

BAB III

PERANCANGAN SISTEM

Dalam bab ini dibahas tentang perancangan perangkat lunak dari sistem, meliputi perancangan *rules*, 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 Metode Penelitian

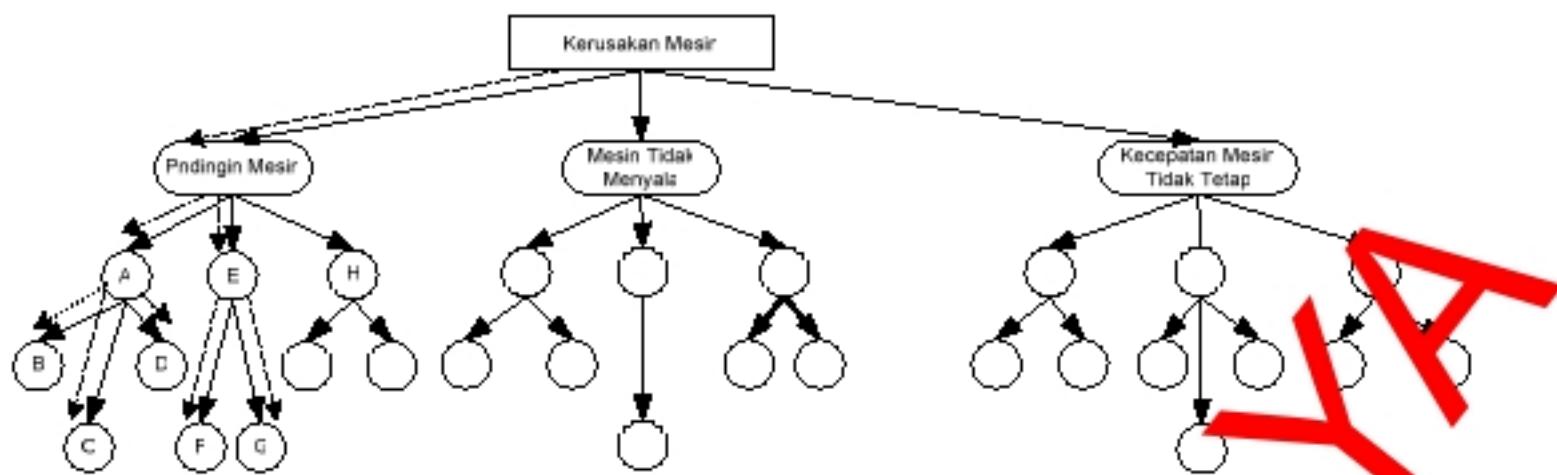
Metode penelitian yang diterapkan dalam penelitian ini menggunakan metode *Forward Chaining* (penelusuran maju) pada mesin inferensi. Alasan menggunakan metode *Forward Chaining* karena sistem pakar ini bekerja dimulai dengan mengajukan pertanyaan-pertanyaan untuk mendapatkan hasil akhir berupa pengecekan kerusakan mesin kapal.

3.1.1 Model yang digunakan

Penelitian yang dilakukan termasuk pada penelitian proyek (pengembangan). Struktur model sistem pakar yang dipakai sebagai dasar pengembangan proyek:

1. *Rule base*

Rule base dibuat dengan menggunakan pohon rule base, kemudian rule base dibangun dengan menggunakan metode *Forward Chaining*. Contoh *rule base* dapat dilihat pada Gambar 3.1.



Gambar 3.1. Pohon *Rule Base* Kerusakan Mesin

Dengan mengikuti arah panah putus-putus maka didapatkan *rule* untuk *rule base* sebagai berikut:

Rule 1 : IF sistem kerusakan = pendingin mesin THEN A

Rule 2 : IF sistem kerusakan = pendingin mesin THEN E

Rule 3 : IF sistem kerusakan = pendingin mesin THEN H

Rule 4 : IF A THEN B

Rule 5 : IF A THEN C

Rule 6 : IF A THEN D

Rule 7 : IF E THEN F

Rule 8 : IF E THEN G

dan seterusnya

Demikian selanjutnya setiap arah ditelusuri, sehingga didapatkan *rule* yang lain maka terbentuklah *rule base*.

2. Model pertanyaan dan jawaban

Model yang digunakan pada mesin inferensi untuk mencari solusi pada sistem perikar ini yaitu menggunakan model pertanyaan dan jawaban.

Sistem akan memberikan pertanyaan kepada pemakai sistem pakar, kemudian pemakai akan memberikan yang sesuai dengan kerusakan yang ada pada mesin kapal yang rusak, lalu mesin inferensi akan memproses jawaban yang diberikan oleh pemakai berdasarkan basis pengetahuan sampai diperoleh hasil akhir berupa pengecekan kerusakan mesin.

3.1.2 Prosedur Penelitian

Langkah-langkah prosedural yang ditempuh dalam membuat proyek, yaitu:

1. Studi pustaka

Dilakukan dengan cara mengumpulkan segala macam informasi secara riset kepustakaan dan mempelajari buku-buku yang berhubungan dengan masalah yang dihadapi, diantaranya: teori trouble shooting, mengenal *Artifical Intelligence* (AI), pemrograman Visual Basic 6.0.

2. Survai

Melakukan wawancara secara langsung untuk mendapatkan data-data yang diperlukan. Data-data tersebut adalah sistem mesin kapal, jenis kerusakan, dan ciri kerusakan mesin kapal.

3. Perancangan sistem

Melakukan perancangan sistem berdasarkan analisa dari hasil studi literatur dan observasi yang meliputi pembuatan diagram alir sistem untuk pakar, user, verifikasi dan *Inference Engine* (Mesin Inferensi).

4. Pembuatan program

Melakukan implementasi terhadap sistem berdasarkan hasil perancangan sistem dengan menggunakan bahasa pemrograman *Visual Basic 6.0*.

5. Evaluasi

Melakukan pengujian terhadap program pada sistem serta mencari kesalahan dari program yang mungkin terjadi sehingga dapat segera diperbaiki.

3.1.3 Evaluasi

Bagian evaluasi ini berkaitan dengan uraian tentang rencana uji coba terhadap produk yang telah dibuat. Oleh karena itu, bagian ini memuat:

a. Desain Uji Coba dan Subjek Coba

Dalam penelitian ini, uji coba produk dilakukan sebagian pada uji kelompok kecil. Yang menjadi populasi atau sampel dalam penelitian ini adalah beberapa orang yang tergabung menjadi kelompok kecil. Produk dari proyek ini akan diujicobakan dan diterapkan kepada PT PAL.

b. Jenis Data dan Instrumen Pengumpulan Data

Jenis data yang diteliti akan dikumpulkan dalam penelitian ini yaitu sistem mesin kapal, jenis kerusakan, ciri kerusakan dan pengecekan kerusakan mesin kapal.

Instrumen pengumpulan data yang digunakan dalam penelitian ini yaitu mempelajari *literature* dan melakuakan penelitian melalui wawancara.

b.1 Jenis data atau variable-variabel yang diteliti dalam penelitian ini adalah sebagai berikut:

1. Sistem-sistem kerusakan mesin

Mengumpulkan data-data untuk sistem-sistem kerusakan mesin kapal dengan cara mencari *literature* yang berhubungan dengan kerusakan mesin dan dengan wawancara langsung kepada pakar atau mekanik.

2. Bagian-bagian kerusakan mesin

Mengumpulkan data-data untuk bagian kerusakan mesin kapal dengan cara mencari *literature* yang berhubungan dengan kerusakan mesin dan dengan wawancara langsung kepada pakar atau mekanik.

3. Jenis-jenis kerusakan mesin

Mengumpulkan data-data untuk jenis-jenis kerusakan mesin kapal dengan cara mencari *literature* yang berhubungan dengan kerusakan mesin dan dengan wawancara langsung kepada pakar atau mekanik.

4. Ciri-ciri kerusakan mesin

Mengumpulkan data-data untuk ciri-ciri kerusakan mesin kapal dengan cara mencari *literature* yang berhubungan dengan kerusakan mesin dan dengan wawancara langsung kepada pakar atau mekanik.

b.2 Instrumen yang akan digunakan pada saat pengumpulan data pada penelitian ini yaitu:

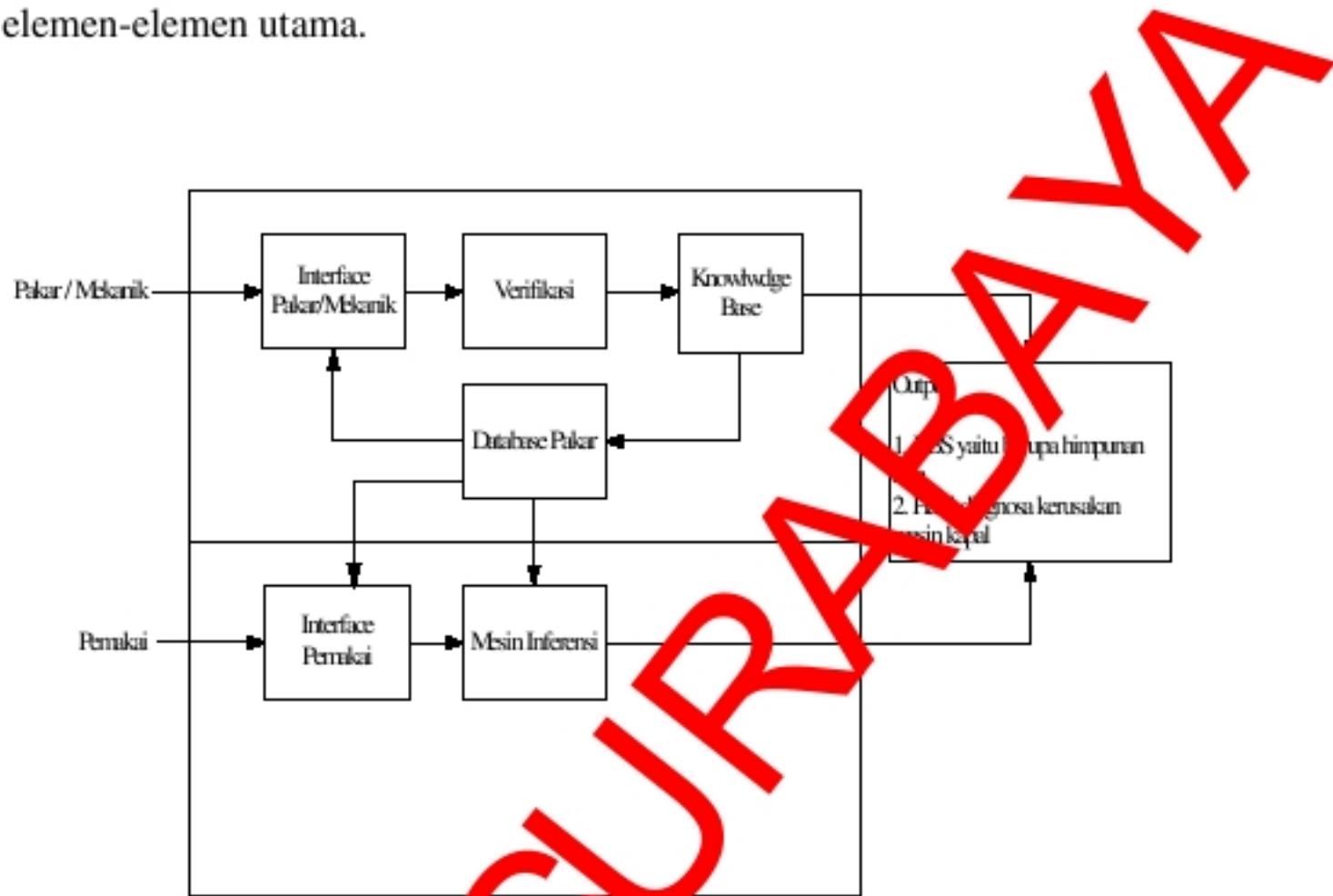
1. Mempelajari *literature* yang berhubungan dengan materi yang akan dibahas.
2. Melakukan penelitian dan mengumpulkan data melalui wawancara, yaitu dengan bertemu langsung kepada pakar atau mekanik mesin kapal.

c. Analisis Hasil Uji Coba

Produk yang telah dibuat dapat diterapkan kepada PT PAL Indonesia. Pada penelitian ini penelusuran dimulai dari mengajukan pertanyaan-pertanyaan sampai dengan penyelesaian akhir.

3.2 Desain Arsitektur

Desain arsitektur seperti terlihat pada Gambar 3.2, menggambarkan hubungan antara elemen-elemen utama.



Gambar 3.2. Desain Arsitektur Sistem Pakar

Berikut adalah penjelasan dari Gambar 3.2 :

a. *Interface Pakar / Mekanik*

Interface Pakar / Mekanik adalah suatu media untuk para pakar menginputkan parameter berupa aturan-aturan(*rule*).

b. *Verifikasi*

Verifikasi merupakan bagian yang digunakan untuk melakukan validasi agar tidak terjadi *redundant rules*, *conflicting rules* dan *subsumed rules*.

c. *Database Pakar*

Database digunakan untuk mengembangkan basis pengetahuan apabila ingin menambah, mengedit, menghapus *rule*, dan menyimpan identitas user.

d. *Knowledge Base*

Knowledge Base merupakan kumpulan dari fakta dan aturan (*rule*) serta *working memory* yang merupakan fakta yang diperoleh oleh sistem selama proses berlangsung, yaitu aturan tentang permasalahan-permasalahan yang telah disusain oleh pakar.

e. *Interface Pemakai*

Interface Pemakai sebagai media oleh pemakai (*user*) untuk berinteraksi dengan sistem, yaitu dengan menginputkan fakta-fakta untuk memperoleh sebuah kesimpulan.

f. *Inference Engine*

Inference Engine yang digunakan adalah *Forward Chaining* karena sistem lebih dulu menggunakan fakta-fakta yang ada, kemudian mencari kesimpulan akhir berupa pengecekan kerusakan mesin kapal.

g. Output

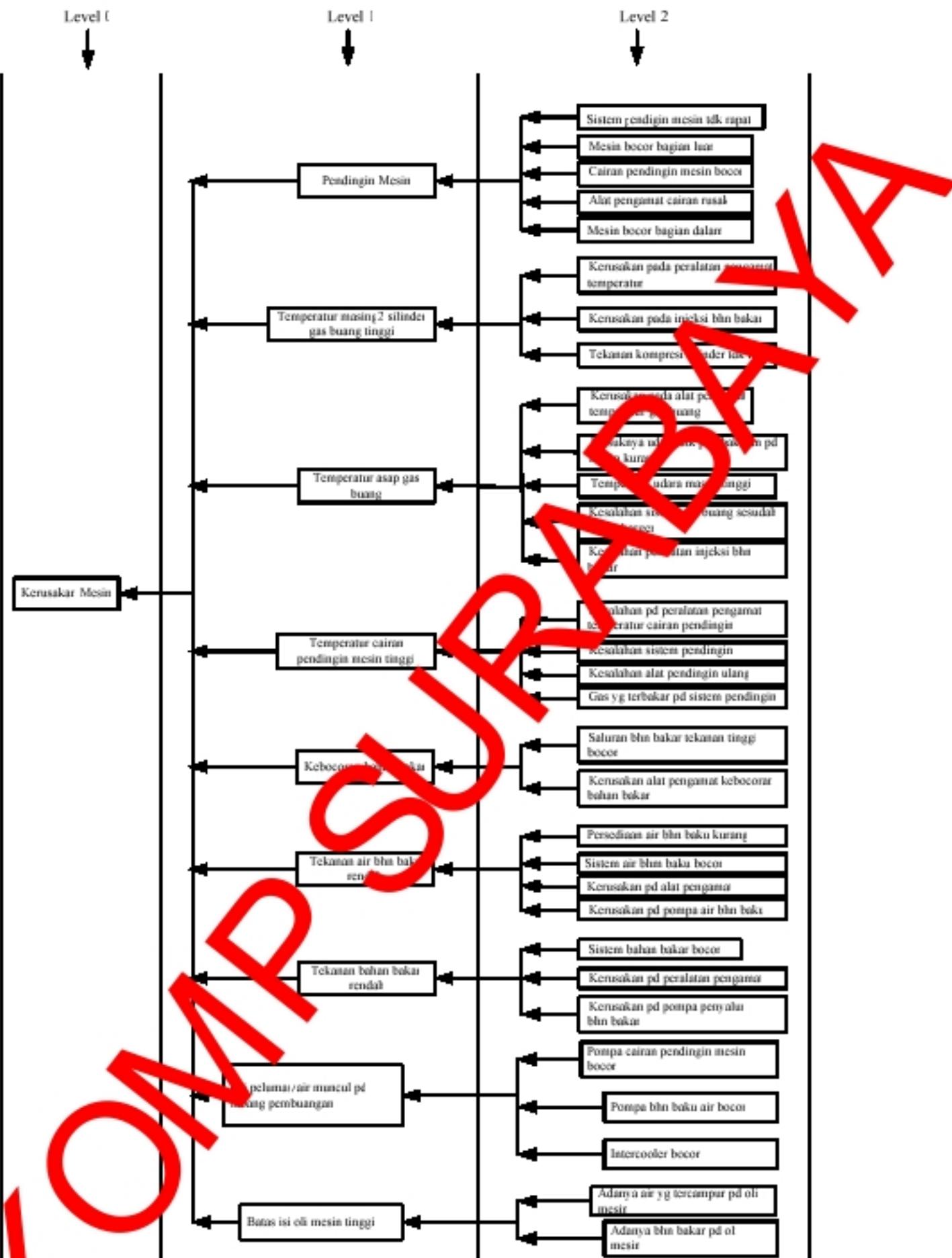
Output dari program akan menunjukkan jawaban dari fakta-fakta yang telah diinputkan oleh pengguna (*user*) berupa ciri-ciri kerusakan mesin kapal.

Data-data yang dimiliki atau dimiliki oleh seorang pakar atau beberapa orang pakar, diverifikasi lalu di definisikan

3.3 Perancangan Aturan Kerusakan Mesin

3.3.1 Perancangan Block Diagram

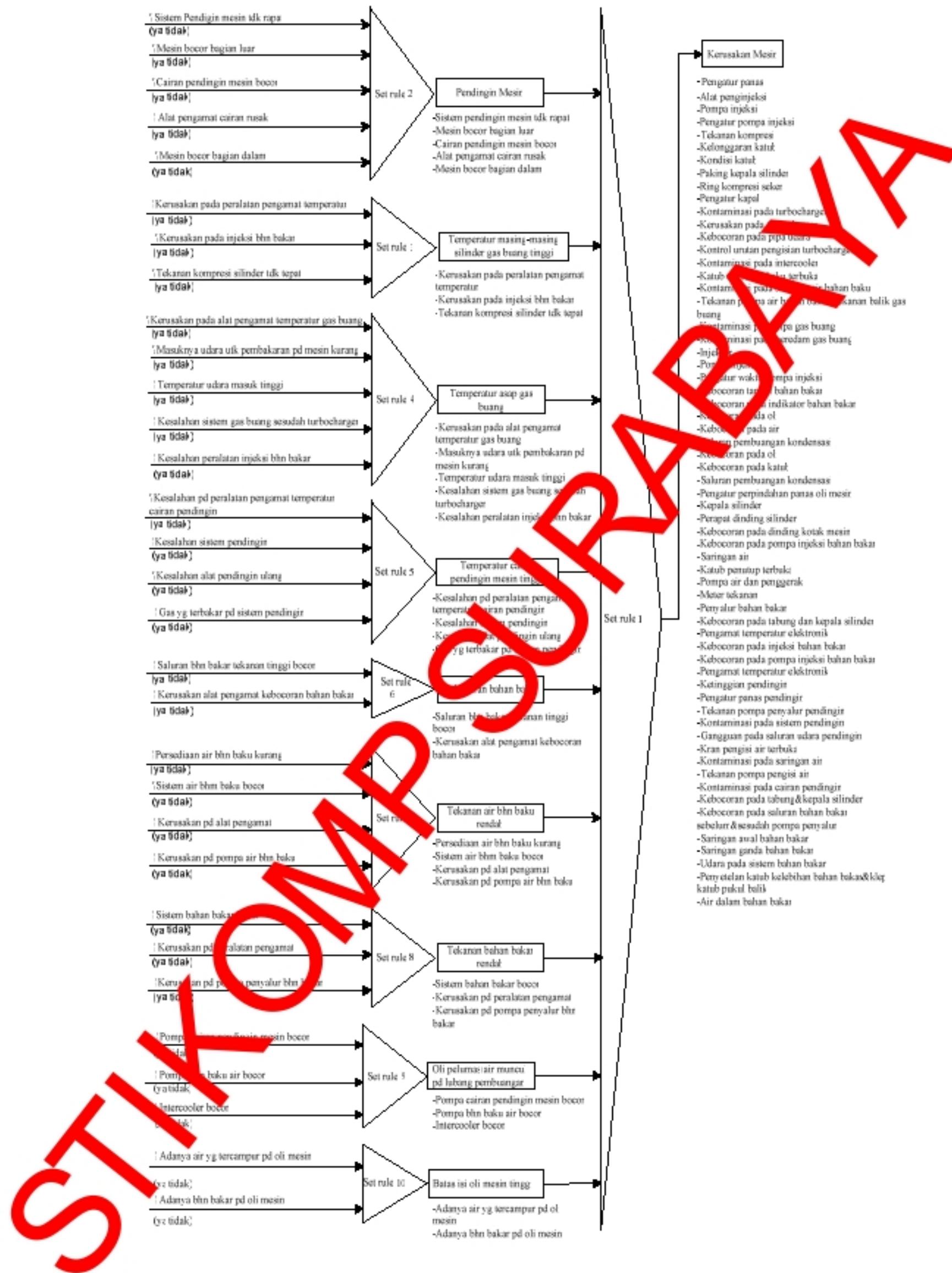
Block diagram sistem pakar untuk mendeteksi jenis kerusakan mesin kapal dapat dilihat pada Gambar 3.3.



Gambar 3.3. *Block Diagram Kerusakan Mesin*

3.3. Perancangan *Dependency Diagram*

Dependency diagram sistem pakar untuk mendeteksi jenis kerusakan mesin kapal dapat dilihat pada Gambar 3.4.



Gambar 3.4. Dependency Diagram Kerusakan Mesin

3.3.3 Perancangan *Decision Table*

Decision table dibuat untuk rekomendasi akhir *Knowledge Base System*. Pada Tabel 3.1 berikut menunjukkan salah satu contoh perancangan *decision table* untuk rule set 9 yaitu oli pelumas atau air muncul pada lubang pembuangan berdasarkan pada perancangan *dependency diagram*.

Tabel 3.1 *Decision Table Rule set 9*

Rule	Pompa cairan pendingin mesin bocor	Pompa bahan baku air bocor	Intercooler bocor	oli pelumas/air muncul pada lubang pembuangan
A1	Y	Y	Y	-
A2	Y	Y	T	-
A3	Y	T	Y	-
A4	Y	T	T	Pompa cairan pendingin mesin bocor
A5	T	Y	Y	-
A6	T	Y	T	Pompa bahan baku air bocor
A7	T		Y	Intercooler Bocor
A8	T	T	T	-

Dalam Tabel 3.1, rancangan *decision table* adalah untuk rangkaian aturan akhir yang terkait dengan dua kondisi, yang masing-masing dapat memiliki sejumlah nilai yang berbeda. Pompa cairan pendingin mesin bocor, kondisi pertama yang hanya memiliki dua nilai: apakah Ya atau Tidak. Begitu pula Pompa bahan baku air bocor, kondisi kedua juga memiliki dua nilai yaitu Ya atau Tidak dan untuk Intercooler bocor juga memiliki dua nilai yaitu Ya atau Tidak.

3.3.4 Perancangan Tabel Reduksi

Pada sistem ini proses perancangan reduksi untuk setiap *decision table* dilakukan secara manual. Perancangan reduksi berdasarkan *decision table* pada Tabel 3.1 menghasilkan parameter seperti pada Tabel 3.2 berikut:

Tabel 3.2 Reduksi *Decision Table Rule set 9*

Rule	Pompa cairan pendingin mesin bocor	Pompa bahan baku air bocor	Intercooler bocor	oli muncul pada lubang pembuangan
B1	Y	T	T	Pompa cairan pendingin mesin bocor
B2	T	Y	T	Pompa bahan baku air bocor
B3	T	T	Y	Intercooler bocor

3.4 Perancangan Sistem

Pada sub bab ini akan dibahas mengenai perancangan sistem untuk mendiagnosa kerusakan mesin kapal yang diimplementasikan dengan menggunakan bahasa pemrograman *Visual Basic 6.0*.

Sebelum sistem pakar ini mulai dirancang, maka tahap pertama yang harus dilakukan adalah memperoleh pengetahuan tentang semua kerusakan mesin kapal. Pengumpulan pengetahuan dilakukan dengan cara melakukan wawancara secara langsung dan membaca buku *literature* yang sesuai dengan bidang masalah.

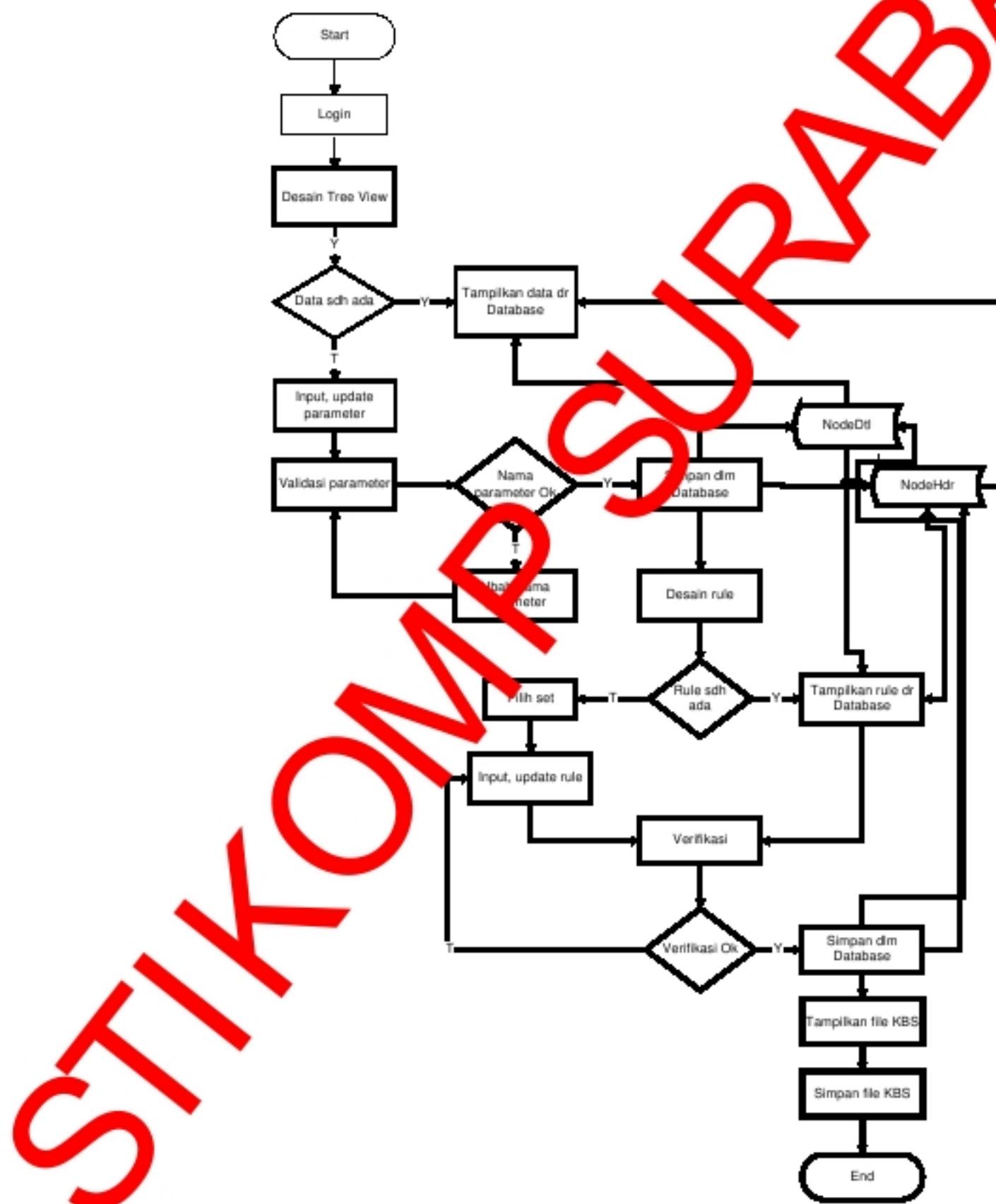
Dalam merancang suatu sistem pakar diperlukan beberapa bagian penting agar pakar dapat bekerja dengan baik, yaitu:

1. Diagram Alir Sistem untuk Pakar.
2. Diagram Alir Sistem untuk *User*.
3. Diagram Alir Sistem untuk Verifikasi.
4. Diagram Alir untuk *Inference Engine* (Mesin Inferensi).

Dengan melaksanakan tahapan tersebut dapat mempermudah dalam pembuatan sistem pakar untuk mendiagnosa kerusakan mesin kapal.

A. Diagram Alir Sistem untuk Pakar

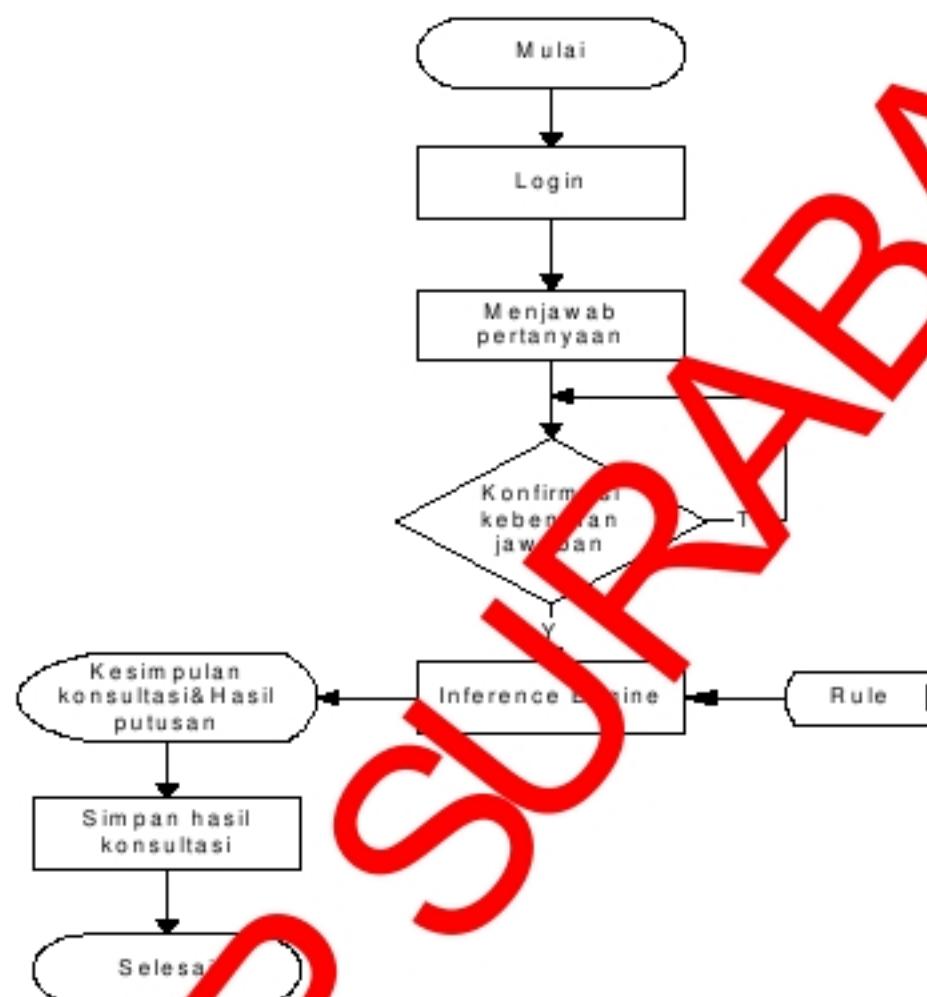
Pada Gambar 3.5 menjelaskan tentang diagram alir sistem untuk desain pakar, memiliki fungsi untuk melakukan proses memasukkan data-data baru mengenai aturan-aturan kerusakan mesin, merubah dan menambahkan data yang telah ada dan juga mengenai proses verifikasi aturan sehingga memberikan aturan dengan benar dan valid.



Gambar 3.5 Diagram Alir Sistem untuk Pakar

B. Diagram Alir Sistem untuk *User*

Pada gambar 3.6 akan digambarkan diagram alir untuk *user*, berfungsi untuk melakukan proses konsultasi hingga menghasilkan suatu output yang benar sesuai dengan *rule* yang ada.



Gambar 3.6 Diagram Alir Sistem untuk *User*

C. Diagram Alir Sistem untuk Verifikasi

Pada Gambar 3.7 menjelaskan proses verifikasi untuk proses *Redundant Rule*, *Conflicting Rule*, dan *Subsumed Rule*. Proses diawali dengan pengecekan list *rule* atau aturan. Pengecekan pertama adalah pengecekan untuk *Redundant*, apakah pada suatu *rule* premis dan *conclusion* ada yang sama. Jika ada akan diperiksa dan ditampilkan pada *display Redundant* dan proses akan dilanjutkan untuk pengecekan selanjutnya. Pengecekan yang kedua adalah pengecekan untuk *Conflicting*, apakah pada suatu *rule* premis sama tetapi *conclusionnya* berlawanan. Jika ada akan

diperiksa dan ditampilkan pada *display Conflicting* dan proses akan dilanjutkan untuk pengecekan selanjutnya. Pengecekan yang ketiga adalah pengecekan untuk *Subsumed*, apakah ada *rule* tersebut mempunyai *constraints* yang lebih atau kurang tetapi mempunyai *conclusion* yang sama. Jika ada akan diperiksa dan ditampilkan pada *display Subsumed*.

1. Redundant rules

Redundant rules terjadi jika ada dua *rule* atau lebih mempunyai *premise* dan *conclusion* yang sama.

Contoh:

- | | |
|-------------|---|
| Rule 1 : If | Batas atau level oli mesin rendah And
Kebocoran mesin bagian luar |
| Then | Pemipaan bagian luar, saluran pembuangan oli,
penyangga peralatan bagian luar, kebocoran pada
kotak mesin dan kotak oli |
| Rule 2 : If | Batas atau level oli mesin rendah And
Kebocoran mesin bagian luar |
| Then | Pemipaan bagian luar, saluran pembuangan oli,
penyangga peralatan bagian luar, kebocoran pada
kotak mesin dan kotak oli |

2. Conflicting rules

Conflicting rules terjadi ketika dua buah *rule* atau lebih mempunyai *premise* yang sama, tetapi mempunyai *conclusion* yang berlawanan.

Contoh:

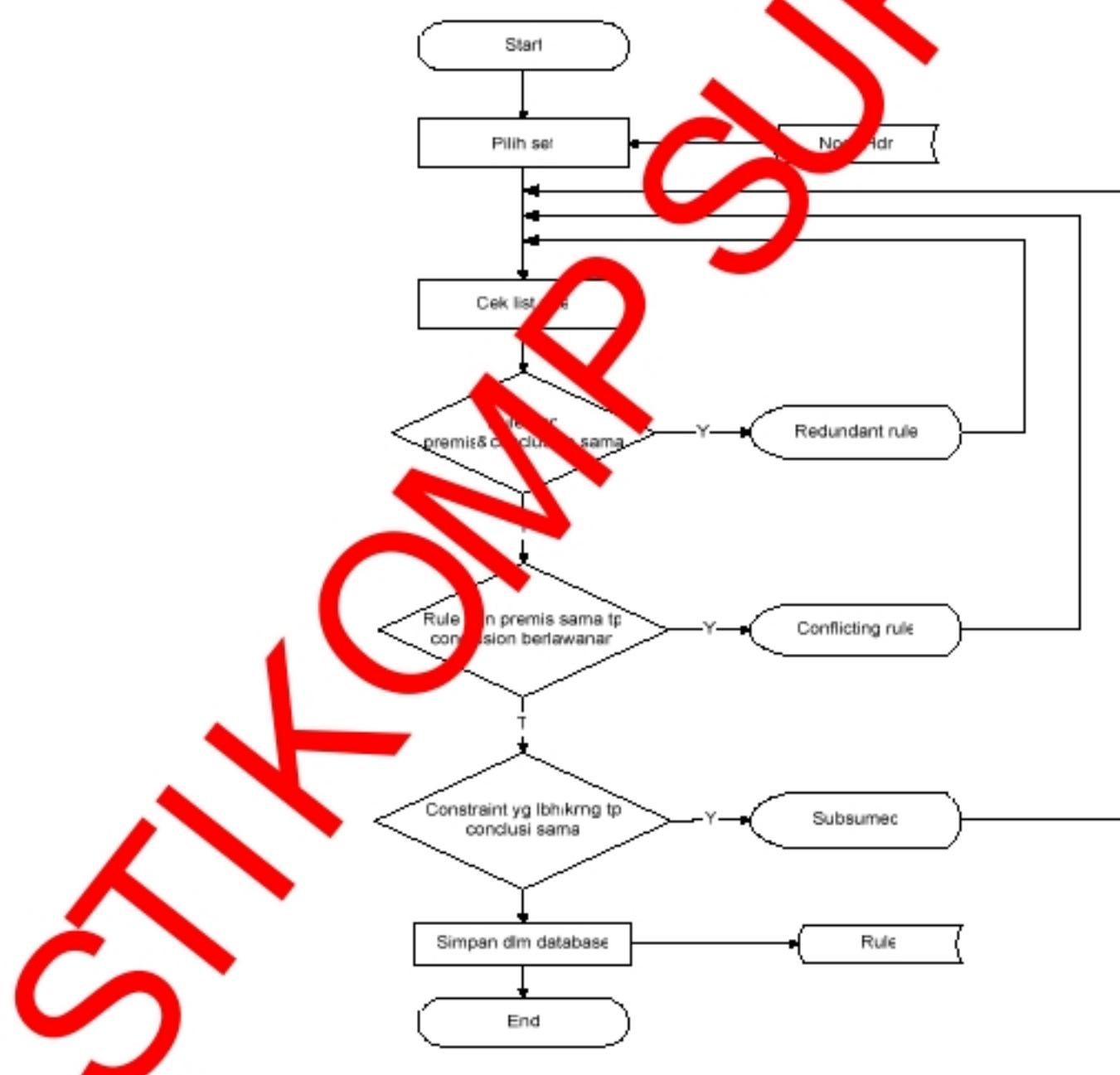
- | | |
|-------------|---|
| Rule 1 : If | Batas atau level oli mesin rendah And
Kebocoran mesin bagian luar |
| Then | Pemipaan bagian luar, saluran pembuangan oli,
penyangga peralatan bagian luar, kebocoran pada
kotak mesin dan kotak oli |
| Rule 2 : If | Batas atau level oli mesin rendah And
Kebocoran mesin bagian luar |
| Then | Pengarah katub, tabung silinder, ring kontrol oli dan
ring kompresi |

3. Subsumed rules

Subsumed rules terjadi jika dua *rule* tersebut mempunyai *constraints* yang lebih atau kurang tetapi mempunyai *conclusion* yang sama.

Contoh:

- Rule 1 : If Batas atau level oli mesin rendah And Kebocoran mesin bagian luar
 Then Pemipaan bagian luar, saluran pembuangan oli, penyangga peralatan bagian luar, kebocoran pada kotak mesin dan kotak oli
- Rule 2 : If Batas atau level oli mesin rendah
 Then Pemipaan bagian luar, saluran pembuangan oli, penyangga peralatan bagian luar, kebocoran pada kotak mesin dan kotak oli



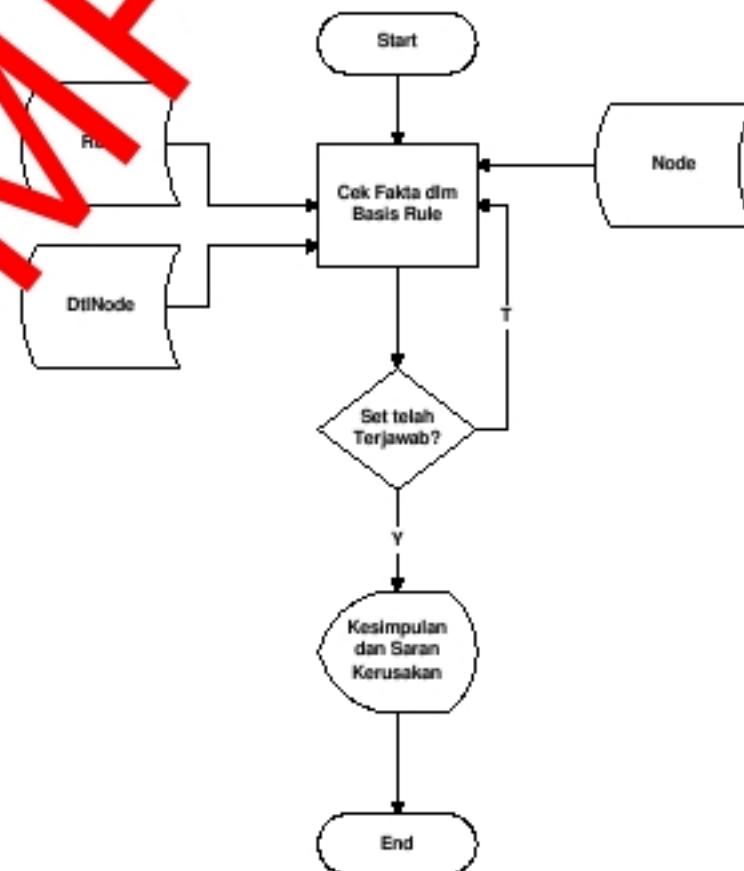
Gambar 3.7 Diagram Alir Sistem Proses Verifikasi

D. Diagram Alir untuk *Inference Engine* (Mesin Inferensi)

Tugas dari *Inference Engine* adalah melakukan proses penelusuran untuk menentukan jawaban yang tepat dengan menggunakan metode pengambilan keputusan yang telah digunakan. *Inference Engine* akan menerima respon data yang didapat dari user, kemudian melakukan proses terhadap basis pengetahuan yang dimiliki.

Pada sistem pakar yang dikembangkan disini akan dibuat pencarian arah maju (*Forward Chaining*). Hal ini dilakukan karena melihat bahwa jumlah gejala-gejala kerusakan (fakta) jauh lebih banyak dibandingkan jumlah bagian kerusakan (kesimpulan).

Pada Gambar 3.8 menjelaskan proses *Inference Engine* dengan menggunakan metode *forward chaining* yaitu penelusuran dari data-data yang ada untuk mencapai suatu konklusi.



Gambar 3.8 Diagram Alir Sistem Proses *Inference Engine*

3.5 Struktur Tabel Data Base

Struktur Tabel merupakan penjabaran dan penjelasan Tabel tersebut, dari fungsi masing-masing tabel sampai masing-masing *field* yang ada dalam tabel. Adapun struktur tabel adalah sebagai berikut:

1. Nama : M_User

Primary Key : UserID

Foreign Key : -

Fungsi : Digunakan menyimpan data *login user* dan *login pakar*.

Tabel 3.3 M_User

Field	Key	Tipe	Lebar	Keterangan
UserID	PK	Integer	10	Kode User
Nama		Text	50	Nama User
Password		Text	50	Password User
Lvl		Text	25	Hak akses User

2. Nama : T_Mesin_Rule

Primary Key : Set

Foreign Key : -

Fungsi : Digunakan menyimpan data premis dan konklusi dari tabel berdasarkan reduksi.

Tabel 3.4 T_Mesin_Rule

Field	Key	Tipe	Lebar	Keterangan
Set	PK	Number	Long Int	Set Node
No		Number	Long Int	No Rule
Rule		Memo		Keterangan Rule

3. Nama : M_Mesin_NodeHdr

Primary Key : Indeks

Foreign Key : Set referensi tabel T_Mesin_Rule (Set)

Fungsi : Digunakan menyimpan data parameter berdasarkan *Tree View* yang telah dibuat.

Tabel 3.5 M_Mesin_NodeHdr

Field	Key	Tipe	Lebar	Keterangan
Indeks	PK	Number	Long Int	Indeks Node
Teks		Text	225	Parameter
Pertanyaan		Memo		Pertanyaan
Parent		Number	Long Int	Parent Node
JmlChild		Number	Long Int	Jml Child Node
Set	FK	Number	Long Int	Set Node
Lvl		Number	Long Int	Level Node

4. Nama : M_Mesin_NodeDtl

Primary Key : -

Foreign Key : Indeks referensi tabel M_Mesin_Node (Indeks)

Fungsi : Digunakan menyimpan option option dari tabel node.

Tabel 3.6 : M_Mesin_NodeDtl

Field	Key	Tipe	Lebar	Keterangan
Indeks	FK	Number	Long Int	Indeks Node
Jawaban		Text	225	Option Jawaban
Putusan		Memo		Keterangan Putusan

3.6 Desain Input Output

Desain *Input Output* yang dibuat berfungsi untuk memudahkan user maupun pakar dalam penggunaan sistem. Form-form yang dirancang meliputi konsep manusia dengan komputer dimana seorang user hanya dengan melihat form, dapat mengerti langkah apa yang akan dilakukan selanjutnya.

1. Login

Form login pada Gambar 3.9 berfungsi untuk masuk ke dalam sistem, pada form login ini menentukan siapa yang melakukan login, apakah pakar atau user.

The form is titled "LOGIN". It contains two text input fields: "Username : Pakar" and "Password : ****". Below the fields are two buttons: "Login" and "Cancel".

Gambar 3.9 Form Login

2. Desain *Tree View*

Form Desain *Tree View* pada Gambar 3.10 berfungsi untuk membuat *tree view* yang merupakan langkah awal membuat aplikasi.

The form is titled "Penentuan Tree View Kerusakan Mesin Kapal". It features a tree view on the left with nodes: "Sistem Mesin" (with "Pendingin mesin" and "Sistem pendingin mesin tlk rapot"), "Mesin bocor bagian luar", and "Temperatur masing2 sistem gas buang tinggi". On the right, there is a section for a question: "Pertanyaan: Mesin bocor bagian luar?" with "Apakah Mesin bocor bagian luar?". Below it, status information is shown: "Parent: Pendingin mesin", "Jml Child: 0", and "Set: 0". A "Jawaban:" section includes a text input field ("tidak") and a checkbox ("ya"). At the bottom are buttons: "Tambah Node", "Hapus Node", "Simpan", and "Keluar".

Gambar 3.10 Form Penentuan *Tree View*

STIKOM SURABAYA

3. Verifikasi

Form Verifikasi berfungsi untuk melakukan proses verifikasi *rule* yang telah dibuat. Verifikasi rule dilakukan agar rule terbebas dari *Redundant rules*, *Conflicting rules*, dan *Subsumed rules*. Form verifikasi dapat dilihat pada Gambar 3.11.

The screenshot shows a window titled 'Verifikasi Rule'. At the top left is a dropdown menu labeled 'Sel' with the value '2 - Pendingin Mesin' and a '...' button. Below this is a section titled 'Detail Rule' containing a table:

	Sistem Pendingin Mesir	Mesin Bocor bag Luar	Pendingin Mesir
1	Ya	Tidak	Sistem Pendingin Mesir
2	Tidak	Ya	Mesin Bocor
3	Tidak	Tidak	Tdk ada Kerusakan

Below the table is a large empty rectangular area. At the bottom are five buttons: 'Tambah Baris', 'Hapus Baris', 'Verifikasi' (highlighted in red), 'Simpar', and 'Keluar'.

Gambar 3.11 Form Verifikasi

4. File KBS

File KBS berisi susunan rule-rule hasil dari verifikasi. Form KBS dapat dilihat pada Gambar 3.12.

The screenshot shows a window titled 'Hasil Verifikasi'. It displays three verified rules:

```

Rule 1
IF (Sistem pendingin mesir = ya) AND (Mesin bocor bag luar = tidak)
THEN Pendingin mesir = Sistem pendingin mesir

Rule 2
IF (Sistem pendingin mesir = tidak) AND (Mesin bocor bag luar = ya)
THEN Pendingin mesir = Mesin bocor bag luar

Rule 3
IF (Sistem pendingin mesir = tidak) AND (Mesin bocor bag luar = tidak)
THEN Pendingin mesir = tidak ada kerusakan
  
```

At the bottom are two buttons: 'Simpar' and 'Keluar'.

Gambar 3.12 Form KBS

5. Diagnosa Kerusakan Mesin Kapal Laut

Form Diagnosa Kerusakan Mesin Kapal Laut pada Gambar 3.13 merupakan interaksi antara user dengan sistem. User menginputkan jawaban dari pertanyaan-pertanyaan yang diberikan.

STIKOM SURABAYA

Diagnosa Kerusakan Mesin Kapal	
Pertanyaan :	<input type="text" value="Apakah Mesin bocor bagian luar?"/>
Pilihan Jawaban :	<input type="checkbox"/> Ya <input type="checkbox"/> Tidak
	<input type="button" value="Back"/> <input type="button" value="Next"/> <input type="button" value="Exit"/> <input type="button" value="Print"/>
Fakta - fakta	<input type="text" value="Mesin bocor bagian luar = ya
Sistem pendingin mesin tdk rapat = tidak"/>
Cek Kerusakan	<input type="text" value="Pemipan bagian luar. Saluran pembuangan untuk pendingin. Kebocoran pada kotak oli mesin"/>

Gambar 3.13 Form Diagnosa Kerusakan Mesin Kapal Laut