

BAB III

PERANCANGAN SISTEM

Dalam bab ini dibahas tentang perancangan perangkat lunak dari sistem, meliputi perancangan aturan penyakit flu burung, perancangan proses, verifikasi dan penjelasan mengenai parameter serta dilengkapi dengan *Block Diagram*, *Dependency Diagram*, *Decision Table*, *Reduced Decision Table*, *Rule Base*, desain arsitektur, diagram alir sistem, dan struktur tabel.

3.1 Perancangan Aturan Penyakit Flu Burung

Perancangan aturan ini diperlukan untuk mengetahui langkah-langkah dalam membangun suatu *Knowledge Based System* (KBS), diantaranya yaitu membuat *Block Diagram*, yaitu mengumpulkan faktor-faktor penting yang dibutuhkan dan dibentuk menjadi sebuah diagram, dalam hal ini adalah tentang penyakit flu burung, setelah itu kita membuat *Dependency Diagram* yang merupakan transformasi dari block diagram. Dari *Dependency Diagram* tersebut selanjutnya kita tentukan *Decision Table* dan *Reduced Decision Table* secara manual.

3.1.1 Block Diagram

Block Diagram diperlukan untuk mengetahui urutan kerja sistem untuk mencapai keputusan. sebagai basis pengetahuan sistem untuk menentukan penyakit flu burung diambil dari parameter Anamnesa serta dari hasil pemeriksaan fisik dari penderita. Berdasarkan parameter-parameter yang ada maka dapat disusun *Dlock Diagram* seperti pada gambar 3.1.

STIKOMMP SURABAYA

Pada Gambar 3.1 dijelaskan bahwa diagnosa penyakit flu burung pada level 1 terdiri dari parameter anamnesa dan Pemeriksaan Fisik. Pada level 2 parameter Anamnesa terdiri dari pertanyaan-pertanyaan mengenai seputar keseharian yang dilakukan pasien atau *User*. Sedangkan parameter pemeriksaan fisik terdiri dari sub parameter demam, nyeri otot, badan lemas, sedang selaput paru, sakit kepala, hidung, mata, pernafasan, dan tenggorokan. Untuk level 3 menjelaskan sub parameter dari level 2 yaitu hidung, mata, pernafasan dan tenggorokan.

Untuk sub parameter hidung dibagi lagi menjadi beberapa pertanyaan seputar hidung yaitu : sekret hidung meningkat & encer, hidung tersumbat, hidung merah dan gatal. Untuk sub parameter mata dibagi lagi menjadi beberapa pertanyaan seputar mata yaitu apakah mata besar, merah dan gatal. Sedangkan untuk sub parameter tenggorokan dibagi menjadi beberapa parameter yaitu : nyeri tenggorokan, gatal tenggorokan dan batuk.

3.1.2 Dependency Diagram

Dependency Diagram merupakan proses transformasi dari dari *Block Diagram* yang sudah kita buat. *Dependency Diagram* digunakan untuk menentukan hubungan antara faktor-faktor penting yang mempengaruhi dalam pembuatan rekomendasi penyakit. *Dependency Diagram* juga berisi aturan-aturan dan jawaban yang digunakan untuk memudahkan pada saat proses verifikasi. Untuk lebih jelasnya dapat dilihat pada Gambar 3.2 yang memberikan penjelasan bahwa hasil diagnosa penyakit flu burung berdasarkan parameter anamnesa dan pemeriksaan fisik, dengan penjabaran tiap parameter pada sub parameter yang terbagi dalam tiap-tiap set.

STIKOMP SURABAYA

Tabel 3.1 Keterangan Parameter Set Gejala Fisik:

No.	Variabel	Parameter
1.	Gejala Fisik 1	Panas > 38 ⁰ C, nyeri tenggorokan, Batuk, sesak nafas
2.	Gejala Fisik 2	Panas > 38 ⁰ C, Sekret hidung meningkat dan encer, hidung merah dan gatal, Batuk, sesak nafas
3.	Gejala Fisik 3	Panas > 38 ⁰ C, Hidung tersumbat, Hidung merah dan gatal, Nyeri Tenggorokan, sesak nafas
4.	Gejala Fisik 4	Panas > 38 ⁰ C, radang tenggorokan, Batuk, sesak nafas
5.	Gejala Fisik 5	Panas > 38 ⁰ C, Hay fever, Hidung merah dan gatal, Batuk, sesak nafas
6.	Gejala Fisik 6	Panas > 38 ⁰ C, nyeri otot, sakit kepala, hay fever, mata Berair, mata merah, mata gatal, sesak nafas, batuk
7.	Gejala Fisik 7	Panas > 38 ⁰ C, nyeri otot, sakit kepala, Sekret hidung meningkat dan encer, hidung merah dan gatal, mata Berair, mata merah, mata gatal, sesak nafas, batuk
8.	Gejala Fisik 8	Panas > 38 ⁰ C, nyeri otot, sakit kepala, Sekret hidung meningkat dan encer, hidung merah dan gatal, mata Berair, mata merah, mata gatal, sesak nafas, radang tenggorokan
9.	Gejala Fisik 9	Panas > 38 ⁰ C, nyeri otot, badan lemas, sakit kepala, hay fever, mata Berair, mata merah, mata gatal, sesak nafas, radang tenggorokan, batuk
10.	Gejala Fisik 10	Panas > 38 ⁰ C, nyeri otot, badan lemas, sakit kepala, Sekret hidung meningkat dan encer, hidung merah dan gatal, mata merah, sesak nafas, batuk
11.	Gejala Fisik 11	Tidak ada gangguan gangguan fisik

Tabel 3.2 Keterangan *conclusion* set anamnesa

No.	Variabel	<i>conclusion</i>
1.	Anamnesa 1	Pernah kontak dengan penderita flu burung konfirmasi dalam 7 hari terakhir, Kontak dengan unggas, babi serta produk mentahnya dalam 7 hari terakhir, Waktu timbulnya gejala 3-7 hari, Pada anak-anak 21 hari, Riwayat keluarga, Riwayat pekerjaan
2.	Anamnesa 2	Pernah kontak dengan penderita flu burung konfirmasi dalam 7 hari terakhir, Kontak dengan unggas, babi serta produk mentahnya dalam 7 hari terakhir

No	Variabel	<i>conclusion</i>
3.	Anamnesa 3	kontak dengan penderita flu burung konfirmasi dalam 7 hari terakhir, Waktu timbulnya gejala 3-7 hari
4.	Anamnesa 4	kontak dengan penderita flu burung konfirmasi dalam 7 hari terakhir, Riwayat pekerjaan
5.	Anamnesa 5	Kontak dengan unggas, babi serta produk mentahnya dalam 7 hari terakhir, riwayat pekerjaan
6.	Anamnesa 6	timbulnya gejala 3-7 hari riwayat keluarga
7.	Anamnesa 7	Tidak ada gangguan

Berikut keterangan dari parameter-parameter set yang lain.

1. Penyakit Flu Burung : Merupakan kesimpulan akhir dari seluruh proses inferensi dan sebagai *Input* diagnosa tingkat penyakit flu burung beserta penyebab dan pengobatannya.
2. Pemeriksaan Fisik : Pemeriksaan secara obyektif yang dilakukan oleh dokter atau petugas paramedis terhadap kondisi fisik pasien sebagai data obyektif.
3. Hidung : Hidung adalah parameter gejala klinis yang terdiri dari sub parameter secret hidung meningkat dan encer, hidung tersumbat, hidung berwarna merah dan gatal.
4. Sekre hidung meningkat : Kondisi hidung yang mengeluarkan cairan secara terus menerus.
5. Hidung tersumbat : Hidung tersumbat diakibatkan oleh pembengkakan jaringan hidung karena pelepasan histamin dan sel mast atau sel mati.
6. Hidung berwarna : Akibat reaksi terhadap allergen, karena terus

- merah dan gatal : digosok maka hidung akan berwarna merah.
7. Mata merah : Keadaan peradangan pada mata khususnya bagian putih mata dan kelopak mata bagian dalam (bagian *conjunctiva*).
8. Mata berair : Keadaan mata yang terus mengeluarkan air.
9. Mata gatal : Keadaan mata yang terus menerus gatal.
10. Sesak nafas : Merupakan kondisi penderita yang sulit bernafas dan rasa sesak di dada.
11. Panas > 38 derajat : Panas >38 derajat adalah sub parameter yang digunakan untuk menentukan suhu tubuh penderita.
12. Nyeri otot : Rasa sakit pada otot tubuh, dimana terdapat hubungan dengan emosional dan selalu ada kerusakan jaringan atau potensial menjadi rusak.
13. Badan lemas : Keadaan dimana badan terasa tidak bertenaga sehingga membuat malas untuk melakukan sesuatu.
14. Nyeri tenggorokan : Rasa sakit pada tenggorokan, yang mengakibatkan batuk atau sakit pada saat digunakan untuk menelan dan selalu ada kerusakan jaringan atau potensial menjadi rusak.

15. Gatal tenggorokan : Rasa yang tidak enak atau rasa gatal pada tenggorokan, yang mengakibatkan batuk.
16. Batuk : Penyakit yang timbul akibat rasa yang tidak enak, gatal atau sakit pada tenggorokan.
17. Sakit kepala : Rasa sakit pada kepala.
18. Anamnesa : Upaya menanyakan riwayat penyakit atau keluhan pasien oleh dokter atau petugas paramedik kepada pasien dengan menggunakan bahasa pasien (dapat dimengerti pasien) sebagai data subyektif. Diantaranya yaitu apakah penderita pernah kontak dengan penderita flu burung konfirmasi dalam 7 hari terakhir, apakah penderita pernah kontak unggas, babi serta produk mentahnya dalam 7 hari terakhir, apakah penderita telah mengalami gejala penyakit ini sejak 3-7 hari terakhir, apakah riwayat keluarga penderita ada yang pernah mengalami penyakit atau gejala yang serupa, dan yang terakhir apakah riwayat pekerjaan penderita berhubungan dengan unggas, laboratorium yang menangani masalah flu burung.

3.1.3 Perancangan Decision Table

Menurut Dologite (1993:23) *decision table* digunakan untuk menunjukkan hubungan suatu nilai-nilai yang menghasilkan beberapa tahap menengah atau rekomendasi akhir dari *Knowledge Based System*. Pada tabel 3.3 berikut ini menunjukkan salah satu contoh perancangan *Decision Table* untuk *Rule set 4* yaitu parameter hidung berdasarkan pada perancangan *Dependency Diagram*.

Tabel 3.3 *Decision Table Rule Set 4*

Step 1 : Plan

Kondisi	Sekret hidung encer dan meningkat	(Ya/Tidak)
	Hidung tersumbat	(Ya/Tidak)
	Hidung gatal dan berwarna merah	(Ya/Tidak)
Baris $2 \times 2 \times 2 = 8$		

Step 2 : Completed Decision Table

Rule	Sekret hidung meningkat dan encer	Hidung tersumbat	Hidung merah dan gatal	Hidung
A1	Y	Y	Y	-
A2	T	Y	T	-
A3	T	T	Y	Hay fever
A4	Y	T	T	Sekret hidung meningkat dan encer
A5	T	Y	Y	Hidung tersumbat, Hidung merah dan gatal
A6	T	Y	T	-
A7	T	T	Y	-
A8	T	T	T	Tidak ada gangguan hidung

Dalam Tabel 3.3 rencana *Decision Table* adalah untuk rangkaian aturan akhir yang terkait dengan tiga kondisi, masing-masing dapat memiliki sejumlah

nilai yang berbeda. Sekret hidung encer dan meningkat, kondisi pertama hanya memiliki dua nilai yaitu : apakah ya atau tidak. Begitu pula dengan hidung tersumbat serta hidung gatal dan berwarna merah, kondisi kedua dan ketiga juga memiliki dua nilai ya atau tidak.

3.1.4 Perancangan Reduksi

Pada sistem ini proses perancangan reduksi untuk setiap *Decision Table* dilakukan secara manual. Perancangan reduksi berdasarkan *Decision Table* pada tabel 3.3 menghasilkan parameter seperti pada tabel 3.4 berikut:

Tabel 3.4 *Reduced Decision Table Rule Set 4*

Step 1 : Plan

Kondisi	Sekret hidung encer dan meningkat	(Ya/Tidak)
	Hidung tersumbat	(Ya/Tidak)
	Hidung gatal dan berwarna merah	(Ya/Tidak)
Baris $2 \times 2 \times 2 = 8$		

Step 2 : Completed *Decision Table*

Rule	Sekret hidung meningkat dan encer	Hidung tersumbat	Hidung merah dan gatal	Hidung
A1	Y	Y	Y	-
A2	Y	Y	T	-
A3	Y	T	Y	Hay fever
A4	Y	T	T	Sekret hidung meningkat dan encer
A5	T	Y	Y	Hidung tersumbat, Hidung merah dan gatal
A6	T	Y	T	-
A7	T	T	Y	-
A8	T	T	T	Tidak ada gangguan

Step 3 : Reduced Decision Table

Rule	Sekret hidung meningkat dan encer	Hidung tersumbat	Hidung merah dan gatal	Hidung
A1	Y	T	Y	Hay Fever
A2	Y	T	T	Sekret hidung meningkat dan encer
A3	T	Y	Y	Hidung tersumbat Hidung merah dan gatal
A4	T	T	T	Tidak ada gangguan hidung

3.1.5 Perancangan Rule Base

Pada pengembangan *Rule Base* telah dipresentasikan dalam bentuk *Blok Diagram* yang kemudian diimplementasikan dalam bentuk list aturan (*Rule*), yaitu struktur berbasis pengetahuan. *Rule Base* pada sistem ini menghasilkan *File* KBS, yaitu *File* text berupa himpunan aturan (*Rule*), sedangkan untuk *Premis* dan *Conclusion* disimpan dalam tabel *Rule Base* pada dasarnya terdiri dari dua bagian pokok yaitu : bagian *If* (*Premis* atau kondisi) dan bagian *Then* (*Conclusion* atau kesimpulan). Pemilihan representasi pengetahuan dengan *Rule Base* didasarkan alasan sebagai berikut:

- Pengembangan sistem pakar menggunakan *Rule Base*.
- Rule Base* dapat dengan mudah dilakukan perubahan, seperti penambahan, penghapusan, dan perubahan aturan.

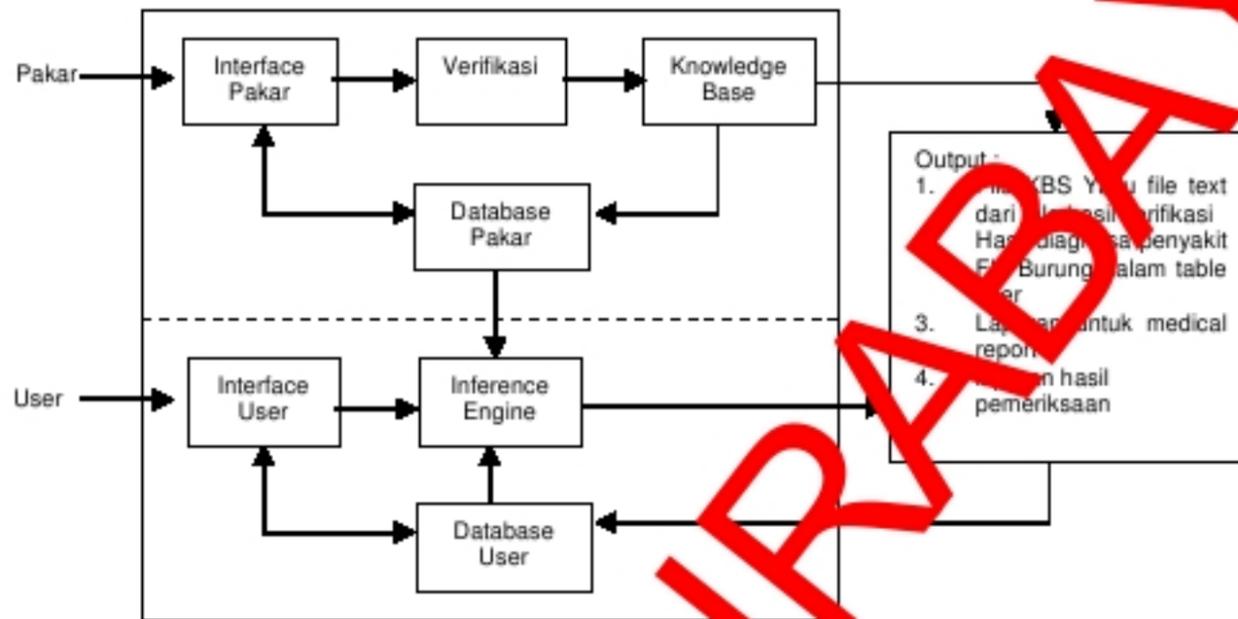
Contoh berikut merupakan struktur basis pengetahuan yang sebelumnya telah dirancang menggunakan *Block Diagram*.

Rule 8

IF Suhu = panas > 2 hr AND Mimisan = tidak
THEN Anamnesa = panas > 2 hr

3.2 Desain Arsitektur

Desain Arsitektur seperti terlihat pada Gambar 3.3, menggambarkan hubungan antara elemen-elemen utama.



Gambar 3.3 Desain Arsitektur untuk Diagnosa Penyakit Flu Burung

Berikut penjelasan dari desain arsitektur:

1. Pakar dalam pembuatan sistem berbasis aturan ini adalah dokter spesialis.
2. User adalah masyarakat umum yang ingin berkonsultasi tentang penyakit flu burung secara *online*.
3. *Interface Pakar* : digunakan oleh pakar untuk mengembangkan sistem, dan dalam merancang sistem tersebut dengan menentukan parameter penyakit flu burung ke *Verifikasi* yang menghasilkan file text KBS, yaitu berupa himpunan aturan (*Rule*).
4. *Verifikasi* : Proses *Verifikasi* dijalankan ketika ada penambahan atau perubahan pada *Rule*, karena *Rule* tersebut sebelumnya sudah ada pada sistem, yang terdapat pada table *Rule*.

- a. *Redundant Rules* adalah suatu *Rule* jika dua *Rule* atau lebih mempunyai *Premise* dan *Conclusion* yang sama.
 - b. *Conflicting Rules* adalah suatu *Rule* jika dua *Rule* atau lebih mempunyai *Premise* yang sama, tetapi mempunyai *Conclusion* yang berlawanan.
 - c. *Circular Rules* adalah suatu keadaan dimana terjadinya proses perulangan dari suatu *Rule*. Ini dikarenakan suatu *Premise* dari salah satu *Rule* merupakan *Conclusion* dari *Rule* yang lain, atau sebaliknya.
 - d. *Subsumed Rules* adalah suatu *Rule* yang mempunyai constraints yang lebih atau kurang tetapi mempunyai *Conclusion* yang sama.
 - e. *Unnecessary IF* conditions adalah kondisi ketika dua *Rule* atau lebih mempunyai *Conclusion* yang sama, tetapi salah satu dari *Rule* tersebut mempunyai *Premise* yang tidak perlu dikondisikan dalam *Rule* karena tidak mempunyai pengaruh apapun.
 - f. *Dead-end Rules* adalah suatu *Rule* yang konklusinya tidak diperlukan oleh *Rule-Rule* lainnya.
5. *Database Pakar* : Digunakan untuk mengembangkan basis pengetahuan apabila pakar ingin menambah, mengupdate, atau menghapus *Rule*. Penamaan *database* pakar sesuai dengan keinginan pakar.

Dalam sistem ini *database pakar* disimpan dalam file *fluBurung.mdb* yang terdiri dari:

- a. Tabel *Node* digunakan untuk menyimpan data parameter berdasarkan *Treeview* yang telah dibuat.
- b. Tabel *DtlNode* digunakan untuk menyimpan *Option-Option* dari tabel *node*.

- c. Tabel *Rule* digunakan untuk menyimpan data *Premis* dan *Conclusion* berdasarkan hasil reduksi.
6. *Interface User* : Sebagai media oleh *user* untuk melihat dan berinteraksi dengan sistem.
7. *Database User* : Digunakan untuk menyimpan data yang diinputkan oleh *User*. *Database user* disimpan dalam file *database.mdb* yang terdiri dari:
- Tabel *User* digunakan untuk menyimpan data yang diinputkan oleh *User* dari hasil konsultasi.
 - Tabel *Pasien* digunakan untuk menyimpan data pasien.
 - Tabel *Password* digunakan untuk menyimpan data *Login User*.
8. *Inference Engine* : Mekanisme inferensi yang digunakan adalah *Forward Chaining* karena sistem lebih dulu mengetahui fakta-fakta yang ada, kemudian mencari kesimpulan sementara sampai akhirnya berhenti setelah menghasilkan sebuah kesimpulan akhir (yaitu suatu diagnosa penyakit flu burung).
- Proses *Forward Chaining* diperlukan dalam mencari solusi berdasarkan *Goal* konsultasi dan *Rule Base* yang ada dalam *Working Memory*. *Goal* konsultasi pada proses *Forward Chaining* adalah hasil diagnosa penyakit flu burung.
9. *Knowledge Base* : Kumpulan fakta dan aturan (*Rule*) serta *Working Memory* yang merupakan hasil dari selama proses system berlangsung, dan aturan tentang diagnosa penyakit flu burung. *Knowledge Base* pada sistem ini, disimpan dalam tabel *Rule*.

10. *Output* : *Output* dari desain pakar adalah *Database* dengan nama tabel (tabel *Rule*) dan file text (KBS.txt) yang berisi himpunan aturan. Penamaan *File* text yang berisi himpunan aturan tidak harus KBS, terserah *User* pada saat proses penyimpanan aturan.

Output dari desain *User* adalah hasil akhir dari proses *Inference* yaitu hasil diagnosa penyakit flu burung dan pengobatannya serta berupa laporan dokter untuk mengetahui hasil pemeriksaan pasien yang menderita penyakit flu burung, dan laporan *Medical Record* untuk mengetahui status pasien.

3.3 Perancangan Proses

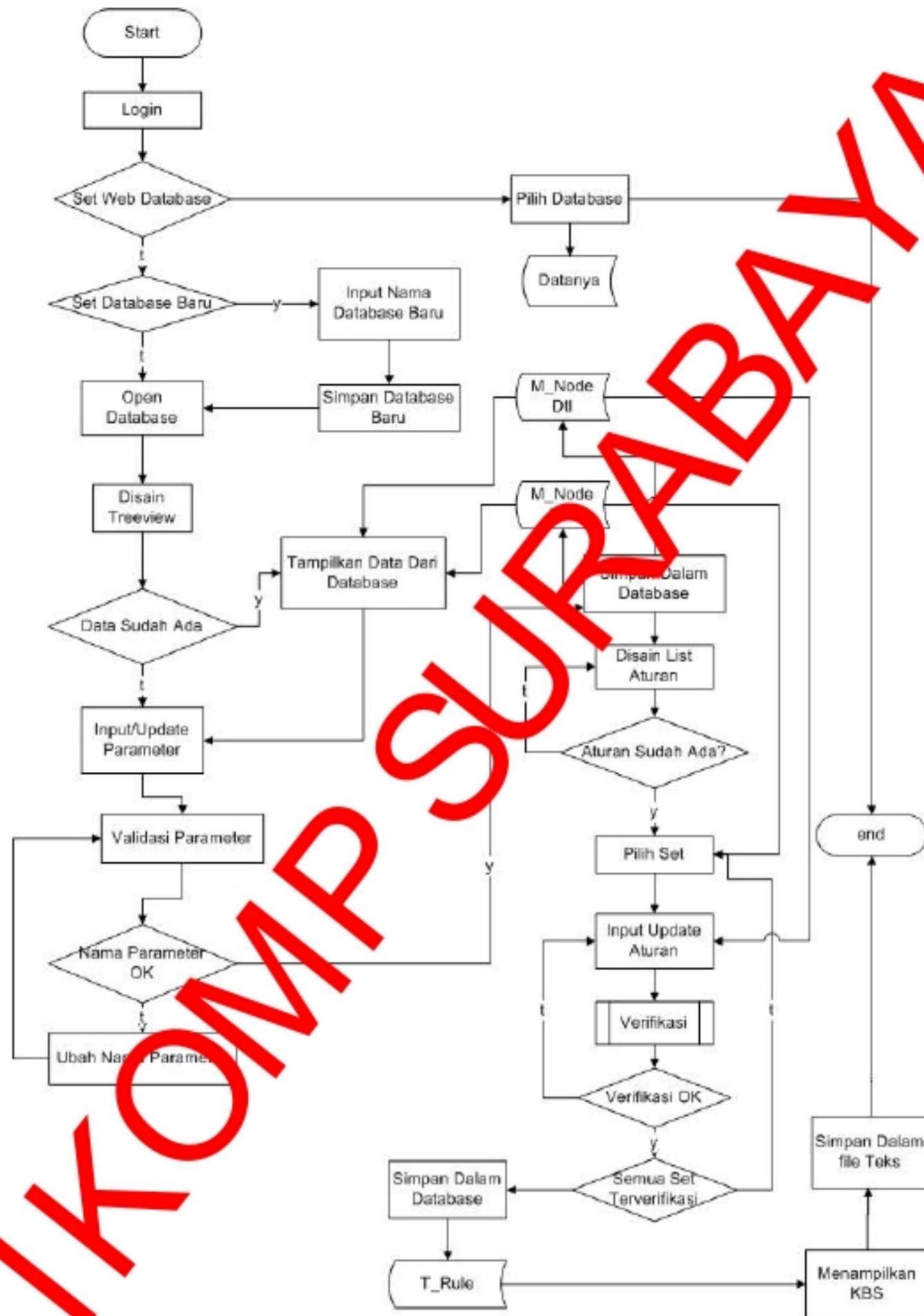
Perancangan proses dalam sistem ini dibagi menjadi dua yaitu perancangan proses untuk sistem secara *Online* dan perancangan proses untuk sistem secara *Offline*.

1. Perancangan proses sistem untuk pakar secara *Offline*
 - a. Diagram Alir Sistem untuk Desain Pakar.
 - b. Diagram Alir Sistem Proses Verifikasi.
 - c. Diagram Alir Sistem Proses *Inference Engine*.
2. Perancangan proses sistem untuk *User* secara *Online* adalah diagram alir untuk desain *User*.

3.3.1 Perancangan Proses Sistem secara Offline

A. Diagram Alir Sistem untuk Desain Pakar.

Pada Gambar 3.4 menjelaskan desain aturan penyakit flu burung oleh pakar untuk mendefinisikan aturan-aturan yang akan digunakan oleh *User* pada halaman *Web Page* untuk proses konsultasi penyakit flu burung.



Gambar 3.4 Diagram Alir Sistem Desain Pakar

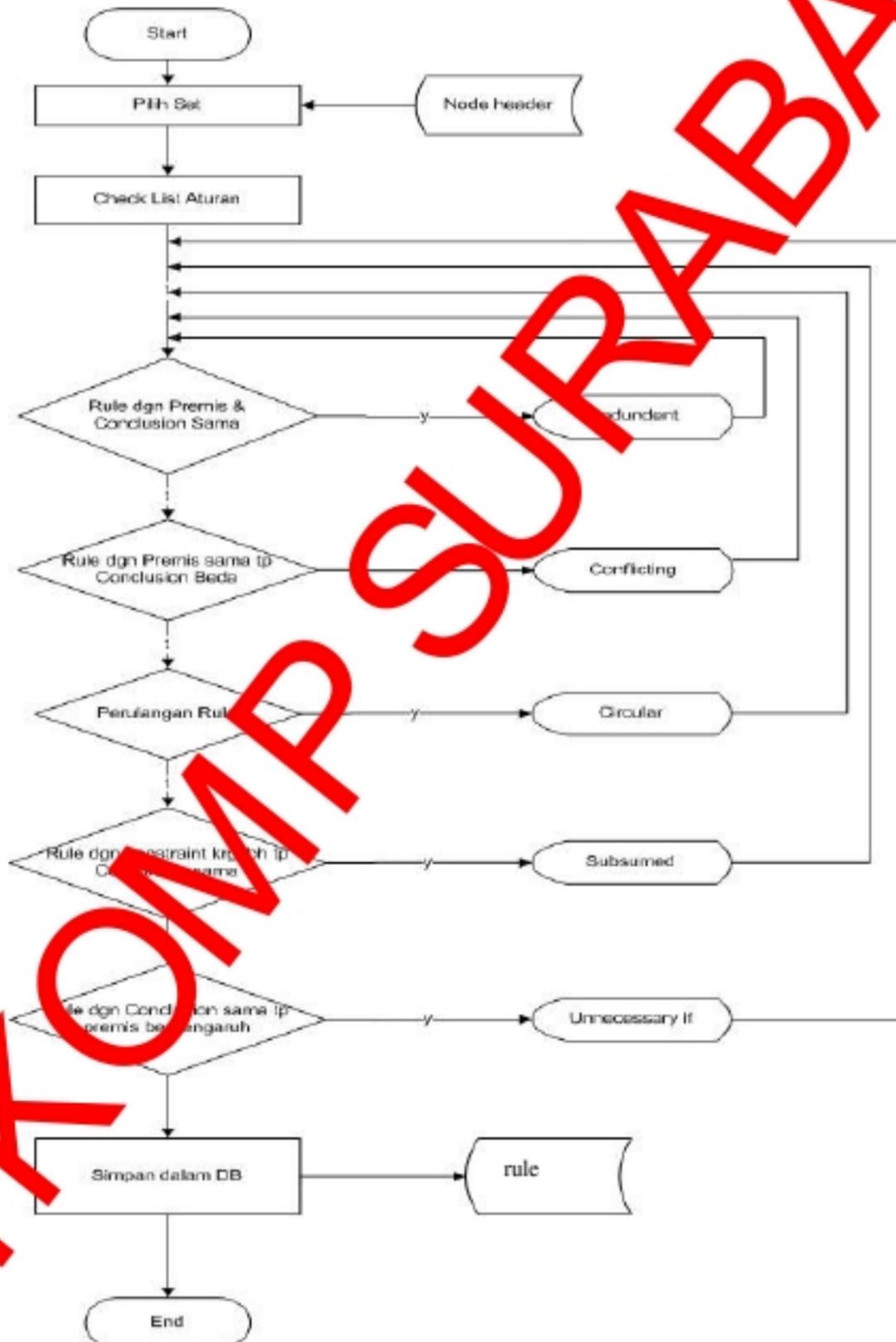
Diagram alir pada gambar 3.4 dijelaskan bahwa pakar harus login terlebih dahulu. Kemudian pakar membuat database baru mengenai penyakit flu burung. Setelah itu dilakukan proses *Open Database* untuk membuat desain *Treeview*. Saat membuat desain *Treeview*, pakar akan *Input* parameter yang digunakan sebagai data. Kemudian sistem akan melakukan validasi parameter agar tidak muncul parameter ganda. Setelah *Treeview* selesai dibuat maka pakar akan membuat *Rule* berdasarkan *Rule Set* yang telah dibuat pada *Treeview*. Setelah *Rule* tersebut dibuat maka sistem akan melakukan proses Verifikasi. Setelah Verifikasi sukses maka *Rule* tersebut ditampilkan dalam bentuk file KBS.

B. Diagram Alir Sistem Proses Verifikasi.

Pada Gambar 3.6 menjelaskan verifikasi yaitu proses untuk *Redundant Rules*, *Conflicting Rules*, *Subsumed Rules*, *Unnecessary IF conditions*, *Dead-end Rules* dan *Circular Rules* dengan penjelasan sebagai berikut:

- a. *Redundant Rules* adalah suatu *Rule* jika dua *Rule* atau lebih mempunyai *Premise* dan *Conclusion* yang sama.
- b. *Conflicting Rules* adalah ketika dua buah *Rule* atau lebih mempunyai *Premise* yang sama, tetapi mempunyai *Conclusion* yang berlawanan.
- c. *Subsumed Rules* Suatu *Rule* dapat dikatakan subsumed jika *Rule* tersebut mempunyai *Constraints* yang lebih atau kurang tetapi mempunyai *Conclusion* yang sama.
- d. *Circular Rules* ialah suatu keadaan dimana terjadinya proses perulangan dari suatu *Rule*. Ini dikarenakan suatu *Premise* dari salah satu *Rule* merupakan *Conclusion* dari *Rule* yang lain, atau sebaliknya.

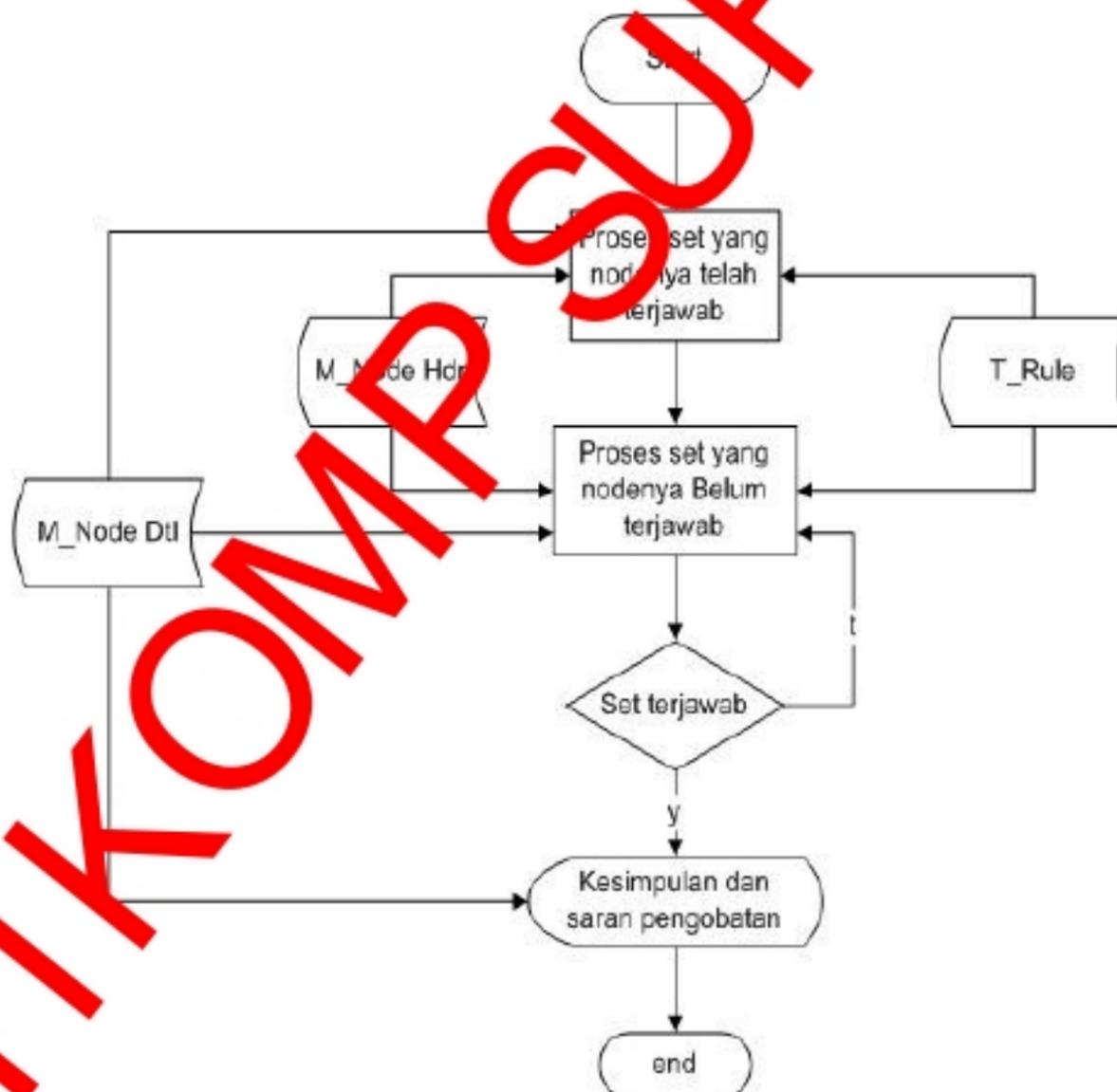
- e. *Unnecessary IF Conditions* adalah ketika dua *Rule* atau lebih mempunyai *Conclusion* yang sama, tetapi salah satu dari *Rule* tersebut mempunyai *Premise* yang tidak perlu dikondisikan dalam *Rule* karena tidak mempunyai pengaruh apapun.



Gambar 3.5 Diagram Alir Sistem Proses Verifikasi

C. Diagram Alir Sistem Proses *Inference Engine*

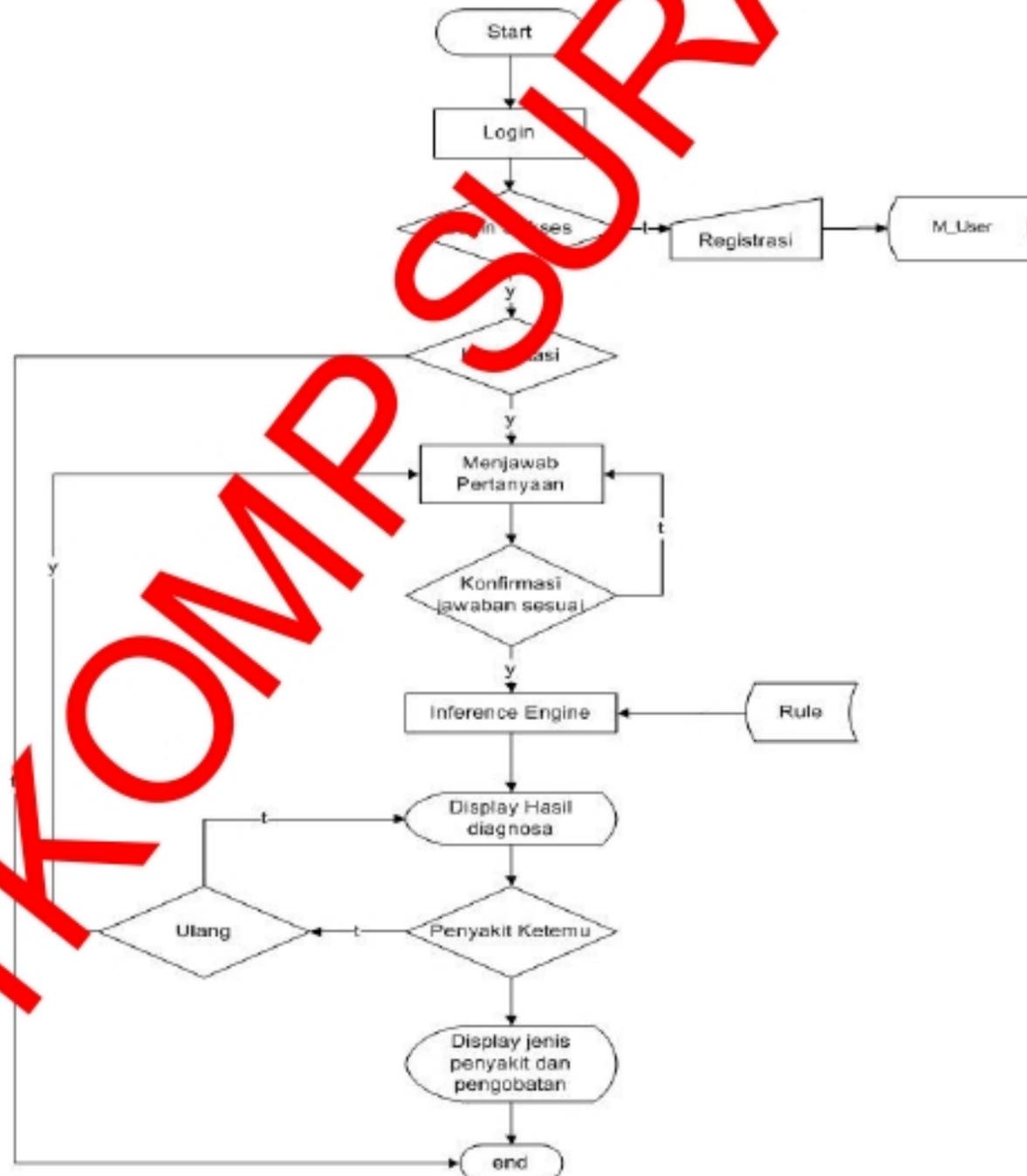
Pada Gambar 3.6 menjelaskan proses *Inference Engine* dengan menggunakan metode *Forward Chaining* yaitu penelusuran dari data-data yang ada untuk mencapai suatu *Conclusion*. Dari gambar 3.6 dijelaskan bahwa dari basis aturan yang disimpan pada tabel *M_NodeHdr*, tabel *M_NodeDtl*, dan tabel *T_Rule* dilakukan penelusuran, untuk set yang terjawab maka *Inference Engine* akan menghasilkan kesimpulan konsultasi berupa penyakit flu burung dan pengobatannya. Jika set tidak terjawab maka *Inference Engine* tidak menghasilkan suatu kesimpulan.



Gambar 3.6 Diagram Alir Sistem Proses *Inference Engine*

3.3.2 Perancangan Proses Sistem secara Online

Pada Gambar 3.7 menjelaskan proses jalannya sistem pada desain *User* dalam melakukan konsultasi tentang penyakit flu burung secara *Online*. Pada gambar 3.7 dijelaskan bahwa *User* akan diberi pertanyaan untuk menelusuri jenis penyakit yang diderita *User* berdasarkan data pada file KBS. Setelah semua pertanyaan terjawab maka sistem akan mengolah data *User* pada *Inference Engine*. Kemudian *User* akan menampilkan list jawaban *User* beserta penyakit yang diderita *User*, pengobatan dan pencegahannya.



Gambar 3.7 Diagram Alir Sistem Desain *User*

3.4 Struktur Tabel

Tabel-tabel yang digunakan pada sistem ini tersimpan dalam file umum.mdb untuk *Database User*, sedangkan database pakar tersimpan dalam file pakar.mdb. Tabel-tabel yang menyusun database tersebut adalah sebagai berikut:

1. Nama tabel : M_NodeHeader
- Primary Key : -
- Foreign Key : -
- Fungsi : Untuk menyimpan data parameter berdasarkan *Treeview* yang telah dibuat.

Tabel 3.5 M_NodeHeader

No	Field	Type	Length	Key	Keterangan
1	Index	Number	Long Integer		Index node
2	Teks	Text	255		Node
3	Pertanyaan	Memo	Long Integer		Parameter
4	Parent	Number	Long Integer		Parent node
5	Jml_child	Number	Long Integer		Jumlah child node
6	Set	Number	Long Integer		Set node
7	Level	Number	Long Integer		Level node
8	UpdatedDate	Date/Time	-		Waktu update
9	UpdatedBy	Text	10		Pelaku update

2. Nama tabel : M_NodeDetil
- Primary Key : -
- Foreign Key : -
- Fungsi : Untuk menyimpan jawaban hasil konsultasi serta pengobatannya

Tabel 3.6 M_NodeDetail

No	Field	Type	Length	Key	Keterangan
1	Index	Number	Long Integer		Index node
2	Jawaban	Text	255		Jawaban penyakit dan pengobatan
3	Putusan	Memo	-		Penyakit

3. Nama tabel : T_Rule

Primary Key : -

Foreign Key : Set referensi tabel M_NodeHeader (set)

Fungsi : Untuk menyimpan *Rule* yang telah terbentuk dari *Treeview* dan hasil *Verifikasi*

Tabel 3.7 M_Rule

No	Field	Type	Length	Key	Keterangan
1	Set	Number	Long Integer	FK	Set node
2	No	Number	Long Integer		
3	Rule	Memo	-		<i>Rule</i> dari <i>Treeview</i>

4. Nama tabel : M_User

Primary Key : -

Foreign Key : -

Fungsi : Untuk menyimpan data login *User*

Tabel 3.8 M_user

No	Field	Type	Length	Key	Keterangan
1	UserId	Number	Long Integer	PK	User Id
2	UserName	Text	50		User Name
3	Password	Text	50		Password
4	Nama	Text	50		Nama User
5	Email	Text	50		Email user
6	Telp	Text	50		No. Telp user

5. Nama tabel : M_keterangan

Primary Key : -

Foreign Key : -

Fungsi : Untuk menyimpan data keterangan untuk halaman

Web pada sistem Online

Tabel 3.9 M_keterangan

No	Field	Type	Length	Key	Keterangan
1	UserId	Number	Long Integer	PK	User Id
2	UserName	Text	50		User Name
3	Password	Text	50		Password
4	Nama	Text	50		Nama User
5	Email	Text	50		Email user

6. Nama tabel : C_Web

Primary Key : -

Foreign Key : -

Fungsi : Untuk menyimpan *Database* yang diset untuk

digunakan pada sistem *Online*

Tabel 3.10 M_Keterangan

No	Field	Type	Length	Key	Keterangan
1	DBWeb	Text	255		Database yang disediakan untuk web page

7. Nama tabel : H_konsultasi

Primary Key : -

Foreign Key : User id referensi tabel M_User (User)

Fungsi : Untuk menyimpan hasil konsultasi *User*

Tabel 3.11 M_Keterangan

No	Field	Type	Length	Key	Keterangan
1	UserId	Number	Long Integer	FK	
2	UserName	Text	255		
3	Detail	Memo	-		

3.5 Desain Input Output

Desain *Input Output* yang digunakan pada sistem ini dibuat agar dapat menggunakan *Mouse* dan *Keyboard* secara maksimal, karena pada dasarnya aplikasi ini berbasis *Windows* yang telah menggunakan *Mouse* dan *Keyboard* dalam mempermudah pengentrian data

Pada sistem ini membuat dua sistem yaitu online dan *Offline* dalam menampilkan form, perancangan dengan menggunakan konsep interaksi manusia dan komputer, sehingga seorang user hanya dengan melihat form akan mudah mengetahui apa yang dilakukan selanjutnya.

3.5.1 Sistem Offline

A. Login

Form ini berfungsi untuk login untuk masuk ke dalam sistem. Form login ini dapat dilihat pada gambar 3.8 dibawah.

Login	
User	<input type="text" value="pakar"/>
Password	<input type="text" value="pakar"/>
	<input type="button" value="Login"/> <input type="button" value="Cancel"/>

Gambar 3.8 Desain Form Login

B. Create Database

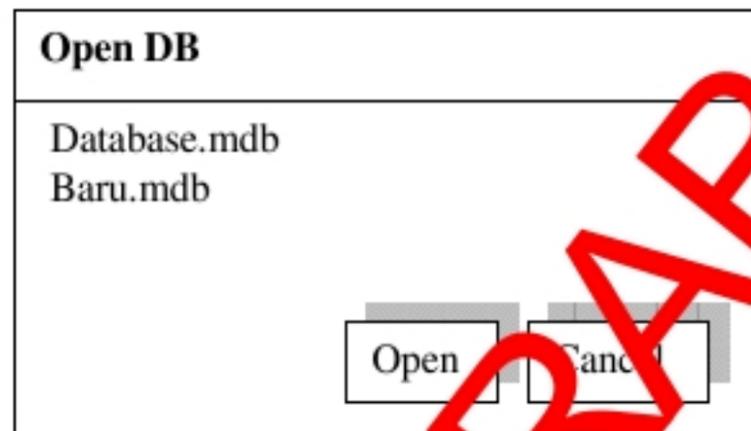
Form *Create Database* berfungsi untuk membuat database baru, sebagai langkah awal membuat *Treeview*. Digunakan apabila pakar membuat sistem berbasis aturan baru mengenai penyakit yang berbeda. Form buat *Database* dapat dilihat pada gambar 3.9.

Create DB	
Database.mdb	
	<input type="button" value="create"/> <input type="button" value="Cancel"/>

Gambar 3.9 Desain Form *Create Database*

C. Open Database

Form *Open Database* berfungsi untuk membuka *Database* yang telah dibuat atau *Database* yang telah ada. Form *Open Database* dapat dilihat pada gambar 3.10.



Gambar 3.10 Desain Form Open Database

D. Desain *Treeview*

Form desain berfungsi untuk membuat *Treeview* yang merupakan langkah awal membuat *Rule* untuk mengentrikan pertanyaan serta pengobatan untuk konsultasi halaman *Web*. Dalam pengisian *Treeview* ini harus disesuaikan dengan *dependency Diagram* yang telah dibuat sebelumnya. *Treeview* juga digunakan untuk menentukan hubungan antara faktor-faktor penting yang mempengaruhi dalam pemberian suatu rekomendasi penyakit. Di dalam *Treeview* juga berisi aturan-aturan dan jawaban yang digunakan untuk memudahkan pada saat proses Verifikasi. Form desain *Treeview* juga terdapat fasilitas untuk menambahkan dan mengurangi *Node* atau parameter, juga terdapat detail *Node* agar pakar lebih mudah dalam pengisiannya. Form desain *Treeview* dapat dilihat pada gambar 3.11.

Desain Treeview

Flu burung

- +----- Gejala Klinis
- +----- Anamnesis

Detail Node

Index : 1
Text : Gejala klinis
PERTANYAAN
Parent : Flu Burung
Jumlah Child : 2
Set : 2
Jawaban :

Pengobatan

Tambah Node

Hapus Node

Save

Exit

Gambar 3.11 Desain Form Desain Treeview

E. Verifikasi

Form Verifikasi berfungsi untuk melakukan proses Verifikasi *Rule* yang telah dibuat. Verifikasi *Rule* dilakukan agar tidak terjadi *Redundant Rules*, *Conflicting Rules*, *Subsumed Rules*. Form Verifikasi dapat dilihat pada gambar 3.12

Verifikasi

Set

Detail Rule			
1	Ya	Ya	Influenza
2	Ya	Tidak	Hay fever

Mail verifikasi

Tambah Baris

Hapus Baris

Verifikasi

Simpan

Keluar

Gambar 3.12 Desain Form Verifikasi

F. File KBS

File KBS adalah *Output* dari hasil Verifikasi yang berupa susunan *Rule-Rule*. Form file KBS dapat dilihat pada gambar 3.13 dibawah.

File KBS
Rule 1 IF Mata berair = Ya AND Mata merah = Tidak AND Mata gatal = Tidak THEN Mata = mata berair Rule 2 IF Mata berair = Tidak AND Mata merah = Ya AND Mata gatal = Tidak THEN Mata = Mata Merah

Gambar 3.13 Desain Form File KBS

G. Form Set Database

Form *Set Database* berfungsi untuk melakukan pengesetan *Database* yang akan diaktifkan pada sistem *Online*. *Database* yang akan diset harus sudah melewati proses rancangan aturan dan Verifikasi, karena apabila tidak maka *Database* yang diset tidak akan dapat digunakan. Gambar form set *Database* dapat dilihat pada gambar 3.14

Set DataBase
Database <input type="text" value="Flu Burung"/>
<input type="text" value="Database.mdb"/> <input type="text" value="Baru.mdb"/>
<input type="button" value="Set"/> <input type="button" value="Keluar"/>

Gambar 3.14 Desain Form *Set Database*

3.5.5 Sistem Online

A. Login Online

Form *login* ini berfungsi untuk login agar bisa masuk ke dalam halaman home pada sistem *Online*. Form *login Online* ini dapat dilihat pada gambar 3.15.

Login	
User	Pakar
Password	Pakar
<input type="button" value="Login"/> <input type="button" value="Cancel"/>	

Gambar 3.15 Desain Form *Login Online*

B. Hasil Konsultasi

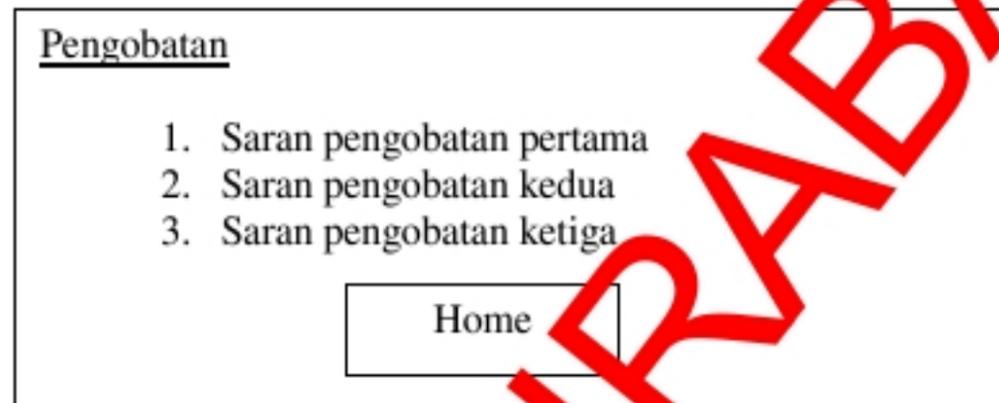
Form hasil konsultasi adalah *Output* dari konsultasi yang telah dilakukan, form ini berisi rangkuman dari hasil jawaban yang diberikan *User* kepada sistem dan jenis penyakit yang ditanyakan. Form hasil konsultasi ini dapat dilihat pada gambar 3.16.

Fakta-fakta	
5. Pertanyaan 1 = ya 6. Pertanyaan 2 = ya 7. Pertanyaan 3 = tidak 8. Pertanyaan 4 = ya	1. Pertanyaan 5 = tidak 2. Pertanyaan 6 = ya 3. Pertanyaan 7 = ya 4. Pertanyaan 8 = tidak
Kesimpulan : Suspect Flu Burung	
<input type="button" value="Pengobatan"/>	

Gambar 3.16 Desain Form Hasil Konsultasi

B. Pengobatan

Form pengobatan adalah *Output* dari form hasil konsultasi yaitu dengan mengklik tombol pengobatan maka akan tampil pengobatan berdasarkan penyakit flu burung yang diderita *User*. Form pengobatan ini dapat dilihat pada gambar 3.17. Untuk kembali ke menu utama maka klik tombol Home



Pengobatan

1. Saran pengobatan pertama
2. Saran pengobatan kedua
3. Saran pengobatan ketiga

Home

Gambar 3.17 Desain Form Pengobatan