buatan

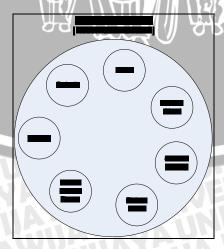
Beberapa bidang penelitian dalam kecerdasan buatan diantaranya

[KUS-06:5]:

- 1. Formal tasks (matematika, games)
- 2. Mundane task (perception, robotics, natural language, common sense, reasoning).
- 3. Expert tasks (financial analysis, medical diagnostics, engineering, scientific analysis, dll).

### 2.2.2 Ruang Lingkup Kecerdasan Buatan

Gambar 2.1 mununjukan bahwa kecerdasan buatan memiliki banyak ruang lingkup atau bidang. Salah satu bidangnya adalah sistem pakar yang menggabungkan pengetahuan dan penelusuran data untuk memecahkan masalah yang secara normal memerlukan keahlian manusia.



**Gambar 2.1** Ruang Lingkup *Artificial Intelegence* (Kecerdasan Buatan) Sumber: [AMU-05:2];

### 2.2.3 Perbandingan Kecerdasan Buatan dengan Kecerdasan Alamiah

Keuntungan kecerdasan buatan dibanding kecerdasan alamiah [KUS-06:6]:

- 1. Lebih Permanen.
- 2. Memberikan kemudahan dalam duplikasi dan penyebaran.
- 3. Relatif lebih murah dari kecerdasan alamiah.
- 4. Konsisten dan teliti.
- 5. Dapat didokumentasi.
- 6. Dapat mengerjakan beberapa tugas dengan lebih cepat dan lebih baik dibanding manusia.

Sedangkan keuntungan kecerdasan alamiah dibanding kecerdasan buatan adalah [KUS-06:7]:

- 1. Bersifat lebih kreatif
- 2. Dapat melakukan proses pembelajaran secara langsung, sementara AI harus mendapatkan masukan berupa simbol dan representasi- representasi
- 3. Fokus yang luas sebagai referensi untuk pengambilan keputusan sebaliknya AI menggunakan fokus yang sempit.

Komputer dapat digunakan untuk mengumpulkan informasi tentang obyek, kegiatan, proses dan dapat memproses sejumlah besar informasi dengan lebih efisien dari yang dapat dikerjakan manusia, tetapi disisi lain manusia, dengan menggunakan insting, dapat melakukan hal yang sulit untuk diprogram pada komputer, yaitu manusia dapat mengenali hubungan antara hal-hal tersebut, menilai kualitas dan menemukan pola yang menjelaskan hubungan tersebut.

Secara konvesial komputer memproses data dengan cara berikut [KUS-06:7]:

### 1. Proses

Yang dikerjakan kalkulasi, mengerjakan operasi-operasi matematis: tambah, kurang, bagi, kali, atau mencari akar. Menyelesaikan rumus/persamaan.

2. Logika

Mengerjakan operasi logika and, or atau *invert*.

3. Penyimpanan

Menyimpan data dan gambar pada file.

4. Retrieve

Mengakses data yang disimpan pada file.

5. Translate

Mengkorvesi data dari suatu bentuk ke bentuk yang lain.

### 6. Sort

Memeriksa data dan menampilkannya dalam urutan yang diinginkan.

### 7. Edit

Melakukan perubahan, penambahan, pengha pusan pada data.

### 8. Monitor

Mengamati event external dan internal dan melakukan tindakan jika kondisi tertentu tercapai. Kontrol, memberikan perintah atau mengendalikan peralatan di luar.

Perbandingan kecerdasan buatan dengan pemrograman konvensional dapat dilihat dalam Tabel 2.1.

Tabel 2.1 Perbandingan AI dengan Pemrograman Konvensional

Dimensi	Kecerdasan Buatan	Pemrograman Konvensional
Proses	Simbolik	Algoritmik
Masukan	Tidak harus lengkap	Harus lengkap
Pencarian	Heuristic	Algoritmik
Majorinterest	Knowledge data	Informasi
Struktur	Terpisah antara kontrol dan <i>knowledge</i>	Kontrol terintegrasi dengan data
Keluaran	Tidak harus lengkap	Harus tetap
Maintenance dan update	Mudah karena menggunakan modul- modul	Umumnya susah dilakukan
Kemampuan pemikiran	Terbatas tetapi dapat ditingkatkan	Tidak ada

Sumber : [KUS-06:8]

### 2.3 Sistem Pakar

Sistem pakar adalah salah satu cabang dari kecerdasan buatan yang membuat penggunaan secara luas *knowledge* yang khusus untuk penyelesaian masalah tingkat manusia yang pakar. Seorang pakar adalah orang yang mempunyai keahlian dalam bidang tertentu, yaitu pakar yang mempunyai *knowledge* atau kemapuan khusus yang orang lain tidak mengetahui atau mampu dalam bidang yang dimilikinya tersebut [AMU-05:3].

Pengetahuan dalam sistem pakar mungkin saja seorang ahli, atau pengetahuan yang umumnya terdapat dalam buku, majalah dan orang yang mempunyai pengetahuan tentang suatu bidang [AMU-05:3].

Seorang pakar dengan sistem pakar mempunyai banyak perbedaan. Perbandingan kemampuan antara kemampuan seorang pakar dengan sebuah sistem pakar dapat dilihat dalam Tabel 2.2 berikut ini:

Tabel 2.2 Perbandingan Kemampuan Seorang Pakar dengan Sistem Pakar

Faktor	Seorang Pakar	Sistem Pakar
Time availability	Hari kerja	Setiap saat
Geografis	Lokal/tertentu	Dimana saja
Keamanan	Tidak tergantikan	Dapat digantikan
Perishable/dapat habis	Ya	Tidak
Perfomansi	Berubah-ubah	Tetap
Kecepatan	Berubah-ubah	Tetap
Biaya	Tinggi	Terjangkau

Sumber: [AMU-05:6]

Tabel 2.3 Perbandingan Sistem Konvensional dengan sistem pakar

Sistem Pakar	
Basis pengetahuan dari mekanisme	
pemrosesan (inferensi)	
Program bisa saja melakukan kesalahan	
Penjelasan (explanation) merupakan	
bagian dari system pakar	
STATE OF THE STATE	
Tidak harus membutuhkan semua input	
data atau fakta	
Perubahan pada kaidah dapat dilakukan	
dengan mudah	
Sistem dapat bekerja hanya dengan	
kaidah yang sedikit	
Eksekusi dilakukan secara heuristic dan	
logis	
Manipulasi efektif pada basis	
pengetahuan yang besar	
Efektivitas adalah tujuan utama	
Data kualitatif	
Represensi pengetahuan dalam	
simbolik	
Menangkap menambah dan	
mendistribusi pertimbangan dan	
pengetahuan	

Sumber: [AMU-05:8]

### 2.3.1 Manfat Sistem Pakar

Manfaat dari sistem pakar yaitu [DWI-03]:

- 1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli
- 2. Bisa melakukan proses secara berulang secara otomatis
- 3. Menyimpan pengetahuan dan keahlian para pakar
- 4. Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka)
- 5. Mampu beroperasi dalam lingkungan yang berbahaya
- Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian
- 7. Tidak memerlukan biaya saat tidak digunakan, sedangkan pada pakar manusia memerlukan biaya sehari-hari.
- 8. Dapat digandakan (diperbanyak) sesuai kebutuhan dengan waktu yang minimal dan sedikit biaya
- 9. Dapat memecahkan masalah lebih cepat daripada kemampuan manusia dengan catatan menggunakan data yang sama.
- 10. Menghemat waktu dalam pengambilan keputusan
- 11. Meningkatkan kualitas dan produktivitasAdapan kelemahan system pakar yaitu [DWI-03] :
- 1. Biaya yang diperlukan untuk membuat, memelihara, dan mengembangkannya sangat mahal
- Sulit dikembangkan, hal ini erat kaitannya dengan ketersediaan pakar di bidangnya
- Sistem pakar tidak 100% benar karena seseorang yang terlibat dalam pembuatan sistem pakar tidak selalu benar. Oleh karena itu perlu diuji ulang secara teliti sebelum digunakan

### 2.3.2 Konsep Umum Sistem Pakar

Konsep dasar sistem pakar terdiri dari beberapa unsur/elemen antara lain :

### 1. Keahlian

Keahlian adalah suatu kelebihan penguasaan pengetahuan di bidang tertentu yang diperoleh dari pelatihan, membaca atau pengalaman. Contoh pengetahuan yang termasuk keahlian adalah [AMU-05:11]:

Fakta-fakta pada lingkup permasalahan tertentu

- Teori-teori pada lingkup permasalahan tertentu
- Prosedur-prosedur dan aturan-aturan yang berkenaan dengan lingkup permasalahan tertentu
- Strategi-strategi global untuk menyelesaikan masalah
- Meta-knowledge

Bentuk-bentuk ini memungkinkan para ahli untuk dapat mengambil keputusan lebih cepat dan lebih baik daripada seseorang yang bukan ahli.

### 2. Ahli

Seorang ahli adalah seseorang yang mampu menjelaskan suatu tanggapan, mempelajari hal-hal baru seputar topik permasalahan (*domain*), menyusun kembali pengetahuan jika dipandang perlu, memecahkan aturan-aturan jika dibutuhkan dan menentukan relevan tidaknya keahlian mereka.

### 3. Pengalihan keahlian

Pengalihan pengetahuan dari para ahli ke komputer untuk kemudian dialihkan lagi ke orang lain yang bukan ahli, merupakan tujuan utama dari sistem pakar. Proses ini memerlukan empat aktivitas yaitu:

- Tambahan pengetahuan (dari para ahli atau sumber-sumber lainnya)
- Representasi pengetahuan ke komputer
- Inferensi pengetahuan
- Pengalihan pengetahuan ke user
   Pengetahuan yang disimpan disebut dengan nama basis pengetahuan.

### 4. Aturan

Sebagian besar sistem pakar komersil dibuat dalam bentuk *rule-based systems* yang mana pengetahuan disimpan dalam aturan-aturan IF-THEN

### 5. Kemampuan menjelaskan

Setiap sistem pakar harus memiliki kemapuan untuk menjelaskan. Fitur inilah yang membedakan sistem pakar dengan sistem konvensional.

Individu yang terlibat dalam lingkungan pengembangan sistem pakar ada tiga macam yaitu [AMU-05:12]:

### 1. Pakar

Pakar adalah orang yang memiliki pengetahuan khusus, pendapat, pengalaman dan metode, serta kemampuan untuk mengaplikasikan keahliannya tersebut guna menyelesaikan masalah.

# BRAWIJAY.

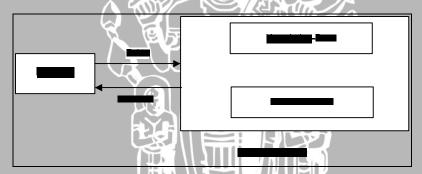
### 2. Perekayasa sistem

Perekayasa sistem adalah orang yang membantu pakar dalam menyusun area permasalahan dengan menginterpretasikan dan mengintegrasikan jawaban-jawaban pakar atas pertanyaan yang diajukan, menggambarkan analogi, mengajukan *counter example* dan menerangkan kesulitan-kesulitan konseptual.

### 3. Pemakai

Sistem pakar memiliki beberapa pemakai yaitu pemakai bukan pakar, pelajar, pembangun sistem pakar yang ingin meningkatkan dan menambah basis pengetahuan dan pakar.

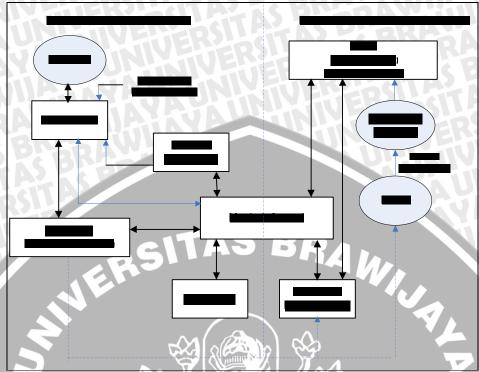
Gambar 2.2 menggambarkan konsep umum suatu sistem pakar. Pengguna menyampaikan fakta atau informasi untuk sistem pakar dan kemudian menerima saran dari pakar atau jawaban. Bagian dalam sistem pakar terdiri dari dua komponen utama, yaitu *knowledge base* yang berisi pengetahuan dan mesin inferensi yang menggambarkan kesimpulan. Kesimpulan tersebut merupakan respons dari sistem pakar atas permintaan user.



Gambar 2.2 Konsep Dasar Fungsi Sistem Pakar Sumber: [AMU-05:4]

### 2.3.3 Struktur Sistem Pakar

Sistem pakar disusun oleh dua bagin utama, yaitu lingkungan pengembang dan lingkungan konsultasi. Lingkungan pengembangan sistem pakar digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar, sedangkan lingkungan konsultasi digunakan oleh pengguna yang bukan pakar untuk memperoleh pengetahuan pakar. Komponen-komponen sistem pakar dalam dua bagian tersebut dapat dilihat dalam Gambar 2.3 berikut ini:



Gambar 2.3 Arsitektur Sistem Pakar Sumber : [AMU-05:14]

Komponen-komponen utama yang terdapat dalam sistem pakar seperti dalam Gambar 2.3 yaitu antarmuka pengguna, basis pengetahuan, akuisisi pengetahuan, mesin inferensi, *workplace*, fasilitas penjelasan, perbaikan pengetahuan [AMU-05:14].

### 2.3.3.1 Antarmuka Pengguna

Antarmuka pengguna merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi. Antarmuka menerima informasi dari pemakai dan mengubahnya ke dalam bentuk yang dapat diterima oleh sistem. Selain itu antarmuka juga menerima informasi dari sistem dan menyajikannya ke dalam bentuk yang dapat dimengerti oleh pemakai. Pada bagian ini terjadi dialog antara program dan pemakai, yang memungkinkan sistem pakar menerima instruksi dan informasi masukan dari pemakai dan juga memberikan informasi keluaran kepada pemakai.

### 2.3.3.2 Basis Pengetahuan

Basis pengetahuan mengandung pengetahuan untuk pemahaman, formulasi dan penyelesaian masalah. Komponen sistem pakar ini disusun atas dua elemen dasar yaitu fakta dan aturan. Fakta merupakan informasi tentang objek dalam area permasalahan tertentu, sedangkan aturan merupakan informasi tentang cara bagaimana memperoleh fakta baru dari fakta yang telah diketahui.

### 2.3.3.3 Akuisisi Pengetahuan

Akuisisi pengetahuan adalah akumulasi, transfer dan tranformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program komputer. Dalam tahap ini perekayasa pengetahuan berusaha menyerap pengetahuan yang selanjutnya ditransfer ke dalam basis pengetahuan. Pengetahuan diperoleh dari pakar, dilengkapi dengan buku, basis data, laporan penelitian dan pengalaman pemakai. Terdapat tiga metode utama dalam akuisisi pengetahuan yaitu [AMU-05:16]:

### Wawancara

Wawancara adalah metode akuisisi yang paling banyak digunakan. Metode ini melibatkan pembicaraan dengan pakar secara langsung dalam suatu wawancara.

### Analisis protokol

Dalam metode akuisisi ini, pakar diminta untuk melakukan suatu pekerjaan dan mengungkapkan pemikirannya dengan menggunakan kata-kata. Pekerjaan tersebut direkam, dituliskan dan dianalisis.

Observasi pada pekerjaan pakar

Dalam metode ini, pekerjaan dalam bidang tertentu yang dilakukan pakar direkam dan diobservasi.

### Induksi aturan dari contoh

Metode ini dibatasi untuk sistem berbasis aturan. Induksi adalah suatu proses penalaran dari khusus ke umum. Suatu sistem induksi aturan diberi contoh-contoh dari suatu masalah yang hasilnya telah diketahui. Setelah diberikan beberapa contoh, sistem induksi aturan tersebut dapat membuat aturan yang benar untuk kasus-kasus contoh. Selanjutnya aturan dapat digunakan untuk menilai kasus lain yang hasilnya tidak diketahui.

Akuisisi pengetahuan dilakukan sepanjang proses pembangunan sistem. Proses akuisisi pengetahuan dibagi kedalam enam tahap yaitu [AMU-05:17]:

### 1. Identifikasi

Identifikasi meliputi penentuan komponen-komponen kunci dalam sistem yang sedang dibangun. Komponen kunci ini adalah perekayasa sistem, pakar, karakteristik masalah, sumber daya dan tujuan.

### 2. Konseptualisasi

Konsep-konsep kunci dan hubungannya yang telah ditentukan pada tahap pertama dibuat lebih jelas dalam tahap konseptualisasi.

### 3. Formalisasi

Tahap ini meliputi pemetaan konsep-konsep kunci, sub masalah dan bentuk aliran yang telah ditentukan dalam tahap-tahap sebelumnya ke dalam representasi formal yang paling sesuai dengan masalah yang ada.

### 4. Implementasi

Tahap ini meliputi pemetaan pengetahuan dari tahap sebelumnya yang telah diformalisasi ke dalam skema representasi pengetahuan yang dipilih.

### 5. Pengujian

Setelah prototipe sistem yang dibangun dalam tahap sebelumnya berhasil menangani dua atau tiga contoh, prototipe tersebut harus menjalani serangkaian pengujian dengan teliti menggunakan beragam contoh masalah

### 6. Revisi prototipe

Pada tahap ini dilakukan perbaikan sistem terhadap tahap-tahap sebelumnya.

### 2.3.3.4 Mesin Inferensi

Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin inferensi adalah program komputer yang memberikan metodologi untuk informasi yang ada dalam basis pengetahuan dan dalam *workplace* untuk menformulasikan kesimpulan.

Ada dua cara yang dapat dikerjakan dalam melakukan inferensi yaitu [AMU-05:19]:

### 1. Forward Chaining (Pelacakan ke depan)

Pada metode ini pencocokan fakta atau pernyataan dimulai dari informasi masukan dan selanjutnya mencoba menggambarkan kesimpulan. Gambar 2.4 menggambarkan proses *forward chaining*.

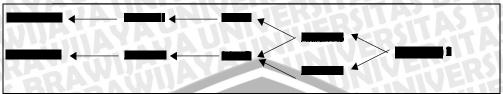


Gambar 2.4 Proses Forward Chaining Sumber: [AMU-05:20]

### 2. Backward Chaining (Pelacakan ke belakang)

Pada metode ini pelacakan dimulai dari tujuan, selanjutnya dicari aturan yang memiliki tujuan tersebut untuk kesimpulannya. Selanjutnya pelacakan menggunakan premis untuk aturan tersebut untuk tujuan baru sebagai kesimpulannya. Proses berlanjut

sampai semua kemungkinan ditemukan. Gambar 2.5 menunjukan proses *backward chaining*.



Gambar 2.5 Proses *Backward Chaining*Sumber: [AMU-05:19]

### 2.3.3.5 Workplace

Workplace merupakan area dari sekumpulan memori kerja (working memory) yang digunakan untuk merekam hasil-hasil antara dan kesimpulan yang dicapai. Ada tiga tipe keputusan yang dapat direkam yaitu [AMU-05:22]:

- 1. Rencana : bagaimana menghadapi masalah
- 2. Agenda: aksi-aksi potensial yang sedang menunggu untuk dieksekusi
- 3. Solusi : calon aksi yang akan dibangkitkan

### 2.3.3.6 Fasilitas Penjelas

Fasilitas penjelas adalah komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Komponen ini menggambarkan penalaran sistem kepada pemakai. Fasilitas penjelasan dapat menjelaskan perilaku sistem pakar dengan menjawab pertanyaan-pertanyaan berikut [AMU-05:22]:

- Mengapa pertanyaan tertentu ditanyakan sistem pakar?
- Bagaimana kesimpulan tertentu diperoleh?
- Mengapa alternative tertentu ditolak?
- Apa rencana untuk memperoleh penyelesaian?

### 2. Perbaikan Pengetahuan

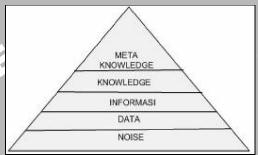
Sistem pakar dapat menganalisis dan mengevaluasi kinerjanya untuk melihat apakah pengetahuan-pengetahuan yang ada masih cocok untuk digunakan di masa mendatang.

### 2.3.3.7 Representasi Pengetahuan

Pengetahuan (knowledge) dapat diklasifikasikan ke dalam tiga kategori yaitu procedural knowledge, declarative knowledge, dan tacit knowledge. Procedural knowledge berkenaan untuk mengetahui bagaimana melakukan sesuatu. Declarative knowledge berkenaan untuk mengetahui sesuatu itu benar atau salah. Sedangkan tacit knowledge atau unconscious knowledge adalah pengetahuan yang tidak dapat diungkapkan dengan bahasa.

Pengetahuan adalah bagian dari suatu hierarki dan ini dapat ditunjukkan dalam Gambar 2.6.

Yang ada pada level paling bawah adalah *noise* (gangguan) yang merupakan data yang masih kabur. Level berikutnya adalah data yang merupakan hal potensial yang paling penting. Data yang sudah diproses adalah informasi yang penting. Berikutnya adalah pengetahuan (*knowledge*) yang menggambarkan informasi sangat khusus. Level paling atas adalah *meta knowledge* yang merupakan *knowledge* dan keahlian.



Gambar 2.6 Hierarki *Knowledge* Sumber: [AMU-05:28];

### 2.3.4 Sistem Produksi

Representasi pengetahuan dengan sistem produksi pada dasarnya berupa aplikasi aturan (rule) yang berupa [AMU-05:74].:

- Antecedent, yaitu bagian yang mengekspresikan situasi atau premis (pernyataan berawalan IF)
- Konsekuen, yaitu bagian yang menyatakan suatu tindakan tertentu atau konklusi yang diterapkan jika suatu situasi atau premis bernilai benar (pernyataan berawalan THEN).
- Konsekuensi atau konklusi yang dinyatakan pada bagian THEN baru dinyatakan benar jika bagian IF pada sistem tersebut juga benar atau sesuai dengan aturan tertentu.

Sistem produksi dapat dibuat dalam beberapa bentuk yang berbeda, yaitu [AMU-05:75]:

- IF Premis THEN Konklusi
   Contoh: IF Pendapatan anda tinggi THEN kesempatan anda untuk diaudit juga tinggi
- Konklusi IF Premis

Contoh: Kesempatan anda diaudit tinggi IF pendapatan anda tinggi

• IF Premis THEN Konklusi ELSE Konklusi

Contoh: IF pendapatan anda tinggi THEN kesempatan anda diaudit tinggi ELSE kesempatan anda diaudit rendah

Kaidah yang lebih kompleks

Contoh: IF tingkat kredit tinggi AND gaji lebih dari \$30,000 OR asset lebih dari \$75,000 AND sejarah pelunasan lancar THEN disetujui peminjaman hingga \$100.000.

Ada dua tipe kaidah (aturan) yang umum dalam kecerdasan buatan, yaitu kaidah pengetahuan dan kaidah inferensi. Kaidah pengetahuan atau kaidah deklaratif menyatakan semua fakta dan hubungan tentang suatu permasalahan. Kaidah inferensi atau kaidah prosedural merupakan nasihat atau saran tentang bagaimana menyelesaikan suatu masalah yang diberikan dengan fakta tertentu yang diketahui.

Kaidah inferensi berisi kaidah tentang kaidah. Tipe kaidah ini dinamakan dengan *metarule*. Perekayasa Pengetahuan memisahkan dua tipe kaidah ini dimana kaidah pengetahuan menjadi basis pengetahuan sedangkan kaidah inferensi menjadi bagian dari mesin inferensi.

### 2.3.5 Logika

Selain dengan sistem produksi, pengetahuan juga dapat direpresentasikan dengan menggunakan logika. Pada dasarnya proses logika adalah proses membentuk kesimpulan atau menarik suatu inferensi berdasarkan fakta yang telah ada. Input dari proses logika berupa premis atau fakta-fakta yang diakui kebenarannya sehingga dengan melakukan penalaran pada proses logika dapat dibentuk suatu inferensi atau kesimpulan yang benar pula [AMU-05:53].

Ada dua penalaran yang dapat dilakukan untuk mendapatkan konklusi yaitu:

### 1. Penalaran Deduktif

Penalaran dimulai dari prinsip umum untuk mendapatkan konklusi yang lebih khusus.

### 2. Penalaran Induktif

Penalaran dimulai dari fakta-fakta khusus untuk mendapatkan kesimpulan umum. Pada penalaran induktif ini, munculnya premis baru bisa menyebabkan gugurnya konklusi yang sudah diperoleh.

### 2.3.5.1 Logika Proposisi

Logika proposisi atau biasa disebut kalkulus proposisi merupakan logika simbolik untuk memanipulasi proposisi. Proposisi merupakan pernyataan pada yang dapat bernilai benar atau salah. P dan Q merupakan contoh dari simbol untuk pernyataan pada logika proposisi. Dua atau lebih proposisi dapat dihubungkan dengan menggunakan operator logika [AMU-05:14]:

- ^ untuk konjugasi (AND/DAN)
- V untuk disjungsi (OR/ ATAU)
- untuk negasi (NOT/TIDAK)
- → untuk implikasi/kondisional (IF...THEN/JIKA...MAKA)
- ↔ untuk equivalensi/bikondisional (IF AND ONLY IF/JIKA DAN HANYA JIKA).

### 2.3.5.2 Logika Predikat

Logika predikat atau kalkulus predikat merupakan logika yang digunakan untuk merepresentasikan masalah yang tidak dapat direpresentasikan oleh logika proposisi. Logika predikat adalah suatu logika yang lebih canggih yang seluruhnya menggunakan konsep dan kaidah proposional yang sama. Logika predikat memberi tambahan kemampuan untuk merepresentasikan pengetahuan dengan sangat cermat dan rinci.

Kalkulus predikat memungkinkan kita untuk memecahkan statemen ke dalam bagian komponen yang disebut obyek, karakteristik obyek, atau beberapa keterangan obyek. Suatu proposisi atau premis dibagi menjadi dua bagian, yaitu ARGUMEN (atau obyek) dan PREDIKAT (keterangan). Argumen adalah individu atau obyek yang membuat keterangan. Predikat adalah keterangan yang membuat argumen dan predikat [AMU-05:62].

### 2.4 Konsep Faktor Kepastiaan

Dalam menghadapi suatu masalah sering ditemukan jawaban yang tidak memiliki kepastian penuh. Ketidakpastian ini dapat berupa probabilitas atau kebolehjadian yang tergantung dari hasil suatu kejadian. Hasil yang tidak pasti disebabkan oleh dua faktor yaitu aturan yang tidak pasti dan jawaban pengguna yang tidak pasti atas suatu pertanyaan yang diajukan oleh sistem. Hal ini sangat mudah dilihat pada sistem diagnosis penyakit dimana pakar tidak dapat mendefinisikan hubungan antara gejala dan penyebabnya secara pasti dan pasien tidak dapat merasakan suatu gejala dengan pasti pula, pada akhirnya akan ditemukan banyak kemungkinan diagnosis [KUS-05:39].

BRAWIJAYA

Ketidakpastian dapat dianggap sebagai suatu kekurangan informasi yang memadai untuk membuat suatu keputusan. Ketidakpastian merupakan suatu permasalahan karena mungkin menghalangi kita dalam membuat suatu keputusan yang terbaik bahkan mungkin dapat menghasilkan suatu keputusan yang buruk. Dalam dunia medis, ketidakpastian mungkin menghalangi pemeriksaan yang terbaik untuk para pasien dan berperan untuk suatu terapi yang keliru. Dalam bisnis, ketidakpastian dapat berarti kerugian uang [AMU-05:149].

Sistem pakar harus mampu bekerja dalam ketidakpastian. Sejumlah teori telah ditemukan untuk menyelesaikan ketidakpastian, termasuk diantaranya probabilitas klasik (*classical probability*) probabilitas Bayes, teori Hartley berdasarkan himpunan klasik, teori Shannon berdasarkan pada probabilitas, teori Dempster-Shafer, teori fuzzy Zadeh, dan faktor kepastian [KUS-05:40].

Ada tiga penyebab ketidakpastian aturan yaitu aturan tunggal, penyelesaian konflik, dan ketidakcocokan antarkonsekuen dalam aturan [KUS-05:40].

Aturan tunggal yang dapat menyebabkan ketidakpastian dipengaruhi oleh tiga hal yaitu kesalahan, probabilitas, dan kombinasi gejala. Kesalahan dapat terjadi karena [KUS-05:40]:

- 1. Ambiguitas
- 2. Ketidaklengkapan data
- 3. Kesalahan informasi
- 4. Ketidakpercayaan terhadap suatu alat
- 5. Adanya bias

Probabilitas disebabkan ketidakmampuan seorang pakar merumuskan suatu aturan secara pasti. Misalnya, jika seseorang mengalami sakit kepala, demam dan bersin-bersin maka ada kemungkinan orang tersebut terserang penyakit flu, tetapi bukan berarti apabila seseorang mengalami gejala tersebut pasti terserang penyakit flu.

Hanya karena aturan tunggalnya benar, belum menjamin suatu jawaban bernilai benar. Hal ini masih dipengaruhi oleh kompatibilitas antar aturan. Inkompabilitas suatu aturan disebabkan oleh beberapa hal, yaitu [KUS-05:41]:

### 1. Kontradiksi aturan, misalnya:

### Aturan 1:

JIKA anak demam

MAKA harus dikompres

Aturan 2:

JIKA anak demam

MAKA jangan dikompres

2. Subsumsi aturan, misalnya:

Aturan 3: JIKA E<sub>1</sub> maka H

Aturan 4: JIKA E<sub>1</sub> dan E<sub>2</sub> MAKA H

Jika hanya E<sub>1</sub> yang muncul, maka masalah tidak akan timbul karena aturan yang digunakan adalah aturan 3, tetapi apabila E1 dan E2 sama-sama muncul maka kedua BRAWINAL aturan (aturan 3 dan 4) sama-sama akan dijalankan.

3. *Redundancy* aturan, misalnya:

Aturan 5: JIKA E<sub>1</sub> DAN E<sub>2</sub> MAKA H

Aturan 6: JIKA E<sub>2</sub> DAN E<sub>1</sub> MAKA H

4. Kehilangan aturan, misalnya:

Aturan 7: JIKA E<sub>4</sub> MAKA H

Ketika E<sub>4</sub> diabaikan maka H tidak pernah tersimpulkan.

5. Penggabungan data, misalnya pada diagnosis kesehatan. Seorang dokter dapat menyimpulkan suatu penyakit tidak hanya berdasarkan anamnesis, tetapi juga hasil tes laboratorium, pemeriksaan kondisi tubuh, sejarah penyakit, dan lain-lain. Untuk itu diperlukan penggabungan semua data untuk dapat menyimpulkan suatu penyakit.

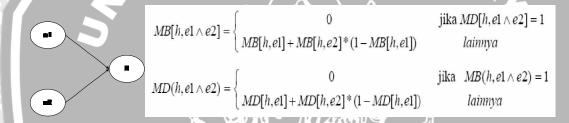
Pemilihan metode penyelesaian konflik dapat juga mempengaruhi hasil penyelesaian akhir atas suatu masalah. Ada suatu sistem yang mendahulukan aturan yang lebih spesifik, misalnya ada aturan 3 dan aturan 4, karena aturan 4 lebih spesifik maka aturan 4 akan dieksekusi terlebih dahulu. Ada juga sistem yang memberi bobot pada aturannya sehingga eksekusi dilakukan terhadap suatu aturan berdasarkan bobot yang dia miliki.

Faktor kepastian diperkenalkan oleh Shortliffe Buchanan dalam pembuatan MYCIN. Faktor kepastian merupakan nilai parameter klinis yang diberikan MYCIN untuk menunjukkan besarnya kepercayaan. Faktor kepastian didefinisikan sebagai berikut:

$$CF(H,E) = MB (H,E) - MD (H,E)$$

- CF(H,E) : certainty factor dari hipotesis H yang dipengaruhi oleh gejala (evidence) E. besarnya CF berkisar antara -1 sampai dengan 1. Nilai -1 menunjukan ketidakpercayaan mutlak sedangkan nilai 1 menunjukan kepercayaan mutlak.
- MB (H,E): ukuran kenaikan kepercayaan terhadap hipotesis H yang dipengaruhi oleh gejala E
- MD (H,E): ukuran kenaikan ketidakpercayaan terhadap hipotesis h yang dipengaruhi oleh gejala E.

Salah satu cara yang di gunakan dalam menentukan CF adalah dengan cara mengkombinasikan *evidence*. Jika e1 dan e2 adalah observasi, maka [DWI-03] :



### Contoh:

• Asih menderita bintik-bintik di wajahnya. Dokter memperkirakan Asih terkena cacar dengan kepercayaan MB[cacar,bintik]=0,80 dan MD[cacar,bintik]=0,01 maka :

CF[cacar, bintik] = 0.80 - 0.01 = 0.79

Jika ada observasi baru bahwa Asih juga panas badan dengan kepercayaan, MB[cacar,panas]=0,7 dan MD[cacar,panas]=0,08 maka:

MB[cacar,bintik panas] = 0.8 + 0.7 \* (1 - 0.8) = 0.94

MD[cacar,bintik panas] = 0.01 + 0.08 \* (1 - 0.01) = 0.0892

CF[cacar, bintik panas] = 0.94 - 0.0892 = 0.8508

Penggabungan kepercayaan dan ketidakpercayaan dalam bilangan yang tunggal memiliki dua kegunaan, yaitu pertama faktor kepastian digunakan untuk tingkatan hipotesis di dalam urutan kepentingan. Sebagai contoh, jika seorang pasien mempunyai gejala tertentu yang menyarankan beberapa kemungkinan penyakit, kemudian penyakit dengan CF tertinggi menjadi urutan pertama dalam urutan pengujian.

### 2.5 Penyakit Kulit

Pada sub bab ini akan dijelaskan mengenai macam-macam penyakit kulit yang dapat didiagnosis oleh sistem pakar ini.

Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian dapat mendiagnosis delapan macam penyakit, yaitu: Campak, Cacar Air, Impetigo, Bisul, Cacar Ular, Veruka, Psoriasis dan Dermatomikosis.

### 2.5.1 Campak

Penyakit campak adalah penyakit yang sangat menular yang disebabkan oleh virus campak. Penularannya terjadi melalui udara ataupun kontak langsung dengan penderita. Virus campak menyebar lewat percikan ludah penderita. Virus cacar air bisa pindah ke tubuh orang sehat lewat bersentuhan langsung dengan cacarnya.

Gejala-gejalanya adalah demam, batuk, pilek dan timbul bercak merah di kulit 3-5 hari setelah anak menderita demam. Bercak mula-mula timbul di pipi di bawah telinga yang kemudian menjalar ke muka, tubuh dan anggota tubuh lainnya. Komplikasi dari penyakit campak adalah pneumonia (radang paru-paru), infeksi telinga, neuritis (radang pada syaraf), artritis (radang sendi) dan ensefalitis (radang otak) yang dapat menyebabkan kerusakan otak yang permanent [JUD-05:23].

### 2.5.2 Cacar Air

Cacar air atau varisela memang merupakan penyakit anak-anak yang sudah ratusan tahun dikenal orang. Diawali dengan gejala melemahnya kondisi tubuh, pusing, demam yang kadang-kadang diiringi batuk, dalam waktu 24 jam timbul bintik-bintik yang berkembang menjadi lesi (mirip kulit yang terangkat karena terbakar) dan terakhir menjadi benjolan-benjolan kecil berisi cairan. Sekitar 250 - 500 benjolan akan timbul menyebar di seluruh bagian tubuh, tidak terkecuali pada muka, kulit kepala, mulut bagian dalam, mata, termasuk bagian tubuh yang paling intim. Namun dalam waktu kurang dari seminggu, lesi ini akan mengering dan bersamaan dengan itu terasa gatal. Dalam waktu 1-3 minggu bekas pada kulit yang mengering akan terlepas.

Virus penyebab penyakit cacar air ini berpindah dari satu orang ke orang lain melalui percikan ludah yang berasal dari batuk atau bersin penderita dan diterbangkan melalui udara atau kontak langsung dengan kulit yang terinfeksi. Komplikasi yang langka tapi bisa terjadi berupa radang paru, radang otak, radang sumsum tulang, kegagalan hati, hepatitis serta sindrom Reye (kelainan pada otak sekaligus hati) [JUD-05:30].

### 2.5.3 Cacar Ular

Walaupun namanya cacar ular, penyakit ini tidak disebabkan oleh ular. Cacar ular adalah nama awam untuk penyakit Herpes Zoster. Penyakit ini merupakan bentuk reaktivasi penyakit cacar air (varisela) yang pernah diderita seseorang sebelumnya.

Perlu diketahui, bila seorang terkena infeksi virus varisela-zoster untuk pertama kali, maka akan timbul penyakit cacar air, setelah sembuh, virus tersebut tidaklah musnah seluruhnya dari tubuhpenderita, melainkan berdiam didalam tubuh manusia, tepatnya diganglion saraf tepi penderitanya. Virus yang berdiam diri didalam tubuh penderita ini dapat sewaktu-waktu muncul kembali dan menyebabkan penyakit yang dinamai Herpes Zoster [ANI-05].

Walau didalam tubuhnya terdapat virus ini, namun kebanyakan orang memang tidak mengalami penyakit Herpes Zoster. Hal ini disebabkan daya tahan tubuh yang baik yang dapat menekan virus ini berkembang. Sebaliknya, pada orang yang daya tahan tubuhnya menurun, tak jarang penyakit ini tiba-tiba muncul menyerang.

Gejala yang terjadi pada penyakit ini awalnya hampir sama dengan cacar air, yaitu terjadi demam dan badan pegal-pegal. Selanjutnya sedikit berbeda dengan penyakit cacar air, walaupun virus penyebabnya sama.

Pada Herpes Zoster, gelembung muncul dalam suatu kelompok yang menyerupai garis lebar deengan dasar kulit kemerahan, yang muncul dari bagian belakang tubuh dan menjalar kearah depan pada salah satu sisi tubuh [MAR-93].

### 2.5.4 Impetigo

Impetigo merupakan jangkitan bakteria yang agak lazim terjadi di permukaan kulit. Ia mudah dan sering terjadi di kalangan kanak-kanak berbanding dengan orang dewasa. Pada amnya, ia disebabkan oleh satu dari dua jenis kuman bakteria [ANO-04]:

- 1. streptococcus Kumpulan A
- 2. staphylococcus aureus.

Ia selalu bermula dibagian yang terdapat rekahan, lelaasan, gigitan serangga, atau dikawasan kulit yang terjangkit ekzema. Lazimnya, impetigo terjadi dibagian muka terutama di antara bibir atas dan hidung, lengan tangan, dan kaki [FER-00].

### 2.5.5 Bisul

Bisul ialah jangkitan pada rambut/bulu roma yang menyebabkan timbulnya nanah atau abses. Bisul biasanya di tandai dengan bengkak berwarna merah dan terasa

BRAWIJAYA

sakit dibagian bawah lapisan kulit. Ia di penuhi nanah dan kelihatan seperti jerawat batu. Nanah atau abses boleh membesar lebih dari saiz bola ping-pong dan terasa sangat sakit. Jangkitan ini selalunya disebabkan oleh kuman *Staphylococcus aureus* yaitu sejenis kuman yang tidak berbahaya. Ia hidup di permukaan kulit dan di kawasan yang terjangkit seperti luka atau jerawat. Oleh itu bisul lazimnya muncul apabila sistem pertahanan badan kita merosot [MAR-93].

Bisul boleh terjadi di mana-mana bahagian badan tetapi ia sering terjadi di bagian yang berbulu dan yang lecet seperti ketiak, kelopak mata, buah dada, punggung, muka, kelangkang dan di belakang tengkuk. Apabila bisul menjadi besar, ia dikenali sebagai karbunkel. Selalunya karbunkel terjadi di belakang tengkuk dan punggung. Karbunkel adalah bisul yang besar atau kelompok bisul kecil yang bergabung bersama. Biasanya ia mudah terjadi di kalangan penderita diabetes [ANO-04].

### **2.5.6** Veruka

Veruka ialah hiperplasia epidermis yang disebabkan Human Papilloma Virus (HPV tipe 1,2,3,4,5,6,7,8,9,10,11,16,18,27,29,31,41) yang termasuk golongan virus papova. Penularanya secara langsung atau autoinokulasi. Familial berperan serta dalam patogenesisnya. Faktor predisposis berupa trauma kulit yang berulang-ulang, kulit yang lembab [SHW-05:12].

### **Bentuk Klisnis**

- 1. Veruka Vulgaris: atau *common warts*; sering dijumpai pada anak-anak Lokalisasi pada anak- anak: pada tempat trauma, dorsum manus dan jari- jari tangan. Berupa papula atau plak permukaan kasar warna kabuan bisa single dan multiple Pada penderita dewasa, lokalisasi di segala tempat kulit tubuh, bibir, lubang hidung, lidah, dan lain- lain yang menyebar secara autoinokulasi. Pada beberapa keadaan dapat timbul veruka yang besar (*mother warts*), yang kemudian diikuti erupsi yang banyak.
- 2. Veruka Plana Juvenilis : papula dengan warna abu- abu atau warna kulit datar penampang 1-3 mm permukaan halus, multiple. Lokalisasi pada dahi, pipi, hidung, dan sekitar mulut dan dorsum tangan. Terdapat pada anak- anak dan dewasa muda. Dapat ditemukan fenomena Koebner.
- 3. Veruka Plantaris : terdapat di telapak kaki terutama daerah yang mengalami tekanan. Nyeri waktu berjalan, lesi berupa plak yang keras dan lunak dibagian tengah berwarna kekuningan, permukaannya licin atau terdapat bitik-bintik perdarahan [SHW-05:13].

### 2.5.7 Psoriasis

Psoriasis ialah sejenis penyakit kulit yang penderitanya mengalami proses pergantian kulit yang terlalu cepat. Penyakit ini secara klinis tidak mengancam jiwa, tidak menular tetapi karena timbulnya dapat terjadi pada bagian tubuh mana sajasehingga dapat menurunkan kualitas hidup serta mengganggu kekuatan mental seseorang bila tidak dirawat dengan baik.

Berbeda dengan pergantian kulit pada manusia normal yang biasanya berlangsung selam tiga sampai empat minggu, proses pergantian kulit pada penderita psoriasis berlangsung secara cepat yaitu sekitar 2-4 hari, (bahkan bias terjadai lebih cepat) pergantian kulit yang banyak dan menebal.

Kulit penderita psoriasis awalnya tampak seperti bintik merah yang makin melebar dan ditumbuhi sisik lebar dan berlapis – lapis. Tumbuhnya tidak selalu di seluruh bagian kulit tubuh kadang – kadang hanya timbul di tempat – tempat tertentu saja, karena bergiliran sel – sel kulit bagian lainya berjalan normal. Psoriasis pada kulit kepala dapat menyerupai ketombe, sedangkan ada lempeng kuku tampak lubang – lubang kecil rapuh atau keruh. Penyakit psoriasis dapat disertai dengan atau tanpa rasa gatal. Kulit dapat membaik seperti kulit normal lainya setelah warna kemerahan, putih atau kehitaman bekas psoriasis. Pada beberapa jenis psoriasis, komplikasi yang diakibatkan dapat menjadi serius, seperti pada psoriasis artropi yaitu psoriasis yang menyerang sendi, psoriasis bernanah, dan terahir seluruh kulit akan menjadi merah disertai badan menggigil (Eritoderma) [SIS-06],

### 2.5.8 Dermatomikosis

Beberapa faktor yang mempengaruhi timbulnya dermatomikosis superfisialis adalah iklim yang panas, higiene sebagian masyarakat yang masih kurang, adanya sumber penularan di sekitarnya, penggunaan obat-obatan antibiotika, steroid dan sitostatika yang meningkat, adanya penyakit kronis dan penyakit sistemik lainnya. Tingginya prevalensi penyakit ini juga dipengaruhi oleh banyaknya hambatan dalam penatalaksanaannya, yaitu lamanya waktu pengobatan, banyaknya kasus yang resisten terhadap obat anti jamur dan efek samping yang ditimbulkan oleh obat anti jamur sistemik [RAH-04].

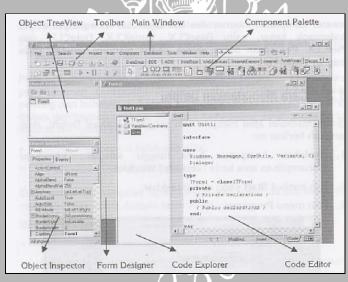
# BRAWIJAYA

### 2.6 Borland Delphi 7

Borland Delphi merupakan suatu bhasa pemrograman yng memberikan berbagai fasilitas pembuat aplikasi visual. Keunggulan bahasa pemrograman ini terletak pada produktivitas, kualitas, pengembangan perangkat lunak, kecepatan kompilasi, pola desain yang menarik serta diperkuat dengan pemrogramannya yang terstruktur. Keunggulan lain dari Delphi adalah dapat digunakan untuk merancang program aplikasi yang memiliki tampilan seperti program aplikasi lain yang berbasis windows.

Khusus untuk pemrograman database, Borland Delphi menyediakan fasilitas objek yang kuat dan lengkap yang memudahkan programmer dalam membuat program. Format database yang dimiliki Delphi adalah format database Paradox, dBase, MS. Access, ODBC, SyBASE, Oracle dan lain-lain. [APM-03:1].

### 2.6.1 Mengenal IDE Delphi



**Gambar 2.7** Lembar Kerja Borland Delphi Sumber : [APM-03:2]

IDE adalah sebuah lingkungan yang berisi tool-tool yang diperlukan untuk desain, menjalankan dan mengetes sebuah aplikasi, disajikan dan terhubung dengan baik sehingga memudahkan pengembangan program. Di Delphi, IDE terdiri atas *Main Window, Component Palette, ToolBar, Form Designer, Code Editor, Code Explorer dan Object TreeView*. Integrasi ini memberi Anda kemudahan dalam mengembangkan aplikasi yang kompleks.

BRAWIJAYA

Main Window adalah bagian utama dari IDE. Main Window mempunyai semua fungsi utama dari program-program Windows lainnya. Main Window dibagi menjadi tiga bagian. yaitu Menu utama, Toolbar dan Component Palette.

Menu Utama. Sama seperti pada program Windows lainnya, menu utama dipakai untuk membuka atau menyimpan file, memanggil *wizard*, menampilkan jendela lain, mengubah option dan lain sebagainya. Setiap pilihan pada menu utama juga dapat dipanggil dengan sebuah tombol pada toolbar.

Dengan toolbar kita dapat melakukan beberapa operasi pada menu utama dengan sebuah klik tunggal. Setiap tombol pada toolbar mempunyai sebuah tooltip yang berisi informasi mengenai fungsi dari tombol tersebut. Selain Component Palette, ada 5 toolbar terpisah pada IDE, yaitu Debug, Desktop, Standard, View, Component Palette dan Custom.

Component Palette adalah toolbar dengan ketinggian ganda, yang berisi page control dengan semua komponennya. Urutan dan tampilan dari page dan komponen pada Component Palette dapat diatur dengan mengklik kanan atau dengan memilih menu Component Configure Component dari menu utama.

Tombol ini akan muncul di setiap page. Tombol ini dipakai untuk menentukan posisi. Jika Anda memilih sebuah item dari sebuah komponen page, tombol *pointer* ini akan berada dalam keadaan *depressed*. Hal ini berarti Anda akan meletakkan komponen yang dipilih pada form. Untuk meletakkan komponen pada form, kita cukup mengklik pada form.

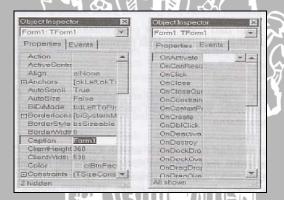
Cara kedua untuk meletakkan komponen pada form adalah dengan mengklik ganda pada komponen. Jika kita telah memilih sebuah komponen dan kita ingin membatalkan (mengembalikannya ke keadaan normal), kita dapat mengklik tombol *pointer*. Dalam hal ini tombol *pointer* berfungsi sebagai tombol *Cancel*. Tombol *pointer* selalu berada di ujung kiri dari component palette.

Form Designer diawali dengan jendela kosong yang memungkinkan kita untuk merancang aplikasi Windows. Dari sini kita dapat menentukan tampilan aplikasi kita. Kita berinteraksi dengan Form Designer dengan cara memilih komponen dari Component Palette dan meletakkannya ke dalam form. Setelah ada komponen dalam form, kita dapat mengatur posisinya atau mengubah ukurannya dengan menggunakan mouse. kita dapat mengubah tampilan dan perilaku komponen dengan menggunakan Object Inspector dan Code Editor.

Object Inspector sangat perlu dalam pembuatan aplikasi. Dengan object inspector, kita dapat mengubah property dari setiap item dengan mudah sekali. Juga, kita dapat mengontrol tindakan yang diambil jika terjadi event. Object Inspector terdiri atas dua tab, yaitu properties tab dan events tab.

Tab *Properties* memberi kita fasilitas untuk melihat dan mengubah properti dari setiap item. Klik pada sebuah form kosong, dan perhatikan atribut-atribut yang ada (sama seperti gambar di atas). Hal yang menarik, jika Anda melihat properti dengan tanda +, berarti properti tersebut mempunyai subproperti. Contohnya. perhatikan properti *Font* yang mempunyai tanda plus. Jika kita klik ganda pada properti Font, akan ditampilkan subpropertinya. seperti *Color, Height. Name* dan lain-lain.

Tab *Events* adalah bagian lain dari *object inspector*. Tab ini berisi *event-event* yang dapat direspon oleh sebuah objek. Klik tab *Events* di sebelah kanan tab *Properties*. Misalnya, kita ingin sesuatu dikerjakan pada saat form ditutup, maka kita menyatakan tindakan tersebut (berupa sebuah procedure) pada *OnClose*.



**Gambar 2.8** Lembar Kerja Object Inspector Sumber: [APM-03:5]

Kita dapat menampilkan *Help* dari *Object Inspector*. Hal ini sangat berguna bagi kita sebagai seorang pemrogram. Jika kita mendapat kesulitan dengan sebuah properti atau *event*, tekan tombol F1 untuk menampilkan *WinHelp*.

Code Editor merupakan bagian terpenting di lingkungan Delphi. Jendela ini dipakai untuk menuliskan program Delphi. Editor Delphi sangat canggih. dengan fasilitas-fasilitas highlight untuk memudahkan menemukan kesalahan, kerangka program sehingga kita tidak perlu menuliskan seluruh program. dan lain-lain.

Title bar berisi nama file yang sedang disunting. Di bagian atas dari jendela kita melihat tab-tab yang menyatakan page-page yang ada. Kita dapat mempunyai beberapa file dalam sebuah aplikasi Delphi, dan tab tersebut dapat menunjukkan hal ini. Di bagian bawah ada tiga item yang perlu diperhatikan. Bagian kiri bawah adalah penunjuk baris/kolom. Hal ini membantu kita untuk mengetahui posisi kursor dalam program. Jika Anda membuat project baru, program yang Delphi berikan kepada kita tidak disimpan, kita harus menyimpan sendiri. Oleh karena program sudah berubah dan belum pernah disimpan, maka kata *Modified* muncul setelah penunjuk. baris/kolom. Item terakhir menunjukkan modus pengetikan, yaitu *Insert/Overwrite*.

Menu lokal dari *code Editor* menyediakan berbagai option untuk memudahkan kita pada saat menuliskan program.

Bagian kiri dari *Code Editor* adalah *Code Explorer*. *Code explorer* dipakai untuk memudahkan navigasi di dalam file unit. Kita dapat menutup code explorer dengan mengklik tanda 'x' pada pojok kanan atas dari *code explorer* (bukan jendela penyunting).

Untuk menampilkan *code explorer*, pilih menu *View Code Explorer*, atau klik kanan pada *code editor* dan pilih *View Explorer*.

Code Explorer berisi struktur pohon yang menampilkan semua type, class, property, method, variabel global dan rutin global yang didefinisikan dalam unit. Juga menampilkan semua unit yang ada di klausa uses. Anda dapat membuka atau menutup sebuah node pada pohon.

### 2.6.2 Form

Inti dari setiap aplikasi Delphi adalah *form. Form* dapat dianggap sebagai sebuah *window*. Delphi memberikan sebuah *form* pada setiap *project* dan menampilkan sebuah *form* kosong pada saat kita memulai Delphi, seperti dalam Gambar 2.7. Akan tetapi kita dapat memilih pilihan *desktop* di menu *Tools|Options. Form* adalah dasar dari aplikasi Windows.

Form dipakai untuk menempatkan komponen, kontrol dan lain-lain. Informasi mengenai form disimpan dalam dua file, yaitu file .dfm (untuk aplikasi VCL) atau .xfm (untuk aplikasi CLX), dan .pas. File .dfm/.xfm berisi informasi mengenai tampilan form, ukuran, lokasi, dan seterusnya. Kita tidak perlu tahu mengenai isi file ini. Cukup kita tahu file itu ada.

BRAWIJAYA

Program untuk mengontrol *form* disimpan di file .pas, atau disebut juga dengan *unit*. Kita akan memakai banyak waktu untuk mengubah *file* ini. Setiap kali kita menambahkan *event handler* atau menambahkan perintah, *file* ini akan diubah, dan Delphi menempatkan kursor di tempat di mana seharusnya kita menambahkan program tersebut.

Form mempunyai properti. Dengan properti tersebut kita dapat mengubah tampilan form dengan mengubah warna, ukuran, lokasi. dan lain-lain. Sebuah form mempunyai sejumlah event handler.

### 2.6.3 Properti

Sebuah objek diakses melalui *properti. Properti* dari objek Delphi mempunyai nilai yang dapat diubah pada saat desain tanpa perlu menuliskan program. Jika kita ingin mengubah *properti* pada saat aplikasi dijalankan, kita harus menuliskan sedikit program.

Pada Delphi, semua *form* dan komponen (*Visual Component*) mempunyai *properti. Form* mempunyai ukuran (tinggi dan lebar), warna *background*, batas, dan juga nama. Kita dapat mengontrol penampilan objek-objek tersebut dengan memanipulasi atribut atau *properti*nya.

### 2.6.4 Basis Data

Secara sederhana *database*(basis data) dapat diungkapkan sebagai suatu pengorganisasian data dengan bantuan komputer yang memungkinkan data dapat diakses dengan mudah dan cepat. Dalam hal ini, pengertian akses dapat mencakup pemerolehan data maupun pemanipulasian data seperti menambah atau menghapus data [KAB-03:2].

Basis Data (*Database*) sendiri dapat didefinisikan dalam sejumlah sudut pandang seperti:

- § Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
- Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansasi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.

Basis Data memiliki prinsip utama yaitu pengaturan data/arsip, dan tujuan utamanya adalah kemudahan dan kecepatan dalam pengambilan kembali data/arsip. Basis Data menonjolkan pengaturan atau pengorganisasian data yang disimpan sesuai dengan fungsi atau jenisnya.

# BRAWIJAY

### 2.6.5 Model Basis Data Relasional

Model basis data relasional sering pula disebut sebagai model relasional atau basis data relasional. Model basis data ini diperkenalkan pertama kali oleh E.F. Codd. Pada model relasional, basis data akan dipilah-pilah ke dalam tabel dua dimensi. Setiap tabel selalu terdiri atas lajur mendatar yang disebut dengan Baris Data ( *Row/Record* ) dan lajur vertikal yang biasa disebut dengan Kolom (*Column/Field*). Di setiap pertemuan antara Baris Data dan Kolom itulah item-item data (satuan data terkecil) ditempatkan. Dalam kehidupan sehari-hari, tabel merupakan bentuk natural dalam menyatakan data atau fakta yang sering kita gunakan. Itulah sebabnya model ini lebih mudah kita terapkan daripada model basis data yang lain [KAB-03:8].

Ada 4 macam *key* yang dapat diterapkan pada suatu tabél dalam model basis data relasional, antara lain :

### a. Superkey

Merupakan satu atau lebih atribut (kumpulan atribut) yang dapat membedakan setiap baris data dalam sebuah tabel secara unik.

### b. Candidate-Key

Merupakan kumpulan atribut minimal yang dapat membedakan setiap baris data dalam sebuah tabel secara unik.

### c. Primary-Key

Merupakan key yang diambil dari beberapa candidate-key, dengan didasari pada :

- Key tersebut lebih sering (lebih natural) untuk dijadikan sebagai acuan.
- Key tersebut lebih ringkas.
- Jaminan keunikan key tersebut lebih baik.

### d. Foreign-Key

Merupakan kunci tamu atau asing dalam suatu tabel dimana kunci ini juga merupakan *primary key* dalam tabel lain yang berelasi.

### 2.6.6 Database Management System

Database Management System (DBMS) merupakan perangkat lunak atau program komputer yang dirancang secara khusus untuk memudahkan pengolahan database. DBMS akan menentukan bagaimana data diorganisasi, disimpan, diubah dan diambil kembali. DBMS juga menerapkan mekanisme pengamanan data, pemakaian data secara bersama serta menunjang keakuratan atau konsistensi data dan sebagainya [KAB-03:18].

BRAWIJAYA

Berdasarkan orientasi pemakaiannya, kita dapat mengelompokkan DBMS dalam 2 kategori yaitu [KAB-03:20]:

- DBMS yang berorientasi untuk satu atau sedikit pemakai.
   DBMS yang termasuk dalam kelompok ini antara lain : MS-Access, dBase/Clipper, FoxBase, Borland-Paradox.
- DBMS yang berorientasi untuk banyak pemakai.
   DBMS yang termasuk dalam kelompok ini antara lain : Oracle, Borland-Interbase,
   MS-SQL Server, CA-OpenIngres, Sybase, Informix, IBM DB2.

### 2.7 Ms. Access 2003

Ms. Access merupakan salah satu contoh produk RDBMS yang sangat popular dilingkungan Windows. RDBMS (*Relational Database Management System*) merupakan salah satu macam DBMS yang menggunakan model basis data relasional atau dalam bentuk tabel-tabel yang saling terhubungkan.

Database Pada Ms.Access disimpan dalam sebuah berkas dengan ekstensi "MDB" Di dalam berkas inilah semua objek yang terkait dengan database, termasuk semua tabel, disimpan. Pada Microsoft Access, istilah kolom yang biasa dipakai pada basis data relasional disebut *field* dan baris biasa disebut *record*.

Setiap *field* yang didefinisikan haruslah memiliki tipe data, seperti propertiproperti yang lain. Tipe data pada Microsoft Access diperlihatkan dalam Tabel 2.4.

**Tabel 2.4** Daftar Tipe Data

Tipe Data	Keterangan	Ukuran
Teks	Merupakan tipe data bawaan yang akan	Maksimum terdiri
S	dipilhkan oleh Microsoft Access ketika	atas 255 karakter
T.	suatu field baru diciptakan. Tipe ini	
超	digunakan untuk menyatakan teks atau	
TUE	data yang bisa mengandung huruf,	
	angka, dan karakter-karakter yang lain	
Charles	seperti tanda * dan &. Contoh	THE REPORT
JIIA! TUA	pemakaiannya antara lain untuk nama	CITAL XS I
XWUATT	orang, alamat, dan bahkan untuk angka	ERSLAT
BRANK	yang tidak dimaksudkan untuk dihitung	<b>JIVENER</b>
LAS BREE	(misalnya nomor telepon)	MINIM

Memo	Merupakan tipe data teks yang	Dapat mencapai
	berukuran besar	65.535 karakter
Number	Merupakan tipe data untuk suatu nilai	1, 2, 4, 8, atau 16
	bilangan yang bisa dihitung. Ada	byte tergantung
	bermacam-macam pilihan lebih lanjut	tipe detilnya
	untuk nilai ini. Tebel 2.2	
	memperlihatkan lebih detil untuk tipe	<b>VAUNTA</b>
25317	ini	
Date/Time	Nilai jam dan tanggal dimulai dari	8 byte
	tahun 100 sampai dengan 9999	
Currency	Merupakan tipe data untuk nilai uang.	8 byte
	Keakuratan sampai 15 digit di sebelah	
	kiri tanda pecahan dan 4 digit di sebelah	Y
	kanan tanda pecahan	
AutoNumber	Menyatakan nilai yang urut. Nilai	4 byte
	dibangkitkan oleh Microsoft Access	2
	dan tidak dapat diubah oleh pemakai	$\Diamond$
Yes/No	Menyatakan data yang hanya memiliki	1 bit
	dua kemungkinan saja. Contoh: benar	J
	atau salah, pria atau wanita	
OLE Object	Menyatakan data objek seperti lembar	Sampai 1 gigabyte
	kerja Excel, dokumen Word yang	
	dihubungkan atau dilekatkan ke	
	Microsoft Access	
Hyperlink	Menyatakan data alamat hyperlink	Sampai 2048
到為人	(misalnya digunakan sebagai alamat	karakter
	Web atau untuk melompat ke Word)	
Lookup Wizard	Memungkinkan berhubungan dengan	Biasanya 4 byte
VASTO	nilai pada table lain melalui fasilitas	- TOBR
	kotak daftar (list box) atau kotak kombo	SITATAS
	(combo box)	ER2LATIT
TO DAY	Sumber : [KAB-03:23]	THURSDAY.

Sumber : [KAB-03:23]

BRAWIJAYA

Pemilihan tipe data number yang tepat akan sangat berguna dalam pengematan memori, kecepatan proses, ketilitian perhitungan dan lain-lain. Tipe data number pada *Microsoft Access* diperlihatkan dalam Tabel 2.5.

**Tabel 2.5** Daftar Pilihan Tipe Data Number

Tipe	Keterangan	Ketelitian	Ukuran
Bilangan			TINE
Byte	Bilangan bulat dari 0 s/d 255	-	1 byte
Decimal	Bilangan real dari -10 <sup>28</sup> -1 s.d	28	12 byte
	10 <sup>28</sup> -1	BRA 1.	
Integer	Bilangan bulat -32.768 s/d	- 7	2 byte
	32.767		
Long Integer	Bilangan bulat dari -	· - ^.	4 byte
5	2.147.483.648 s/d		
	2.147.483.647	Freschille Comment	
Single	Dari -3.402823E38 s/d –	なの意	4 byte
	1,401298 x 10 <sup>-45</sup> untuk nilai		
	negatif dan dari 1.401298 x 10 <sup>-</sup>		
	<sup>45</sup> s/d 3.402823 x 10 <sup>38</sup> untuk		
	nilai positif		
Double	Dari -1.79769313486231 x 10 <sup>308</sup>	15	8 byte
	s/d -4,94065645841247 x 10 <sup>-324</sup>	MITTER	
	untuk nilai negatif dan dari		
$\mathfrak{L}$	4,94065645841247 x 10 <sup>-324</sup> s/d		
17/	1.79769313486231 x 10 <sup>308</sup>	<b>9</b>	
14年	untuk nilai positif		
Replication	Untuk membentuk pengenal	Tak ada	16 byte
ID	yang unik dalam suatu field		
MAGI	yang dijadikan kunci primer		TOBE

Sumber : [KAB-03:24]

### 2.8 Diagram Arus Data

Data Flow Diagram (DFD) adalah sebuah teknis grafis yang menggambarkan aliran informasi dan transformasi yang diaplikasikan pada saat data bergerak dari input menjadi output. DFD sering digunakan untuk menggambarkan sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan. DFD merupakan dokumentasi dari sistem yang baik. Komponen-komponen dari diagram arus data [JOG-95:201]:

a. Kesatuan luar (external entity) atau Batas Sistem (boundary)

Setiap sistem pasti mempunyai batas sistem yang memisahkan suatu sistem dengan lingkungan luarnya. Sistem akan menerima *input* dan menghasilkan *output* kepada lingkungan luarnya. Dalam hal ini, sejumlah *input* dapat menjadi hanya satu *output* ataupun sebaliknya. Kesatuan luar merupakan kesatuan di luar lingkungan sistem yang dapat berupa orang, organisasi atau sistem yang lainnya yang berada di lingkungan luarnya yang akan memberikan *input* atau menerima *output* dari sistem.

### b. Arus Data (data flow)

Komponen ini direpresentasikan dengan menggunakan tanda panah yang menuju ke atau dari proses. Alir data ini menunjukkan arus data yang dapat berupa masukan untuk sistem atau hasil dari suatu proses sistem. Beberapa kesalahan yang harus dihindari dalam membuat DFD (*Data Flow Diagram*) antara lain adalah:

- 1. Jangan menghubungkan satu tempat penyimpanan dengan tempat penyimpanan yang lain.
- 2. Jangan membuat hubungan langsung tempat penyimpanan dengan entitas luar.
- 3. Jangan membuat hubungan langsung entitas luar dengan entitas luar yang lain.

### c. Proses (process)

Proses menunjukkan transformasi dari masukan sampai keluaran. Suatu proses terjadi karena adanya arus data yang masuk dan hasil dari proses adalah juga merupakan arus data yang mengalir. Suatu proses harus menerima arus data dan menghasilkan arus data. Beberapa hal yang harus dihindari dalam menggambarkan suatu proses adalah:

1. Proses yang mempunyai masukan tetapi tidak memiliki keluaran, kesalahan ini disebut *Blank Hole* (lubang hitam).

2. Proses yang menghasilkan keluaran tetapi tidak mendapatkan masukan atau proses yang menghasilkan keluaran lengkap dengan data terbatas, kesalahan ini disebut dengan *miracle* (ajaib).

### d. Penyimpanan Data (data store)

Penyimpanan data digunakan untuk menyimpan data hasil proses maupun menyediakan data untuk diproses pada waktu yang lain. Beberapa ketentuan dalam menggambarkan penyimpanan data:

- 1. Hanya proses saja yang dapat berhubungan dengan penyimpanan data.
- 2. Arus data yang menuju ke simpanan data dari suatu proses menunjukan proses *update* terhadap data yang tersimpan di simpanan data.
- 3. Arus data yang berasal dari simpanan data ke suatu proses menunjukan bahwa proses tersebut menggunakan data yang ada di simpanan data.

**SIMBOL** NAMA **FUNGSI** Merupakan kesatuan di luar sistem yang memberikan Entitas Luar input dan output kepada sistem. Menunjukkan arus data yang dapat berupa masukan atau Alir Data hasil dari proses sistem. transformasi Menunjukkan dari masukan sampai keluaran. **Proses** Penyimpanan menyimpan data hasil proses maupun menyediakan data untuk diproses di waktu lain. Data

Tabel 2.6 Simbol Data Flow Diagram

Sumber : [JOG-95:234]

Terdapat dua bentuk DFD yaitu diagram arus data fisik (physical data flow diagram) dan diagram arus data logika (logical data flow diagram). Diagram arus data fisik lebih tepat digunakan untuk menggambarkan sistem yang ada (sistem yang lama) dan lebih ditekankan pada bagaimana proses dari sistem diterapkan (dengan cara bagaimana, oleh siapa dan dimana), termasuk proses-proses manual sedangkan diagram arus data logika lebih tepat digunakan untuk menggambarkan sistem yang akan diusulkan (sistem baru) dan lebih ditekankan logika dari kebutuhan sistem, yaitu proses-proses apa secara logika dibutuhkan oleh sistem. Simbol-simbol dalam DFD tersebut dapat dilihat dalam Tabel 2.6.

Diagram konteks merupakan aliran yang menggambarkan hubungan antara sistem dengan kesatuan luar di luar sistem, yang direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem. Aliran dalam diagram konteks memodelkan masukan ke sistem dan keluaran dari sistem. Aliran data hanya digambarkan dan hanya diperlukan untuk mendeteksi kejadian dalam lingkungan sistem dan harus memberikan respon atau membutuhkan untuk menggambarkan transportasi antara sistem dan eksternal entitas.

### 2.8.1 Entity-Relationship Model (ER Model)

Pada *Model Entity-Relationship*, semesta data yang ada di 'Dunia Nyata' diterjemahkan/ditransformasikan dengan memanfaatkan sejumlah perangkat konseptual menjadi sebuah diagram data, yang disebut sebagai *Diagram Entity-Relationship* (Diagram E-R) [FAT-99:46].

Komponen-komponen dalam Model Entity-Relationship adalah sebagai berikut:

- Entitas (Entity) dan Himpunan Entitas (Entitas Sets)
   Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain. Sedangkan sekelompok entitas yang sejenis dan berada dalam lingkup yang sama membentuk sebuah Himpunan Entitas (Entity
- Sets).

  2. Atribut (Attributes/Properties)

Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik (*properti*) dari entitas tersebut. Penetapan atribut bagi sebuah entitas umumnya memang didasarkan pada fakta yang ada.

- 3. Relasi (*Relationship*) dan Himpunan Relasi (*Relationship Sets*)
  - Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Sedangkan kumpulan semua relasi di antara entitas-entitas yang terdapat pada himpunan entitas-himpunan entitas tersebut membentuk Himpunan Relasi (*Relationship Sets*).
- 4. Kardinalitas/Derajat Relasi

Kardinalitas Relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain.

Kardinalitas Relasi yang terjadi di antara dua himpunan entitas (misalnya A dan B) dapat berupa :

§ Satu ke Satu (One to One)

Berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A

§ Satu ke Banyak (One to Many)

Berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi sebaliknya, di mana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

§ Banyak ke Satu (Many to One)

Berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.

§ Banyak ke Banyak (Many to Many)

Berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan B, dan demikian juga sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

Model Entity-Relationship yang berisi komponen-komponen Himpunan Entitas dan Himpunan Relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari 'Dunia Nyata' yang kita tinjau, dapat digambarkan dengan lebih sitematis dengan menggunakan Diagram Entity-Relationship.

### 2.9 Pengujian

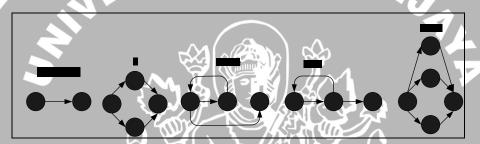
Pengujian *white box* adalah metode desain *test case* yang menggunakan struktur kontrol desain prosedural untuk memperoleh *test case*. Dengan menggunakan metode pengujian *white box*, perekayasa sistem dapat melakukan *test case* yang: (1) memberikan jaminan bahwa semua jalur independen pada suatu modul telah digunakan paling tidak satu kali; (2) menggunakan semua keputusan logis pada sisi *true* dan *false*; (3) mengeksekusi semua loop pada batasan mereka dan pada batasan operasional [PRE-02:30].

### 2.9.1 Pengujian Basis Path

Metode *basis path* ini memungkinkan desainer test case mengukur kompleksitas logis dari desain prosedural dan menggunkannya sebagai pedoman untuk menetapkan *basis set* dari jalur eksekusi. Test case yang dilakukan untuk menggunakan *basis set* tersebut dijamin untuk menggunakan setiap statemen di dalam program paling tidak sekali selama pengujian.

### 2.9.2 Notasi Diagram Alir

Grafik alir berfungsi sebagai piranti yang berguna untuk memahami aliran kontrol dan untuk menggambarkan metode basis path. Grafik alir menggambarkan aliran kontrol logika yang menggunakan notasi yang ditunjukkan dalam Gambar 2.9.



Gambar 2.9 Notasi Diagram Aliran Sumber: [PRE-02:530]

Pada gambar diatas lingkaran disebut dengan simpul grafik alir atau *node* merepresentasikan satu atau lebih statemen prosedural. Anak panah pada gambar diatas menggambarkan *edges* atau *links*, merepresentasikan aliran kontrol dan analog dengan anak panah bagan alir. *Edge* harus berhenti pada suatu simpul, meskipun bila simpul tersebut tidak merepresentasikan statemen prosedural. Area yang dibatasi oleh *edge* dan simpul disebut *region*. Pada saat menghitung *region*, kita memasukkan area di luar grafik dan menghitungnya sebagai sebuah region.

### 2.9.3 Kompleksitas Siklomatis

Kompleksitas siklomatis adalah metriks perangkat lunak yang memberikan pengukuran kuantitatif terhadap kompleksitas logis suatu program. Bila metriks ini digunkan dalam konteks metode pengujian basis path, maka nilai yang terhitung untk kompleksitas siklomatis menentukan jumlah jalur independen dalam basis set suatu program dan memberi batas atas bagi jumlah pengujian yang harus dilakukan untuk memastikan bahwa semua statemen telah dieksekusi sedikitnya satu kali.

BRAWIJAY/

Jalur independen adalah jalur yang melalui program yang mengintroduksi sedikitnya satu rangkaian statemen proses baru atau suatu kondisi baru. Bila dinyatakan dengan terminologi grafik alir, jalur inependen harus bergerak sepanjang paling tidak satu edge yang tidak dilewatkan sebelum jalur tersebut ditentukan.

Kompleksitas dihitung dalam salah satu dari tiga cara berikut :

- 1. Jumlah region grafik alir sesuai dengan kompleksitas siklomatis.
- 2. *Kompleksitas siklomatis*, V(G), untuk grafik alir ditentukan sebagai V(G) = E N + 2 di mana E adalah jumlah edge grafik alir dan N adalah jumlah simpul grafik alir.
- 3. Kompleksitas siklomatis, V(G), untuk grafik alir G juga ditentukan sebagai V(G) = P + I diaman P adalah jumlah simpul predikat yang diisikan dalam grafik alir G.

### 2.9.4 Test Case

Metode pengujian *basis path* dapat diaplikasikan pada desain prosedural atau kode sumber. Langkah – langkah untuk melakukan *test case* adalah sebagi berikut :

- 1. Dengan menggunakan desain atau kode sebagai dasar, menggambarkan sebuah grafik alir yang sesuai.
- 2. Menentukan kompleksitas siklomatis dari garfik alir resultan.
- 3. Menentukan sebuah basis set dari jalur independen secara linier.
- 4. Menyiapkan test case yang akan memaksa adanya eksekusi setiap basis set

### 2.9.5 Turing Test

Turing test merupakan sebuah metode pengujian yang dibuat oleh Alan Turing. Proses uji ini melibatkan seorang penanya (human intrrogator) dan dua obyek yang ditanyai. Yang satu adalah seorang manusia dan satunya adalah sebuah mesin yang akan diuji. Penanya tidak dapat melihat langsung pada obyek yang ditanyai. Penanya diminta untuk membedakan mana jawaban komputer mana jawaban manusia berdasarkan jawaban kedua obyek tersebut. Jika penanya tidak dapat membedakan mana jawaban mesin mana jawaban manusia maka Turing berpendapat bahwa mesin yang diuji tersebut diasumsikan cerdas [KUS-05:5].

Pada sistem "Turing Test", human interrogator harus bisa menilai apakah respon

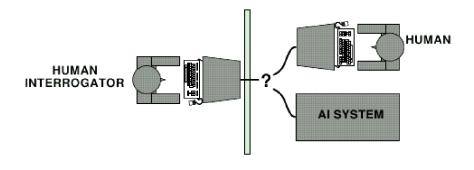
yang diberikan berasal dari manusia (*human*) atau AI system. Pemrograman yang bisa melalui Test ini harus memiliki kemampuan berikut [VIC-02]:

- 1. Natural Language Processing: mampu berkomunikasi dgn bahasa manusia
- 2. Knowledge Representation: pengetahuan awal

BRAWIJAYA

- 3. Atomated Reasoning: mengolah pengetahuan awal untuk menjawab dan membuat kesimpulan baru.
- 4. Machine Learning : kemampuan beradaptasi, mendeteksi dan mengeksplorasi pola.

Keseluruhan dari turing test melibatkan juga computer vision (berinteraksi dengan objek melalui penglihatan) dan robotics (berinteraksi lewat gerak).



Gambar 2.10 metode Turing Test Sumber: [VIC-02]

Gambar 2.10 menjelaskan sebuah metode "Turing Test". Ada dua buah ruangan, A dan B. salah satu ruang berisi komputer, yang lain berisi manusia. Pengintrogasi berada diluar dan tidak tahu ruangan mana yang berisi komputer dan mana berisi manusia. Dia dapat mengajukan pertanyaan melalui sebuah teletype dan akan menerima jawaban dari A dan B. pengintrogasi kemudian harus mengidentifikasi apakah A atau B yang adalah manusia. Untuk melewati turing test komputer harus bisa menipu pengintregator sehingga ia percaya computer itu adalah manusia [NIM-02].

# BRAWIJAYA

### BAB III METODE PENELITIAN

Pada bab ini akan dibahas berbagai tahapan penelitian yang digunakan untuk menyelesaikan tugas akhir. Dalam penyusunan tugas akhir ini diperlukan data, informasi dan metode dalam penyelesaiannya. Langkah – langkah yang perlu dilakukan untuk dapat merealisasikan perangkat lunak yang akan dibuat adalah sebagai berikut:

SBRAWIUA

- § Observasi
- § Studi literatur
- § Perancangan
- § Implementasi
- § Pengujian dan analisis
- § Pengambilan kesimpulan
- § Penulisan laporan

### 3.1 Observasi

Observasi dilakukan dengan wawancara langsung terhadap seorang pakar yang dalam hal ini adalah seorang dokter sepesialis kulit untuk mendapatkan informasi tentang penyakit-penyakit yang banyak menyerang pada kulit manusia serta gejalagejalanya dan tindakan yang perlu dilakukan.

### 3.2 Studi Literatur

Melakukan studi literatur yakni mengumpulkan dan mempelajari bahan pustaka yang dibutuhkan dalam penyelesaian tugas akhir. Bahan pustaka tersebut membahas tentang:

- Kecerdasan Buatan
- 2. Sistem Pakar
- 3. Borland Delphi 7
- 4. Ms. Access 2003
- 5. Gangguan dan Penyakit Kulit
- 6. Perancangan
  - Diagram Arus Data
  - Entity Relationship Model (ER Model)
- 7. Pengujian
  - White BoxTesting

- Black boxTesting
- Turing Test

### 3.3 Perancangan

Melakukan analisis dan perancangan perangkat lunak sebagai dasar untuk melakukan implementasi. Adapun tahap-tahap dalam perancangan adalah membuat analisis sistem, perancangan sistem pakar yang meliputi penyusunan rules dan inference tree dan perancangan perangkat lunak.

### 3.4 Implementasi

Melakukan implementasi dari hasil perancangan yang telah didefinisikan sebelumnya kedalam bentuk kode (*coding*) dengan bahasa pemrograman Borland Delphi7 serta Ms Access sebagai sarana untuk penyimpanan databasenya.

### 3.5 Pengujian dan analisis

Melakukan pengujian perangkat lunak yang telah dibuat. Ditujukan untuk menemukan kesalahan, sehingga perangkat lunak akan terbebas dari segala kesalahan pada saat dieksekusi oleh pengguna. Pengujian meliputi pengujian basis data, pengujian algoritma dengan teknik white box testing dan dilakukan pengujian sistem dengan metode black box testing, serta dilakukan pengujian tingkat kepakaran sistem yang dibuat dengan menggunakan turing test.

### 3.6 Pengambilan Kesimpulan

Menarik kesimpulan yang dapat diambil dari hasil perancangan, implementasi, dan pengujian aplikasi Sistem Pakar untuk mendiagnosis gangguan dan penyakit kulit pada manusia dengan menggunakan faktor kepastian, serta saran yang menurut penulis sangat diperlukan untuk pengembangan aplikasi ini pada masa yang akan datang.

### 3.7 Penulisan Laporan

Melakukan dokumentasi skripsi yang sudah dibuat, sehingga dokumentasi ini dapat berguna dan diharapkan dapat dikembangkan menjadi lebih baik.

# BRAWIJAYA

### BAB III METODE PENELITIAN

Pada bab ini akan dibahas berbagai tahapan penelitian yang digunakan untuk menyelesaikan tugas akhir. Dalam penyusunan tugas akhir ini diperlukan data, informasi dan metode dalam penyelesaiannya. Langkah – langkah yang perlu dilakukan untuk dapat merealisasikan perangkat lunak yang akan dibuat adalah sebagai berikut:

SBRAWIUA

- § Observasi
- § Studi literatur
- § Perancangan
- § Implementasi
- § Pengujian dan analisis
- § Pengambilan kesimpulan
- § Penulisan laporan

### 3.8 Observasi

Observasi dilakukan dengan wawancara langsung terhadap seorang pakar yang dalam hal ini adalah seorang dokter sepesialis kulit untuk mendapatkan informasi tentang penyakit-penyakit yang banyak menyerang pada kulit manusia serta gejalagejalanya dan tindakan yang perlu dilakukan.

### 3.9 Studi Literatur

Melakukan studi literatur yakni mengumpulkan dan mempelajari bahan pustaka yang dibutuhkan dalam penyelesaian tugas akhir. Bahan pustaka tersebut membahas tentang:

- Kecerdasan Buatan
- 9. Sistem Pakar
- 10. Borland Delphi 7
- 11. Ms. Access 2003
- 12. Gangguan dan Penyakit Kulit
- 13. Perancangan
  - Diagram Arus Data
  - Entity Relationship Model (ER Model)
- 14. Pengujian
  - White BoxTesting

- Black boxTesting
- Turing Test

### 3.10 Perancangan

Melakukan analisis dan perancangan perangkat lunak sebagai dasar untuk melakukan implementasi. Adapun tahap-tahap dalam perancangan adalah membuat analisis sistem, perancangan sistem pakar yang meliputi penyusunan rules dan inference tree dan perancangan perangkat lunak.

### 3.11 Implementasi

Melakukan implementasi dari hasil perancangan yang telah didefinisikan sebelumnya kedalam bentuk kode (*coding*) dengan bahasa pemrograman Borland Delphi7 serta Ms Access sebagai sarana untuk penyimpanan databasenya.

### 3.12 Pengujian dan analisis

Melakukan pengujian perangkat lunak yang telah dibuat. Ditujukan untuk menemukan kesalahan, sehingga perangkat lunak akan terbebas dari segala kesalahan pada saat dieksekusi oleh pengguna. Pengujian meliputi pengujian basis data, pengujian algoritma dengan teknik white box testing dan dilakukan pengujian sistem dengan metode black box testing, serta dilakukan pengujian tingkat kepakaran sistem yang dibuat dengan menggunakan turing test.

### 3.13 Pengambilan Kesimpulan

Menarik kesimpulan yang dapat diambil dari hasil perancangan, implementasi, dan pengujian aplikasi Sistem Pakar untuk mendiagnosis gangguan dan penyakit kulit pada manusia dengan menggunakan faktor kepastian, serta saran yang menurut penulis sangat diperlukan untuk pengembangan aplikasi ini pada masa yang akan datang.

### 3.14 Penulisan Laporan

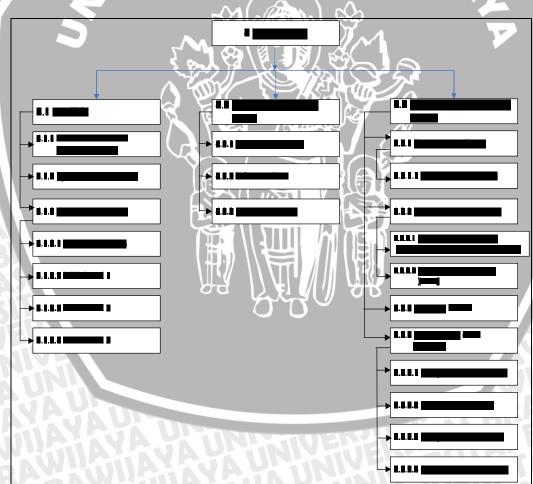
Melakukan dokumentasi skripsi yang sudah dibuat, sehingga dokumentasi ini dapat berguna dan diharapkan dapat dikembangkan menjadi lebih baik.

### BAB IV PERANCANGAN

### 4.1 Analisis Sistem

Pada skripsi ini dirancang suatu sistem pakar untuk mendiagnosis gangguan dan penyakit kulit pada manusia. Perancangan yang dilakukan meliputi tiga tahap. Proses analisis kebutuhan dilakukan pada tahap pertama dan perancangan sistem dilakukan pada tahap kedua, sedangkan tahap ketiga adalah perancangan perangkat lunak. Proses analisis meliputi analisis kebutuhan perangkat lunak dan pemodelan perangkat lunak yaitu dengan membuat DFD.

Perancangan yang dilakukan dapat digambarkan dengan diagram pohon seperti dalam gambar 4.1 berikut:



Gambar 4.1 Diagram Pohon Perancangan

Sumber : Perancangan

## BRAWIJAY

### 4.1.1 Analisis Kebutuhan Perangkat Lunak

Pengembangan sebuah perangkat lunak bertujuan untuk menghasilkan perangkat lunak yang dapat memenuhi kebutuhan *user*. setiap pengembangan sebuah sistem perangkat lunak memerlukan adanya dokumentasi terhadap kebutuhan–kebutuhan *user*. tahap analisis kebutuhan dilakukan dengan memodelkan kebutuhan dalam *Data Flow Diagram* (DFD). *Data Flow Diagram* bertujuan untuk menggambarkan kebutuhan-kebutuhan fungsional yang harus disediakan oleh sistem agar sesuai dengan kebutuhan *user*.

Kebutuhan sistem pakar tersebut antara lain :

- 1. Pengguna dapat melakukan konsultasi (sesi tanya jawab) dengan memasukkan gejala-gejala fisik yang dialaminya untuk mendapatkan informasi gangguan dan penyakit kulit yang diderita dan cara mengatasinya serta faktor kepastiannya [SP\_010].
- 2. Pakar dapat menentukkan jenis gangguan dan penyakit kulit, gejala-gejalanya, aturan-aturan yang digunakan untuk mendiagnosis setiap gangguan dan penyakit kulit berdasarkan gejala-gejalanya serta memberi solusi dari setiap gangguan dan penyakit kulit [SP\_020].
- 3. *User* pakar harus melakukan *login* untuk dapat mengakses sistem secara keseluruhan sedangkan *user* bukan pakar dapat langsung mengakses tanpa perlu *login* terlebih dahulu[SP\_030].
- 4. Sistem diimplementasikan dengan bahasa pemrograman Borland Delphi 7 dan basis data Ms Access 2003 [SP\_040].
- 5. Sistem operasi yang digunakan adalah sistem operasi Windows [SP\_050].

### 4.1.2 Spesifikasi Kebutuhan

Spesifikasi kebutuhan dibutuhkan untuk menjelaskan kebutuhan perangkat lunak yang telah didefinisikan sebelumnya secara lebih detail dan tepat yang akan menjadi dasar bagi perancangan dan implementasi.

### Definisi:

 Pengguna dapat melakukan konsultasi (sesi tanya jawab) dengan memasukkan gejala-gejala fisik yang dialami untuk mendapatkan informasi gangguan dan penyakit kulit dan cara mengatasinya serta faktor kepastianya [SP\_010].

### Spesifikasi:

- 1.1 Pengguna harus dapat melakukan konsultasi dengan sistem pakar yaitu dengan diberikan pertanyaan oleh sistem dan harus dijawabnya.[SP\_011]
- 1.2 Pengguna harus mendapatkan informasi tentang gangguan dan penyakit kulit yang dialami.[SP\_012]
- 1.3 Pengguna harus mendapatkan informasi faktor kepastian gangguan dan penyakit yang dialami.[SP\_013]
- 1.4 Pengguna harus mendapat informasi bagaimana mengatasi gangguan dan penyakit kulit tersebut. [SP\_014]

### Definisi:

2. Pakar dapat menentukkan jenis gangguan dan penyakit kulit, gejalagejalanya, aturan-aturan yang digunakan untuk mendiagnosis setiap gangguan dan penyakit kulit berdasarkan gejala-gejalanya serta memberi solusi dari setiap gangguan dan penyakit kulit [SP\_020].

### Spesifikasi:

- 2.1 Pakar harus dapat menentukkan gangguan dan penyakit kulit serta gejalagejalanya. [SP\_021]
- 2.2 Pakar harus dapat menentukan aturan-aturan yang digunakan untuk mendiagnosis gangguan dan penyakit kulit berdasarkan gejala-gejala yang ada. [SP\_022]
- 2.3 Pakar dapat menentukan solusi dari setiap gangguan dan penyakit kulit. [SP\_023]

### **Definisi:**

3. *User* pakar harus melakukan *login* untuk dapat mengakses sistem secara keseluruhan sedangkan *user* bukan pakar dapat langsung mangakses tanpa perlu login terlebih dahulu. [SP\_030].

### Spesifikasi:

- 3.1 *User* pakar harus memberikan masukan *username* dan password yang terdaftar pada sistem. [SP\_031]
- 3.2 *User* pakar dapat melanjutkan jika login yang diberikan adalah valid atau benar [SP\_032]
- 3.3 Sistem akan memberikan pesan kesalahan saat login yang diberikan tidak valid dan user pakar tidak dapat melanjutkan.[SP\_033]

3.4 *User* bukan pakar dapat langsung mengakses tanpa perlu login terlebih dahulu. [SP\_034]

### Definisi:

4. Sistem diimplementasikan dengan bahasa pemrograman Borland Delphi 7 dan basis data Ms Access 2003 [SP\_040].

### Spesifikasi:

- 4.1 Bahasa pemrograman yang digunakan dalam implementasi sistem pakar ini adalah Borland Delphi 7. [SP\_ 041]
- 4.2 Database yang digunakan dalam pembuatan sistem ini adalah Ms. Access 2003 [SP\_042]

### Definisi:

5. Sistem operasi yang digunakan adalah sistem operasi Windows. [SP\_050]

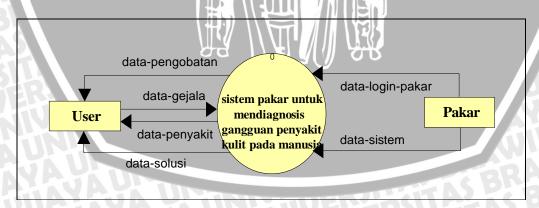
### Spesifikasi:

5.1 Sistem yang dibuat akan menggunakan Microsoft Windows XP. [SP\_051]

### 4.1.3 Data Flow Diagram

Data Flow Diagram (DFD) merupakan salah satu alat yang digunakan untuk menggambarkan keseluruhan sistem Pakar yang akan dibuat. DFD yang pertama kali dibuat adalah Context Diagram.

### 4.1.3.1 Diagram Konteks



Gambar 4.2 Diagram Konteks Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian

Sumber : Perancangan

BRAWIJAYA

Berdasarkan Gambar 4.2 proses "Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian" mempunyai data masukan berupa:

1. data-login-pakar

login\_pakar diberikan oleh user pakar untuk dilakukan proses validasi login, parameter yang digunakan adalah:

a. username

berisi masukan username yang diberikan oleh pakar.

b. Password

Berisi masukan *password* yang diberikan pakar

2. data-sistem

data-sistem diberikan oleh pakar berisi data-data yang diperlukan oleh sistem pakar seperti data aturan, gangguan dan penyakit kulit, dan solusi.

BRAM

3. data-gejala

data-gejala diberikan oleh user.

Berdasarkan Gambar 4.2 proses "Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian" mempunyai data keluaran berupa:

1. data--penyakit

berisi informasi tentang gangguan dan penyakit kulit yang diderita.

2. data-pengobatan

berisi informasi tentang obat untuk mengatasi gangguan dan penyakit kulit.

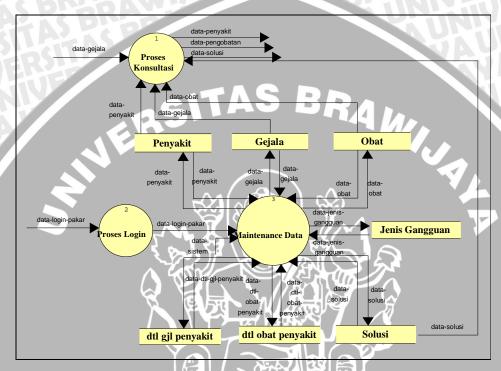
3. data-solusi

berisi informasi tentang solusi untuk mengatasi gangguan dan penyakit kulit.

Untuk masuk ke dalam sistem tersebut *user* pakar harus memasukkan login terlebih dahulu. Jika *data-login-pakar* yang dimasukan sudah benar maka *user* pakar dapat memasukan data yang diperlukan oleh sistem. Jika *user* pakar salah dalam memasukan login maka akan muncul pesan kesalahan dan proses akan tetap berada pada *proses login*. Hal ini dikarenakan akuisisi pengetahuan harus benar-benar dilakukan oleh orang yang yang berhak. *User* bukan pakar dapat langsung masuk kedalam sistem tanpa harus melakukan proses login terlebih dahulu. Pilihan-pilihan proses yang akan didapatkan pengguna bukan pakar yaitu: konsultasi dan Tips.

### 4.1.3.2 DFD Level 1

DFD level 1 menggambarkan proses-proses yang lebih detail dari diagram konteks untuk Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian. DFD level 1 digambarkan dalam Gambar 4.3.



Gambar 4.3 DFD Level 1 Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian

Sumber: Perancangan

Gambar 4.3 merupakan DFD Level 1 dari Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian dimana sistem tersebut dibagi menjadi tiga proses, yaitu : proses Login, proses Konsultasi, proses Maintenance Data. Dari diagram tersebut, maka dapat dijabarkan lebih detail proses-proses tersebut sebagai berikut:

### 1. Proses Login (proses 1)

Proses ini merupakan proses untuk mendapatkan nilai login yang telah diberikan oleh *user* (pakar). Nilai login yang diberikan adalah *usename* dan *password*. Proses ini mempunyai satu data masukan yaitu *data-login-pakar*. Dan mempunyai satu data keluaran yaitu *data-login-pakar*.

# BRAWIJAYA

### 2. Proses Konsultasi

Proses ini merupakan proses untuk memasukkan data gejala dari *user* dan sistem akan memberikan gangguan dan penyakit kulit dari data gejala yang dimasukkan tersebut serta pengobatan.

BRAWIUAL

Proses ini mempunyai masukan:

- a. data-gejala
- b. data-jenis-penyakit
- c. data-pengobatan
- d. data-solusi

### Proses ini mempunyai keluaran:

- a. data-jenis-penyakit
- b. data-pengobatan
- c. data-solusi

### Data ini menggunakan data store:

- a. data-gejala
- b. data-penyakit
- c. data-pengobatan
- d. data-solusi

### 3. Proses Maintenance Data

Proses ini merupakan proses untuk memasukan data dan memperbarui data.

Proses ini mempunyai data masukan:

- a. data-penyakit
- b. data-pengobatan
- c. data-login-pakar
- d. data-sistem
- e. data-dtl-gjl-penyakit
- f. data-dtl-obat-penyakit
- g. data-gejala
- h. data-jenis-gangguan
- i. data-solusi

### Proses ini mempunyai data keluaran:

- a. data-penyakit
- b. data-pengobatan
- c. data-dtl-gjl-penyakit

BRAWIJAY

- d. data-dtl-obat-penyakit
- e. data-gejala
- f. data-jenis-gangguan
- g. data-solusi

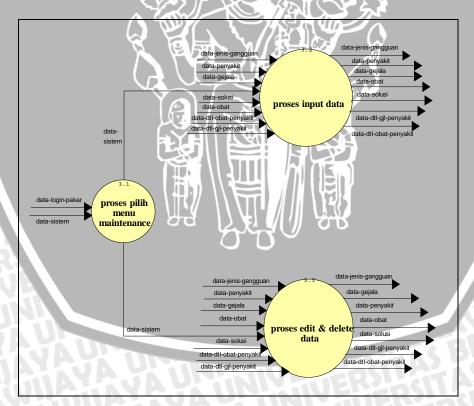
Data ini menggunakan data store:

- a.solusi
- b.jenis penyakit
- c.pengobatan
- d.gejala
- e.dtl gjl penyakit
- f. dtl obat penyakit

### SITAS BRAWIUA

### 4.1.3.3 DFD Level 2

Proses Maintenance Data dapat dijelaskan menjadi lebih detail lagi melalui DFD Level 2 dalam Gambar 4.4.



**Gambar 4.4** DFD Level 1 Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian

Sumber : Perancangan

Gambar 4.4 merupakan DFD Level 2 proses Maintenance Data dimana proses tersebut dibagi menjadi proses-proses yang lebih spesifik, yaitu : proses pilih menu maintenance, proses input data, proses edit dan delete data. Dari diagram tersebut, maka dapat dijabarkan lebih detail proses-proses tersebut sebagai berikut:

### 1. Proses Pilih Menu Maintenance

Proses ini merupakan proses yang dilalui pertama pada proses maintenance data. Proses ini mengambil data berupa login pakar. Apabila pakar telah melakukan login dengan benar maka pakar dapat memberikan data-data yang dibutuhkan sistem RAWINA dengan menambah data atau mengubah dan menghaput data yang sudah ada.

Proses ini mempunyai data masukan:

- a. data-login-pakar
- b. data-sistem

Proses ini memunyai data keluaran:

- a. data-sistem
- 2. Proses Input Data

Proses ini merupakan proses yang digunakan untuk memasukkan data diberikan oleh pakar. Proses ini akan berjalan apabila pakar telah melakukan login.

Proses ini mempunyai data masukan:

- data-penyakit
- data-solusi
- data-sistem
- data-pengobatan
- data-gejala
- data-dtl-obat-penyakit
- data-dtl-gjl-penyakit
- data-jenis-gangguan

Proses ini mempunyai data keluaran:

- data-penyakit
- data-solusi
- data-sistem
- data-pengobatan
- data-gejala
- data-dtl-obat-penyakit
- data-dtl-gjl-penyakit

h. data-jenis-gangguan

### 3. Proses Edit dan Delete Data

Proses ini untuk menghapus atau mengubah data yang ada di dalam sistem. Proses ini hanya dapat dilakukan oleh pakar.

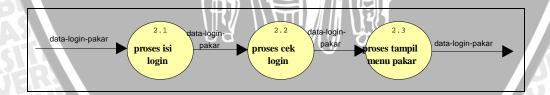
BRAWINAL

Proses ini mempunyai data masukan:

- a. data-penyakit
- b. data-solusi
- c. data-sistem
- d. data-pengobatan
- e. data-gejala
- f. data-dtl-obat-penyakit
- g. data-dtl-gjl-penyakit
- h. data-jenis-gangguan

Proses ini mempunyai data keluaran:

- a. data-penyakit
- b. data-solusi
- c. data-sistem
- d. data-pengobatan
- e. data-gejala
- f. data-dtl-obat-penyakit
- g. data-dtl-gjl-penyakit
- h. data-jenis-gangguan



Gambar 4.5 DFD Level 2 Proses Login Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia

Sumber: Perancangan

Gambar 4.5 merupakan DFD Level 2 proses login dimana proses tersebut dibagi menjadi proses yang lebih spesifik, yaitu : proses isi login, proses cek login, proses tampil menu pakar.

Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut :

### 1. Proses Isi login

Proses ini mendapat masukan *data-login-pakar* dan menghasilkan *data-login-pakar* untuk digunakan pada proses selanjutnya sesuai dengan data masukan yang diterima. Proses Isi Login merupakan proses yang pertama dilakukan dalam proses login.

Proses ini mempunyai data masukan:

a. data-login-pakar

Proses ini mempunyai data keluran:

a. data-login-pakar

### 2. Proses Cek Login

Proses ini merupakan proses memeriksa masukan *data-login-pakar*. Apabila data yang dimasukan sudah benar maka proses akan berjalan menuju proses selanjutnya.

BRAN

Proses ini mempunyai data masukan:

a. data-login-pakar

Proses ini mempunyai data keluaran:

a. data-login-pakar

### 3. Proses Tampil Menu Pakar

Proses ini merupakan proses yang dilakukan setelah proses cek login dilakukan dan data yang dimasukkan sudah benar. Pakar akan mendapatkan menu untuk menambah, mengubah atau menghapus data yang sudah ada didalamnya.

Proses ini mempunyai data masukan:

a. data-login-pakar

Proses ini mempunyai data keluaran:

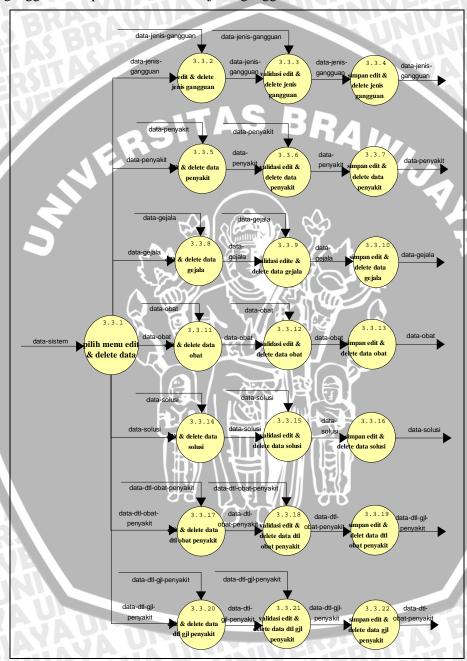
a. data-login-pakar

b.

### 4.1.3.4 DFD Level 3

Gambar 4.6 merupakan DFD Level 3 Proses Pilih Menu Edit dan Delete Data dimana proses tersebut dapat dibagi menjadi proses yang lebih spesifik lagi, yaitu : proses edit dan delete data solusi, proses validasi edit dan delete solusi, simpan edit dan delete solusi, proses edit dan delete data penyakit, proses validasi edit dan delete penyakit, simpan edit dan delete penyakit, proses edit dan delete data pengobatan, proses validasi edit dan delete pengobatan, proses edit dan delete data gejala, proses validasi edit dan delete gejala, simpan edit dan delete

gejala, proses edit dan delete data dtl gjl penyakit, proses validasi edit dan delete dtl gjl penyakit, simpan edit dan delete dtl gjl penyakit, proses edit dan delete dtl obat penyakit, proses validasi edit dan delete dtl obat penyakit, simpan edit dan delete dtl obat penyakit, proses edit dan delete jenis gangguan, proses validasi edit dan delete jenis gangguan, simpan edit dan delete jenis gangguan.



**Gambar 4.6** DFD Level 3 Proses Pilih Menu Edit dan Delete Data Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian

Sumber : Perancangan

Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

Proses Edit dan Delete Data jenis gangguan

Proses edit dan delete data jenis gangguan dilakukan untuk mengubah atau menghapus data jenis gangguan yang tersimpan dalam data store jenis gangguan.

Proses ini mempunyai masukan:

a. data-jenis-gangguan

Proses ini mempunyai keluaran:

- a. data-jenis-gangguan
- BRAWIVAL 2. Proses Validasi Edit dan Delete Data jenis gangguan

Proses ini mempunyai masukan:

a. data-jenis-gangguan

Proses ini mempunyai keluaran:

- a. data-jenis-gangguan
- 3. Simpan Edit dan Delete Data jenis gangguan

Proses ini mempunyai masukan:

a. data-jenis-gangguan

Proses ini mempunyai keluaran:

- a. data-jenis-gangguan
- 4. Proses Edit dan Delete Data Penyakit

Proses edit dan delete data penyakit dilakukan untuk mengubah atau menghapus data penyakit yang tersimpan dalam data store penyakit.

Proses ini mempunyai masukan:

a. data-penyakit

Proses ini mempunyai keluaran:

- a. data-penyakit
- 5. Proses Validasi Edit dan Delete Data Penyakit

Proses ini mempunyai masukan:

a. data-penyakit

Proses ini mempunyai keluaran:

- a. data-penyakit
- Simpan Edit dan Delete Data Penyakit

Proses ini mempunyai masukan:

a. data-penyakit

Proses ini mempunyai keluaran:

a. data-penyakit

Proses Edit dan Delete Data Pengobatan

Proses edit dan delete data pengobatan dilakukan untuk mengubah atau menghapus data peengobatan yang tersimpan dalam data store pengobatan.

Proses ini mempunyai masukan:

a. data-pengobatan

Proses ini mempunyai keluaran:

- a. data-pengobatan
- BRAWIUAL 8. Proses Validasi Edit dan Delete Data Pengobatan

Proses ini mempunyai masukan:

a. data-pengobatan

Proses ini mempunyai keluaran:

- a. data-pengobatan
- 9. Simpan Edit dan Delete Data Pengobatan

Proses ini mempunyai masukan:

a. data-pengobatan

Proses ini mempunyai keluaran:

- a. data-pengobatan
- 10. Proses Edit dan Delete Data Gejala

Proses edit dan delete data gejala dilakukan untuk mengubah atau menghapus data gejala yang tersimpan dalam data store gejala.

Proses ini mempunyai masukan:

a. data-gejala

Proses ini mempunyai keluaran:

- a. data-gejala
- 11. Proses Validasi Edit dan Delete Data Gejala

Proses ini mempunyai masukan:

a. data-gejala

Proses ini mempunyai keluaran:

- a. data-gejala
- 12. Simpan Edit dan Delete Data Gejala

Proses ini mempunyai masukan:

a. data-gejala

Proses ini mempunyai keluaran:

a. data-gejala

### 13. Proses Edit dan Delete Data Dtl Gjl Penyakit

Proses edit dan delete data dtl gjl penyakit dilakukan untuk mengubah atau menghapus data detail gejala penyakit yang tersimpan dalam *data store* dtl gjl penyakit.

Proses ini mempunyai masukan:

a. data-dtl-gjl-penyakit

Proses ini mempunyai keluaran:

- a. data-dtl-gjl-penyakit
- 14. Proses Validasi Edit dan Delete Data Dtl Gjl Penyakit

Proses ini mempunyai masukan:

a. data-dtl-gjl-penyakit

Proses ini mempunyai keluaran:

a. data-dtl-gjl-penyakit

15. Simpan Edit dan Delete Data Dtl Gjl Penyakit

Proses ini mempunyai masukan:

a. data-dtl-gjl-penyakit

Proses ini mempunyai keluaran:

a. data-dtl-gjl-penyakit

16. Proses Edit dan Delete Data Dtl Obat Penyakit

Proses edit dan delete data dtl obat penyakit dilakukan untuk mengubah atau menghapus data detail obat penyakit yang tersimpan dalam *data store* dtl obat penyakit.

Proses ini mempunyai masukan:

a. data-dtl-obat-penyakit

Proses ini mempunyai keluaran:

- a. data-dtl-obat-penyakit
- 17. Proses Validasi Edit dan Delete Data Dtl Obat Penyakit

Proses ini mempunyai masukan:

a. data-dtl-obat-penyakit

Proses ini mempunyai keluaran:

- a. data-dtl-obatpenyakit
- 18. Simpan Edit dan Delete Data Dtl Obat Penyakit

Proses ini mempunyai masukan:

a. data-dtl-obat-penyakit

Proses ini mempunyai keluaran:

- a. data-dtl-obat-penyakit
- 19. Proses Edit dan Delete Data Solusi

Proses edit dan delete data solusi dilakukan untuk mengubah atau menghapus data solusi yang tersimpan dalam *data store* solusi.

BRAWINAL

Proses ini mempunyai masukan:

a. data-solusi

Proses ini mempunyai keluaran:

- a. data-solusi
- 20. Proses Validasi Edit dan Delete Data Solusi

Proses ini mempunyai masukan:

a. data-solusi

Proses ini mempunyai keluaran:

- a. data-solusi
- 21. Simpan Edit dan Delete Data Dtl Obat Solusi

Proses ini mempunyai masukan:

a. data-solusi

Proses ini mempunyai keluaran:

a. data-solusi

Gambar 4.7 merupakan DFD Level 3 Proses Pilih Menu Input dimana proses tersebut dapat dibagi menjadi proses yang lebih spesifik lagi, yaitu : proses Input data tanya gejala, proses validasi data jenis gangguan, simpan data jenis gangguan, proses Input data penyakit, proses validasi data penyakit, simpan data penyakit, proses Input data data pengobatan, proses validasi data pengobatan, simpan data pengobatan, proses Input data gejala, proses validasi data gejala, simpan data gejala, proses Input data dtl gjl penyakit, proses validasi data dtl gjl penyakit, simpan data dtl gjl penyakit, proses Input data dtl obat penyakit, proses validasi data dtl obat penyakit, simpan data dtl obat penyakit, proses Input data solusi, proses validasi data solusi, simpan data solusi. Penjabaran dari proses-proses tersebut dapat dijelaskan sebagai berikut:

1. Proses Input Data jenis gangguan

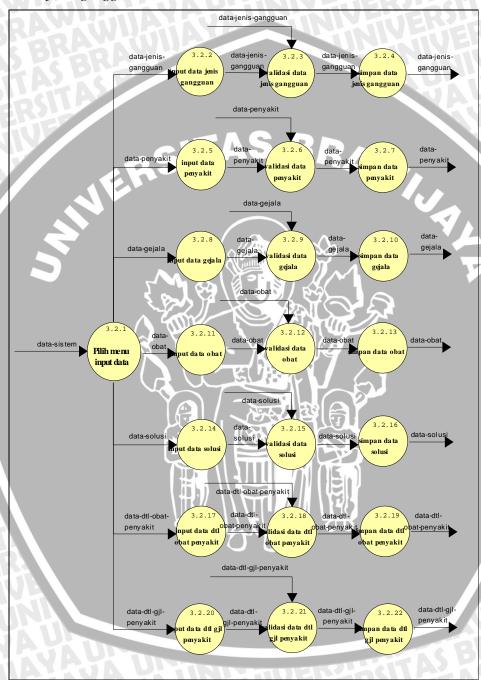
Proses Input data jenis gangguan dilakukan untuk menambahkan data jenis gangguan dan disimpan dalam *data store* jenis gangguan.

Proses ini mempunyai masukan:

a. data-jenis-gangguan

Proses ini mempunyai keluaran:

a. data-jenis-gangguan



**Gambar 4.7** DFD Level 3 Proses Pilih Menu Input Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian

Sumber: Perancangan

2. Proses Validasi Data jenis gangguan

Proses ini mempunyai masukan:

a. data-jenis-gangguan

Proses ini mempunyai keluaran:

- a. data-jenis-gangguan
- 3. Simpan Data jenis gangguan

Proses ini mempunyai masukan:

a. data-jenis-gangguan

Proses ini mempunyai keluaran:

a. data-jenis-gangguan

4. Proses Input Data Jenis Penyakit

Proses input data jenis penyakit dilakukan untuk menambahkan data jenis penyakit

BRAM

dan disimpan dalam data store jenis penyakit.

Proses ini mempunyai masukan:

a. data-jenis-penyakit

Proses ini mempunyai keluaran:

- a. data-jenis-penyakit
- 5. Proses Data Jenis Penyakit

Proses ini mempunyai masukan:

a. data-jenis-penyakit

Proses ini mempunyai keluaran:

- a. data-jenis-penyakit
- 6. Simpan Data Jenis Penyakit

Proses ini mempunyai masukan:

a. data-jenis-penyakit

Proses ini mempunyai keluaran:

- a. data-jenis-penyakit
- 7. Proses Input Data Pengobatan

Proses Input data pengobatan dilakukan untuk menambahkan data peengobatan dan disimpan dalam *data store* pengobatan.

Proses ini mempunyai masukan:

a. data-pengobatan

Proses ini mempunyai keluaran:

a. data-pengobatan

Proses Validasi Data Pengobatan

Proses ini mempunyai masukan:

a. data-pengobatan

Proses ini mempunyai keluaran:

- a. data-pengobatan
- 9. Simpan Data Pengobatan

Proses ini mempunyai masukan:

a. data-pengobatan

Proses ini mempunyai keluaran:

- a. data-pengobatan
- 10. Proses Input Data Gejala

Proses input data gejala dilakukan untuk menambahkan data gejala dan disimpan

BRAM

dalam data store gejala.

Proses ini mempunyai masukan:

a. data-gejala

Proses ini mempunyai keluaran:

- a. data-gejala
- 11. Proses Validasi Data Gejala

Proses ini mempunyai masukan:

a. data-gejala

Proses ini mempunyai keluaran:

- a. data-gejala
- 12. Simpan Data Gejala

Proses ini mempunyai masukan:

a. data-gejala

Proses ini mempunyai keluaran:

- a. data-gejala
- 13. Proses Input Data Dtl Gjl Penyakit

Proses input data dtl gjl penyakit dilakukan untuk menambahkan data detail gejala penyakit dan disimpan dalam data store dtl gjl penyakit.

Proses ini mempunyai masukan:

a. data-dtl-gjl-penyakit

Proses ini mempunyai keluaran:

a. data-dtl-gjl-penyakit







14. Proses Validasi Data Dtl Gjl Penyakit

Proses ini mempunyai masukan:

a. data-dtl-gjl-penyakit

Proses ini mempunyai keluaran:

- a. data-dtl-gjl-penyakit
- 15. Simpan Data Dtl Gjl Penyakit

Proses ini mempunyai masukan:

a. data-dtl-gjl-penyakit

Proses ini mempunyai keluaran:

a. data-dtl-gjl-penyakit

16. Proses Input Data Dtl Obat penyakit

Proses input data dtl obat penyakit dilakukan untuk menambahkan data detail obat penyakit dan disimpan dalam *data store* dtl obat penyakit.

Proses ini mempunyai masukan:

a. data-dtl-obat-penyakit

Proses ini mempunyai keluaran:

a. data-dtl-obat-penyakit

17. Proses Validasi Data Dtl Obat Penyakit

Proses ini mempunyai masukan:

a. data-dtl-obat-penyakit

Proses ini mempunyai keluaran:

a. data-dtl-obat-penyakit

18. Simpan Data Dtl Obat Penyakit

Proses ini mempunyai masukan:

a. data-dtl-obat-penyakit

Proses ini mempunyai keluaran:

a. data-dtl-obat-penyakit

19. Proses Input Data Solusi

Proses input data solusi dilakukan untuk menambahkan data solusi dan disimpan dalam data store solusi.

Proses ini mempunyai masukan:

a. data-solusi

Proses ini mempunyai keluaran:

a. data-solusi

20. Proses Validasi Data Solusi

Proses ini mempunyai masukan:

a. data-solusi

Proses ini mempunyai keluaran:

a. data-solusi

### 21. Simpan Data Solusi

Proses ini mempunyai masukan:

a. data-solusi

Proses ini mempunyai keluaran:

a. data-solusi

### SBRAWA

### 4.2 Perancangan

Perancangan sistem pakar dibuat untuk memberikan gambaran sistem pakar yang akan dibuat. Perancangan sistem pakar ini dibagi menjadi tiga yaitu Perancangan Rules, Inference tree, dan tabel keputusan.

### 4.2.1 Perancangan Rules

Sistem pakar ini menggunakan mesin inferensi runut maju (farward chaining) yaitu dengan penalaran dimulai dengan fakta merunut maju ke jalur yang mengarahkan ke tujuan.

Dalam sistem pakar ini terdapat 8 aturan. Aturan-aturan tersebut didapatkan dari berbagai sumber diantaranya Dokter, Penyakit Kulit yang sering diderita dan Tindakan Darurat, Macam-macam Penyakit Kulit dan Pencegahannya. Aturan-aturan tersebut antara lain dapat dilihat dalam Tabel 4.1

Tabel 4.1 Aturan yang terdapat dalam Sistem Pakar

RUL	E	IF	THEN
	1	A1, A2, A3, A4, A5, A6, A7, A8, A9, A10, A11, A17	P1
	2	A2, A4, A5, A12, A13, A14, A15, A16, A17	P2
	3	A17, A18, A19, A29, A31, A37, A39	P3
	4	A21, A22, A24, A30	P4
	5	A2, A4, A5, A12, A13, A14, A17, A20, A25	P5
	6	A17, A23, A26, A27, A28	P6
	7	A17, A23, A27, A31, A32, A36, A37	P7
	8	A33, A34, A35, A38, A40	P8

Sumber: Dirangkum dari Berbagai Sumber

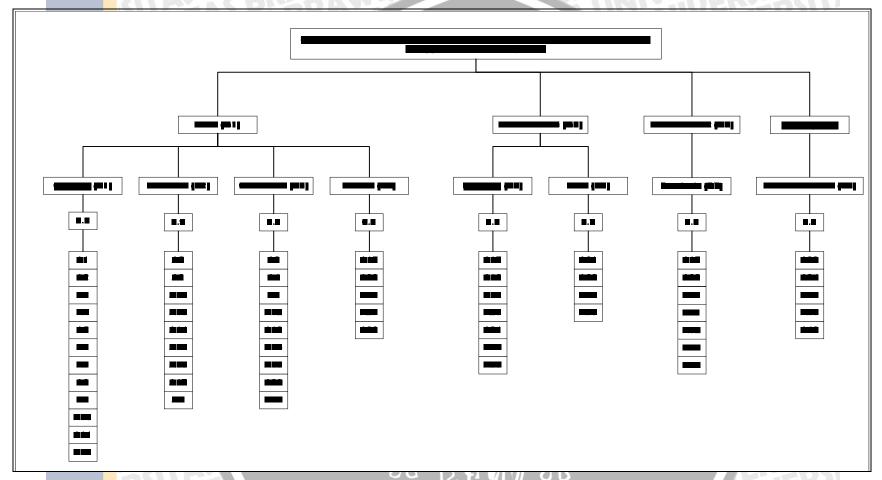
Tabel 4.2 kode gejala yang terdapat dalam Sistem Pakar

A1	Nama_Gejala
	Nyeri Tengorokan
A2	Hidung Meler
A3	Batuk
A4	Nyeri Otot
A5	Demam
A6	Mata Merah
A7	Fotofobia
A8	Timbul Ruam di Muka Seperti Jerawat
A9	Pada Langit-langit Mulut Timbul Bintik Putih
A10	Mata Berair
A11	Bernafas Dengan Cepat
A12	Cepat Merasa Lelah
A13	Nyeri Sendi
A14	Sakit Kepala
A15	Kemerahan Pada Kulit Dada/Perut
A16	Bintik Berisi Cairan Jernih Pada Seluruh Tubuh
A17	Timbul Bintik
A18	Bintik Sekitar Hidung dan Mulut Cepat Lepuh
A19	Bintik Pada Muka/tangan dan Kepala
A20	Muncul Benjolan Kecil yang Melingkar Separuh Bagian di Tubuh
A21	Timbul Bengkak Berwarna Merah
A22	Terasa Sakit di Bawah Kulit Bengkak
A23	Bintik Semakin Membesar
A24	Ujung Bengkak Melunak Dan Bernanah
A25	Pernah Menderita Cacar Sebelumnya
A26	Jumlah Satu dan Bisa Bertambah
A27	Bintik Terasa Kasar
A28	Tidak Nyeri
A29	Bintik Semakin Meluas
A30	Bengkak Semakin Membesar
A31	Bintik Berkeruping
A32	Bintik Bersisik Tebal Berlapis-lapis
A33	Timbul Bercak Merah
A34	Timbul Dibagian Tubuh Tertentu
A35	Bercak Merah ditutupi Sisik Halus
A36	Bintik Terasa Tebal
A37	Bintik Berwarna Merah
A38	Bercak Merah Semakin Lebar / Meluas dan Semakin Tebal
A39	Bintik Melepuh dan Bernanah
A40	Bercak Terasa Gatal

Sumber: Dirangkum dari Berbagai Sumber

### **4.2.2** Inference Tree

Inference Tree digunakan untuk mempermudah menggambarkan aturan yang ada dalam sistem. Inference tree pada sistem pakar ini digambarkan dalam Gambar 4.8.



Gambar 4.8 Diagram Tree Gangguan dan Penyakit Kulit

Sumber : Perancangan

AXA

# BRAWIJAY.

# 4.2.3 Tabel Keputusan

Tabel Keputusan digunakan untuk mengolah data pada *knowledge base*. Data yang tersimpan akan digunakan sebagai pengambil keputusan, dengan cara memberi nilai pada setiap gejala dan penyakit yang ada. Tabel keputusan pada sistem pakar ini digambarkan dalam Tabel 4.3.

Tabel 4.3 Tabel Keputusan Dari Knowledge Base

9	Gejal / Penyakit	P1	P2	P3	P4	P5	P6	P7	P8	Jml
	A1	Χ								1
N	A2	Χ	Χ			Χ				3
	A3	Χ								1
	A4	Χ	Χ			Χ				3
4	A5	X	Χ	A	7	X		1		3
	A6	Χ								1
Ī	A7	Χ							M	1
	A8	Χ								1
	A9	Χ								1
	A10	Χ								1
	A11	Χ	X	26		2	66			1
Þ	A12		Χ			Χ				2
1	A13	7	X	10		Χ	46			2
	A14		Χ			Χ				2
ı	A15		X		))		1	R.	1	1
	A16		Χ							1
Ī	A17	X	X	Χ	M	Х	Х	Χ	分	6
	A18									1
Ī	A19		3,	X		ř	P	$\sim$	$\lambda J$	1
	A20					Χ				1
	A21	Us.	1 /	_	Χ	4()	リド	DI.		1
	A22				Χ					1
Ī	A23		5.				Χ	Х		2
	A24				Χ					1
ı	A25	I F		114	-	Χ	111	317		1
	A26						Χ			1
Ī	A27	ᅫ	4	II	#1	// //	X	X		2
	A28						Χ			1
Ī	A29			X	J					1
V	A30				Х					1
	A31			Χ				Χ		2
	A32							Х		1
1	A33								Х	1
	A34								Х	1
1	A35								Χ	1
ζ,	A36							Х		1
1	A37	J	1	Х	311	VE		X		2
1	A38								Х	1
	A39	A		Х	71	1		ZII	V.	1
1	A40								Х	1
d	Jmi	12	9	7	4	9	5	7	5	

Sumber : Dirangkum dari Berbagai Sumber

#### 4.3 Perancangan Perangkat Lunak

Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian dibuat dengan menggunakan bahasa pemrograman Borland Delphi 7 dan menggunakan basis data Ms Access 2003. Sistem Operasi yang digunakan dalam perancangan aplikasi ini adalah Microsoft Windows XP.

Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian dirancang agar mampu:

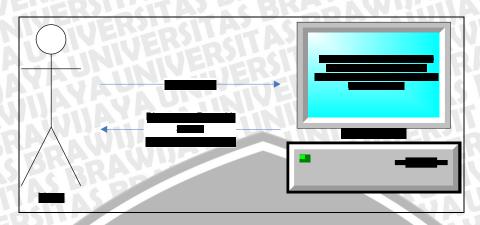
- 1. Melakukan konsultasi dengan *user*
- 2. Memberikan informasi gangguan dan penyakit kulit dari hasil konsultasi dengan user disertai faktor kepastian
- 3. Memberikan informasi cara mengatasi gangguan dan penyakit kulit tersebut.

# 4.3.1 Perancangan Sistem

Perancangan sistem merupakan tahap awal dari perancangan perangkat lunak. Perancangan ini dilakukan untuk mengetahui kondisi sistem secara umum. Perancangan sistem ini meliputi diagram blok sistem.

# 4.3.1.1 Diagram Blok Sistem

Diagram blok sistem menggambarkan setiap blok atau bagian dari dari sistem aplikasi. Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian dirancang untuk dapat memberikan informasi kepada pengguna, gangguan dan Penyakit Kulit yang diderita disertai dengan faktor kepastiannya. Pengguna memberi masukkan berupa gejala-gejala penyakit kulit. Dari gejala-gejala tersebut sistem pakar akan memberikan informasi mengenai gangguan dan penyakit kulit, faktor kepastian serta solusi untuk mengatasi gangguan dan penyakit kulit tersebut. Sistem pakar ini dapat dapat digambarkan dalam diagram blok seperti yang diperlihatkan dalam Gambar 4.9.



Gambar 4.9 Diagram Blok Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian

Sumber : Perancangan Sumber : Perancangan

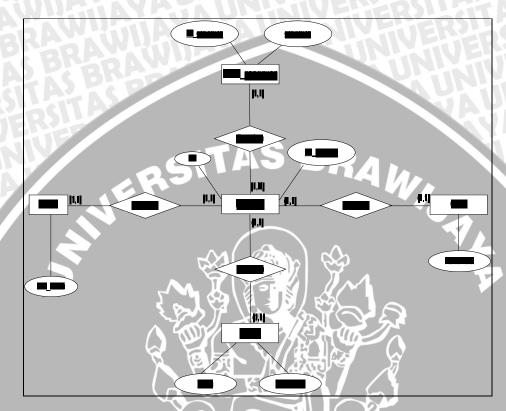
# 4.3.2 Perancangan Basis Data

Perancangan basis data dilakukan agar Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia memiliki basis data yang efisien dalam penggunaan ruang penyimpanan, mudah mengubah data yang sudah ada.



# 4.3.2.1 Hubungan antar Entitas Pembentuk Sistem dalam Diagram sistem-ER

Diagram-ER digunakan untuk memodelkan struktur data dan hubungan antar entitas. Untuk menjelaskan hubungan antar entitas satu dengan entitas lain dalam sistem pakar ini, akan ditunjukkan dalam diagram-ER dalam Gambar 4.10 dibawah ini.

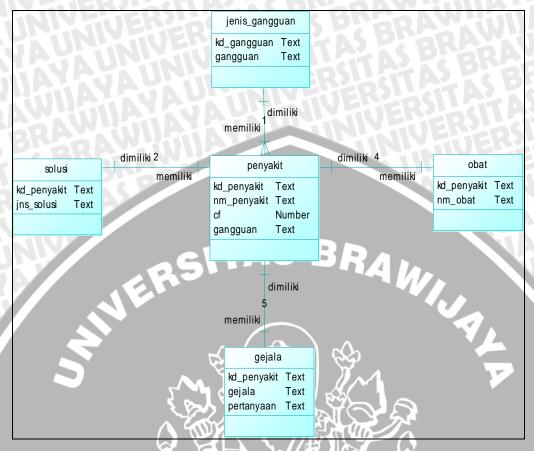


Gambar 4.10 ERD Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian

Sumber: Perancangan

## 4.3.2.2 Model Relasi antar Tabel

Model relasi antar table yang terbentuk beserta fieldnya ditunjukan dalam gambar 4.11.



Gambar 4.11 Model Relasi Antar Tabel

Entitas jenis\_gangguan, data\_penyakit, dtl\_gjl\_penyakit, gejala, dtl\_obat\_penyakit, pengobatan, solusi, masing-masing memiliki struktur tabel basis data. Rancangan struktur tabel basis data yang digunakan pada tujuh tabel tersebut diperlihatkan pada Tabel 4.4 dan seterusnya.

Tabel 4.4 Definisi Rancangan Struktur Tabel data\_penyakit

Field	Tipe Data	Lebar Field	Indeks
kd_penyakit	Text	3	Primary key
nm_ penyakit	Text	50	A
cf	Number	Double	AN

Pada Tabel data\_penyakit digunakan untuk menyimpan data penyakit. *Field* kd\_penyakit digunakan untuk menyimpan *index* suatu penyakit dan merupakan *primary key* dari Tabel data\_penyakit.

Tabel 4.5 Definisi Rancangan Struktur Tabel dtl\_gejala\_penyakit

Field	Tipe Data	Lebar Field	Indeks
kd_penyakit	Text	3	Primary key
kd_gejala	Text	3 = = = 5	Primary key

Tabel dtl\_gejala\_penyakit terdiri atas dua *field*, yaitu *field* kd\_penyakit dan kd\_gejala. *Field* kd\_penyakit digunakan untuk menyimpan *index* suatu data penyakit dan merupakan *primary key* yang diambil dari Tabel data\_jenis\_penyakit. *Field* kd\_gejala digunakan untuk menyimpan *index* suatu data gejala merupakan *primary key* yang diambil dari Tabel gejala. Kedua *field* tersebut selain merupakan *primary key* juga berpredikat sebagai *foreign key*, karena data yang tersimpan di dalamnya bergantung dari data tersimpan di dalam Tabel data\_penyakit dan gejala.

Tabel 4.6 Definisi Rancangan Struktur Tabel Gejala

Field	Tipe Data	Lebar Field	Indeks
kd_gejala	Text	3	Primary key
gejala	Text Text	50	$\tilde{\Lambda}$

Tabel gejala terdiri atas dua *field*, yaitu *field* kd\_gejala dan gejala. *Field* kd\_gejala digunakan untuk menyimpan *index* suatu gejala dan merupakan *primary key* dari Tabel gejala. Tabel gejala digunakan untuk menyimpan data gejala.

Tabel 4.7 Definisi Rancangan Struktur Tabel jenis\_gangguan

Field	Tipe Data	Lebar Field	Indeks
kd_jns_gangguan	Text	3140 20	Primary key
Jns_gangguan	Memo		

Tabel jenis\_gangguan terdiri atas dua *field*, yaitu *field* kd\_jns\_gangguan dan jns\_gangguan. *Field* kd\_jns\_gangguan digunakan untuk menyimpan *index* suatu jenis gangguan dan merupakan *primary key* dari Tabel jenis\_gangguan. Tabel jenis\_gangguan digunakan untuk menyimpan data jenis gangguan.

**Tabel 4.8** Definisi Rancangan Struktur Tabel obat

	VALLETT VALLE		
Field	Tipe Data	Lebar Field	Indeks

$\triangleleft$
$\triangleleft$
<

kd_obat	Text	3 15 2 3	Primary key
nm_obat	Text	50	SBRARA
MAYAGA		HERSLY	TADRER

Tabel obat mempunyai dua *field*, yaitu *field* kd\_obat dan nm\_obat. Field kd\_obat merupakan *primary key* dari tabel obat. Tabel ini digunakan untuk menyimpan data obat.

Tabel 4.9 Definisi Rancangan Struktur Tabel dtl\_obat\_penyakit

	Field	Tipe Data	Lebar Field	Indeks
	kd_obat	Text	3	Primary key
	kd_penyakit	Text	3	Primary key
4			$\alpha$	

Tabel dtl\_obat\_penyakit terdiri atas dua field, yaitu field kd\_obat dan kd\_penyakit . Field kd\_obat digunakan untuk menyimpan index suatu data obat merupakan primary key yang diambil dari Tabel obat. Field kd\_penyakit digunakan untuk menyimpan index suatu data penyakit dan merupakan primary key yang diambil dari Tabel data\_penyakit. Kedua field tersebut selain merupakan primary key juga berpredikat sebagai foreign key, karena data yang tersimpan di dalamnya bergantung dari data tersimpan di dalam Tabel data\_penyakit dan data\_obat.

Tabel 4.10 Definisi Rancangan Struktur Tabel Solusi

Field	Tipe Data	Lebar Field	Indeks
kd_penyakit	Text	3	Primary key
Solusi	Memo		

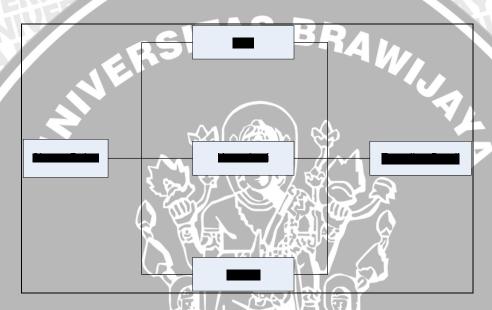
Tabel solusi terdiri atas dua *field*, yaitu *field* kd\_penyakit dan solusi. *Field* kd\_penyakit digunakan untuk menyimpan *index* suatu data penyakit dan merupakan *primary key* yang diambil dari Tabel data\_penyakit. *Field* solusi berisi solusi bagaimana mengatasi suatu gangguan dan penyakit kulit.

## 4.3.3 Site Map Sistem

Site map digunakan untuk memberikan penjelasan tentang menu-menu yang terdapat pada sistem yang dibuat.

Gambaran *site map* sistem yang akan dibuat terdapat dalam Gambar 4.12 dan Gambar 4.13. Pertama kali *user* dan pakar akan masuk ke halaman yang berisi pilihan login. User yang ingin melakukan konsultasi hanya akan mengakses menu Konsultasi, *Tips dan Loqout*. Menu *File* berisi *data master*, yang hanya dapat berfungsi jika pakar telah melakukan *login*, dengan mengisi nama pakar dan pasword. Menu Tips untuk memberi informasi jika memerlukan bantuan.

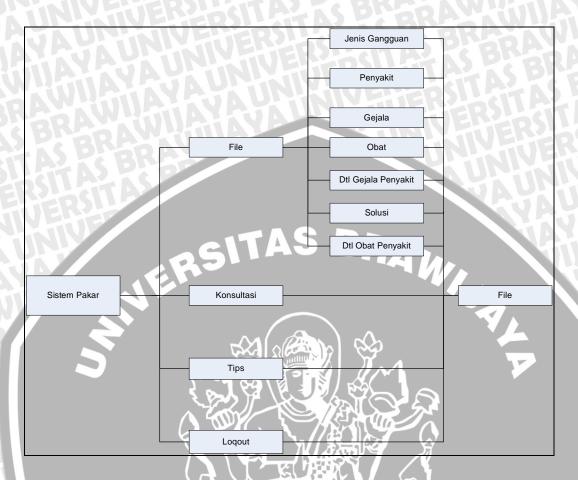
Sitemap Sistem pakar ini dibedakan menjadai dua yaitu sitemap pengguna bukan pakar dan sitemap pakar.



**Gambar 4.12** Site map Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian untuk User

#### Sumber: Perancangan

User bukan pakar dapat langsung mengakses sistem tanpa harus mengisi nama user dan password terlebih dahulu. User dapat memilih beberapa menu diantaranya konsultasi dan Tips. Menu konsultasi digunakan untuk melakukan konsultasi sedangkan menu Tips berisi tata cara menggunakan program sistem pakar ini. User pakar dapat mengakses sistem secara keseluruhan dengan mengisi data login terlebi dahulu. Pakar dapat mengakses menu yang tidak dapat diakukan oleh user bukan pakar. Site map pakar dapat dilihat dalam gambar 4.13.



**Gambar 4.13** Site map Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia dengan Menggunakan Faktor Kepastian untuk Pakar

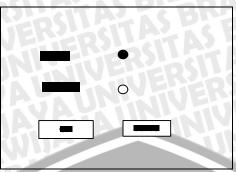
Sumber: Perancangan

# 4.3.4 Perancangan User Interface

Interface merupakan bagian yang menghubungkan antara sistem dengan user. Dengan demikian suatu perancangan interface yang baik akan memudahkan user untuk berinteraksi dengan sistem yang dibuat.

# 4.3.4.1 Tampilan Awal Sistem

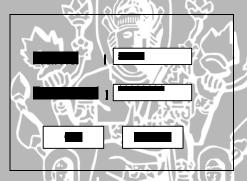
Tampilan ini merupakan tampilan awal ketika sistem diakses. User dan pakar harus memilih salah satu untuk bisa mengakses sistem. Rancangan dari menu tersebut dapat dilihat dalam Gambar 4.14.



Gambar 4.14 Rancangan menu login

Sumber: Perancangan

User pakar harus mengisi data login pakar terlebih dahulu sebelum masuk kedalam sistem, yaitu naman pakar dan pasword. User pakar memiliki hak akses secara penuh untuk mengolah data. Rancangan dari menu tersebut dapat dilihat dalam gambar 4.15



Gambar 4.15 Rancangan menu login pakar

Sumber : Perancangan

# 4.3.4.2 Tampilan Menu File

Menu file terbagi menjadi beberapa menu yaitu menu : data jenis gangguan, data Penyakit, Gejala, Detail Gejala Penyakit, Obat, solusi, Detail Obat Penyakit, User Akses. Rancangan Tampilan Menu File dapat dilihat dalam Gambar 4.16

File Kon	sultasi Tips Loqout
Jenis Gangguan Penyakit Gejala Obat Solusi Gejala Penyakit Obat Penyakit User Akses Exit	SISTEM PAKAR UNTUK MENDIAGNOSIS GANGGUAN DAN PENYAKIT KULIT PADA MANUSIA MENGGUNAKAN FAKTOR KEPASTIAN
Sir	STAS BD

Gambar 4.16 Rancangan Menu File

Sumber : Perancangan

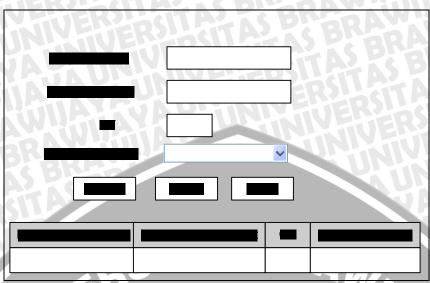
Menu jenis gangguan digunakan untuk memasukkan, mengubah atau menghapus data jenis gangguan yang akan ditampilkan dalam sistem pakar ini. Data tersebut kemudian akan tersimpan dalam *data store* jenis gangguan. Rancangan Menu Pertanyaan Penyakit dapat dilihat dalam Gambar 4.17.



Gambar 4.17 Rancangan Menu Pertanyaan Penyakit

Sumber: perancangan

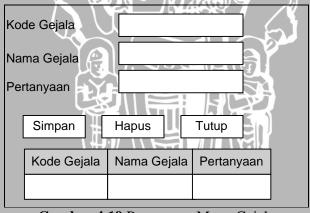
Menu data penyakit digunakan untuk memasukkan, mengubah atau menghapus data penyakit. Data tersebut kemudian akan tersimpan dalam *data store* data penyakit. Rancangan Menu data Penyakit dapat dilihat dalam Gambar 4.18.



Gambar 4.18 Rancangan Menu data Gangguan

Sumber : Perancangan

Menu gejala digunakan untuk memasukkan, mengubah atau menghapus data gejala dan pertanyaan yang akan ditanyakan kepada user. Data tersebut kemudian akan tersimpan dalam *data store* gejala. Rancangan Menu Gejala dapat dilihat dalam Gambar 4.19



Gambar 4.19 Rancangan Menu Gejala

Sumber: Perancangan

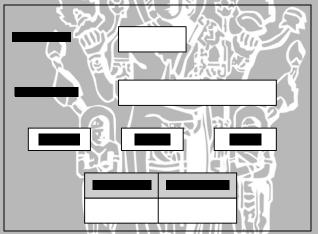
Menu detail gejala penyakit digunakan untuk memasukkan, mengubah atau menghapus data detail gejala penyakit. Data tersebut kemudian akan tersimpan dalam data store detail gejala penyakit. Rancangan Menu Detail Gejala Penyakit dapat dilihat dalam Gambar 4.20

			•	1
				1
				1
			•	I
	•	1	1	

Gambar 4.20 Rancangan Menu Data detail Gejala Gangguan

Sumber: perancangan

Menu obat digunakan untuk memasukkan, mengubah atau menghapus data obat. Data tersebut kemudian akan tersimpan dalam *data store* pengobatan. Rancangan Menu obat dapat dilihat dalam Gambar 4.21



Gambar 4.21 Rancangan Menu Obat

Sumber : Perancangan

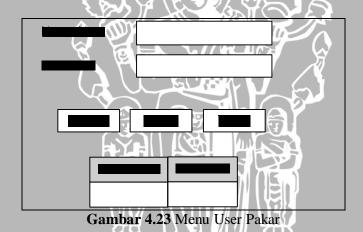
Menu detail obat penyakit digunakan untuk memasukkan, mengubah atau menghapus data detail obat penyakit. Data tersebut kemudian akan tersimpan dalam data store detail obat penyakit. Rancangan Menu detail obat penyakit dapat dilihat dalam Gambar 4.22

$\Lambda \sim \Lambda$	na Penyakit e Obat	<u> </u>	<u>▼</u> )	S
Nam	na Obat [			
	Simpan	lapus	Tutup	
S	lama Penyakit	Kode Obat	Nama Obat	

Gambar 4.22 Rancangan Menu Detail Obat Penyakit

Sumber: Perancangan

Menu user pakar digunakan untuk memasukkan, mengubah atau menghapus data user pakar yang boleh memberi data-data masukan untuk sistem. Rancangan Menu User Pakar dapat dilihat dalam Gambar 4.23



Sumber: Perancangan

# 4.3.4.3 Tampilan Menu Proses

Menu Konsultasi dilgunakan untuk melakukan konsultasi dengan sistem pakar secara langsung. Rancangan menu tersebut dapat dilihat dalam Gambar 4.24.

	_			
·				
•				

Gambar 4.24 Rancangan Menu Konsultasi

Sumber: Perancangan

# 4.3.4.4 Tampilan Hasil Konsultasi

Hasil konsultasi ini akan menghasilkan tampilan berupa hasil diagnosa sistem pakar. Untuk lebih jelasnya form ini dapat dilihat dalam Gambar 4.25.

	~	
	~	
	1 0,000 M	
 1		,

Gambar 4.25 Rancangan Tampilan Hasil Diagnosa

Sumber: Perancangan

#### BAB V

#### Article II. **IMPLEMENTASI SISTEM**

Implementasi merupakan proses transformasi hasil perancangan perangkat lunak ke dalam kode (coding) sesuai dengan sintaks dari bahasa pemrograman yang digunakan. Implementasi sistem akan diawali dengan pembuatan algoritma dan mengimplementasikan algoritma menjadi kode pemrograman sesungguhnya.

#### 5.1 **Tampilan Awal**

Tampilan ini merupakan tampilan awal ketika sistem diakses. User dan pakar harus memilih salah satu untuk bisa mengakses sistem. Rancangan dari menu tersebut dapat dilihat dalam Gambar 5.1.



Gambar 5.1 Tampilan Awal Sistem Sumber: Implementasi

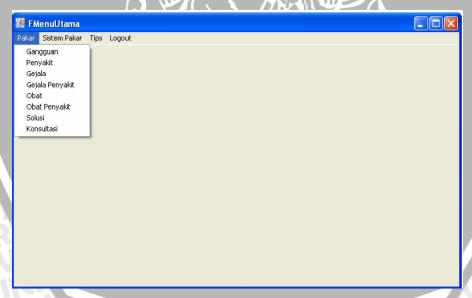
User pakar harus mengisi data login pakar terlebih dahulu sebelum masuk kedalam sistem, yaitu nama pakar dan pasword. User pakar memiliki hak akses secara penuh untuk mengolah data. Rancangan dari menu tersebut dapat dilihat dalam gambar 5.2.



Gambar 5.2 Tampilan Login Pakar

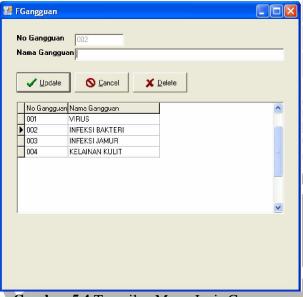
# 5.2 Menu File

Menu file terbagi menjadi beberapa menu yaitu menu : data jenis gangguan, data Penyakit, Gejala, Detail Gejala Penyakit, Obat, solusi, Detail Obat Penyakit, User Akses. Rancangan Tampilan Menu File dapat dilihat dalam Gambar 5.3.



Gambar 5.3 Tampilan Menu File

Sumber : Implementasi



Gambar 5.4 Tampilan Menu Jenis Gangguan

Menu jenis gangguan digunakan untuk memasukkan, mengubah atau menghapus data jenis gangguan yang akan ditampilkan dalam sistem pakar ini. Data tersebut kemudian akan tersimpan dalam *data store* jenis gangguan. Algoritma proses Simpan dan hapus dapat dilihat dalam gambar dibawah ini.

Gambar 5.5 Algoritma Proses Simpan Jenis gangguan

Sumber: Implementasi

```
procedure TFGangguan.CmdDeleteClick(Sender: TObject);
    Kode_Penyakit:string;
    with Gangguan do
    begin
         FMenuutama.ADOConn.BeginTrans;
        Sqlcmd:='belete * From T_Gangguan Where Kode_Gangguan='+ Quotedstr(Kode_Gangguan) +'';
Ffenuutama.ADOConn Execute(sqlcmd);
        sqlcmd:='Select Top 1 * From T_Penyakit Where Kode_Gangguan= '+ Quotedstr(Kode_Gangguan) +' orde
Search.Close;
         search.SOL.Clear:
         search.SQL.Add(sqlcmd);
         search.Open;
         While not search.IsEmpty do
         begin
             Kode Penyakit: =Search.Fieldbyname('Kode Penyakit').AsString;
             Sqlcmd:='Delete * From T Gejala Penyakit Where Kode Penyakit='+ Quotedstr(Kode Penyakit) + FMenuutama.ADOConn.Execute(sqlcmd);
             sqlcmd:='Select Top 1 * From T_Penyakit Where Kode_Penyakit > '+ Quotedstr(Kode_Penyakit) - Search.Close;
             search.SQL.Clear;
             search.SQL.Add(sqlcmd);
             search.Open;
         end:
         Sqlcmd:='Delete * From T_Penyakit Where Kode_Gangguan='+ Quotedstr(Kode_Gangguan) +'';
         FMenuutama.ADOConn.Execute(sqlcmd);
         FMenuutama. ADOConn. CommitTrans;
         setting Awal;
```

Gambar 5.6 Algoritma Proses Hapus Jenis gangguan

Kode Penyakit 001			Gambar Kerusal	kan	
Na	ama Penyakit	CAMPAK			
CF	Penyakit	0.8			
Na	ma File Gb			1 1	
M =	ama Gangguan	Labrio.			
Nid	illa Ganyyuan	VIRUS	_		
, <u></u>					
				w 1	
	✓ Update	O Cano	el 🗶	<u>D</u> elete	
		<u></u>	~· /	<u>=</u> 0.0.0	
_		G gand		25555	
	Mama Gangguan				
F	Nama Gangguan		No Penyakit	Nama Penyakit	^
	VIRUS		No Penyakit	Nama Penyakit CAMPAK	
•	VIRUS VIRUS		No Penyakit 001 003	Nama Penyakit CAMPAK CACAR AIR	^
<u> </u>	VIRUS		No Penyakit	Nama Penyakit CAMPAK	
•	VIRUS VIRUS VIRUS		No Penyakit 001 003 005	Nama Peryakit CAMPAK CACAR AIR CACAR ULAR	
	VIRUS VIRUS VIRUS VIRUS	RI	No Penyakit 001 003 005 006	Nama Penyakit CAMPAK CACAR AIR CACAR ULAR VERUKA	
	VIRUS VIRUS VIRUS VIRUS INFEKSI BAKTEI	RI RI	No Penyakit 001 003 005 006	Nama Penyakit CAMPAK CACAR AIR CACAR ULAR VERUKA BISUL	
	VIRUS VIRUS VIRUS VIRUS INFEKSI BAKTEI INFEKSI BAKTEI	RI RI	No Penyakit 001 003 005 006 002 004	Nama Penyakit CAMPAK CACAR AIR CACAR ULAR VERUKA BISUL IMPETIGO	

**Gambar 5.7** Tampilan Menu Penyakit

Sumber : Implementasi

Menu data penyakit digunakan untuk memasukkan, mengubah atau menghapus data penyakit. Data tersebut kemudian akan tersimpan dalam *data store* data penyakit. Algoritma proses Simpan dan hapus dapat dilihat dalam gambar dibawah ini.

Gambar 5.8 Algoritma Proses Simpan Penykit

```
procedure TFPenyakit.CmdDeleteClick(Sender: TObject);
begin

| FMenuutama.ADOConn.BeginTrans;
Sqlcmd:='DELETE * FROM Penyakit UHERE Kode_Penyakit = '+ QuotedStr(Penyakit.Kode_Penyakit) +' ';
FMenuutama.ADOConn.Execute(sqlcmd);
Sqlcmd:='DELETE * FROM GEJALAPenyakit UHERE Kode_Penyakit = '+ QuotedStr(Penyakit.Kode_Penyakit) +'
FMenuutama.ADOConn.Execute(sqlcmd);
FMenuutama.ADOConn.CommitTrans;
Setting_Awal;
end;
```

Gambar 5.9 Algoritma Proses Hapus Penyakit

Sumber: Implementasi



Gambar 5.10 Tampilan Menu Gejala

Sumber : Implementasi

Menu gejala digunakan untuk memasukkan, mengubah atau menghapus data gejala dan pertanyaan yang akan ditanyakan kepada user. Data tersebut kemudian akan tersimpan dalam data store gejala. Algoritma proses Simpan dan hapus dapat dilihat dalam gambar dibawah ini.

```
procedure TFGejala.CmdSaveClick(Sender: TObject);
begin
   With Gejala do
   begin
       Kode_Gejala:=txtKode_Gejala.Text;
       Nama Gejala:=TxtNama Gejala.Text;
       Pertanyaan:=TxtPertanyaan.Text;
       if CmdSave.Caption='&Save' Then
           Sqlcmd:='INSERT INTO T_GEJALA VALUES('+ QuotedStr(Kode_Gejala) +', '+
                   ''+ QuotedStr(Nama Gejala) +','+ QuotedStr(Pertanyaan) +',0,0)'
           Sqlcmd:='UPDATE T GEJALA SET Nama Gejala = '+ QuotedStr(Nama Gejala) +', ' +
                   'PERTANYAAN = '+ QuotedStr(Pertanyaan) +' WHERE Kode_Gejala = '+ QuotedStr(Kode_Geja
        FMenuutama.ADOConn.BeginTrans;
       FMenuutama. ADOConn. Execute (solomd):
        Fmenuutama.ADOConn.CommitTrans;
    end:
    setting_lval;
end:
```

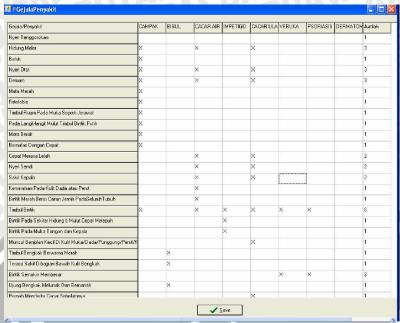
Gambar 5.11 Algoritma Proses Simpan Gejala

Sumber: Implementasi

```
procedure TFGejala.CmdDeleteClick(Sender: TObject);
begin
   FMenuutama.ADOConn.BeginTrans;
   Sqlcmd:='DELETE * FROM T GEJALA WHERE Kode Gejala = '+ QuotedStr(gejala.Kode Gejala) +' ';
   FMenuutama.lDOConn.Execute(sqlcmd);
   Sqlcmd:='DELETE * FROM T_GEJALA_PENYAKIT WHERE Kode_Gejala = '+ QuotedStr(gejala.Kode_Gejala) +'
   FMenuutama.ADOConn.Execute(sqlcmd);
   FMenuutama.ADOConn.CommitTrans;
   Setting Awal;
procedure TFGejala.CmdCancelClick(Sender: TObject);
   Setting_lwal;
end;
end.
```

Gambar 5.12 Algoritma Proses Hapus Gejala

Sumber: Implementasi



Gambar 5.13 Tampilan Menu Detail Gejala Penyakit

Menu detail gejala penyakit digunakan untuk memasukkan, mengubah atau menghapus data detail gejala penyakit. Data tersebut kemudian akan tersimpan dalam data store detail gejala penyakit. Algoritma penyimpanan dan penghapusan data diperlihatkan dalam Gambar dibawah ini.

```
i,j:integer;
begin
    FMenuUtama.adoconn.begintrans;
    Sqlcmd:='Delete * FROM T Gejala Penyakit';
    FMenuUtama.ADOConn.Execute(sqlcmd);
    FMenuUtama.adoconn.CommitTrans;
    FMenuUtama.adoconn.begintrans;
    For j:=1 to Tabel_GPenyakit.RowCount-2 do
         For i:=1 To Tabel_GPenyakit.ColCount-2 do
         begin
               \  \  if \  \, Tabel\_GPenyakit.Cells[i,j] = {}^{i}X^{i} \  \, then \\
                   GejalaPenyakit.Kode Gejala:=Get KodeGejala(Tabel GPenyakit.Cells[0.jl):
                   GejalaPenyakit.Kode_Penyakit:=Get_KodePenyakit(Tabel_GPenyakit.Cells[i,0]);
                   GejalaPenyakit.Kode_Gejala_Penyakit:=GejalaPenyakit.Kode_Penyakit+GejalaPenyakit.Kode_Ge
sqlcmd:='INSERT INTO T_GEJALA_Penyakit VALUES ('+ quotedstr(GejalaPenyakit.Kode_Gejala_F
''+ quotedstr(GejalaPenyakit.Kode_Penyakit) +','+ Quotedstr(GejalaPenyakit.Kode_Gejala)
                   FMenuUtama.ADOConn.Execute(sqlcmd);
              end:
   For j:=1 to Tabel GPenyakit.RowCount-2 do
         GejalaPenyakit.Kode Gejala:=Get KodeGejala(Tabel GPenyakit.Cells[0,j]);
         SQLCMD:='Update T_Gejala set score='+ Tabel_GPenyakit.Cells[Tabel_GPenyakit.Colcount-1,j] +' Wh
         FMenuUtama.ADOConn.Execute(sqlcmd);
```

Gambar 5.14 Algoritma Proses Simpan Detail Gejala Penyakit

Sumber: Implementasi

Gambar 5.15 Tampilan Menu Obat

Menu obat digunakan untuk memasukkan, mengubah atau menghapus data obat. Data tersebut kemudian akan tersimpan dalam *data store* pengobatan. Algoritma proses Simpan dan hapus dapat dilihat dalam gambar dibawah ini.

```
procedure TFObat.CmdSaveClick(Sender: TObject);
begin
    With Obat do
    begin
        Kode Obat:=txtKode Obat.Text;
        Nama_Obat:=TxtNama_Obat.Text;
        if {\tt CmdSave.Caption='\&Save'} Then
            Sqlcmd:='INSERT INTO T_Obat VALUES('+ QuotedStr(Kode_Obat) +', '+
                    ''+ QuotedStr(Nama Obat) +')'
        else
            Sqlcmd:='UPDATE T_Obat SET Nama_Obat = '+ QuotedStr(Nama_Obat) +' '
                    'UHERE Kode_Obat = '+ QuotedStr(Kode_Obat) +'';
        FMenuutama.ADOConn.BeginTrans;
        FMenuutama.ADOConn.Execute(sqlcmd);
        Fmenuutama.ADOConn.CommitTrans;
    setting_Awal;
```

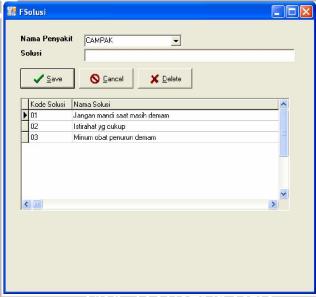
Gambar 5.16 Algoritma Proses Simpan Obat

Sumber : Implementasi

BRAWIJAYA

```
procedure TFObat.CmdDeleteClick(Sender: TObject);
begin
    FMenuutama.ADOConn.BeginTrans;
    Sqlcmd:='DELETE * FROM T_Obat WHERE Kode_Obat = '+ QuotedStr(Obat.Kode_Obat) +' ';
    FMenuutama.ADOConn.Execute(sqlcmd);
    Sqlcmd:='DELETE * FROM T_Obat_PENYAKIT UHERE Kode_Obat = '+ QuotedStr(Obat.Kode_Obat) +' ';
    FMenuutama.ADOConn.Execute(sqlcmd);
    FMenuutama.ADOConn.CommitTrans;
    Setting_Awal;
end;
end.
```

Gambar 5.17 Algoritma Proses Hapus Obat



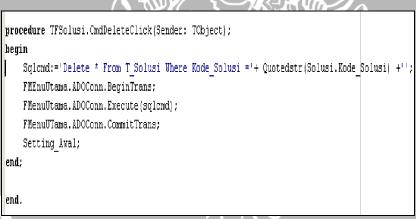
Gambar 5.18 Tampilan Menu Solusi

Sumber: Implementasi

Menu Solusi digunakan untuk menyimpan, mengubah, dan menghapus data solusi yang tersipan dalam *data store* solusi. Algoritma proses Simpan, hapus, dan ubah dapat dilihat dalam gambar dibawah ini.

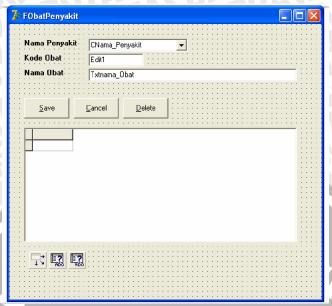
```
procedure TFSolusi.CmdSaveClick(Sender: TObject);
begin
    Solusi.Nama Solusi:=TxtSolusi.Text;
    if cmdsave.Caption='&Save' then
       Solusi.Kode_Penyakit:=Get_Kode_Penyakit(CNama_Penyakit.Text);
        Solusi.Kode_Solusi:=Get_Kode_Solusi(Solusi.Kode_Penyakit);
        Sqlcmd:='Insert Into T Solusi Values('+ Quotedstr(Solusi.Kode Solusi) +','+ Quotedstr(Solusi.Kod
             ''+ Quotedstr(Solusi.Nama_Solusi)+')';
        Sqlcmd:='Update T_Solusi set Nama_Solusi='+ Quotedstr(Solusi.Nama_Solusi) +' Where Kode_Solusi =
    FMEnuUtama.ADOConn.BeginTrans;
   FMenuUtama.ADOConn.Execute(sqlcmd);
    FMenuUTama.ADOConn.CommitTrans;
    Setting Awal;
```

Gambar 5.19 Algoritma Proses Simpan Solusi



Gambar 5.20 Algoritma Proses Hapus Solusi

Sumber: Implementasi



Gambar 5.21 Tampilan Menu Detail Obat Penyakit

Menu Detail Obat penyqkit digunakan untuk menyimpan, mengubah, dan menghapus data detail obat penyakit yang tersipan dalam data store dtl obat penyakit. Algoritma proses Simpan dan hapus dapat dilihat dalam gambar dibawah ini.

```
procedure TFObatPenyakit.CmdSaveClick(Sender: TObject);
   Obat Penyakit.Kode Penyakit:=Get Kode Penyakit(cNama Penyakit.Text);
   Obat_Penyakit.Kode_Obat:=TxtKode_Obat.Text;
   Obat_Penyakit.Kode_OP:=Obat_Penyakit.Kode_Penyakit + Obat_Penyakit.Kode_Obat;
   Sqlcmd:='Insert into T_Obat_Penyakit Values('+ Quotedstr(Obat_Penyakit.Kode_OP) +', ' +
       '+ Quotedstr(Obat Penyakit.Kode Penyakit) +','+ Quotedstr(Obat Penyakit.Kode Obat) +')';
   FMenuutama.ADOConn.BeginTrans;
   FMenuutama.ADOConn.Execute(sqlcmd);
   FMenuutama.ADOConn.CommitTrans;
   setting_Awal;
procedure TFObatPenyakit.CmdCancelClick(Sender: TObject);
   Setting_Awal;
end;
end.
```

Gambar 5.22 Algoritma Proses Simpan Detail Obat penyakit

Sumber: Implementasi

```
procedure TFCbatPenyakit.CmdDeleteClick(Sender: TCbject);

begin

Sqlcmd:='Delete * From T Cbat Penyakit Where Kode Penykit ='+ Quotedstr(Penyakit.Kode penyakit| +'';

FMEnuUtama.ADCConn.BeginTrans;

FMenuUtama.ADCConn.Execute(sqlcmd);

FMenuUtama.ADCConn.CommitTrans;

Setting_Awal;

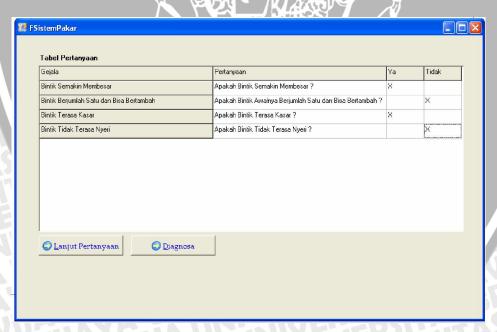
end;

end.
```

Gambar 5.23 Algoritma Proses Hapus Detail Obat Penyakit

## 5.3 Menu Proses

Menu Konsultasi dilgunakan untuk melakukan konsultasi dengan sistem pakar secara langsung. Pertanyaan yang diberikan oleh sistem harus dijawab oleh user, dengan cara memberi inputan *ya* atau *tidak* pada setiap pertanyaan.



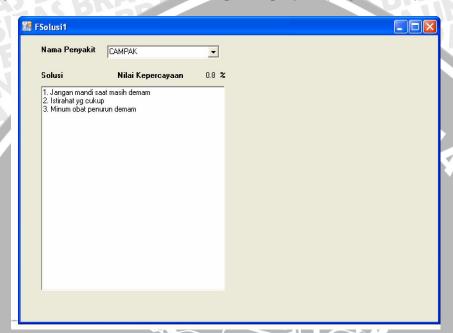
Gambar 5.24 Tampilan Form Konsultasi

Sumber : Implementasi

# BRAWIJAYA

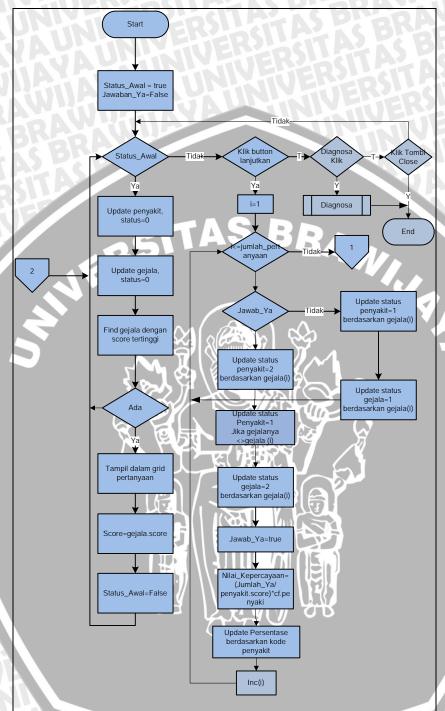
# 5.4 Tampilan Hasil Konsultasi

Hasil konsultasi ini akan ditampilkan setelah proses pencarian penyakit telah ditemukan oleh sistem. Dari hasil konsultasi user akan mengetahui nama penyakit yang diderita dan solusi yang dianjurkan oleh dokter kulit, user juga dapat mengetahui obat yang baik untuk dikonsumsi dan nilai kepastian penyakit yang dideritanya.

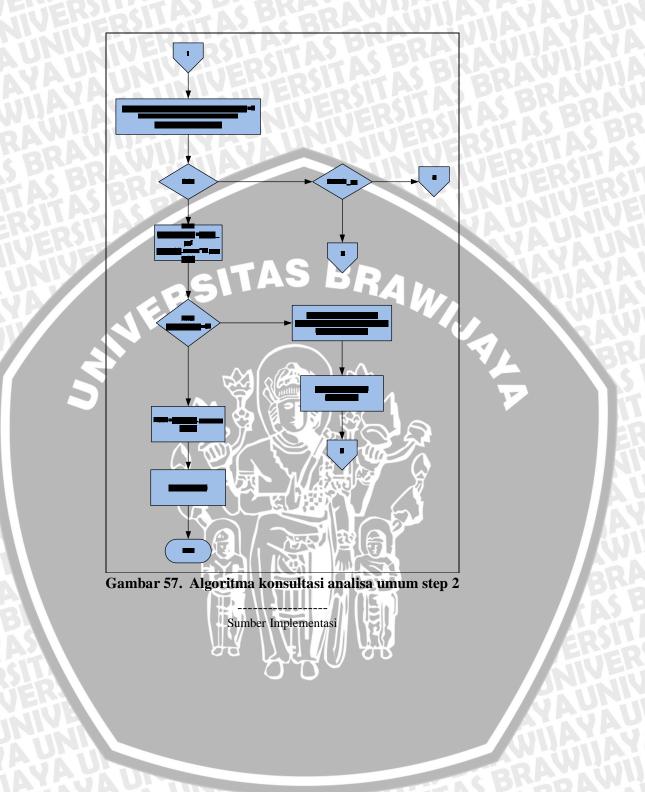


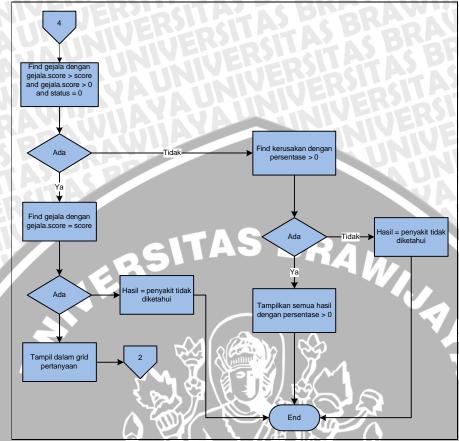
Gambar 5.25 Tampilan Hasil Konsultasi

Sumber : Implementasi

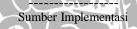


Gambar 56. Algoritma konsultasi analisa umum step 1

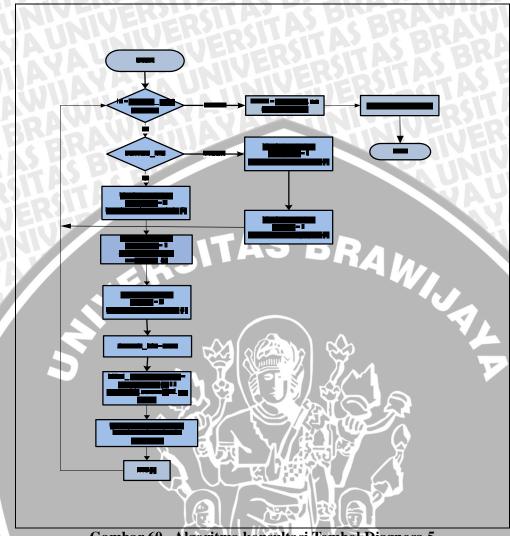




Gambar 59. Algoritma konsultasi analisa umum step 4



III



Gambar 60. Algoritma konsultasi Tombol Diagnosa 5

# BAB VI PENGUJIAN DAN ANALISIS

Bab ini membahas mengenai pengujian dan analisis terhadap implementasi Sistem Pakar Untuk Mendiagnosis Gangguan dan Penyakit Kulit Pada Manusia. Proses pengujian dilakukan melalui tiga tahapan (strategi) yaitu pengujian unit, pengujian integrasi dan pengujian validasi sedangkan pengujian basis data dilakukan secata terpisah. Proses analisis dilakukan dengan membandingkan sistem aplikasi yang dengan teori yang ada sehingga didapatkan suatu kesimpulan.

## 6.1 Pengujian

Proses pengujian dilakukan melalui tiga tahapan (strategi) yaitu pengujian unit, pengujian integrasi dan pengujian validasi.

# 6.1.1 Pengujian Unit

## 6.1.1.1 Pengujian

Pengujian unit dilakukan untuk menguji unsur terkecil dari sistem pakar.

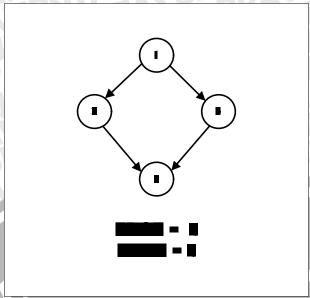
Dengan membuatan PDL (Pengujian Jalaur Dasar) untuk desain *test case* dengan simpul – simpul yang diidentifikasikan.

#### Pengujian Unit: Simpan Data jenis Gangguan

Gambar 6.1 PDL Proses Simpan Jenis Gangguan

Sumber : Pengujian

BRAWIJAYA



Gambar 6.2 Grafik Aliran Dari Proses Simpan Jenis Gangguan

Sumber: Pengujian

Dari gambar diatas diperoleh kompleksitas siklomatis sebagai berikut:

$$V(G) = E - N + 2$$

$$V(G) = 4 - 4 + 2$$

$$V(G) = 2$$

Dengan demikian diperoleh 2 buah basis set dari jalur independent yaitu:

Jalur 1: 1 - 2 - 4

Jalur 2: 1 - 3 - 4

Tabel 6.1 Test case untuk pengujian penyimpanan data data

3	data	Hasil yang diharapkan	Hasil yang diperoleh		
	1	Sistem melakukan proses	sistem penyimpanan data dan		
Ì		penyimpanan data dan menampilkan	menampilkan data yang telah		
	H	data yang telah tersimpan	tersimpan		

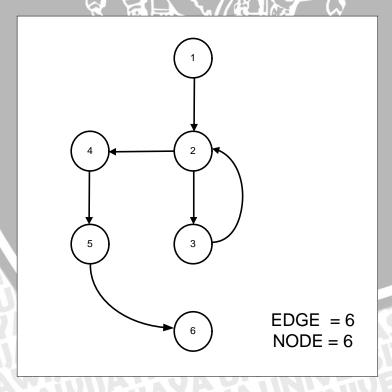
Sumber: Pengujian

## Pengujian Unit: Delete Data jenis Gangguan

```
Kode_Penyakit:string;
begin
   with Gangguan do _
   begin
      - FMenuutama.ADOConn.BeginTrans;
       Sqlcmd:='Delete * From T_Gangguan Where Kode_Gangguan='+ Quotedstr(Kode_Gangguan) +'';
       FMenuutama. ADOConn. Execute (sqlcmd);
       sqlcmd:='Select Top 1 * From T_Penyakit Where Kode_Gangguan= '+ Quotedstr(Kode_Gangguan) +' or
       Search.Close:
       search.SQL.Clear;
       search.SQL.Add(sqlcmd);
       search.Open;
       While not search. IsEmpty do -
           - Kode_Penyakit:=Search.Fieldbyname('Kode_Penyakit').AsString;
           Sqlcmd:='Delete * From T_Gejala_Penyakit Where Kode_Penyakit='+ Quotedstr(Kode_Penyakit) +
           FMenuutama.ADOConn.Execute(sqlcmd);
           sqlcmd:='Select Top 1 * From T Penyakit Where Kode Penyakit > '+ Quotedstr(Kode Penyakit)
            Search.Close;
           search.SQL.Clear;
           search.SQL.Add(sqlcmd);
           search.Open;
       end;
       Sqlcmd:='Delete * From T_Penyakit Where Kode_Gangguan='+ Quotedstr(Kode_Gangguan) +'';
   FMenuutama.ADOConn.Execute(sqlcmd);
       FMenuutama.ADOConn.CommitTrans;
       setting_Awal;
    end;
end:
end.
```

Gambar 6.3 PDL Proses Hapus Jenis Gangguan

Sumber: Pengujian



Gambar 6.4 Grafik Aliran Dari Proses Hapus Jenis Gangguan

Sumber : Pengujian

BRAWIJAYA

$$V(G) = E - N + 2$$

$$V(G) = 6 - 6 + 2$$

$$V(G) = 2$$

Dengan demikian diperoleh 2 buah basis set dari jalur independent yaitu:

Jalur 1: 
$$1 - 2 - 4 - 5 - 6$$

Jalur 2: 
$$1 - 2 - 3 - 2 - 4 - 5 - 6$$

Tabel 6.2 Test case untuk pengujian penghapusan data data

	data	Hasil yang diharapkan Hasil yang diperoleh	
	1	Sistem melakukan proses sistem menghapus data d	lan
X		penghapusan data dan menampilkan menampilkan data yang mas	sih
		data yang masih tersimpan tersimpan	

Sumbar: Pengujian

Dari hasil pengujian yang dilakukan, dapat dilihat bahwa status hasil pengujian seluruhnya adalah *succes*. Dengan demikian maka dapat disimpulkan bahwa dalam setiap prosedur tidak terdapat kesalahan.

## 6.1.1.2 Hasil dan Analisis

Dari hasil pengujian unit yang dilakukan pada unit simpan dan delete data jenis gangguan didapatkan:

- 1. seluruh prosedur untuk menyimpan data gangguan memiliki status vailiditas succes.
- 2. seluruh prosedur untuk menghapus data gangguan memiliki status validitas succes.

Dengan demikian dapat disimpulkan bahwa seluruh unit pada unit jenis gangguan adalah valid

## 6.1.2 Pengujian Integrasi

## 6.1.2.1 Pengujian

Pengujian integrasi diterapkan pada proses yang mengintegrasikan fungsionalitas dari beberapa prosedur untuk melakukan sebuah operasi tertentu. Pada pengujian integrasi yang dijadikan sebagai Obyek uji adalah prosedur yang menggabungkan kinerja dari prosedur yang lain.

Dalam laporan tugas akhir ini dicantumkan beberapa hasil pangujian integrasi yang diterapkan pada proses penyimpanan data, penghapusan data dan perubahan data yang dilakaukan oleh user pakar.

Test case untuk proses pencatatan data dapat dilihat pada table 6.3.

Tabel 6.3 Test case untuk pengujian intergrasi pencatatan data

No	Kasus Uji	Hasil Yang Diinginkan	Hasil Yang Didapatkan
1	Sistem telah dijalankan pada kondisi user telah melakukan login sebagai user pakar dan telah memilih menu file	Sistem menampilkan daftar sub menu yang terdapat dalam menu file.	Sistem menampilkan daftar sub menu yang terdapat dalam menu file.
2	User pakar memilih sub menu jenis gangguan	Sistem menampilkan halaman sub menu jenis gangguan	Sistem menampilkan halaman sub menu jenis gangguan.
3	•	Sitem akan memeriksa apakah data yang dimasukan telah lengkap dan benar, jika terdapat kesalahan maka sistem akan menampilkan peringatan kesalahan.	sistem akan menampilkan pesan kesalahan. Berupa peringatan bahwa data yang dimasukan belum lengkap.
4	User pakar mengulangi proses pengisian data dengan benar	Sistem memriksa data yang dimasukan jika benar maka system melakukan proses penyimpanan data dan menampilkan pesan penyimpanan data telah dilakukan	Sistem melakukan penyimpanan data dam menampilkan pesan penyimpanan data.

Test case untuk proses penghapusan data dapat dilihat pada table 6.4.

Tabel 6.4 Test case untuk pengujian intergrasi pencatatan data

No	Kasus Uji	Hasil yang diinginkan	Hasil yang didapatkan
1	Sistem telah dijalankan	Sistem menampilkan daftar	Sistem menampilkan daftar
	pada kondisi user telah	sub menu yang terdapat	sub menu yang terdapat
	melakukan login	dalam menu file	dalam menu file.
	sebagai user pakar dan		VALUE
V\:	telah memilih menu file		
2	User pakar memilih sub	Sistem menampilkan halaman	Sistem menampilkan
	menu jenis gangguan	sub menu jenis gangguan dan	halaman sub menu jenis
		data jenis gangguan.	gangguan.dan data jenis
			gangguan
3	User pakar memilih data	Sitem akan melakukan	Sitem akan melakukan
	yang akan dihapus dan	penghapus data dan	penghapus data dan
	tekan tombol hapus	menampilkan pesan bahawa	menampilkan pesan bahawa
		data telah terhapus	data telah terhapus

Sumber : Pengujian

## 6.1.2.2 Hasil dan Analisis

Dari hasil pengujian intergasi didapatkan hasil sebagai berikut:

- 1. pada pengujian integrasi pencatatan data seluruh hasil yang didapatkan sesuai dengan hasi yang diharapkan
- 2. pada pengujian integrasi penghapusan data seluruh hasil yang didapatkan sesuai dengan hasi yang diharapkan

Dengan demikian dapat disimpulkan bahwa seluruh hasil pengujian integrasi adalah valid.

## BRAWIJAY

## 6.1.3 Pengujian Verifikasi

## 6.1.3.1 Pengujian

Pengujian validasi digunakan untuk mengetahui apakah sistem yang dibangun sudah benar sesuai dengan yang dibutuhkan. Item-item yang telah dirumuskan dalam daftar kebutuhan dan merupakan hasil analisis kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode pengujian *Black Box*, karena tidak memerlukan untuk berkonsentrasi terhadap alur jalannya algoritma program dan lebih ditekankan untuk menemukan konformitas antara kinerja sistem dengan daftar kebutuhan.

Untuk mengetahui kesesuaian antara kebutuhan dengan kinerja sistem, pada setiap kebutuhan (*requirement*) dilakukan proses pengujian dengan kasus uji masingmasing.

Tabel 6.4 Test case untuk pengujian validasi Menu Konsultasi

Nama Kasus Uji	Konsultasi
Objek Uji	Kebutuhan Menu Konsultasi
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas konsultasi yang akan memberikan hasil diagnosa kepada user tentang penyakitnya.
Prosedur Pengujian	Sistem dijalankan pada menu utama
Hasil yang diharapkan	Sistem menampilkan menu konsltasi yang berisi beberapa pertanyaan dari sistem.
Hasil yang didapatkan	Sistem menampilkan menu konsltasi yang berisi beberapa pertanyaan dari sistem.

Tabel 6.5 Test case untuk pengujian validasi Menu Login Tidak Sah

Nama Kasus Uji	Login tidak sah
Objek Uji	Kebutuhan Login User Pakar
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas <i>log-in</i> bagi User Pakar untuk melakukan
SPERRAM	tugas pencatatan, validasi data jenis gangguan, penyakit, gejala,
TAZK BK	obat, solusi, dtl obat penyakit, dtl gejala penyakit dan hak akses.

Prosedur Pengujian	Sistem dijalankan ketika program dieksekusi
YAUAUNI	User memilih salah satu sebagai user bukan pakar atau sebagai user pakar
MATAYAS	User pakar mengisikan user_id dan password
AWHITTAY	User menekan tombol login
Hasil Yang Diharapkan	Sistem memriksa user_id dan password yang dimasukan, jika
AS BRARA	user_id dan password telah dimasukan, selanjutnya sistem
KITALKEP	memeriksa apakah user_id yang dimasukan berupa karakter
RESILL.	yang valid, jika salah maka sistem menampilkan pesan
Valti-	kesalahan.
Hasil Yang Didapatkan	Sistem menampilkan pesan kesalahan dan kembali ke menu
1	login.

Tabel 6.6 Test case untuk pengujian validasi Menu Login Sah

Nama Kasus Uji	Login sah
Objek Uji	Kebutuhan Login Use bukan Pakar dan User Pakar
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas <i>log-in</i> bagi User Pakar untuk melakukan tugas pencatatan, validasi data jenis gangguan, penyakit, gejala, obat, solusi, dtl obat penyakit, dtl gejala penyakit dan hak akses.
Prosedur Pengujian	<ul> <li>Sistem dijalankan ketika program dieksekusi</li> <li>User memilih salah satu sebagai user bukan pakar atau sebagai user pakar</li> <li>User pakar mengisikan user_id dan password</li> <li>User menekan tombol login</li> </ul>
Hasil Yang Diharapkan	Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika benar maka sistem menampilkan menu utama user pakar.
Hasil Yang Didapatkan	Sistem menampilkan menu utama bagi user pakar.

Tabel 6.7 Test case untuk pengujian validasi Log out

Nama Kasus Uji	Log out
Objek Uji	Kebutuhan Log out
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas <i>log out</i> bagi semua <i>user</i> yang telah melakukan proses <i>log in</i>
Prosedur Pengujian	Sistem dijalankan pada halaman utama     User memilih menu <i>log out</i>
Hasil Yang Diharapkan	Sistem melakukan penghapusan session dan menampilkan halaman <i>log in</i> .
Hasil Yang Didapatkan	Sistem melakukan penghapusan session dan menampilkan halaman <i>log in</i> .

Tabel 6.8 Test case untuk pengujian validasi input jenis gangguan

Nama Kasus Uji	Input jenis gangguan
Objek Uji	Kebutuhan input jenis gangguan
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>jenis gangguan</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	Sistem telah dijalankan pada form jenis gangguan     User pakar mengisikan semua data yang dibutuhkan
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan pesan bahwa penyimpanan data telah dilakukan.
Hasil Yang Didapatkan	Sistem menyimpan data dan menampilkan pesan penyimpanan telah dilakukan.

Tabel 6.9 Test case untuk pengujian validasi input jenis gangguan dengan kesalahan

Nama Kasus Uji	Input jenis gangguan dengan kesalahan
Objek Uji	Kebutuhan input jenis gangguan
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>jenis gangguan</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	Sistem telah dijalankan pada form jenis gangguan     User pakar mengisikan sebagian data yang dibutuhkan
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input masih ada yang Belem terisi, maka sistem akan menampilkan pesan kesalahan dan kembali ke form <i>jenis gangguan</i> .
Hasil Yang Didapatkan	Sistem menampilkan pesan kesalahan dan kembali ke form jenis gangguan.

Tabel 6.10 Test case untuk pengujian validasi input penyakit

Nama Kasus Uji	Input penyakit	
Objek Uji	Kebutuhan input penyakit	
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>penyakit</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .	
Prosedur Pengujian	Sistem telah dijalankan pada form <i>penyakit</i> User pakar mengisikan semua data yang dibutuhkan	
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan pesan bahwa penyimpanan data telah dilakukan	
Hasil Yang Didapatkan	Sistem menyimpan data dan menampilkan pesan penyimpanan telah dilakukan.	

Tabel 6.11 Test case untuk pengujian validasi input penyakit dengan kesalahan

Nama Kasus Uji	Input penyakit dengan kesalahan
Objek Uji	Kebutuhan input penyakit
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>penyakit</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	Sistem telah dijalankan pada form <i>penyakit</i> User pakar mengisikan sebagian data yang dibutuhkan
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input masih ada yang Belem terisi, maka sistem akan menampilkan pesan kesalahan dan kembali ke form <i>penyakit</i> .
Hasil Yang Didapatkan	Sistem menampilkan pesan kesalahan dan kembali ke form penyakit

Tabel 6.12 Test case untuk pengujian validasi input gejala

Nama Kasus Uji	Input gejala
Objek Uji	Kebutuhan input gejala
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>gejala</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	<ul> <li>Sistem telah dijalankan pada form <i>gejala</i></li> <li>User pakar mengisikan semua data yang dibutuhkan</li> </ul>
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan pesan bahwa penyimpanan data telah dilakukan
Hasil Yang Didapatkan	Sistem menyimpan data dan menampilkan pesan penyimpanan telah dilakukan.

Tabel 6.13 Test case untuk pengujian validasi input gejala dengan kesalahan

Nama Kasus Uji	Input gejala dengan kesalahan	
Objek Uji	Kebutuhan input gejala	
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>gejala</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .	
Prosedur Pengujian	<ul> <li>Sistem telah dijalankan pada form gejala</li> <li>User pakar mengisikan sebagian data yang dibutuhkan</li> </ul>	
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input masih adayang Belem terisi, maka sistem akan menampilkan pesar kesalahan dan kembali ke form <i>gejala</i> .	
Hasil Yang Didapatkan	Sistem menampilkan pesan kesalahan dan kembali ke form gejala	

Tabel 6.14 Test case untuk pengujian validasi input obat

Nama Kasus Uji	Input obat
Objek Uji	Kebutuhan input obat
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>obat</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	Sistem telah dijalankan pada form <i>obat</i> User pakar mengisikan semua data yang dibutuhkan
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan pesan bahwa penyimpanan data telah dilakukan
Hasil Yang Didapatkan	Sistem menyimpan data dan menampilkan pesan penyimpanan telah dilakukan.

Tabel 6.15 Test case untuk pengujian validasi input obat dengan kesalahan

Nama Kasus Uji	Input <i>obat</i> dengan kesalahan
Objek Uji	Kebutuhan input obat
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>obat</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	<ul> <li>Sistem telah dijalankan pada form <i>obat</i></li> <li>User pakar mengisikan sebagian data yang dibutuhkan</li> </ul>
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input masih ada yang Belem terisi, maka sistem akan menampilkan pesan kesalahan dan kembali ke form <i>obat</i> .
Hasil Yang Didapatkan	Sistem menampilkan pesan kesalahan dan kembali ke form <i>obat</i> .

Tabel 6.16 Test case untuk pengujian validasi input solusi

Nama Kasus Uji	Input solusi
Objek Uji	Kebutuhan input solusi
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>solusi</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	Sistem telah dijalankan pada form <i>solusi</i> User pakar mengisikan semua data yang dibutuhkan
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan pesan bahwa penyimpanan data telah dilakukan
Hasil Yang Didapatkan	Sistem menyimpan data dan menampilkan pesan penyimpanan telah dilakukan.

Tabel 6.17 Test case untuk pengujian validasi input solusi dengan kesalahan

Nama Kasus Uji	Input solusi dengan kesalahan	
Objek Uji	Kebutuhan input solusi	
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>solusi</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .	
Prosedur Pengujian	Sistem telah dijalankan pada form <i>solusi</i> User pakar mengisikan sebagian data yang dibutuhkan	
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input masih ad yang Belem terisi, maka sistem akan menampilkan pesa kesalahan dan kembali ke form <i>solusi</i> .	
Hasil Yang Didapatkan	Sistem menampilkan pesan kesalahan dan kembali ke form solusi.	

Tabel 6.18 Test case untuk pengujian validasi input dtl obat penyakit

Nama Kasus Uji	Input dtl obat penyakit
Objek Uji	Kebutuhan input dtl obat penyakit
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>dtl obat penyakit</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	Sistem telah dijalankan pada form <i>dtl obat penyakit</i> User pakar mengisikan semua data yang dibutuhkan
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan pesan bahwa penyimpanan data telah dilakukan
Hasil Yang Didapatkan	Sistem menyimpan data dan menampilkan pesan penyimpanan telah dilakukan.

Tabel 6.19 Test case untuk pengujian validasi input dtl obat penyakit dengan kesalahan

Nama Kasus Uji	Input dtl obat penyakit dengan kesalahan
Objek Uji	Kebutuhan input dtl obat penyakit
Tujuan	Untuk menguji validitas kinerja dari sistem yaitu untuk menyediakan fasilitas input <i>dtl obat penyakit</i> . Fasilitas ini hanya dapat diakse oleh <i>user pakar</i> yang telah melakukan <i>login</i> .
Prosedur Pengujian	Sistem telah dijalankan pada form dtl obat penyakit     User pakar mengisikan sebagian data yang dibutuhkan
Hasil Yang Diharapkan	Sistem memeriksa imput yang diberikan, jika input masih ada yang Belem terisi, maka sistem akan menampilkan pesan kesalahan dan kembali ke form dtl obat penyakit.
Hasil Yang Didapatkan	Sistem menampilkan pesan kesalahan dan kembali ke form dtl obat penyakit.

## 6.1.3.2 Hasil dan Analisis

Dari kasus uji yang telah dilaksanakan, didapatkan hasil pengujian sebagaimana ditampilkan dalam table 6.20.

Tabel 6.21 Hasil Test case pengujian validasi

No	Nama Kasus Uji	Hasil yang diharapkan	Hasil yang didapatkan	Status Validitas
1	Konsultasi	Sistem menampilkan	Sistem menampilkan	Valid
		menu konsltasi yang	menu konsltasi yang	
		berisi beberapa	berisi beberapa	
14:		pertanyaan kepada	pertanyaan kepada	
H	<b>注</b> 】	user.	user.	
2	Log in sah	Sistem memriksa	Sistem menampilkan	Valid
	AU	user_id dan password	menu utama bagi user	- GOA
4	NAH	yang dimasukan, jika	pakar.	AS PER
MI		war id dan nagawand		
	TO MERCE	user_id dan password		
RA		telah dimasukan,	UNIVE	4431
RA	RAWI		UNIVIVE	STERNIER VERER

⋖
<□ .
$\rightarrow$

user_id yang dimasukan berupa karakter yang valid, jika benar maka sistem menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan penghapusan session dan menampilkan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	dimasukan berupa karakter yang valid, jika benar maka sistem menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan Sistem melakukan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan Sistem memeriksa jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan menampilkan pesan penyimpanan telah dilakukan.	1		111111	SCII BY AC	DL TOVA	
karakter yang valid, jika benar maka sistem menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	karakter yang valid, jika benar maka sistem menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa sistem memampilkan besar jid dan password yang dimasukan, jika user id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan ladar menampilkan ladar an menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan ladar dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan ladar dan menampilkan ladar dan menampilkan ladar data dan menampilkan ladar menampilkan ladar data dan menampilkan ladar data dan menampilkan ladar dan menampilkan ladar data dan menampilkan ladar menampilkan ladar data dan menampilkan ladar			NATT:	user_id yang	V PLEU	
jika benar maka sistem menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan bertupat karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	jika benar maka sistem menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah naka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user laporan laporan aktivitas user laporan aktivitas user laporan l			NUNIT	dimasukan berupa	TITAS PER	RAYNU
jika benar maka sistem menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupat karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	jika benar maka sistem menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah naka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user laporan laporan aktivitas user laporan aktivitas user laporan l			TUAUN	karakter yang valid,	CHATA2	& BKS
menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user jid dan password pesan kesalahan dan kembali ke menu login.  1 Log out Sistem melakukan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  2 Input jenis gangguan session memeriksa jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	menampilkan halaman utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan Sistem melakukan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan	Ų				JEKS 651L	TAS D
utama user pakar.  3 Log in tidak sah Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  5 Input jenis yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	utama user pakar.  3 Log in tidak sah  Sistem memriksa user id dan password yang dimasukan, jika user id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out  Sistem melakukan pesan kesalahan  Sistem melakukan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  5 Input jenis Sistem memeriksa imput yang diberikan data dan menampilkan pesan penyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan			NUTTIA		MIVEHER	
3 Log in tidak sah  Sistem memriksa user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out  Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan  Input jenis gangguan  Sistem memeriksa Sistem menampilkan data dan menampilkan pesan penyimpanan telah dilakukan.	3 Log in tidak sah  Sistem memriksa user id dan password yang dimasukan, jika user id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out  Sistem melakukan penghapusan session dan menampilkan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan  Sistem meneriksa imput lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan basis data dan menamp			DAWKI		THAT	ERSTON
user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis simput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	user_id dan password yang dimasukan, jika user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang vatid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan lanan l		P	BRAN	utama user pakar.		JULIE
yang dimasukan, jika user_id_dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	yang dimasukan, jika kembali ke menu login.  user_id_dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan		3	Log in tidak sah	Sistem memriksa	Sistem menampilkan	Valid
user_id_dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	user_id dan password telah dimasukan, selanjutnya sistem memeriksa apakah user_id yang dimasukan berupa karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log-in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan laporan aktivitas user			-dita	user_id dan password	pesan kesalahan dan	
karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan	1	V		yang dimasukan, jika	kembali ke menu <i>login</i> .	
karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan lagoran aktivitas user pakar kedalam basis data dan menampilkan		M		user_id dan password	RD.	
karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan		1		telah dimasukan,	DRAM	
karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan lagoran aktivitas user pakar kedalam basis data dan menampilkan				1	- "	/.
karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan lagoran aktivitas user pakar kedalam basis data dan menampilkan						
karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan lagoran aktivitas user pakar kedalam basis data dan menampilkan						Y
karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	karakter yang valid, jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan lagoran aktivitas user pakar kedalam basis data dan menampilkan					Sp CS	
jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	jika salah maka sistem menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan			7			
menampilkan pesan kesalahan.  4 Log out  Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	menampilkan pesan kesalahan.  4 Log out Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan				5 Park 14 / 1/2 3		
kesalahan.  4 Log out  Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan  imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	kesalahan.  4 Log out  Sistem melakukan penghapusan session penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan  imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan basis data dan menampilkan basis data dan menampilkan data dan menampilkan basis data dan menampilkan						
4 Log out  Sistem melakukan Sistem melakukan penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan  Sistem memeriksa sistem menyimpan data dan menampilkan pesan penyimpanan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	4 Log out  Sistem melakukan penghapusan session dan menampilkan halaman log-in.  5 Input jenis gangguan  Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan				menampilkan pesan		
penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan	١			kesalahan.		
penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	penghapusan session penghapusan session dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan		4	τ ,			X7 1° 1
dan menampilkan halaman log in.  5 Input jenis Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	dan menampilkan halaman log in.  5 Input jenis gangguan imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan basis data dan menampilkan laporan aktivitas user pakar kedalam basis data dan menampilkan	l	4	Log out		太川野(A)	vand
halaman log in.  Input jenis Sistem memeriksa jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	halaman log-in.  Input jenis Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan						
5 Input jenis Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	5 Input jenis Sistem memeriksa imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan data dan menampilkan	1	N		dan menampilkan	dan menampilkan	
imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan		М		halaman <i>log in</i> .	halaman <i>log in</i> .	
imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	imput yang diberikan, jika input lengkap dan benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan		5	Input	Sistem	Sistam	Volid
jika input lengkap dan benar maka sistem telah dilakukan.  menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	jika input lengkap dan benar maka sistem telah dilakukan.  menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan	À			M-3 1 / M-		valid
benar maka sistem menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	benar maka sistem telah dilakukan.  menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan	3		gangguan			
menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user	menyimpan data kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan		13:	3.		1 1 1	
kedalam basis data, sistem menuliskan laporan aktivitas user	kedalam basis data, sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan		H	UE		telah dilakukan.	
sistem menuliskan laporan aktivitas user	sistem menuliskan laporan aktivitas user pakar kedalam basis data dan menampilkan		1	III	menyimpan data		
laporan aktivitas user	laporan aktivitas user pakar kedalam basis data dan menampilkan				kedalam basis data,		
N TEAL OF THE REPORT OF THE RE	pakar kedalam basis data dan menampilkan			AUA U	sistem menuliskan		AC BR
TYPOTHER STATE OF THE STATE OF	data dan menampilkan			PHATA	laporan aktivitas user	TUENZESSI	THAS E
pakar kedalam basis	K DPSOAPSIUNESAVAYS INVESTUEN			WILLIAM	pakar kedalam basis	HIVER	251451
data dan menampilkan	K DPSOAPSIUNESAVAYS INVESTUEN			PANI	data dan menampilkan	LITTINIX	VER213
K DESOARSIDIES AVASSANIS SUEP	I A N Y TO KEN A LIMIT LARY PO TA TINY A TIL		5 1	BRA		YAJA UN	KIVES
	TIPEAS DEBORY, INTERIAVA YA TINI	Ĺ		B	Francisco	HAYRJA	

$\overline{}$

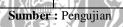
<u> </u>			BR46AW	
		penyimpanan data telah	AS PEBR	
V	NU-MIN	dilakukan.	CHAPACE.	
6	Input jenis	Sistem memeriksa	Sistem menampilkan	Valid
H	gangguan	imput yang diberikan,	pesan kesalahan dan	
	dengan	jika input masih ada	kembali ke form jenis	ERSIL ST
	kesalahan	yang Belem terisi,	gangguan.	HULLY
TA	2 REBR	maka sistem akan		HTIL
	TATAS	menampilkan pesan		AL VUIS
+17	RULL	kesalahan dan kembali		A CONT
+		ke form jenis	DA	TA
		gangguan.	PRAIL	
			V	
7	Input <i>penyakit</i>	Sistem memeriksa	Sistem menyimpan	Valid
		imput yang diberikan,	data dan menampilkan	Y
		jika input lengkap dan		
	7	benar maka sistem	telah dilakukan.	
		menyimpan data		
		kedalam basis data,		
		sistem menuliskan		
		laporan aktivitas user		
		pakar kedalam basis		
		data dan menampilkan		
1.		pesan bahwa		
		penyimpanan data telah		
		dilakukan		
8	Input <i>penyakit</i>	Sistem memeriksa	Sistem menampilkan	Valid
	dengan	imput yang diberikan,	pesan kesalahan dan	Valid
13:	kesalahan	jika input masih ada	kembali ke form	
TT	VE.	yang Belem terisi,	penyakit	
		maka sistem akan		
At	AU	menampilkan pesan		- 50 A
	AVAU	kesalahan dan kembali		AS PE
	MAH	ke form penyakit.	KITTER	
9	Input g <i>ejala</i>	Sistem memeriksa	Sistem menyimpan	Valid
	7 8-7	imput yang diberikan,	data dan menampilkan	NUAT
	AS D			

\		111111	SCHIPLAC	DI SOAW	
	T		jika input lengkap dan	pesan penyimpanan	
	J		benar maka sistem	telah dilakukan.	
	4		menyimpan data	2651124	
N			kedalam basis data,	VEH ERDLY	
			sistem menuliskan	NATUE	
	B		laporan aktivitas user		
	A		pakar kedalam basis		
	45	TAPAG	data dan menampilkan		
	4	55112	pesan bahwa		
	<u>V</u> :		penyimpanan data telah		
	K		dilakukan	BRAW	
	10	Input gejala	Sistem memeriksa	Sistem menampilkan	Valid
À		dengan	imput yang diberikan,	pesan kesalahan dan	
4	4	kesalahan	jika input masih ada	kembali ke form <i>gejala</i>	
1		5	yang Belem terisi,		
			maka sistem akan		
			menampilkan pesan		
			kesalahan dan kembali		
١			ke form gejala.		
1	11	Input obat	Sistem memeriksa	Sistem menyimpan	Valid
			imput yang diberikan,	data dan menampilkan	
	1		jika input lengkap dan	pesan penyimpanan	
	M		benar maka sistem	telah dilakukan.	
			menyimpan data		
			kedalam basis data,		
			sistem menuliskan		
	4	2.	laporan aktivitas user	)	
	H	JE	pakar kedalam basis		
	1		data dan menampilkan		
			pesan bahwa		
			penyimpanan data telah		AS BR
	W		dilakukan		
1	12	Input obat	Sistem memeriksa	Sistem menampilkan	Valid
		dengan	imput yang diberikan,	pesan kesalahan dan	
L			PEOAWN	HALLUA	

4		kesalahan	jika input masih ada	kembali ke form <i>obat</i>	
\		Resalaliali		Kemban ke form obai	
	V		yang Belem terisi,	SITALAS	PARA
1	41		maka sistem akan	HERSLLSTV.	DRAB
	4		menampilkan pesan	ATTERNATION OF	ATTARA K
			kesalahan dan kembali	WHITTEH	ARSILA
	B		ke form obat.		HILL
	13	Input solusi	Sistem memeriksa	Sistem menyimpan	Valid
	P	Latita	imput yang diberikan,	data dan menampilkan	VAUN
i			jika input lengkap dan	pesan penyimpanan	
			benar maka sistem	telah dilakukan.	
Y	H		menyimpan data	DRAM	
			kedalam basis data,		
			sistem menuliskan		NAL.
		3	laporan aktivitas user		V.
			pakar kedalam basis		
ı		7	data dan menampilkan		
			5 824 14 1C3		
۱			penyimpanan data telah		
			dilakuk		
١	14	Input solusi	Sistem memeriksa	Sistem menampilkan	Valid
N		dengan	imput yang diberikan,		
١		kesalahan	jika input masih ada		
	М		yang Belem terisi,		
			maka sistem akan		
			menampilkan pesan		
j			kesalahan dan kembali	TAM SR	
	4		ke form <i>solusi</i> .	<b>U</b>	
	3	25	Ke form solust.		
	15	Input dtl obat	Sistem memeriksa	Sistem menyimpan	Valid
		penyakit	imput yang diberikan,	data dan menampilkan	
			jika input lengkap dan	pesan penyimpanan	FRRA
	H		benar maka sistem	telah dilakukan.	AL PER
			menyimpan data	WHITELY	SITA
1			kedalam basis data,	UNITIVE	HERSLY
			sistem menuliskan	VAULTIN	MATTER
			laporan aktivitas user	HAYAJA	
		ATAD A	BREDAW		AULT

$\overline{}$
$\triangleleft$

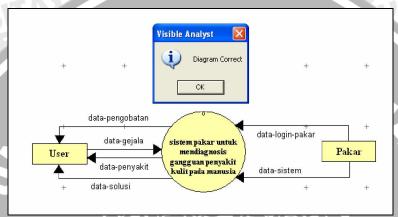
		pakar kedalam basis data dan menampilkan pesan bahwa penyimpanan data telah dilakukan	AS BY BRASILAS IN AS BY BRASIL	RAWI S BRAY S BRAY STAS BR STAS BRAY ERSTA
16	Input dtl obat	Sistem memeriksa	Sistem menampilkan	Valid
15	penyakit dengan	imput yang diberikan,	pesan kesalahan dan	
	kesalahan	jika input masih ada	kembali ke form dtl	DAUL
V):		yang Belem terisi,	obat penyakit.	
		maka sistem akan	RD.	T/P
		menampilkan pesan	MAIN	
4		kesalahan dan kembali	BRAW	
		ke form dtl obat		
	3	penyakit.	(A)	<b>—</b>



## 6.2 Pengujian Data Flow Diagram

Pengujian perancangan Data Flow Diagram bertujuan untuk mengetahui apakah implementasi perancangan Data Flow Diagram yang dilakukan telah sesuai dengan *Diagram konteks*. Pengujian perancangan Data Flow Diagram dilakukan dengan menggunakan perangkat lunak Visible Analist 7.6.

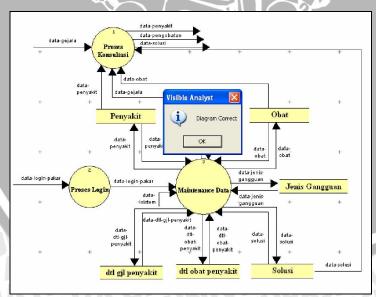
1. hasil pengujian Diagram Konteks menggunakan visible analist 7.6



Gambar 6.5 Hasil pengujian Diagram Konteks menggunakan visible analist 7.6

Sumber: Pengujian

2. Hasil pengujian Diagram Maintenance Data Level 1

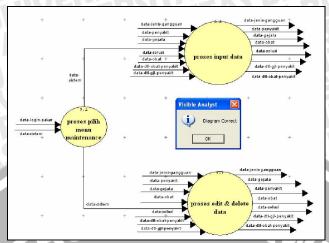


**Gambar 6.6** Hasil pengujian Diagram Maintenance Data Level 1 menggunakan visible analist 7.6

Sumber: Pengujian

BRAWIJAYA

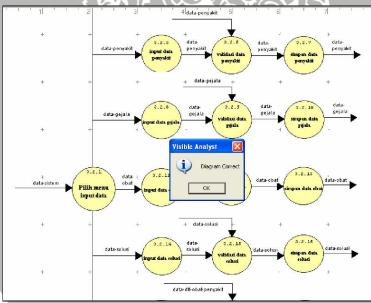
## 3. Hasil pengujian Diagram Pilih Maintenance Data Level 2



**Gambar 6.7** Hasil pengujian Diagram Maintenance Data Level 2 menggunakan visible analist 7.6

Sumber: Pengujian

## 4. Hasil pengujian Diagram Pilih Maintenance Data Level 3



**Gambar 6.8** Hasil pengujian Diagram Pilih Menu Input Data Level 3 menggunakan visible analist 7.6

Sumber: Pengujian

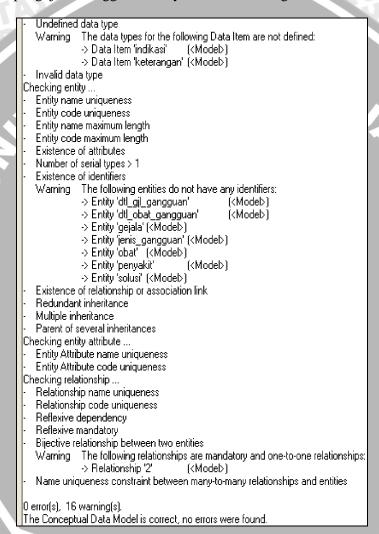
Dari hasil pengujian dapat dilihat bahwa tidak terdapat kesalahan atau error. Dengan demikian dapat dinyatakan bahwa Data Flow Diagram yang digunakan sudah valid dan Diagram telah dinyatakan Correct.

# BRAWIJAYA

## 6.3 Pengujian Basis Data

Pengujian perancangan basis data bertujuan untuk mengetahui apakah implementasi perancangan basis data yang dilakukan telah sesuai dengan *entity* relationship diagram. Pengujian perancangan basis data dilakukan dengan menggunakan perangkat lunak Sybase PowerDesigner 12.5.

## 1. Hasil pengujian menggunakan Sybase Power Designer 12.5



Gambar 6.9 Hasil pengujian menggunakan Sybase Power Designer 12.5

## BRAWIJAY

## 6.4 Pengujian Faktor kepastian

Pengujian Faktor Kepastian dilakukan untuk mengetahui kemampuan dari sistem pakar ini dalam mengitung nilai faktor kepastian. Dalam pengujian ini digunakan salah satu aturan yang terdapat dalam sistem pakar ini. Aturan tersebut yaitu:

R1 Campak dengan CF 0,8

JIKA timbul bintik

DAN Hidung meler

**DAN Batuk** 

DAN Demam

DAN Nyeri Tenggorokan

DAN Timbul ruam pada muka seperti jerawat

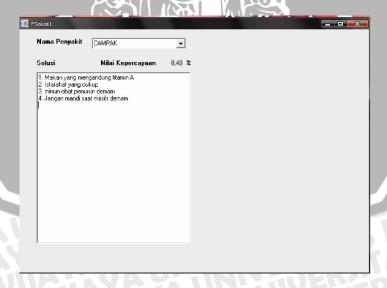
DAN Cepat merasa lelah

Dari *User* diketaui bahwa *user* mengalami gejala timbul bintik, hidung meler, batuk, demam, nyeri tenggorokan, timbul ruam pada muka dan cepat lelah.

Dari tiga gejala yang terdapat dalam aturan, penderita hanya mengalami dua gejala. Maka, CF = 7/12 \* 0.8 = 0.48 %

Jadi, penderita menderita cacar air dengan tingkat kepastian 48 %.

Ketika sistem pakar diberi input seperti diatas didapatkan asil seperti pada Gambar 6.10.



Gambar 6.10 Tampil Hasil Diagnosa Pengujian Tingkat Kepastian

## BRAWIJAY/

## **6.4 Turing Test**

Pengujian turing Test dilakukan dengan memberikan beberapa pertanyaan yang kemudian akan dijawab oleh sistem dan seorang pakar. Kedua jawaban tersebut dibandingkan dan dari situ dapat dihitung tingkat akurasi sistem. Dalam Sistem Pakar untuk Mendiagnosis Ganggan dan penyakit kulit pada manusia ini pakar adalah seorang dokter kulit, yaitu dr. Aunur Rofiq .SpKK. Jawaban sistem dan pakar dapat dilihat dalam Tabel 6.22.

Tabel 6.22 Perbandingan Jawaban Pakar dan Sistem

111:	RU	ndingan sawaban rakar da	
No	Gejala GII	Hasil Diagnosa	
	ERS	Pakar	Sistem
1	• Demam		4,
	Mata merah		
	Timbul ruam pada muka seperti jerawat	Campak	Campak
	Bernafas dengan cepat		2
	Batuk		Q
2	• Demam		
	Timbul bintik	1 3 Re (2)	
	Bintik merah berisi cairan jernih pada seluruh tubuh	Cacar Air	Cacar Air
	Sakit kepala		
	Cepat merasa lelah		
3	• Timbal bintik	17/11/11/11/18	
	Bintik semakin meluas	200	
	Bintik Berkeruping	Impetigo	Impetigo
	Bintik berwarna merah		
	Bintik melepuh dan bernanah		
4	Timbul bengkak berwarna merah	HNIVHEE	STATASE
RA	Bengkak semakin membesar	Bisul	Bisul
	Ujung bengkak melunak dan bernanah	IJAYAYAU	NUNIVER

$\rightarrow$
$\triangleleft$

5	<ul> <li>Hidung Meler</li> <li>Demam</li> <li>Cepat lelah</li> <li>Nyeri sendi</li> <li>Sakit kepala</li> <li>Timbul bintik</li> <li>Pernah mendaerita cacar</li> </ul>	Cacar Ular	Cacar Air
6	<ul> <li>Timbul bintik</li> <li>Bintik semakin membesar</li> <li>Jumlah satu dan bias bertambah</li> <li>Bintik terasa kasar</li> <li>Tidak nyeri</li> </ul>	AS Veruka	Veruka
7	<ul><li>Timbul bintik</li><li>Bintik terasa kasar</li><li>Bintik semakin membesar</li><li>Bintik berwarna merah</li></ul>	Veruka	Psoriasis
8	<ul> <li>Timbul bercak merah</li> <li>Timbul dibagian tubuh tertentu</li> <li>Bercak merah ditutupi sisik halus</li> <li>Bercak terasa gatal</li> </ul>	Dermatomikosis	Dermatomikosis

Sumber: Pengujian dan Analisis

Dari enam belas variasi gejala yang diberikan empat belas diantaranya sistem dan pakar menjawab sama. Sistem tersebut memiliki tingkat keakuratan 87,5%.

### **BAB VII**

## PENUTUP

## 7.1 Kesimpulan

Kesimpulan yang didapat dari perancangan, implementasi, dan pengujian aplikasi Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia Menggunakan Faktor Kepastian antara lain:

- Sistem Pakar untuk mendiagnosis Gangguan dan Penyakit Kulit pada Manusia Menggunakan Faktor Kepastian dapat mendignosis gangguan dan penyakit pada Manusia dan solusi untuk mengatasi penyakit kulit tersebut dengan menerima masukkan gejala dari user.
- 2. Aturan-aturan (*rule*) yang ada dalam sistem pakar ini dapat diperbaharui oleh pakar secara mudah dengan mengisikan tabel gejal penyakit pada data file.
- 3. Pada pengujian basis data menggunakan PowerDesigner 12.5 dapat disimpulkan bahwa ER-diagram yang dibuat tidak ditemukan kesalahan.
- 4. Pada pengujian DFD data menggunakan Visible Analyst 7.6 dapat disimpulkan bahwa DFD yang dibuat tidak ditemukan kesalahan dan dinyatakan correct.
- 5. Dari Pengujian test turing didapatkan hasil tingkat keakurasian sistem pakar sebesar 87,5%.
- 6. Untuk mendapatkan nilai faktor kepastian, diharapkan user dapat menjawab semua pertanyaan yang terdapat dalam sistem pakar

## 7.2 Saran

Saran yang dapat diberikan untuk pengembangan Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia menggunakan Faktor kepastian:

- 1. Sistem Pakar untuk Mendiagnosis Gangguan dan Penyakit Kulit pada Manusia Menggunakan Faktor Kepastian dapat dikembangkan dalam bentuk web service sehingga dapat diakses oleh oleh lebih banyak user
- 2. Dalam memberi pertanyaan kepada pengguna dapat diberikan keterangan lebih jelas mengenai suatu gejala sehingga pengguna lebih memahami bagaimana mendiagnosa suatu gejala.

BRAWIJAY

- 3. Dalam penghitungan CF, perlu dikembangkan lebih lanjut untuk mendapatkan CF yang mendekati nilai sesungguhnya.
- 4. Basis Pengetahuan harus diolah sebaik mungkin agar dapat menghasilkan diagnosis yang tepat.
- 5. solusi dan obat yang diberikan sebaiknya disesuaikan dengan setiap gejala yang timbul pada suatu penyakit.

