

SKRIPSI

SISTEM PAKAR DIAGNOSA KERUSAKAN HANDPHONE NOKIA E-SERIES MENGGUNAKAN METODE FORWARD CHAINING



Disusun Oleh :

HARRY AULIA RAHMAN

05.12.507

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2010**

3040

WISUDA DEPARTEMEN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
KEMENTERIAN TEKNIK KOMUNIKASI DAN INFORMATIKA
SARANGGANI DEKAT ETEN 2-1

0013001

WISUDA 2014

WISUDA 2014

LOKASI CHITING

WISUDA E-SEMESTER MENGGUNAKAN METODE
SISTEM BUKU DAN BAHAN KEMAHIRAN HANYA

SKRIPSI

LEMBAR PERSETUJUAN

**SISTEM PAKAR DIAGNOSA KERUSAKAN HANDPHONE
NOKIA E-SERIES MENGGUNAKAN METODE
FORWARD CHAINING**

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Komputer Dan Informatika Strata Satu (S-1)*

Disusun Oleh :

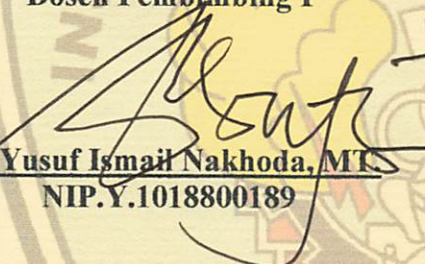
HARRY AULIA RAHMAN

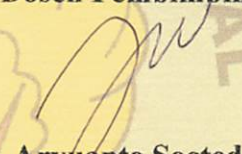
NIM : 05.12.507

Diperiksa dan Disetujui

Dosen Pembimbing I

Dosen Pembimbing II


Ir. Yusuf Ismail Nakhoda, MT.
NIP.Y.1018800189


Dr. Eng. Aryanto Soetedjo, ST, MT
NIP.Y.1030800417

Mengetahui

Ketua Jurusan Teknik Elektro S-1




Ir. Yusuf Ismail Nakhoda, MT.
NIP.Y.1018800189

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

2010

SISTEM PAKAR DIAGNOSA KERUSAKAN HANDPHONE NOKIA E-SERIES MENGGUNAKAN METODE FORWARD CHAINING

Harry Aulia Rahman

Jurusan Teknik Elektro S-1, Konsentrasi Teknik Komputer dan Informatika
Fakultas Teknologi Industri, Institut Teknologi Nasional Malang
Jl. Raya Karanglo Km 2 Malang
Email : harryaulia@gmail.com

Dosen Pembimbing : I. Ir. Yusuf Ismail Nakhoda,MT.

II. Dr. Eng. Aryuanto Soetedjo,ST,MT.

Abstrak

Layanan handphone atau telepon seluler telah menjadi kebutuhan penting, yang dalam tahun-tahun mendatang tidak mustahil akan menggeser layanan telepon tradisional. Dengan kemampuannya yang semakin berkembang, ternyata media telepon seluler mempunyai ruang gerak perkembangan teknologi yang tidak terbendung lagi. Perancangan Dan Pembuatan Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode Fordward chaining ini merupakan jawaban awal terhadap keingintahuan masyarakat tentang servis Nokia E-Series dan perkembangan teknologi telekomunikasi handphone secara umum untuk dapat dijadikan sebagai bahan acuan dasar dalam mengantisipasi kerusakan handphone Nokia E-Series.

Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode Forward Chaining ini memakai teknik representasi pengetahuan berbasis aturan atau rule, dan menggunakan Depth-First Search dalam penelusurannya, dari gejala-gejala hingga menghasilkan solusi bagaimana cara memecahkan suatu persoalan pada kerusakan handphone Nokia E-Series.

Kata kunci : Sistem Pakar, Fordward chaining, Nokia E-Series

Abstract

Mobile or cellular phone service has become an important requirement, which in future years is not impossible to shift the traditional phone service. With its ability to progressively expand, mobile phones have turned out to media space technology developments that are not be stopped. Design And Construction Damage Diagnosis Expert System Mobile Nokia E-Series Using Fordward Method chaining is beginning to answer the public's curiosity about the service Nokia E-Series and the development of mobile telecommunication technology in general to be used as basic reference material in anticipation of damage to the phones Nokia E- Series.

Diagnosis Expert System Mobile Nokia E-Series Using these methods Forward chaining using rules-based knowledge representation techniques or rules, and use the Depth-First Search in search, from symptoms to produce a solution how to solve a problem on mobile phones Nokia E-damage Series

Key Word: Expert System, Forward Chaining, Nokia E-Series

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa yang telah memberikan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode Forward Chaining” ini dengan lancar. Skripsi ini merupakan persyaratan kelulusan Studi di Jurusan Teknik Elektro S-1 Konsentrasi Teknik Komputer dan Informatika ITN Malang dan untuk mencapai gelar Sarjana Teknik.

Keberhasilan penyelesaian laporan skripsi ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penyusun menyampaikan terima kasih kepada :

1. Prof. Dr. Ir. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Yusuf Ismail Nakhoda, MT. selaku Ketua Jurusan Teknik Elektro S-1.
3. Bapak Ir. Yusuf Ismail Nakhoda, MT. selaku Dosen Pembimbing I.
4. Bapak Dr. Eng. Aryunto Soetedjo, ST, MT. selaku Dosen Pembimbing II.
5. Ayah dan Ibu serta saudara-saudara kami yang telah memberikan do'a restu, dorongan, semangat, dan biaya.
6. Rekan-rekan Instruktur di Laboratorium Jaringan Komputer ITN Malang.
7. Semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Penyusun telah berusaha semaksimal mungkin dan menyadari sepenuhnya akan keterbatasan pengetahuan dalam menyelesaikan laporan ini. Untuk itu penyusun mengharapkan saran dan kritik yang membangun dari pembaca demi kesempurnaan laporan ini.

Harapan penyusun semoga laporan skripsi ini memberikan manfaat bagi perkembangan ilmu pengetahuan dan pembaca.

Malang, Agustus 2010

Penyusun

DAFTAR ISI

LEMBAR PERSETUJUAN	i
ABSTRAKSI	ii
KATA PENGANTAR	iii
DAFTAR ISI	v
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan	2
1.4. Batasan Masalah	3
1.5. Metode Penelitian.....	3
1.6. Sistematika Penulisan.....	4
BAB II DASAR TEORI	7
2.1. Konsep Dasar Telepon Seluler	7
2.1.1. Prinsip Kerja Handphone	7
2.1.2. Perkembangan Hardware Handphone	8
2.1.3. Sejarah Nokia E-Series	8
2.1.4. Jenis Gejala dan Kerusakan Handphone	9
2.2. Sistem Pakar	11
2.2.1. Definisi Sistem Pakar	11

2.2.2. Latar Belakang Sistem Pakar	13
2.2.3. Lingkungan Operasional Sistem Pakar	14
2.2.4. Ciri – ciri Sistem Pakar	14
2.2.5. Keuntungan Sistem Pakar	15
2.2.6. Kelemahan Sistem Pakar	16
2.2.7. Konsep Dasar Sistem Pakar	16
2.2.8. Bentuk Sistem Pakar	18
2.2.9. Struktur Sistem Pakar	18
2.2.10. Basis Pengetahuan (Base Knowledge)	21
2.2.11. Motor Inferensi (Inference Engine)	22
2.2.12. Permasalahan Yang Disentuh Sistem Pakar	24
2.3. Perancangan DataBase	25
2.4. Contex Diagram	26
2.5. Data Flow Diagram	26
2.6. Diagram Entity Relationship	28
2.7. Bordland Delphi 7	31
2.8. InterBase 6.5	32
BAB III ANALISA DAN PERANCANGAN	34
3.1. Blok Diagram Sistem Pakar	35
3.2. Tree Sistem Pakar	36
3.3. Perancangan Rule Base	37
3.4. Mesin Inferensi	37
3.5. Pemilihan Inferensi	38

3.6. Forward Chaining	38
3.7. Perancangan Perangkat Lunak	39
3.8. Diagram Konteks	39
3.9. DFD Level 1	40
3.10. ER Diagram	41
3.11. Flowchart Sistem	42
3.12. Tabel Database	42
3.12.1. Tabel Gejala	42
3.12.2. Tabel Kerusakan	43
3.12.3. Tabel Rule	43
3.13. Site Map	43
3.14. Layout Program	45
BAB IV IMPLEMENTASI DAN PENGUJIAN	48
4.1. Implementasi Program	48
4.2. Penjelasan Program	48
4.2.1. Form Menu Utama	49
4.2.2. Form Gejala	49
4.2.3. Form Kerusakan	50
4.2.4. Form Rule	51
4.3. Pengujian Sistem	53
4.3.1. Pengujian Sistem Pertama	54
4.3.2. Pengujian Sistem Kedua	62
BAB V KESIMPULAN DAN SARAN	65

5.1. Kesimpulan	65
5.2. Saran	65
DAFTAR PUSTAKA	66
LAMPIRAN	67

DAFTAR GAMBAR

2.1. Konsep dasar fungsi sistem pakar	17
2.2. Struktur sistem pakar	19
2.3. Diagram Forward Chaining	22
2.4. Diagram Backward Chaining	23
2.5. Depth-First Search	23
2.6. Breadh-First Search	24
2.7. Relasi satu ke satu	29
2.8. Relasi satu ke banyak	30
2.9. Relasi banyak ke satu	30
2.10. Relasi banyak ke banyak	31
3.1. Blok Diagram SP Kerusakan HP Nokia E-Series	35
3.2. Tree SP Kerusakan HP Nokia E-Series	37
3.3. Blok doagram Mesin Inferensi Kerusakan HP Nokia E-Series	38
3.4. Contex Diagram	40
3.5. DFD level 1	40
3.6. ER Diagram	41
3.7. Flowchart Sistem	42
3.8. Site Map Sistem Pakar Kerusakan HP Nokia E-Series	44
3.9. Desain Layout Menu Utama	45
3.10. Desain layout Gejala	46
3.11. Desain layout Kerusakan	46
3.12. Desain layout Rule	47

3.13. Desain layout Diagnosa Kerusakan Handphone.....	47
4.1. Form Menu Utama	49
4.2. Form Gejala	50
4.3. Form Kerusakan	51
4.4. Form Rule	52
4.5. Form Diagnosa	53
4.6. Rule Diagnosa Kerusakan No Signal Pada HP E-51	54
4.7. Rule Diagnosa Kerusakan No Charge Pada HP E-51	55
4.8. Rule Diagnosa Kerusakan No Voice Pada HP E-51	57
4.9. Rule Diagnosa Kerusakan No Radio Pada HP E-51	58
4.10. Rule Diagnosa Kerusakan No Signal Pada HP E-60	59
4.11. Rule Diagnosa Kerusakan No Charger Pada HP E-60	59
4.12. Rule Diagnosa Kerusakan No Radio Pada HP E-60	60
4.13. Rule Diagnosa Kerusakan Kamera Fault Pada HP E-60.....	61
4.14. Rule Diagnosa Kerusakan No Charger Pada HP E-61	62
4.15. Rule Diagnosa Kerusakan Kamera Fault Pada HP E-71.....	63
4.16. Rule Diagnosa Kerusakan No Voice Pada HP E-90	64

DAFTAR TABEL

2.1. Sistem Pakar Yang Terkenal	13
2.2. Perbedaan seorang pakar dengan sistem pakar	14
2.3. Simbol-simbol Data Flow Diagram	27
2.4. Notasi E-R Diagram	28
3.1. keterangan tree.....	37
3.2. Gejala	42
3.3. Kerusakan	43
3.4. Rule.....	43
4.1 Hasil Pengujian	64

BAB I

PENDAHULUAN

1.1. Latar Belakang

Teknologi handphone dan komputer, tampaknya semakin berkembang dengan pesat terutama handphone bertipe Nokia E-Series. Rasanya tiap detik, teknologi ini melesat menuju ke sebuah fungsi yang semakin canggih.

Sistem telepon seluler (Ponsel) atau handphone (HP) ini sebenarnya telah mengalami evolusi yang panjang, sebelum melesat perkembangannya seperti sekarang ini. Handphone atau ponsel Nokia E-Series tidak hanya sarana komunikasi suara seperti pada awal-awal teknologi seluler ini ditemukan, akan tetapi saat ini telah menjadi sebuah piranti canggih, sebagai media komunikasi yang mempunyai fungsi sangat luas karena berkembangnya teknologi *wireless* dan teknologi transmisi data.

Layanan handphone atau telepon seluler telah menjadi kebutuhan penting, yang dalam tahun-tahun mendatang tidak mustahil akan menggeser layanan telepon tradisional. Dengan kemampuannya yang semakin berkembang, ternyata media telepon seluler mempunyai ruang gerak perkembangan teknologi yang tidak terbendung lagi.

Dengan peralihan kecanggihan dalam teknologi ini tentu saja berpengaruh juga terhadap harga dari handphone Nokia E-Series ini sendiri. Sehingga apabila ada kerusakan handphone, dalam perbaikannya

harus dengan ketelitian yang sangat tinggi untuk menghindari kesalahan dari diagnosa kerusakan.

Perancangan Dan Pembuatan Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode *Forward chaining* ini merupakan jawaban awal terhadap keingintahuan masyarakat tentang servis Nokia E-Series dan perkembangan teknologi telekomunikasi handphone secara umum untuk dapat dijadikan sebagai bahan acuan dasar dalam mengantisipasi kerusakan Handphone Nokia E-Series.

1.2. Rumusan Masalah

Dari latar belakang yang telah diuraikan di atas dapat dirumuskan permasalahannya yaitu “Bagaimana cara Merancang dan Membuat Program Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode *Forward Chaining*”.

1.3. Tujuan

Ditinjau dari permasalahan diatas, tujuan dari Skripsi ini sebagai berikut Merancang dan Membuat Program Sistem Pakar Diagnosa Kerusakan Handphone E-Series Menggunakan Metode *Forward Chaining*.

Adapun tujuan detail yang ingin dicapai dalam pelaksanaan Skripsi adalah sebagai berikut:

1. Memberikan solusi bagaimana cara memecahkan suatu persoalan pada kerusakan handphone Nokia E-Series.

2. Membantu teknisi untuk mengetahui sumber masalah dari kerusakan handphone.
3. Membantu teknisi untuk mengetahui kemungkinan-kemungkinan ada kerusakan yang lain.
4. Menyakinkan para teknisi dalam memutuskan kerusakan apa yang terjadi sesungguhnya pada handphone Nokia E-Series tersebut.

1.4. Batasan Masalah

Pengetahuan mengenai servis handphone Nokia E-Series yang sangat kompleks dan luas, apalagi kalau sudah berkaitan dengan perkembangannya saat ini, dilihat dari pentingnya pengetahuan dan informasi mengenai dasar - dasar servis Telepon Seluler yang umum, maka pembahasan ditekankan pada :

1. Hanya membahas, mengulas dan menangani diagnosa serta solusi perbaikan handphone Nokia E-Series.
2. Pengenalan jenis-jenis kerusakan handphone Nokia E-Series yang biasa atau umum terjadi.

1.5. Metodologi Penelitian

Metodologi yang diterapkan dalam pengerjaan Skripsi ini adalah sebagai berikut:

1. Studi pustaka/*litelature* dilakukan dengan melakukan studi mengenai sistem pakar, kerusakan handphone Nokia E-Series mencari melalui literatur-literatur seperti buku (*textbook*), *paper*, dan sumber ilmiah

lain seperti *website*, artikel, dan dokumen teks sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan obyek penelitian.

2. Analisis sistem pakar dengan melakukan analisis terhadap sistem pakar berupa studi kelayakan, spesifikasi kebutuhan, teknik akuisisi pengetahuan, dan penentuan pakar.
3. Perancangan sistem pakar yang akan dikembangkan berdasarkan hasil analisis yang telah dilakukan, termasuk akuisisi pengetahuan untuk mendapatkan rancangan *rule* yang akan diimplementasikan. Akuisisi pengetahuan dilakukan dengan cara wawancara kepada pakar, studi literatur, dan analisa kasus.
4. Pengujian sistem pakar, yaitu melakukan pengujian hasil implementasi sistem pakar. Pengujian dilakukan pula oleh pakar untuk menguji kebenaran sistem pakar.

1.6. Sistematika Pembahasan

Penyusunan laporan Skripsi ini menggunakan kerangka pembahasan yang terbentuk dalam susunan bab, yang dapat dijelaskan sebagai berikut :

BAB I : Pendahuluan

Bab ini merupakan dasar penyusunan laporan Skripsi yang di dalamnya berisi tentang latar belakang masalah, rumusan masalah, tujuan Skripsi, batasan masalah, dan metodologi penelitian.

BAB II : Dasar Teori

Bab ini berisi penjelasan tentang tinjauan singkat mengenai materi dan Sistem Pakar yang digunakan dalam program Perancangan dan Pembuatan Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode *Forward Chaining*.

BAB III: Analisis dan Perancangan

Bab ini berisi tentang tahap analisis yaitu identifikasi dan analisis masalah dan analisis kebutuhan sistem untuk menyelesaikan masalah yang dihadapi berdasarkan teori pada Bab II dan Bab III. Bab ini juga berisi hasil perancangan yaitu proses kelanjutan dari tahap analisis meliputi proses akuisisi pengetahuan.

BAB IV : Implementasi dan Pengujian

Bab ini berisi tentang proses pembuatan program “Perancangan dan Pembuatan Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode *Forward Chaining*” serta meliputi hasil pengujian sistem.

BAB V : Penutup

Bab ini berisi tentang kesimpulan dan saran-saran mengenai program Perancangan dan Pembuatan Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode *Forward Chaining*.

BAB II

DASAR TEORI

Dalam bab ini membahas tentang sumber-sumber informasi yang berhubungan dengan “Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode *Forward Chaining*” dengan menggunakan media atau teori - teori yang mendukung pembuatan program tersebut.

2.1. Konsep Dasar Telepon Seluler

Sistem seluler adalah sistem yang jenius sebab sistem ini membagi suatu kawasan dalam beberapa sel yang kecil. Hal ini digunakan untuk memastikan bahwa frekuensi dapat meluas sehingga mencapai ke semua wilayah.

2.1.1. Prinsip Kerja Handphone

Prinsip kerja handphone pada dasarnya merupakan prinsip dari komunikasi *wireless* yang menggunakan kanal radio terpisah untuk berkomunikasi dengan *cell site*, dan mengantarkan informasi berupa data dan suara pengguna melalui sinyal pembawa (*carrier*).

Perambatan sinyal informasi tersebut menggunakan berbagai macam metode akses, yaitu GSM, CDMA, dan UMTS (3G). Pengguna dapat saling berkomunikasi melalui identitas pengguna berupa numerik SIM (*Subscribe Identity Module*). SIM merupakan piranti registrasi

pengguna melalui handphone di dalam suatu *cell* untuk diteruskan ke BTS (*Base Tranceiver Station/ Tower*) terdekat agar dapat diketahui keberadaan *account*-nya pada *server* utama BTS.

2.1.2. Perkembangan *Hardware Handphone*

Perkembangan teknologi seluler saat ini semakin pesat, menawarkan fungsi handphone tidak lagi sekedar sarana komunikasi, tetapi menawarkan juga fitur lain, seperti televisi tuner atau kamera digital dengan resolusi tinggi ditambah adanya kapasitas memori yang besar menjadikan handphone sebagai perangkat mobile yang canggih dan pintar. Hingga saat ini, layanan yang beroperasi di Indonesia untuk generasi handphone telah masuk pada generasi ketiga atau yang lebih dikenal 3G - 3,5G seperti handphone Nokia E-Series.

2.1.3. Sejarah Nokia E-Series

E-Series (Seri E) adalah sekelompok produk telepon genggam dari Nokia yang mendukung berbagai perangkat *digital multimedia* seperti pemutar musik, pembuatan video, fotografi, *mobile gaming* dan layanan internet. Semua Nokia seri E ini mempunyai setidaknya satu teknologi nirkabel dengan kecepatan tinggi seperti 3G, HSDPA atau LAN *Wireless*.

Sejarah peluncuran E-Series^[6] :

1. 12 Oktober 2005 : E60, E61, E70
2. 18 Mei 2006 : E 50
3. 12 Februari 2007 : E 61, E61i, E90

4. 10 JUNI 2008 : E66, E71
5. 2009 : E72, E52, E63

Fitur Unggulan Nokia seri E :

Nokia seri E ini berusaha merangkum semua fitur yang dibutuhkan oleh para konsumen. Fitur utama Nokia seri E adalah kamera yang bertaraf 2 MP ke atas, pengambilan video dengan handphone yang lebih lama dengan kualitas baik, fitur *game* yang sangat digemari dan yang terakhir konektivitas dengan kecepatan tinggi seperti 3G, HSDPA dan LAN *Wireless*.

Sistem Operasi E-Series :

Nokia seri E yang pertama, E70 dan E62, menggunakan sistem operasi *Symbian 6.0. 3rd Edition* (rilis awal) Namun pada seri-seri selanjutnya, Nokia berganti menggunakan sistem operasi *Symbian S60 3rd Edition Feature Pack 1*, Seri E yang terbaru memasukkan sistem operasi *Symbian S60 3rd Edition Feature Pack 2* yang telah direvisi sehingga mempunyai paket fitur^[7].

2.1.4. Jenis Gejala dan Kerusakan Handphone

Adapun jenis kerusakan pada handphone yang umum terjadi antara lain^[4]:

Mati total

Mati total adalah gangguan dimana semua perangkat handphone tidak berfungsi. mati total meliputi beberapa gejala sebagai berikut :

- IC CPU mati

- IC Power mati
- *Switch* On/Off tidak berfungsi

No Charger

No Charger adalah kondisi dimana handphone tidak ada tegangan pengisian pada konektor ponsel dan baterai. *No Charger* meliputi beberapa gejala sebagai berikut :

- Tidak ada arus listrik yang mengisi baterai
- Konektor ponsel tidak mampu menyimpan arus listrik

No Ring

No ring adalah kondisi handphone yang tidak mengeluarkan nada dering saat menerima panggilan masuk atau saat menerima pesan. *No ring* meliputi beberapa gejala sebagai berikut:

- Audio mengalami gangguan
- Buzzer rusak
- IC Power rusak

No Signal

No Signal adalah kondisi handphone dimana tidak terdapat sinyal untuk melakukan panggilan atau menerima panggilan. *No Signal* meliputi beberapa gejala sebagai berikut :

- Gangguan pada antena
- Gangguan pada unit pemancar
- Gangguan pada unit penerima

Kamera Fault

Kamera *Fault* adalah dimana fitur kamera pada handphone mengalami gangguan *software* dan *hardware*. Kamera Fault meliputi beberapa gejala sebagai berikut :

- Gambar tampak gelap di LCD saat memasuki menu kamera
- Pada saat mengambil gambar kamera handphone tidak dapat fokus

Keterangan-keterangan diatas merupakan sekilas tentang berbagai macam kerusakan dan gejala handphone yang umum terjadi, selanjutnya data akan berjenjang dan tidak disebutkan disini karena banyaknya data, data di entrikan di program sistem pakar.

2.2 Sistem Pakar

Perancangan dan pembuatan Sistem Pakar Diagnosa Kerusakan handphone E-Series ini menggunakan metode sistem pakar. Bidang teknik kecerdasan buatan yang paling populer saat ini adalah sistem pakar. Ini disebabkan penerapannya diberbagai bidang, baik dalam pengembangan ilmu pengetahuan dan terutama dibidang bisnis telah terbukti sangat membantu dalam pengambilan keputusan. Sistem pakar juga merupakan bidang teknik kecerdasan buatan yang paling luas penerapannya.

2.2.1 Definisi Sistem Pakar

Secara umum, sistem pakar (*expert system*) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para

ahli^[8]. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para ahli. Dengan sistem pakar ini, orang awampun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli. Bagi para ahli, sistem pakar ini juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman.

Ada beberapa definisi tentang sistem pakar, antara lain:

- a) Menurut Durkin: Sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan oleh seorang pakar.
- b) Menurut Ignizio: Sistem pakar adalah suatu model dan prosedur yang berkaitan, dalam suatu domain tertentu, yang mana tingkat keahliannya dapat dibandingkan dengan keahlian seorang pakar.
- c) Menurut Giarratano dan Riley: Sistem pakar adalah suatu sistem komputer yang bisa menyamai atau meniru kemampuan seorang pakar.
- d) Menurut Martin dan Oxman: 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 bidang tersebut.

Sistem pakar pertama kali dikembangkan oleh komunitas AI pada pertengahan tahun 1960. Sistem pakar yang muncul pertama kali adalah *General-Purpose Problem Solver* (GPS) yang dikembangkan oleh Newell dan Simon. GPS (dan program-program yang serupa) ini mengalami

kegagalan dikarenakan cakupannya terlalu luas sehingga terkadang justru meninggalkan pengetahuan-pengetahuan penting yang seharusnya disediakan.

Sampai saat ini sudah banyak sistem pakar yang dibuat, beberapa contoh seperti pada tabel 2.1 di bawah ini :

Tabel 2.1 Sistem Pakar yang terkenal^[8].

Sistem Pakar	Kegunaan
MYCIN	Diagnosa Kerusakan penyakit radang pembuluh darah
DENDRAL	Mengidentifikasi struktur molekular campuran yang tidak dikenal
XCON& XCEL	Membantu konfigurasi sistem komputer besar
SOPHIE	Analisis sirkuit elektronik
PROSPECTOR	Digunakan didalam geologi untuk membantu mencari dan menemukan deposit
FOLIO	Membantu memberikan keputusan bagi seorang manajer dalam hal stok broker dan investasi
DELTA	Pemeliharaan lokomotif listrik diesel

2.2.2 Latar Belakang Sistem Pakar

Sistem pakar mempunyai beberapa latar belakang yang mempengaruhinya. Latar belakang tersebut diantaranya sebagai berikut:

- a) Para pakar biasanya jumlahnya sedikit dan terkonsentrasi dikota-kota besar oleh karena itu biasanya jasa pakar mahal
- b) Umur pakar terbatas sehingga jika pakar tersebut tidak tersedia lagi (meninggal) maka kepakarannyapun ikut hilang dan diperlukan waktu bertahun-tahun dengan biaya besar untuk menghasilkan pakar bidang tersebut.

Seorang pakar dengan sistem pakar mempunyai banyak perbedaan antara lain, seperti pada table 2.2 di bawah ini :

Tabel 2.2 Perbedaan seorang pakar dengan sistem pakar^[8].

Faktor	Human Expert	Expert System
Time availability	Hari kerja	Setiap saat
Geografis	Lokal/tertentu	Dimana saja
Keamanan	Tidak tergantikan	Dapat diganti
Perishable/dapat habis	Ya	Tidak
Performansi	Variable	Konsisten
Kecepatan	Variable	Konsisten
Biaya	Tinggi	Terjangkau

2.2.3 Lingkungan Operasional Sistem Pakar

Lingkungan operasional dalam sistem pakar dibagi menjadi dua, yaitu:

1. *ON-LINE*

Sisten pakar dapat langsung mengambil data dari sistem yang sedang didiagnosis. Sebagai contoh sistem pakar untuk mendeteksi kerusakan komponen pada mobil, pesawat atau komputer.

2. *OFF-LINE*

Sistem pakar dalam memperoleh informasi dilakukan melalui layar monitor dengan cara melakukan tanya jawab dengan pemakai atau client.

2.2.4 Ciri-ciri Sistem Pakar

Sistem pakar memiliki ciri-ciri sebagai berikut:

- Terbatas pada domain keahlian tertentu.
- Dapat memberikan penalaran untuk data-data yang tidak pasti.
- Dapat mengemukakan rangkaian alasan-alasan yang diberikannya dengan cara yang dapat dipahami.
- Berdasarkan pada kaidah/rule tertentu.
- Dirancang untuk dapat dikembangkan secara bertahap.

- Keluarannya bersifat anjuran.
- Sistem dapat mengaktifkan kaidah secara searah yang sesuai, dituntun oleh dialog dengan pemakai.

2.2.5 Keuntungan Sistem Pakar

Secara garis besar, banyak manfaat yang dapat diambil dengan adanya sistem pakar, antara lain :

- Membuat orang awam, bekerja selayaknya seorang pakar.
- Meningkatkan produktivitas akibat meningkatnya kualitas hasil pekerjaan, peningkatan kualitas disebabkan oleh meningkatnya efisiensi kerja.
- Menghemat waktu kerja.
- Menyederhanakan pekerjaan.
- Merupakan arsip yang terpercaya dari sebuah keahlian, sehingga bagi pemakai sistem pakar akan seolah-olah berkonsultasi langsung dengan sang pakar, meskipun mungkin sang pakar telah meninggal.
- Memerluas jangkauan, dari keahlian seorang pakar. Sistem pakar yang telah disahkan, akan sama saja artinya dengan seorang pakar yang tersedia dalam jumlah besar (dapat diperbanyak dengan kemampuan yang persis sama), dapat diperoleh dan dipakai dimana saja.

2.2.6 Kelemahan Sistem Pakar

Disamping memiliki beberapa keuntungan, sistem pakar juga memiliki beberapa kelemahan, antara lain :

- Biaya yang diperlukan untuk membuat dan memeliharanya sangat mahal.
- Sulit dikembangkan. Hal ini tentu saja erat hubungannya dengan ketersediaan pakar dibidangnya.
- Sistem pakar tidak 100% bernilai benar.

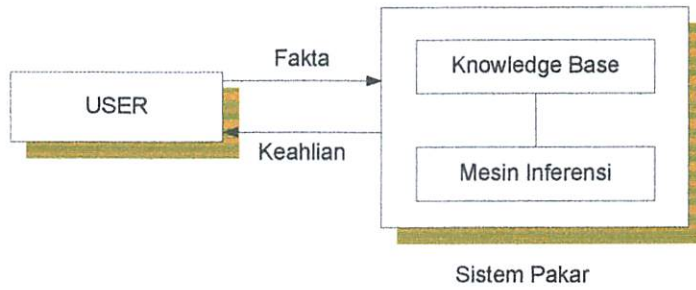
2.2.7 Konsep Dasar Sistem Pakar

Konsep dasar sistem pakar mengandung keahlian, ahli, pengalihan keahlian, inferensi, aturan dan kemampuan menjelaskan.

Keahlian adalah suatu kelebihan penguasaan pengetahuan dibidang tertentu yang diperoleh dari pelatihan, membaca atau pengalaman. Contoh bentuk pengetahuan yang termasuk keahlian adalah:

- Fakta-fakta pada lingkup permasalahan tertentu.
- Teori-teori pada lingkup permasalahan tertentu
- Prosedur-prosedur dan aturan-aturan berkenaan dengan lingkup permasalahan tertentu.
- Strategi-strategi global untuk menyelesaikan masalah.
- *Meta- knowledge* (pengetahuan tentang pengetahuan).

Konsep dasar fungsi sistem pakar dapat di lihat pada gambar 2.1 di bawah ini :



Gambar 2.1 Konsep dasar fungsi sistem pakar^[8]

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

Terdapat tiga orang yang terlibat dalam lingkungan sistem pakar, yaitu :

- **Pakar**, adalah orang yang memiliki pengetahuan khusus, pendapat, pengalaman dan metode, serta kemampuan untuk mengSystem Pakarkan keahliannya tersebut guna menyelesaikan masalah
- **Knowledge enginer**, 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.
- **Pemakai**, sistem pakar memiliki beberapa pemakai, yaitu: pemakai bukan pakar, pelajar, pembangun sistem pakar yang ingin meningkatkan dan menambah basis pengetahuan, dan pakar sendiri.

2.2.8 Bentuk Sistem Pakar

Ada 4 bentuk sistem pakar, yaitu :

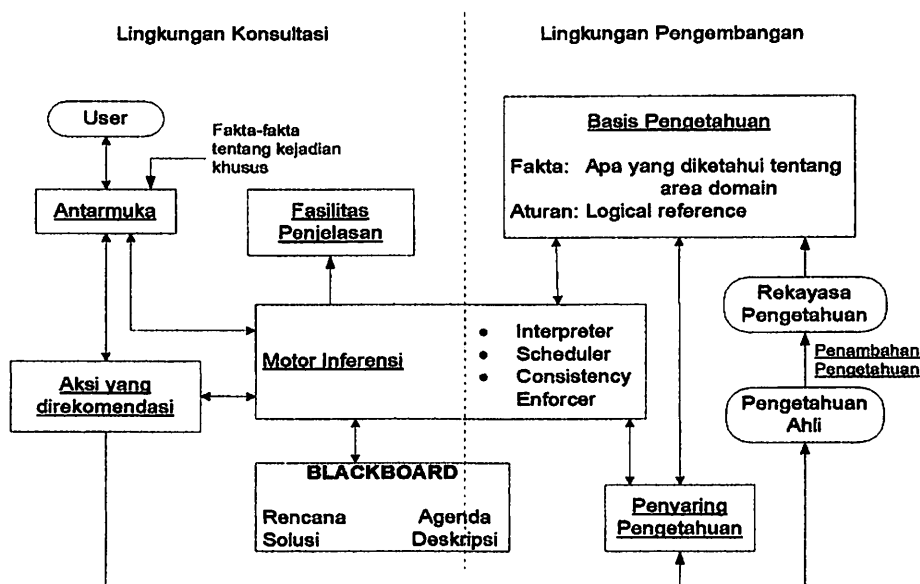
1. Berdiri sendiri. Sistem pakar jenis ini merupakan *software* yang berdiri-sendiri tidak tergabung dengan *software* yang lainnya.
2. Tergabung. Sistem pakar jenis ini merupakan bagian program yang terkandung didalam suatu algoritma (*konvensional*), atau merupakan program dimana didalamnya memanggil algoritma suburtin lain (*konvensional*).
3. Menghubungkan ke *software* lain. Bentuk ini biasanya merupakan sistem yang menghubungkan ke suatu paket program tertentu, misalnya dengan DBMS.
4. Sistem Mengabdi. Sistem pakar merupakan bagian dari komputer khusus yang dihubungkan dengan suatu fungsi tertentu. Misalnya sistem pakar yang digunakan untuk membantu menganalisis data radar.

2.2.9 Struktur Sistem Pakar

Sistem pakar terdiri dari 2 bagian pokok, yaitu : lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*).

Lingkungan pengembangan digunakan sebagai pembangun sistem pakar baik dari segi pembangun komponen maupun basis pengetahuan. Lingkungan konsultasi digunakan oleh seseorang yang bukan ahli untuk berkonsultasi.

Struktur sistem pakar dapat di lihat pada gambar 2.2 di bawah ini :



Gambar 2.2 Struktur sistem pakar^[8]

Komponen-komponen yang ada pada sistem pakar adalah sebagai berikut :

1. Sub sistem penambahan pengetahuan. Bagian ini digunakan untuk memasukkan pengetahuan, mengkonstruksi atau memperluas pengetahuan dalam basis pengetahuan. Pengetahuan itu bisa berasal dari: ahli, buku, basis data, penelitian dan gambar.
2. Basis pengetahuan. Berisi pengetahuan-pengetahuan yang dibutuhkan untuk memahami, memformulasikan dan menyelesaikan masalah.
3. Motor inferensi (*inference engine*). Program yang berisi metodologi yang digunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis pengetahuan dan *blackboard*, serta digunakan untuk memformulasikan konklusi. Ada 3 elemen utama dalam motor inferensi, yaitu :

- *Interpreter* : mengeksekusi item-item agenda yang terpilih dengan menggunakan aturan-aturan dalam basis pengetahuan yang sesuai.
 - *Scheduler* : akan mengontrol agenda.
 - *Consistency enforcer* : akan berusaha memelihara kekonsistenan dalam mempresentasikan solusi yang bersifat darurat.
4. *Blackboard* Merupakan area dalam memori yang digunakan untuk merekam kejadian yang sedang berlangsung termasuk keputusan sementara. Ada 3 tipe keputusan yang dapat direkam, yaitu :
- Rencana : bagaimana menghadapi masalah.
 - Agenda : aksi-aksi yang potensial yang sedang menunggu untuk dieksekusi.
 - Solusi : calon aksi yang akan dibangkitkan.
5. Antarmuka. Digunakan untuk media komunikasi antara user dan program.
6. Subsistem penjelasan. Digunakan untuk melacak respon dan memberikan penjelasan tentang kelakuan sistem pakar secara interaktif melalui pertanyaan :
- Mengapa suatu pertanyaan ditanyakan oleh sistem pakar?
 - Bagaimana konklusi dicapai?
 - Mengapa ada alternatif yang dibatalkan?
 - Rencana apa yang digunakan untuk mendapatkan solusi?
7. Sistem penyaring pengetahuan. Sistem ini digunakan untuk mengevaluasi kinerja sistem pakar itu sendiri untuk melihat apakah

pengetahuan-pengetahuan yang ada masih cocok untuk digunakan dimasa mendatang.

2.2.10 Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan berisi pengetahuan-pengetahuan dalam penyelesaian masalah, tentu di dalam domain tertentu. Ada 2 bentuk pendekatan basis pengetahuan yang sangat umum digunakan, yaitu :

a. Penalaran berbasis aturan (*Rule-Based Reasoning*)

Pada penalaran berbasis aturan, pengetahuan direpresentasikan dengan menggunakan aturan berbentuk: IF-THEN. Bentuk ini digunakan apabila kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu, dan si pakar dapat menyelesaikan masalah tersebut secara berurutan. Disamping itu, juga digunakan apabila dibutuhkan penjelasan tentang jejak (langkah – langkah) pencapaian solusi.

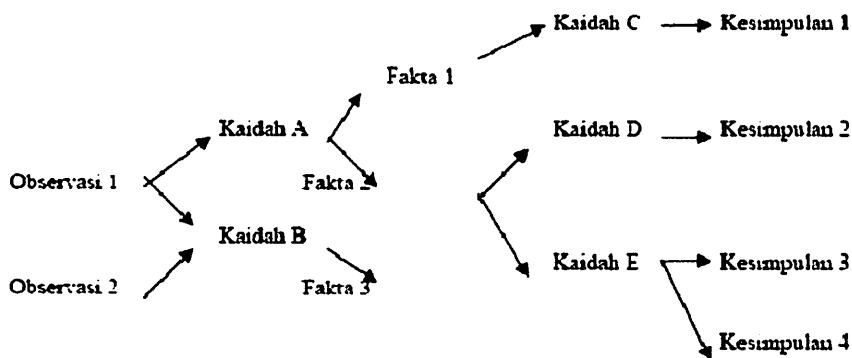
b. Penalaran berbasis kasus (*Case-Based Reasoning*).

Pada penalaran berbasis kasus, basis pengetahuan akan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang terjadi sekarang (fakta yang ada). Bentuk ini akan digunakan apabila user menginginkan untuk tahu lebih banyak lagi pada kasus-kasus yang hampir sama (mirip). Selain itu, bentuk ini juga digunakan apabila kita telah memiliki sejumlah situasi atau kasus tertentu dalam basis pengetahuan.

2.2.11 Motor Inferensi (*Inference Engine*)

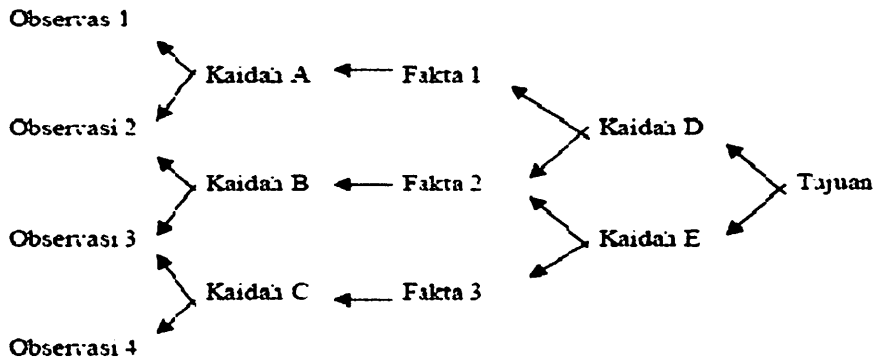
Ada 2 cara yang dapat dikerjakan dalam melakukan inferensi, yaitu:

1. **Forward Chaining.** Pencocokan fakta atau pernyataan dimulai dari bagian sebelah kiri (*IF* dulu). Dengan kata lain, penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis, seperti pada gambar 2.3 di bawah ini :



Gambar 2.3 Diagram *Forward Chaining*

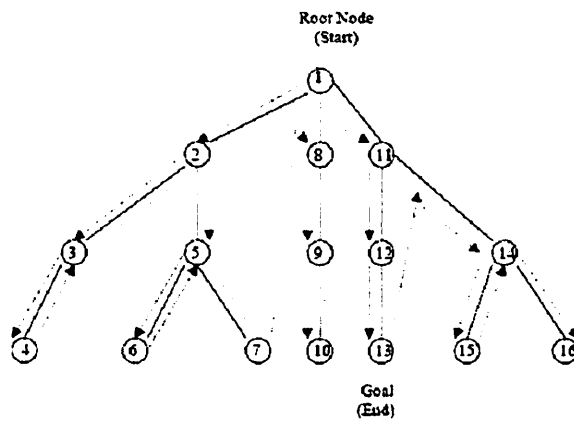
2. **Backward Chaining.** Pencocokan fakta atau pernyataan dimulai dari bagian sebelah sebelah kanan (*THEN* dulu). Dengan kata lain, penalaran dimulai dari hipotesis terlebih dahulu, dan untuk menguji kebenaran hipotesis tersebut harus dicari fakta-fakta yang ada dalam basis pengetahuan, seperti pada gambar 2.4 di bawah ini :



Gambar 2.4 Diagram *Backward Chaining*

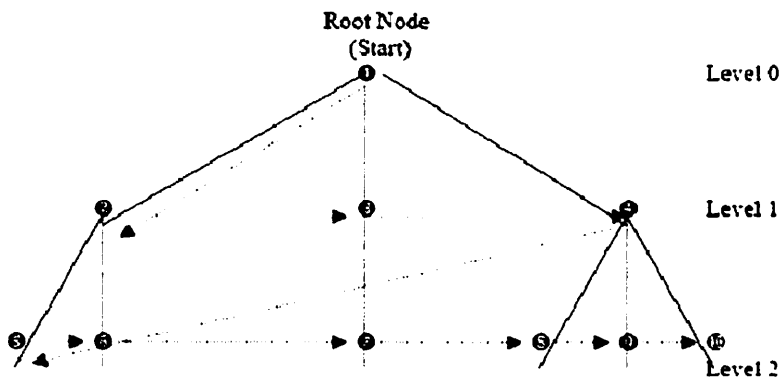
Kedua metode inferensi tersebut, dipengaruhi oleh dua macam teknik penelusuran yaitu :

- a. *Depth-First Search*. Penelusuran kaidah secara mendalam dari simpul akar bergerak menurun ke tingkat dalam yang berurutan, seperti pada gambar 2.5 di bawah ini :



Gambar 2.5 *Depth-First Search*

- b. *Breadth-First Search*. Penelusuran bergerak dari simpul akar, simpul yang ada pada setiap tingkat diuji sebelum pindah ke tingkat selanjutnya, seperti pada gambar 2.6 di bawah ini :



Gambar 2.6 *Breadth-First Search*

2.2.12 Permasalahan Yang Disentuh Oleh Sistem Pakar

Ada beberapa masalah yang menjadi area luas Sistem Pakar sistem pakar, antara lain :

- a. Interpretasi. Pengambilan keputusan dari hasil observasi, termasuk diantaranya: pengawasan, pengenalan ucapan, analisis citra, interpretasi sinyal, dan beberapa analisis kecerdasan.
- b. Prediksi. Termasuk diantaranya: peramalan, prediksi demografis, peramalan ekonomi, prediksi lalu lintas, estimasi hasil, militer, pemasaran, atau peramalan keuangan.
- c. Diagnosis. Termasuk diantaranya: medis, elektronis, mekanis, dan diagnosis perangkat lunak.
- d. Perancangan. Termasuk diantaranya: *layout* sirkuit dan perancangan bangunan.
- e. Perencanaan. Termasuk diantaranya: perencanaan keuangan, komunikasi, militer, pengembangan produk, *routing*, dan manajemen produk.

- f. *Monitoring*. Misalnya: *Computer-Aided Monitoring Systems*.
- g. *Debugging*. Memberikan resep obat terhadap suatu kegagalan.
- h. Perbaikan.
- i. Instruksi. Melakukan instruksi untuk diagnosis, *debugging*, dan perbaikan kinerja.
- j. Kontrol. Melakukan kontrol terhadap interpretasi, prediksi, perbaikan, dan *monitoring* kelakuan sistem.

2.3 Perancangan Database

Database merupakan salah satu komponen yang penting di sistem informasi, karena berfungsi sebagai basis penyedia informasi bagi pemakainya. *Database* adalah sekumpulan data (arsip) yang saling berhubungan dan disimpan sedemikian rupa tanpa ada pengulangan yang tidak perlu, untuk memenuhi berbagai kebutuhan. Sistem basis data (*database system*) ini adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan yang lain dan membuatnya tersedia untuk beberapa Sistem Pakar yang bermacam-macam didalam suatu organisasi.

Database yang dibentuk diharapkan memiliki sifat-sifat sebagai berikut :

- Efisien, artinya untuk proses penyimpanan data tidak memerlukan banyak tempat penyimpanan dan waktu pemrosesan data yang cepat.
- Efektif, artinya untuk menambah dan menyisipkan dan menghapus data dapat dilakukan dengan mudah dan sederhana.

- Fleksibel, artinya *database* dapat diakses dengan mudah, dinamis dan tidak tergantung sepenuhnya pada Sistem Pakar-Sistem Pakar tertentu.
- Integritas data, artinya data yang ada tidak memiliki banyak redundansi (redundansi data yang minimal).

Oleh karena itu proses pembentukan *database* merupakan tahapan yang sangat menentukan bagi terciptanya sistem informasi yang baik. Untuk mendukung proses pembentukan *database* tersebut ada beberapa peralatan yang digunakan seperti *contex diagram* (CD), *data flow diagram* (DFD), dan *entity relationship diagram* (ERD).

2.4 Contex Diagram


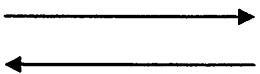

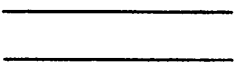
Contex Diagram merupakan pendekatan terstruktur yang mencoba untuk menggambarkan sistem pertama kali secara garis besar (disebut dengan top level) dan memecah-mecahnya menjadi bagian yang lebih terinci. Satu *contex diagram* selalu mengandung satu dan hanya satu proses saja, proses ini mewakili proses dari seluruh sistem. *Contex diagram* ini menggambarkan hubungan *input/output* antara sistem dengan kesatuan luar.

2.5 Data Flow Diagram(DFD)

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut akan disimpan. DFD merupakan alat yang digunakan pada metodologi pengembangan

sistem yang terstruktur. Simbol- simbol yang digunakan untuk menggambarkan DFD adalah seperti pada table 2.3 di bawah ini :

Tabel 2.3 Simbol – simbol Data Flow Diagram

Nama	Keterangan
	Entitas Luar dari Sistem
	Data Flow / Aliran Data
	Proses
	Data Store / Penyimpanan

Keterangan :



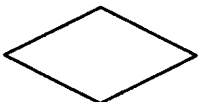

- Entitas luar atau *boundary* (batas sistem) merupakan kesatuan (*entity*) di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luar yang akan memberikan input atau menerima output dari sistem.
- *Data Flow* (aliran data) menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau hasil dari proses sistem
- Proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.
- *Data Store* digunakan untuk memodelkan kumpulan data atau paket data.

2.6 Diagram *Entity-Relationship* (Diagram E-R).

Pada model relasional, basis data akan dikelompokkan kedalam berbagai tabel dua dimensi, disetiap pertemuan baris dan kolom item-item data (satuan data terkecil) ditempatkan. 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 ditinjau, dapat digambarkan dengan lebih sistematis dengan menggunakan Diagram E-R.

Notasi simbolik diagram E-R yang dapat kita gunakan adalah seperti pada tabel 2.3 di bawah ini :

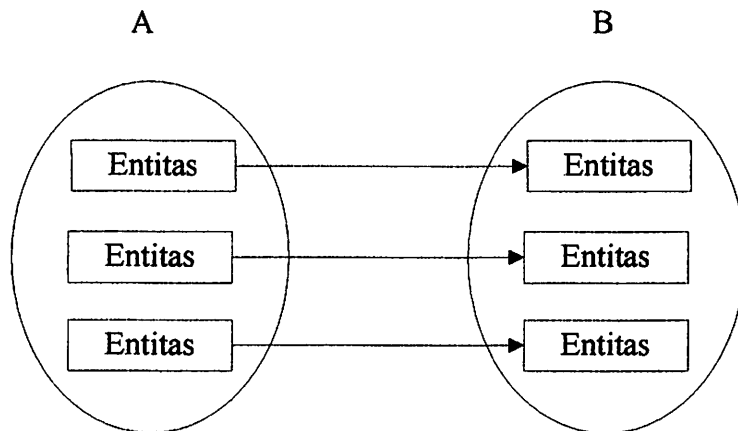
Tabel 2.4 Notasi E-R Diagram

Simbol	Keterangan
	Entitas merupakan kesatuan di lingkungan luar sistem
	Atribut adalah keterangan yang menyertai entitas
	Relasi adalah proses yang meghubungkan satu sentitas dengan entitas lain
	Link adalah garis yang menghubungkan antar entitas dengan relasi

Kardinalitas relasi dapat dinyatakan dengan banyaknya garis cabang atau dengan angka (1 dan 1 untuk relasi satu ke satu, 1 dan N untuk relasi satu ke banyak atau N dan N untuk relasi banyak ke banyak).

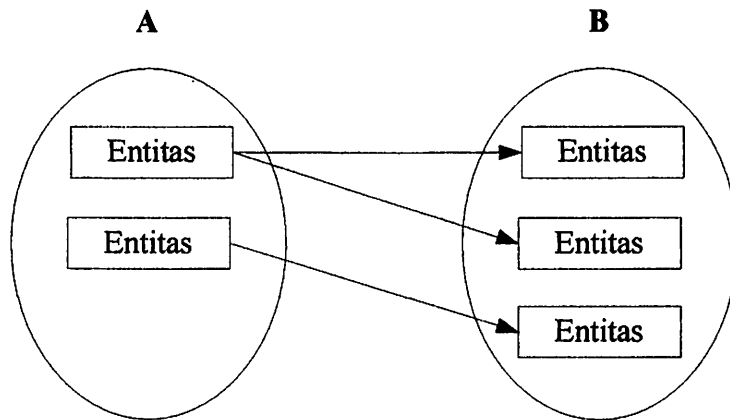
Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Adapun kardinalitas yang terjadi antara dua himpunan entitas dapat berupa :

- **Satu ke satu (*one to one*)**, yang 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 A berhubungan dengan paling banyak dengan entitas A, seperti pada gambar 2.8 di bawah ini :



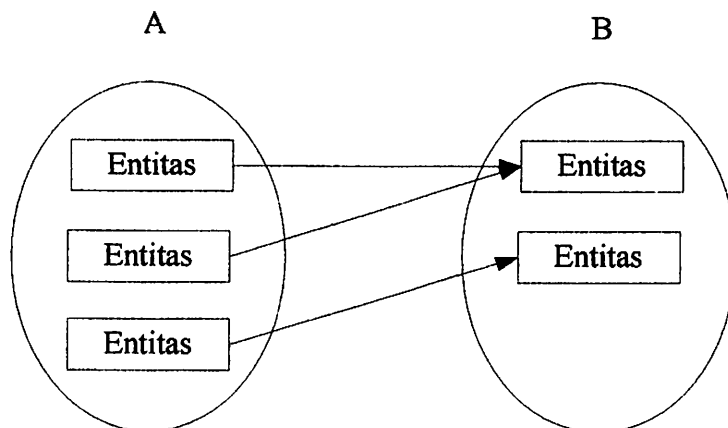
Gambar 2.7 Relasi satu ke satu

- **Satu ke banyak (*one to many*)**, yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, akan tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A, seperti pada gambar 2.9 di bawah ini :



Gambar 2.8 Relasi Satu ke Banyak

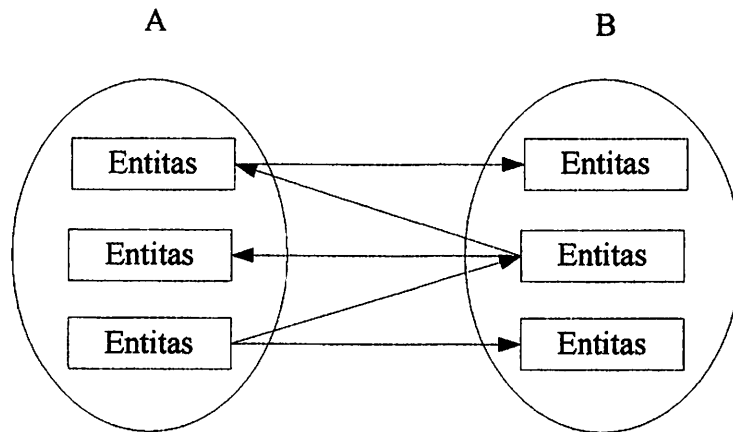
- **Banyak ke satu (*many to one*)**, yang berarti setiap pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B, akan tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B, seperti pada gambar 2.10 di bawah ini :



Gambar 2.9 Relasi Banyak ke Satu

- **Banyak ke banyak (*many to many*)**, yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian juga sebaliknya, dimana setiap

entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A, seperti pada gambar 2.11 di bawah ini :



Gambar 2.10 Relasi Banyak ke Banyak

Kardinalitas relasi satu ke banyak dan banyak ke satu dapat dianggap sama, karena tinjauan kardinalitas relasi selalu dilihat dari satu sisi (dari himpunan entitas A ke himpunan entitas B dan dari himpunan entitas B ke himpunan entitas A).

2.7 Borland Delphi 7

Borland Delphi merupakan *event driven programming* (pemrograman kendali kejadian) artinya program menunggu sampai adanya respon dari pemakai berupa *event* atau kejadian tertentu (tombol diklik, menu dipilih, dan lain-lain). Ketika *event* terdeteksi, kode yang berhubungan dengan *event* (prosedur *event*) akan dijalankan^[2].

Delphi merupakan sebuah piranti pengembangan Sistem Pakar berbasis *Windows* yang dikeluarkan oleh Borland *International*. Perangkat lunak ini sangat terkenal di lingkungan pengembang Sistem Pakar karena mudah untuk dipelajari dan dapat digunakan untuk menangani berbagai hal, dari Sistem Pakar matematika, permainan (*games*), hingga *database*. Pada penanganan *database*, Delphi menyediakan *fasilitas* yang memungkinkan pemrogram dapat berinteraksi dengan database seperti interbase, dBase, Paradox, Oracle, MySQL, dan Access.

2.8 Interbase 6.5 SQL Server

Interbase merupakan program Sistem Pakar *database* untuk menangani dan mengelola database oleh sebuah perangkat lunak yang sangat terkenal, yaitu Borland. Interbase apat ditemukan dalam satu paket dengan program Dephi. Tujuannya adalah agar pemakai lebih mudah mengakses data. Delphi merupakan program Sistem Pakar *database* berbasis *Windows* yang menyertakan banyak komponen untuk mengakses *database* dan mempresentasikan isi dari informasi. Interbase *server* adalah suatu program Sistem Pakar untuk para administrator sistem yang bertanggung jawab untuk pengoprasian dan pemeliharaan *server database*. Interbase bermanfaat untuk pengembang Sistem Pakar yang ingin memahami lebih banyak tentang Interbase.

Interbase dipakai untuk mengelola database Interbase lokal atau *remote*. Interbase memiliki beberapa fasilitas dalam pengelolaan *database*, yaitu dapat:

- Mengatur sekuritas *database* (menambah user, mengubah *password*, dan membuang *user*).
- Membuat backup *database*.
- Melakukan pemeliharaan *database*, menampilkan *database*, dan menjalankan operasi-operasi lain yang berhubungan dengan *database*.
- Memulai dan mengakhiri hubungan *database*.

BAB III

ANALISA DAN PERANCANGAN

Pada bab ini dijelaskan mengenai analisis dan perancangan sistem aplikasi. Analisis ditujukan untuk memberikan gambaran secara umum terhadap aplikasi. Hal ini berguna untuk menunjang perancangan aplikasi yang akan dikembangkan sehingga kebutuhan akan aplikasi tersebut dapat diketahui sebelumnya. Kemudian hasil analisis akan menjadi dasar untuk melakukan perancangan atau desain aplikasi sesuai kebutuhan sistem.

Dalam merancang aplikasi pada proyek akhir ini terlebih dahulu dilakukan pembuatan desain proses, desain data, serta desain antar muka aplikasi. Desain proses berguna untuk mengintegrasikan semua proses yang terjadi dalam aplikasi yang akan dibuat. Desain data berguna untuk mengetahui data apa saja yang dibutuhkan dalam proses yang akan dikerjakan. Sedangkan perancangan antarmuka berfungsi sebagai antar muka interaksi antara pengguna dengan sistem aplikasi yang dibuat, sehingga pengguna dapat mengoperasikan aplikasi yang dibuat.

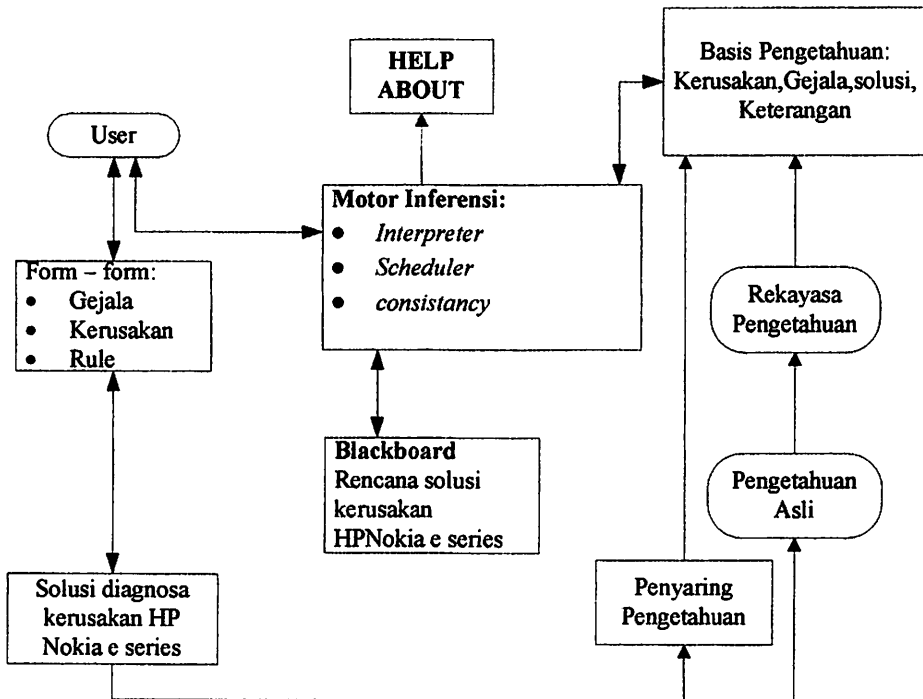
Materi Penelitian

Data diambil dari Nokia *Service Center*, *Twin Service Center* dan literature-literatur seperti buku (*textbook*), paper, dan sumber ilmiah lain seperti *website*, artikel, dan dokumen teks, yaitu :

- Data gejala-gejala kerusakan pada handphone Nokia E-Series.
- Data penanganan kerusakan pada handphone Nokia E-Series.

3.1 Blok diagram sistem pakar

Blok Diagram Sistem Pakar Kerusakan Handphone Nokia E-Series dapat di lihat pada gambar 3.1 di bawah ini :



Gambar 3.1 Blok Diagram Sistem Pakar Kerusakan Handphone Nokia E-Series

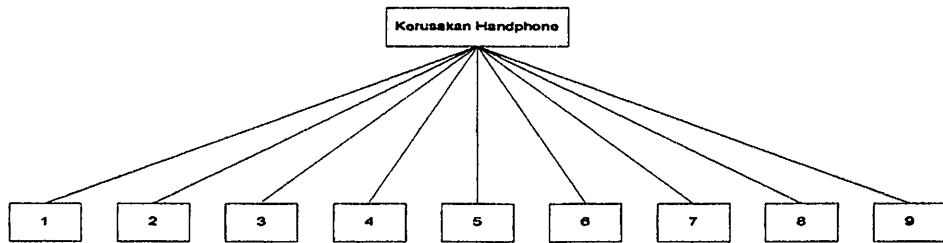
Dengan penjelasan masing-masing komponen sebagai berikut :

1. Sub sistem penambahan pengetahuan. Bagian ini digunakan untuk memasukkan pengetahuan, mengkonstruksi atau memperluas pengetahuan dalam basis pengetahuan antara lain kerusakan, gejala kerusakan, cara pengecekan dan solusi.
2. Motor inferensi (*inference engine*). Program yang berisi metodologi yang digunakan untuk melakukan penalaran terhadap informasi-informasi dalam basis pengetahuan dan blackboard, serta digunakan untuk memformulasikan konklusi. Ada 3 elemen utama dalam motor inferensi, yaitu :

- *Interpreter* : mengeksekusi masukan dari teknisi tentang gejala yang muncul dengan menggunakan aturan-aturan dalam basis pengetahuan yang sesuai.
 - *Scheduler* : akan mengontrol agenda dari gejala-gejala yang diinputkan.
 - *Consistency* : akan berusaha memelihara kekonsistenan dalam mempresentasikan solusi kerusakan handphone Nokia E-Series.
3. *Blackboard*. Merupakan area dalam memori yang digunakan untuk merekam kejadian yang sedang berlangsung termasuk keputusan sementara. Ada 3 tipe keputusan yang dapat direkam, yaitu :
- Rencana : Teknisi memasukkan gejala kerusakan yang terlihat.
 - Agenda : Reaksi dari sistem pakar untuk menanggapi aksi dari gejala yang sedang dimasukkan oleh teknisi.
 - Solusi : Program sistem pakar memberikan cara pengecekan, keterangan dan solusi yang sesuai dengan *base knowledge*.
4. *Form-form* digunakan untuk media komunikasi antara user dan program antara lain *form* diagnosa kerusakan handphone Nokia E-Series.

3.2 *Tree* Sistem Pakar

Tree sistem pakar diagnosa kerusakan handphone Nokia E-Series, seperti pada gambar 3.2 di bawah ini :



Gambar 3.2 *Tree* Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series

Keterangan dari *tree* Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series ini dapat di lihat pada tabel 3.1 di bawah ini :

Table 3.1 Table Keterangan *Tree*

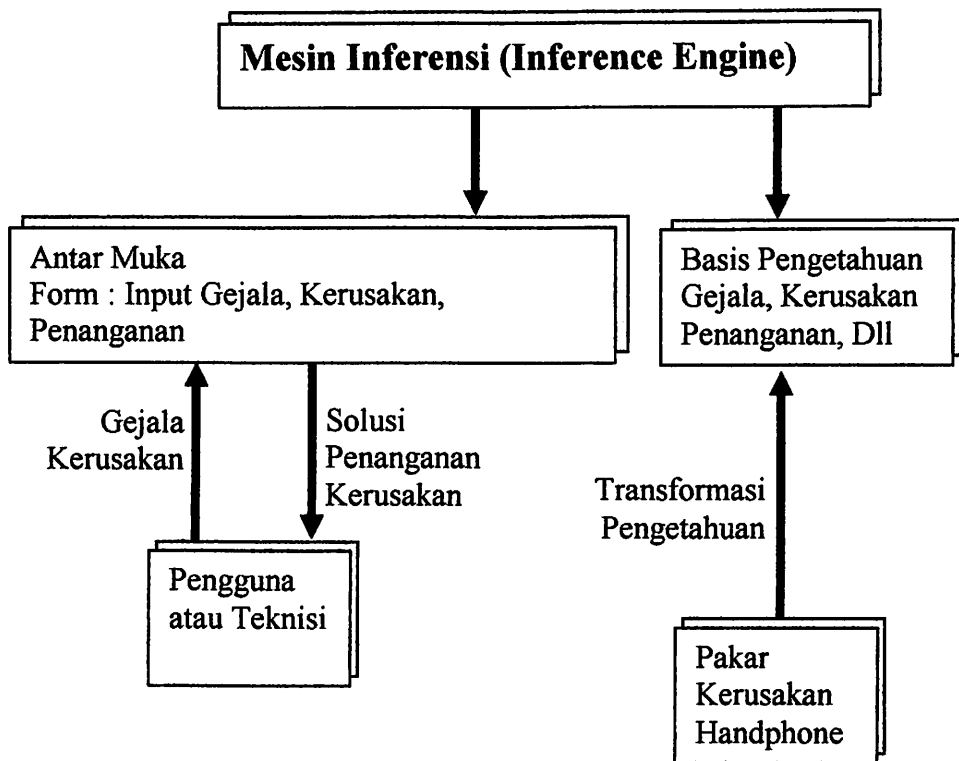
Kode	Kerusakan
1	Mati Total
2	No Signal
3	No Charger
4	No Voice
5	No Ring
6	Kamera Fault
7	No Radio
8	Kerusakan SIM CARD
9	Keypad Error

3.3 Perancangan *Rule Base* (Basis Aturan)

Digunakannya teknik representasi pengetahuan berbasis aturan atau rule dikarenakan lebih mudah dipahami oleh *knowledge engineer* dibandingkan dengan teknik representasi pengetahuan yang lain.

3.4 Mesin Inferensi

Blok Diagram Mesin Inferensi Kerusakan Handphone Nokia E-Series dapat di lihat pada gambar 3.3 :



Gambar 3.3 Blok Diagram Mesin Inferensi Kerusakan Handphone Nokia E-Series

3.5 Pemilihan Inferensi

Dalam sebuah implementasi sistem pakar, perlu dipilih metode penelusuran fakta apakah menggunakan *forward chaining* atau *backward chaining*, pemilihan ini berfungsi untuk memberikan kemudahan dalam menyusun mesin inferensi.

3.6 *Forward Chaining*

Forward chaining merupakan pencocokkan fakta atau pernyataan dimulai dari bagian sebelah kiri (*IF* dulu). Dengan kata lain, penalaran dimulai dari fakta yakni gejala-gejala kerusakan handphone terlebih dahulu untuk memperoleh kesimpulan berupa kerusakan komponen apa yang dialami pada handphone.

Mengapa menggunakan *forward chaining* :

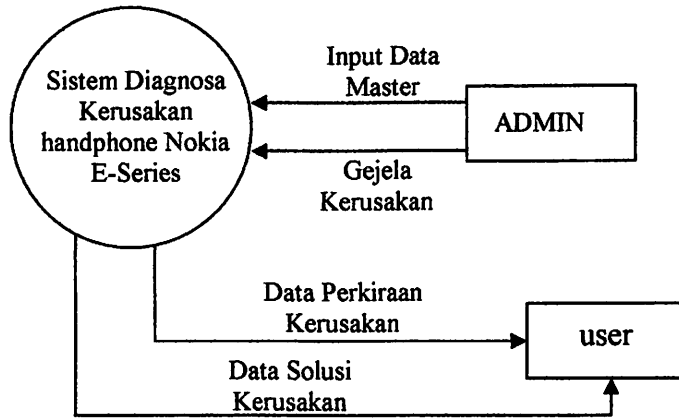
- Karena adanya perencanaan, monitoring, kontrol.
- Data memandu, penalaran step by step terhadap kerusakan handphone yang umum sampai kepada kerusakan yang paling tepat.
- Bekerja ke depan untuk mendapatkan solusi apa yang mengikuti fakta.
- Pencarian menggunakan *Depth-First Search* sehingga bisa mudah step by step.

3.7 Perancangan Perangkat Lunak

Perancangan perangkat lunak digunakan untuk mempermudah gambaran pembuatan program dengan menggunakan fungsional model antara lain *context diagram*, *data flow diagram*, *er-model* dan *flowchart*.

3.8 Diagram Konteks

Context Diagram adalah alur data informasi dan proses sistem informasi kerusakan handphone. Dalam *Context Diagram* ini terdapat beberapa alur data yang masuk dan keluar dari berbagai *entity* yang terkait dengan proses kerusakan handphone atau sistem informasi tersebut. *Context Diagram* bisa juga disebut dengan *Top level* atau *Level* teratas. Dari *Context Diagram* ini kemudian akan digambarkan dengan lebih rinci lagi ke *DFD level 1* yang biasa disebut dengan *overview diagram*, seperti pada gambar 3.4 di bawah ini :

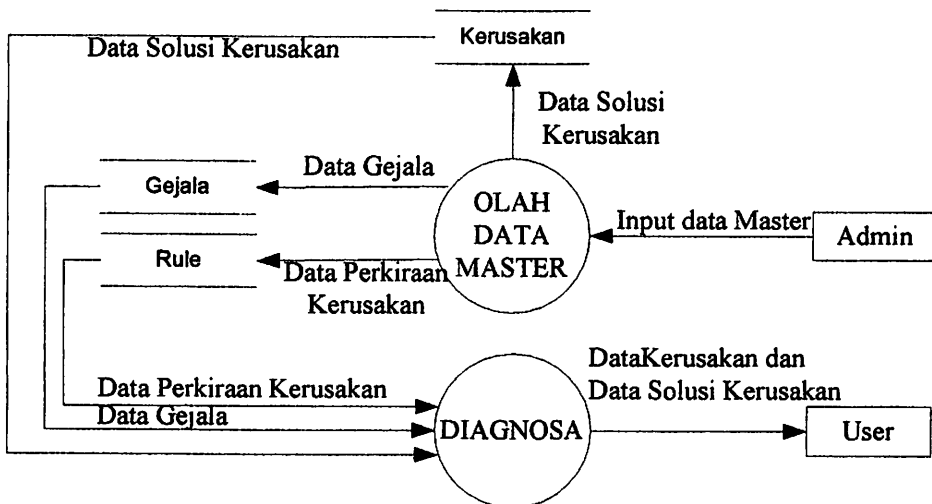


Gambar 3.4 *Contex Diagram*

Dalam sistem diagnosa kerusakan handphone Nokia E-Series secara umum dapat digambarkan pada *Context Diagram* tersebut. Kegiatannya yaitu menginputkan gejala kerusakan dan *rule* pada mesin inferensi, setelah *knowledgebase* tersusun teknisi dapat menginputkan kerusakan handphone untuk diagnosa, , seperti pada gambar 3.5 di bawah ini :

3.9 DFD Level 1

DFD level 1 ini merupakan penjabaran dari *Contex Diagram*.

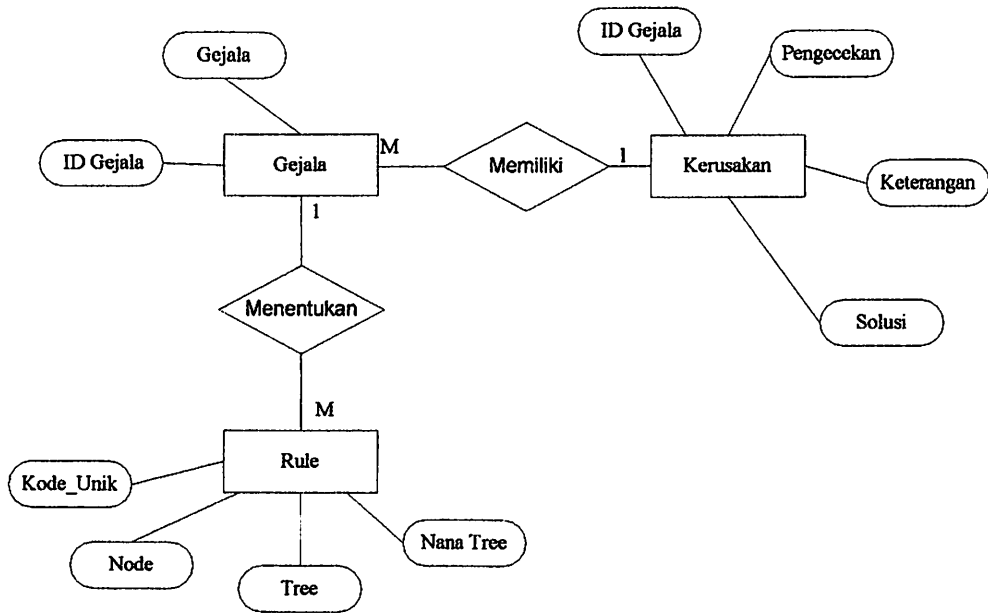


Gambar 3.5 *DFD level 1*

3.10 Entity Relationship Model (ER Diagram)

Relasi antar tabel pada *database* sistem pakar kerusakan handphone

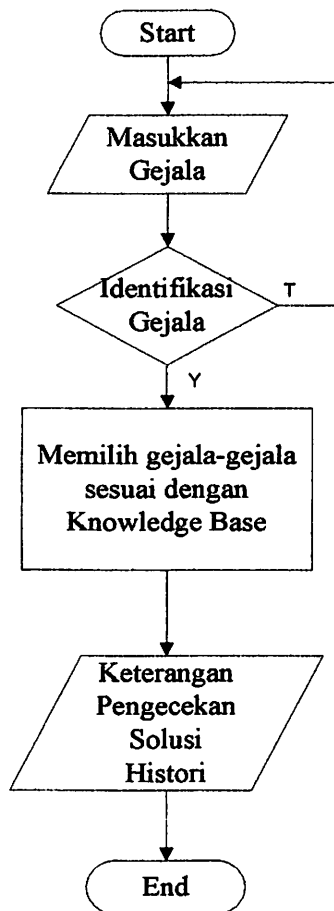
Nokia E-Series seperti yang ditunjukkan pada gambar 3.6 berikut ini :



Gambar 3.6 ER Diagram

3.11 Flowchart Sistem

Flowchart sistem dapat di lihat , seperti pada gambar 3.7 di bawah ini :



Gambar 3.7 Flowchart Sistem

3.12 Tabel Database

Table *database* digunakan untuk menyimpan data gejala, kerusakan dan *rule*.

3.12.1 Tabel Gejala

Tabel 3.2 berfungsi untuk menyimpan data gejala pada handphone :

Tabel 3.2 Gejala

No	Field Name	Type	Size	Kunci
1	ID Gejala	Char	10	√
2	Gejala	Char	100	

3.12.2 Tabel Kerusakan

Tabel 3.3 berfungsi untuk menyimpan data kerusakan dan solusi mengatasinya pada handphone :

Tabel 3.3 Kerusakan

No	Field Name	Type	Size	Kunci
1	ID_Gejala	Char	10	√
2	Keterangan	String	100	
3	Pengecekan	String	100	
4	Solusi	String	100	

3.12.3 Tabel Rule

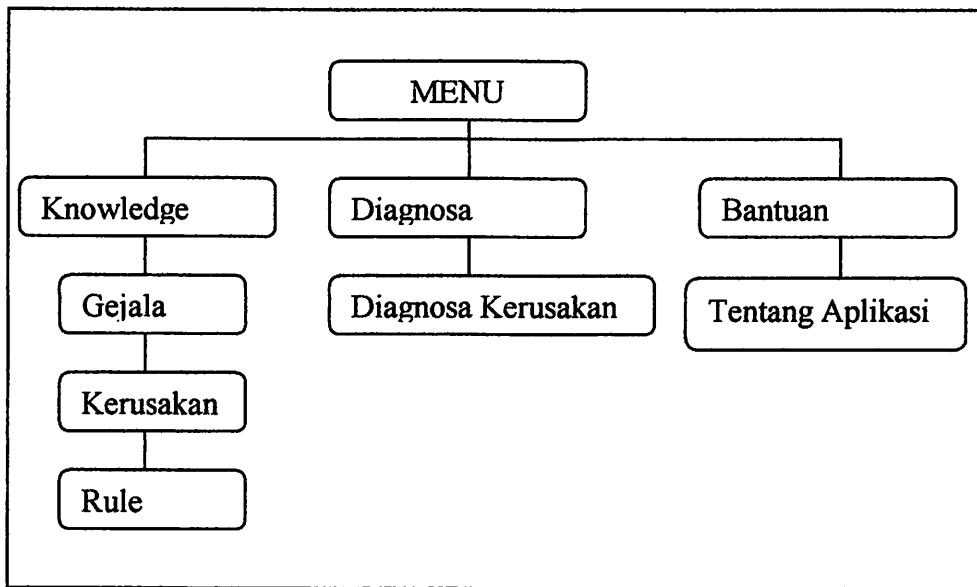
Tabel 3.4 berfungsi untuk menyimpan data *rule* :

Tabel 3.4 Rule

No	Field Name	Type	Size	Kunci
1	Kode_unik	Char	20	√
2	Nodes	Char	10	
3	Tree	Char	10	
4	Nama Tree	Char	100	

3.13 Site Map

Site map digunakan untuk memberikan penjelasan tentang menu-menu yang terdapat dalam sistem yang akan dibuat. Dari site map tersebut dapat diketahui alur-alur pengaksesan dari tiap-tiap halaman, seperti pada gambar 3.8 di bawah ini :



Gambar 3.8 *Site Map* Sistem Pakar Kerusakan Handphone Nokia E-Series

Berikut ini penjelasan *site map* dari Sistem Pakar Kerusakan Handphone Nokia E-Series secara keseluruhan :

1. *Knowledge Base* (basis pengetahuan)

a. Gejala

Digunakan untuk mensetting data gejala antara lain : isi gejala, edit gejala, laporan gejala dan hapus gejala.

b. Kerusakan

Digunakan untuk mensetting data kerusakan antara lain : isi kerusakan, edit kerusakan, laporan kerusakan dan hapus kerusakan.

c. *Rule*

Digunakan untuk mensetting data *rule* antara lain : isi *rule*, edit *rule*, laporan *rule* dan hapus *rule*.

2. Diagnosa

Diagnosa Kerusakan

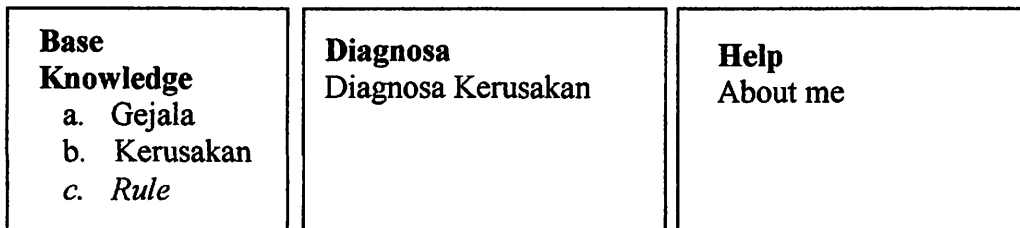
Digunakan untuk mendiagnosa kerusakan handphone dengan cara memilih gejala yang timbul selanjutnya menelusuri sesuai dengan alur yang telah diprogram.

3. Help

About Me

3.14 Layout Program

Layout merupakan desain tampilan program sistem pakar kerusakan handphone Nokia E-Series yang akan dibuat, seperti pada gambar 3.9 di bawah ini :



Gambar 3.9 Desain *Layout* Menu Utama

Gambar dari desain *Layout* Gejala dapat dilihat seperti pada gambar 3.10 di bawah ini :

The image shows a rectangular form titled "Isi Gejala". It contains two input fields: a shorter one labeled "Kode" and a longer one labeled "Gejala". Below these fields is a rounded rectangular button labeled "Simpan".

Gambar 3.10 Desain *Layout* Gejala

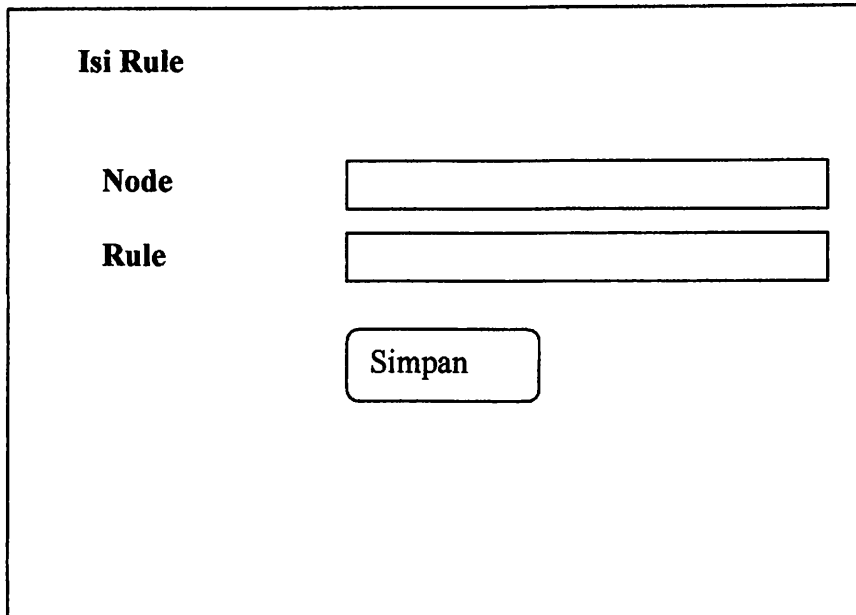
Gambar dari desain *Layout* Kerusakan dapat dilihat seperti pada gambar 3.11 di bawah ini :

The image shows a rectangular form titled "Isi Kerusakan". It contains two input fields: a shorter one labeled "Kode" and a longer one labeled "Kerusakan". Below these fields is a rounded rectangular button labeled "Simpan".

Gambar 3.11 Desain *Layout* Kerusakan

Gambar dari desain *Layout Rule* dapat dilihat seperti pada gambar

3.12 di bawah ini :



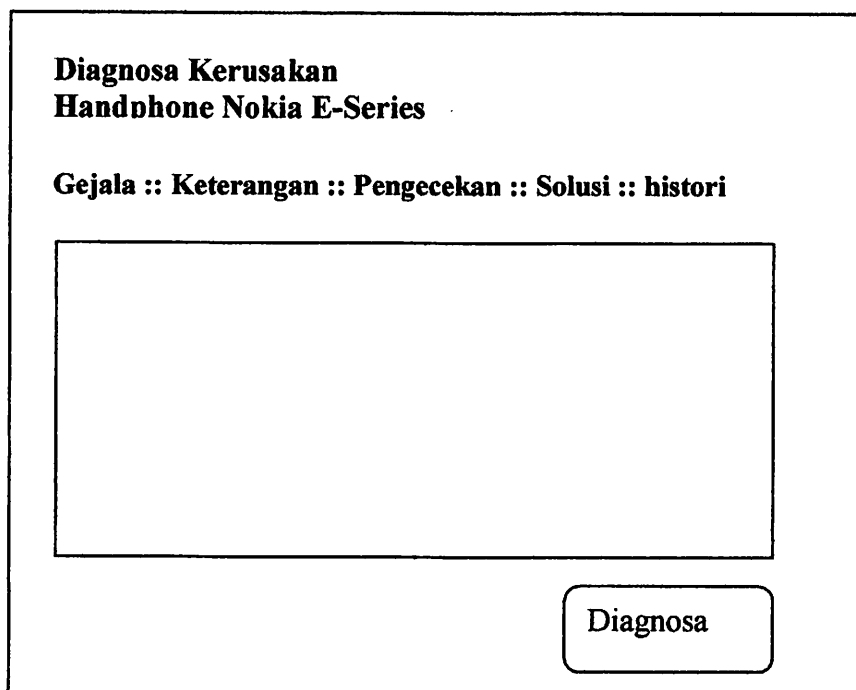
Isi Rule

Node

Rule

Gambar 3.12 Desain *Layout Rule*

Gambar dari desain *Layout Diagnosa Kerusakan Handphone* dapat dilihat seperti pada gambar 3.12 di bawah ini :



**Diagnosa Kerusakan
Handphone Nokia E-Series**

Gejala :: Keterangan :: Pengecekan :: Solusi :: histori

Gambar 3.13 Desain *layout Diagnosa Kerusakan Handphone*

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Implementasi Program

Dalam bab ini akan dibahas mengenai penerapan Pemrograman *Windows* yang menggunakan Borland Delphi 7.0 serta Interbase 6.5 sebagai media dalam pembuatan program aplikasi diagnosa kerusakan Handphone Nokia E-Series.

Penerapan program Perancangan dan Aplikasi Diagnosa kerusakan Handphone E-Series ini akan membantu teknisi handphone dalam mendiagnosa kerusakan handphone, dimana kerusakan handphone membutuhkan ketelitian yang sangat tinggi.

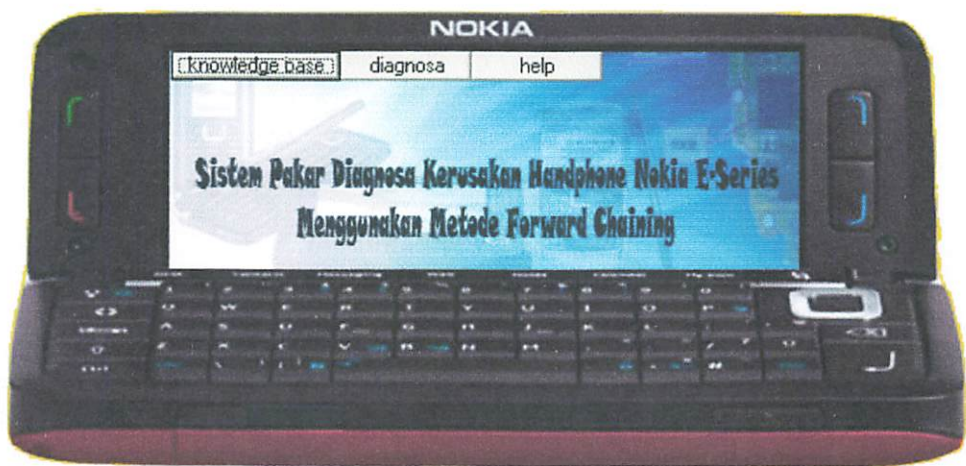
Program ini akan mendata semua kemungkinan gejala kerusakan serta solusi untuk mengatasi kerusakan handphone secara umum. seorang teknisi handphone dapat mengetahui perkiraan solusi dari gejala kerusakan handphone.

4.2 Penjelasan Program

Didalam menu-menu program ini hanya menerangkan tentang fungsi diagnosa kerusakan handphone sedangkan perintah pemrosesan program dapat dilihat pada lampiran *script* program. *form-form* yang ada dalam Perancangan dan Aplikasi Diagnosa Kerusakan Handphone E-Series Menggunakan Metode *Forward Chaining* dapat dilihat pada lampiran dibawah ini.

4.2.1 *Form* Menu Utama

Form ini digunakan untuk menampilkan menu-menu yang dapat digunakan *user* antara lain: Menu *Knowledge base*, Menu *Diagnosa*, , menu *Help*. Menu *Knowledge base* terdiri dari Menu *Gejala*, Menu *Kerusakan*, Menu *Rule*. Menu *Diagnosa* terdiri dari Menu *Diagnosa Kerusakan Handphone*, , Menu *Help* terdiri dari Menu *About*, seperti pada gambar 4.1 di bawah ini :



Gambar 4.1 *Form* Menu Utama

4.2.2 *Form* Gejala

Form ini merupakan *form* inputan gejala kerusakan Handphone yang terdiri dari dua pilihan yaitu *Isi/Edit Data* dan *View Data* dan *Delete*. Pada *form* ini berfungsi untuk menginputkan dan menghapus id gejala dan gejalanya, seperti pada gambar 4.2 di bawah ini :

Isi/Koreksi data Laporan/Delete

SETTING DATA GEJALA KERUSAKAN

Id Gejala

Gejala

Simpan

Gambar 4.2 *Form* Gejala

4.2.3 *Form* Kerusakan

Form Kerusakan merupakan uraian dari gejala dan solusi kerusakan Handphone. Pada *Form* ini terdapat Inputan Gejala, Inputan Keterangan, Inputan Pengecekan, Inputan Solusi, seperti pada gambar 4.3 di bawah ini :

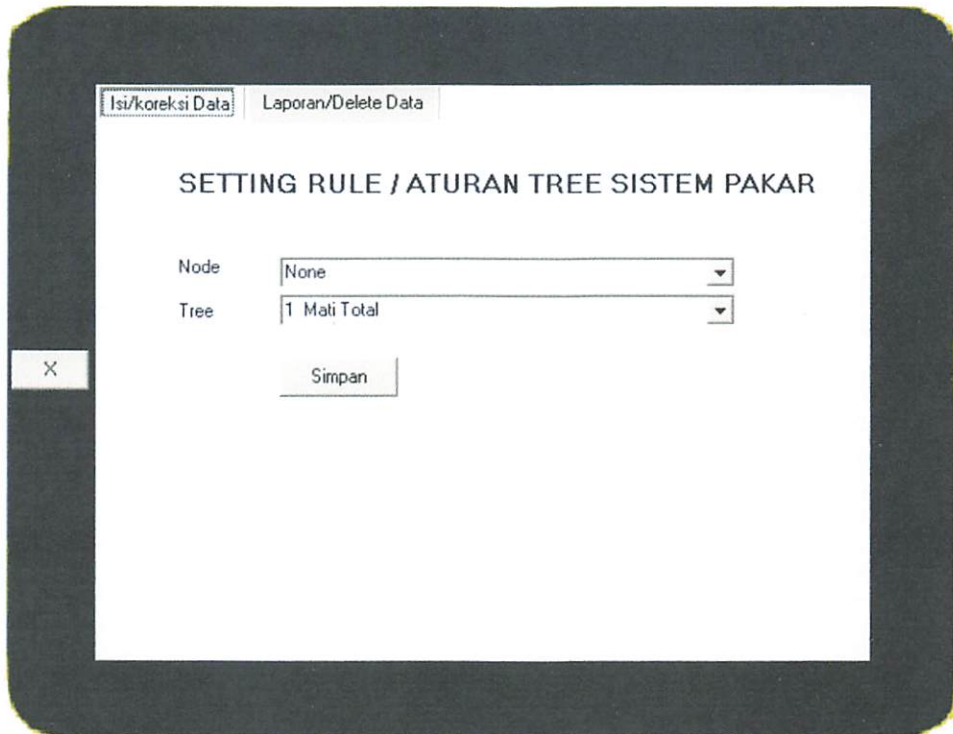
Gejala	1. Mati Total
Keterangan	Mati total adalah gangguan dimana HP tidak dapat berfungsi sama sekali
Pengecekan	Cek IC UI lalu Cek Swith On/off berikutnya Cek LCD dan IC Power. atau bisa saja kerusakan terjadi pada bagian-bagian IC UI maka Cek Vibrator, Buzzer, LED. cek juga Kaki-kaki Konektor seluruh bagian yang ada pada PCB dengan menggunakan Power Supply
Solusi	Buka casing pastikan kerusakan dan pilih salah satu kerusakan Pada Sub Menu mati Total

Simpan

Gambar 4.3 *Form Kerusakan*

4.2.4 *Form Rule*

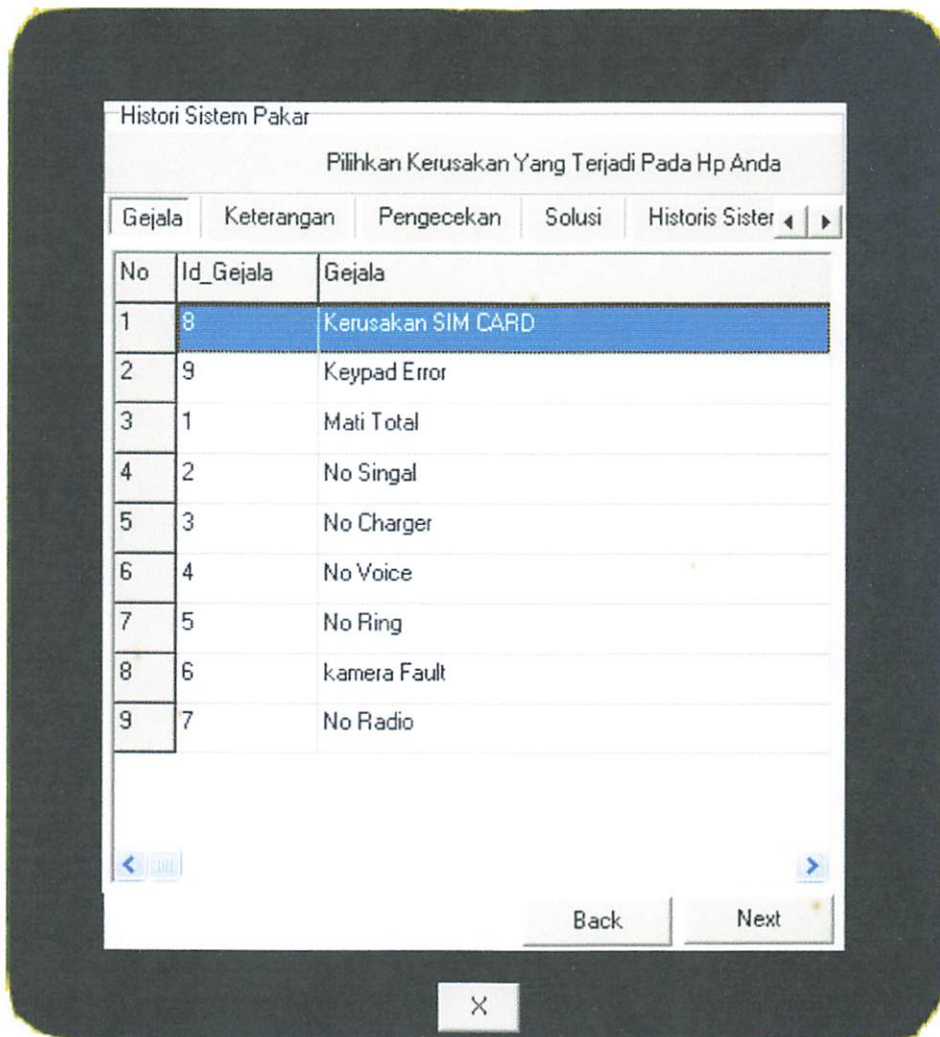
Form Rule Merupakan penentuan *tree* dari diagnosa kerusakan handphone Nokia E-Series, di form ini yang menentukan sebuah gejala untuk menjadi *node* dari *tree* dan sebaliknya. Di form ini juga dapat difungsikan untuk menghapus *node* dan *tree* yang sudah diinputkan, seperti pada gambar 4.4 di bawah ini :



Gambar 4.4 *Form Rule*

4.2.5 *Form Diagnosa*

Form diagnosa merupakan hasil dari diagnosa serta *tree* kerusakan handphone dari gejala yang dikeluhkan pelanggan. Di form ini dapat diketahui keterangan, pengecekan dan solusi dari gejala yang sudah diinputkan, seperti pada gambar 4.5 di bawah ini :



Gambar 4.5 Form Diagnosa

4.3 Pengujian Sistem

Pengujian sistem program utama sistem pakar untuk diagnosa kerusakan Handphone Nokia E-Series di lakukan untuk mengetahui keakuratan system dalam pengambilan keputusan kerusakan yang secara khusus pada Handphone Nokia E-Series berdasarkan gejala - gejala umum yang di inputkan ke program oleh pengguna(*user*).

Pengujian sistem dilakukan terhadap dua subject orang yang berprofesi sebagai teknisi handphone pemula. Pengujian system ini di uji

dengan dua objek handphone yang di sengajakan kerusakannya,yaitu pada tipe Nokia E-51 dan Nokia E60 dan tiga objek handphone lainnya yang bener kerusakannya (kerusakan tidak di sengaja).

Spesifikasi komputer yang digunakan untuk pembuatan sistem ini adalah :

Hardware:

- Processor : Intel Pentium IV
- Hardisk : 120 GHz
- Memory : 256 Mb
- Printer : Canon Pixma IP1700
- Keyboard, mouse, dan monitor

Software

Software yang digunakan dalam Perancangan & Sistem Pakar Diagnosa Kerusakan HP E-Series menggunakan metode forward chaining adalah :

- Windows XP Profesional Service pack 2
- Borland Delphi 7.0
- Interbase 6.5
- Photosop Cs 3
- Microsoft Word 2007
- Microsoft Visio 2007

4.3.1 Pengujian Sistem Pertama

Pada pengujian sistem yang pertama ini di lakukan diagnosa yang kerusakannya di sengaja oleh penguji.Di uji dengan dua objek yang

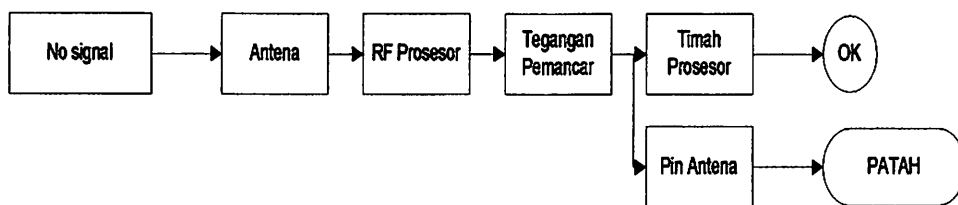
berbeda yaitu Nokia E-51 dan Nokia E60,dan dengan tempat pengujian yang berbeda yaitu Wijaya Cell dan Yoedz Cell.

- **Pengujian Sistem di Counter Wijaya Cell**

Kerusakan yang di diagnosa :

1. No Signal

Pemeriksaan kerusakan dengan diagnosa kerusakan *No Signal* dilakukan dengan rule program yang telah ditentukan pada object handphone Nokia E-51 dengan hasil pemeriksaaan, seperti pada gambar 4.6 di bawah ini :



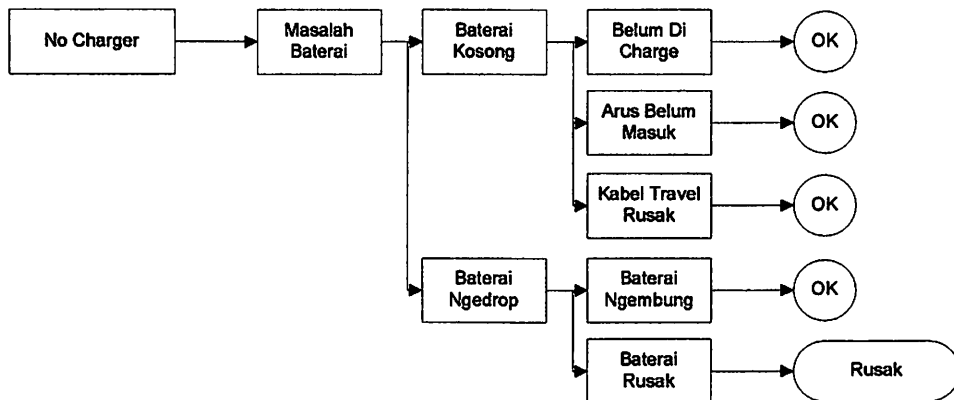
Gambar 4.6 Rule Diagnosa Kerusakan *No Signal* Pada Handphone Nokia E-51

Keterangan :

Setelah di lakukan diagnosa,di temukan masalah pada bagian Pin Antena (Patah),kemudian di lakukan perbaikan sesuai dengan solusi,handphone kembali dalam keadaan normal.

2. No Charger

Pemeriksaan kerusakan dengan diagnosa kerusakan *No Charger* dilakukan dengan rule program yang telah ditentukan pada object handphone Nokia E-51 dengan hasil pemeriksaaan, seperti pada gambar 4.7 di bawah ini :



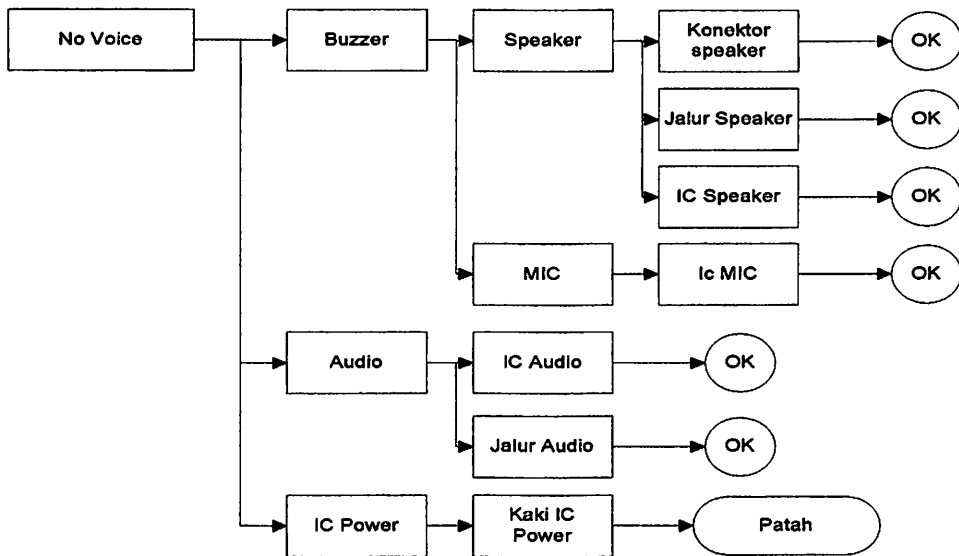
Gambar 4.7 Rule Diagnosa Kerusakan *No Charger* Pada Handphone Nokia E-51

Keterangan:

Setelah di lakukan diagnosa,di temukan masalah pada bagian Baterai (rusak),kemudian di lakukan perbaikan sesuai dengan solusi,handphone kembali dalam keadaan normal.

3.No Voice

Pengecekan kerusakan dengan diagnosa kerusakan *No Voice* dilakukan dengan rule program yang telah ditentukan pada object handphone Nokia E-51 dengan hasil pemeriksaaan, seperti pada gambar 4.8 di bawah ini :



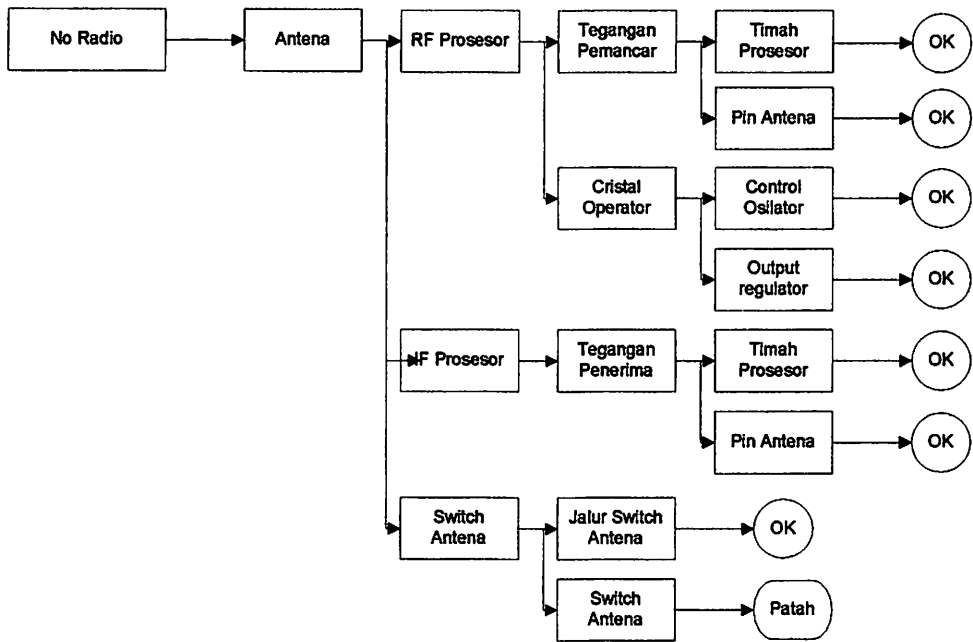
Gambar 4.8 Rule Diagnosa Kerusakan *No Voice* Pada Handphone Nokia E-51

Keterangan:

Setelah di lakukan diagnosa,di temukan masalah pada bagian kaki *IC Power* (Patah),kemudian di lakukan perbaikan sesuai dengan solusi,handphone kembali dalam keadaan normal.

4.No Radio

Pemeriksaan kerusakan dengan diagnosa kerusakan *No Radio* dilakukan dengan rule program yang telah ditentukan pada object handphone Nokia E-51 dengan hasil pemeriksaaan, seperti pada gambar 4.9 di bawah ini :



Gambar 4.9 Rule Diagnosa Kerusakan *No Radio* Pada Handphone Nokia E-51

Keterangan:

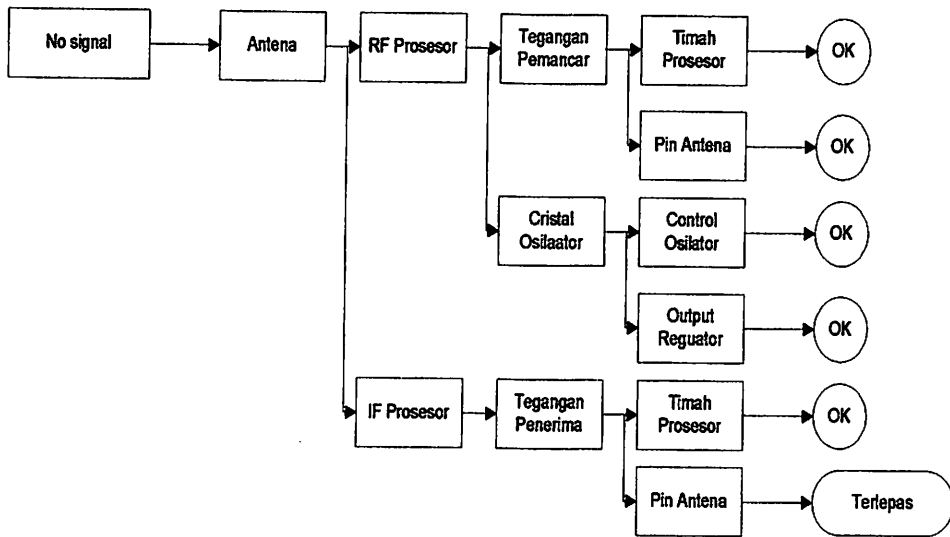
Setelah di lakukan diagnosa,di temukan masalah pada bagian *switch* antena (Patah),kemudian di lakukan perbaikan sesuai dengan solusi,handphone kembali dalam keadaan normal.

- **Pengujian Sistem di *Counter Yoedz Cell***

Kerusakan yang di diagnosa :

1. ***No Signal***

Pemeriksaan kerusakan dengan diagnosa kerusakan *No Signal* dilakukan dengan rule program yang telah ditentukan pada object handphone Nokia E-60 dengan hasil pemeriksaan, seperti pada gambar 4.10 di bawah ini :



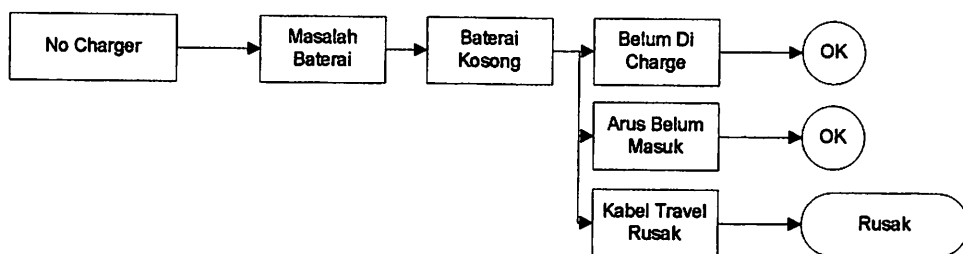
Gambar 4.10 Rule Diagnosa Kerusakan *No Signal* Pada Handphone Nokia E-60

Keterangan:

Setelah di lakukan diagnosa,di temukan masalah pada bagian Pin Antena (terlepas),kemudian di lakukan perbaikan sesuai dengan solusi,handphone kembali dalam keadaan normal.

2. No Charger

Pemeriksaan kerusakan dengan diagnosa kerusakan *No Charge* dilakukan dengan rule program yang telah ditentukan pada object handphone Nokia E-60 dengan hasil pemeriksaan, seperti pada gambar 4.11 di bawah ini :



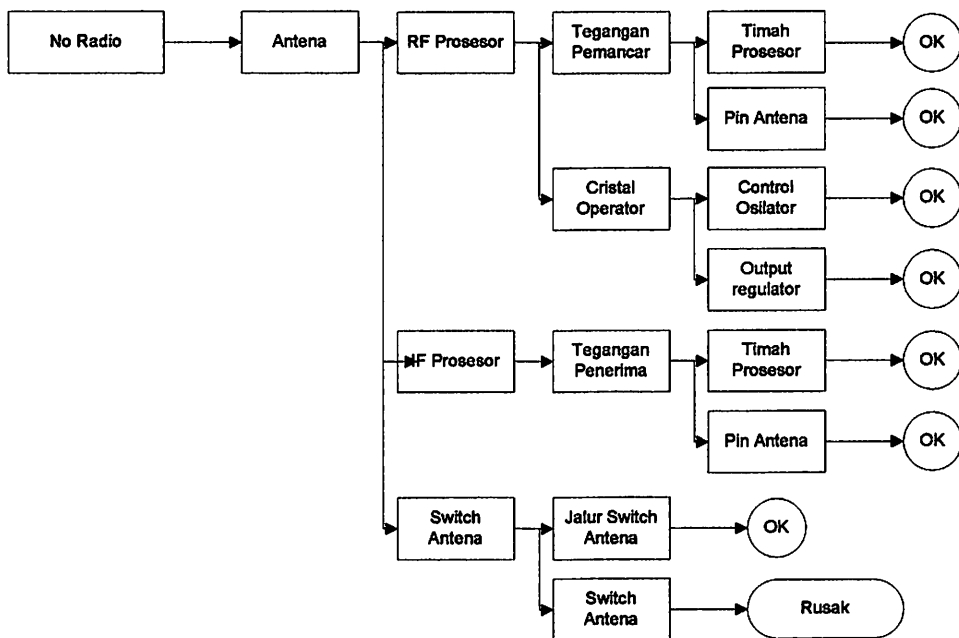
Gambar 4.11 Rule Diagnosa Kerusakan *No Charger* Pada Handphone Nokia E-60

Keterangan:

Setelah di lakukan diagnosa,di temukan masalah pada bagian kabel travel (rusak),kemudian di lakukan perbaikan sesuai dengan solusi,handphone kembali dalam keadaan normal.

3.No Radio

Pemeriksaan kerusakan dengan diagnosa kerusakan *No Radio* dilakukan dengan rule program yang telah ditentukan pada object handphone Nokia E-60 dengan hasil pemeriksaan, seperti pada gambar 4.12 di bawah ini :



Gambar 4.12 Rule Diagnosa Kerusakan *No Radio* Pada Handphone Nokia E-60

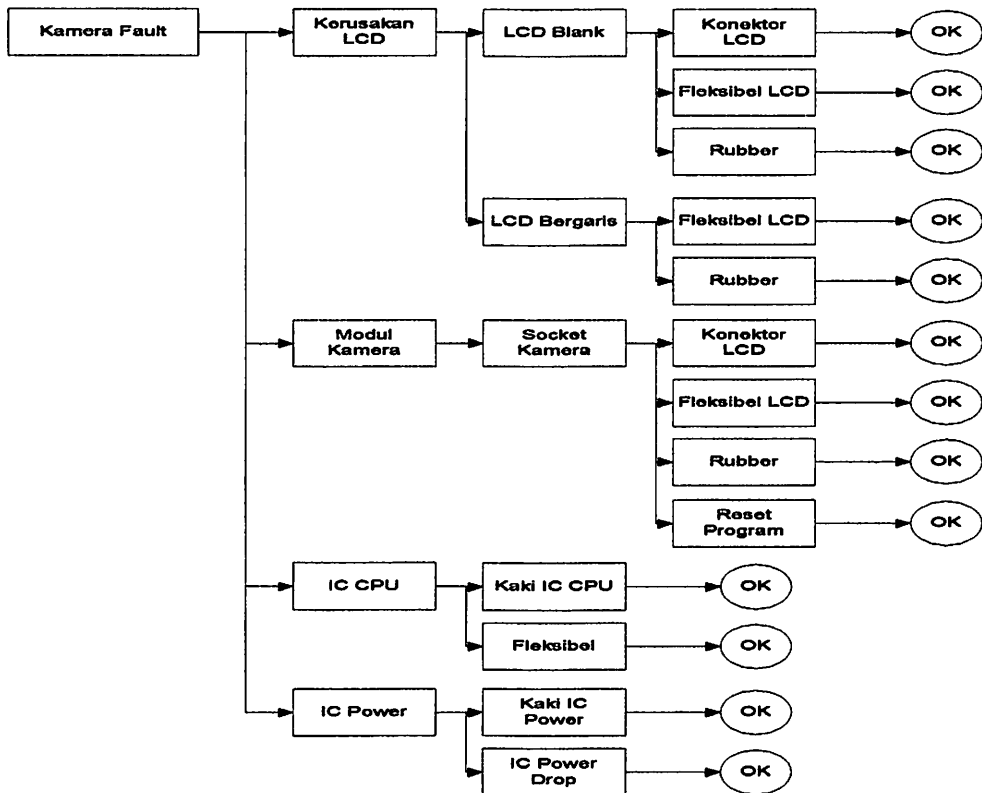
Keterangan:

Setelah di lakukan diagnosa,di temukan masalah pada bagian *Switch* Antena (rusak),kemudian di lakukan perbaikan sesuai dengan solusi,handphone kembali dalam keadaan normal.

4.Kamera Fault

Pemeriksaan kerusakan dengan diagnosa kerusakan Kamera *Fault*

dilakukan dengan rule program yang telah ditentukan pada object handphone Nokia E-60 dengan hasil pemeriksaaan, seperti pada gambar 4.13 di bawah ini :



Gambar 4.13 Rule Diagnosa Kerusakan Kamera *Fault* Pada Handphone Nokia E-60

Keterangan:

Pada diagnosa ini sistem tidak dapat menemukan kerusakan secara khusus,maka pada pengujian ini handphone masih belum dalam keadaan normal.Letak kerusakan sebenarnya adalah pada kameranya.

4.3.2 Pengujian Sistem Kedua

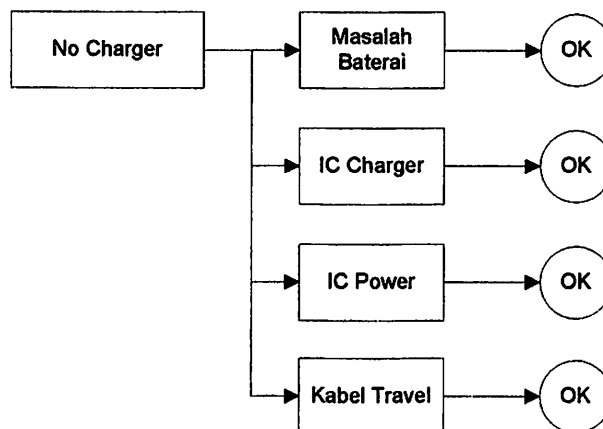
Pada pengujian sistem yang kedua ini dilakukan diagnosa yang kerusakannya tidak ada unsur kesengajaan, handphone yang di uji adalah handphone customer pada *Wijaya Cell* dan *Yoedz Cell* yang pada saat di adakan pengujian pada kondisi rusak.

- **Pengujian Sistem di *Counter Wijaya Cell***

Kerusakan yang di diagnosa :

1. **Pengujian pada Handphone Nokia E-61**

Sesuai dengan keluhan dari pelanggan bahwa handphone dalam kondisi tidak ada arus yang masuk ke baterai maka di lakukan diagnosa dengan *rule*, seperti pada gambar 4.14 di bawah ini :



Gambar 4.14 Rule Diagnosa Kerusakan *No Charger* Pada Handphone Nokia E-61

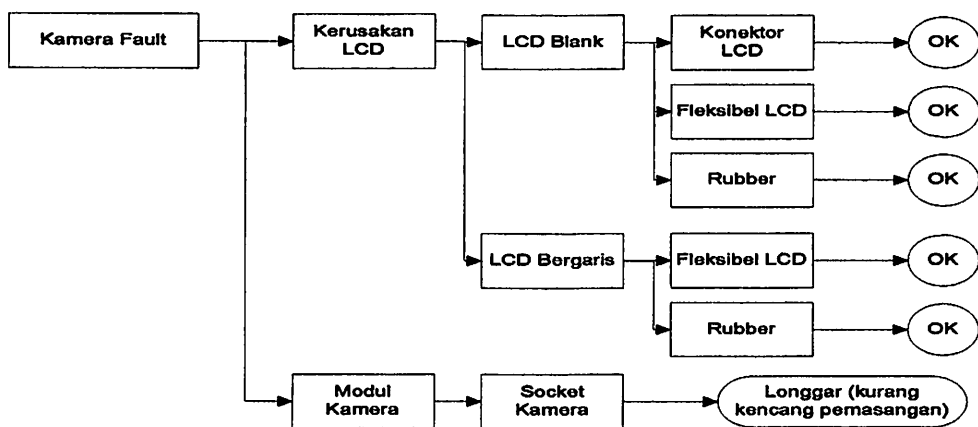
Keterangan:

Pada diagnosa ini sistem tidak dapat menemukan kerusakan secara khusus, maka pada pengujian ini handphone masih belum dalam keadaan

normal. Letak kerusakan sebenarnya adalah *socket* inputan dari kabel travel.

2. Pengujian pada Handphone Nokia E-71

Sesuai dengan keluhan dari pelanggan bahwa handphone dalam kondisi kamera rusak maka dilakukan diagnosa dengan *rule*, seperti pada gambar 4.15 di bawah ini :



Gambar 4.15 Rule Diagnosa Kerusakan Kamera Fault Pada Handphone Nokia E-71

Keterangan:

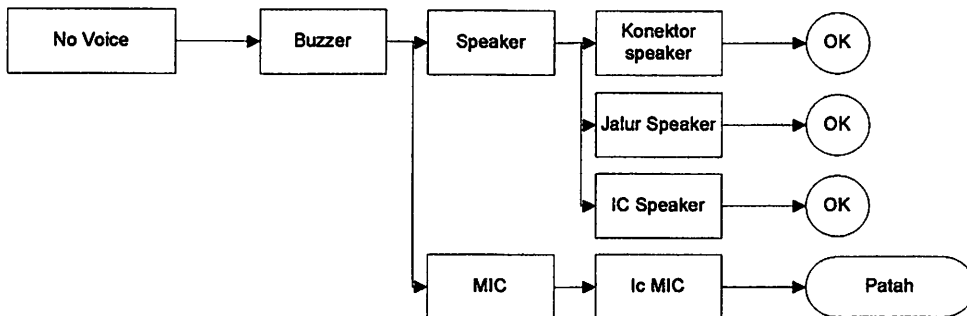
Setelah dilakukan diagnosa, ditemukan masalah pada bagian *socket* kamera (longgar), kemudian dilakukan perbaikan sesuai dengan solusi, handphone kembali dalam keadaan normal.

- **Pengujian Sistem di Counter Yoedz Cell**

Kerusakan yang di diagnosa :

1. Pengujian pada Handphone Nokia E-90

Sesuai dengan keluhan dari pelanggan bahwa handphone dalam kondisi pada saat menelpon, suara tidak dapat di dengarkan oleh telepon yang di tuju, maka di lakukan diagnosa dengan *rule*, seperti pada gambar 4.16 di bawah ini :



Gambar 4.16 *Rule* Diagnosa Kerusakan *No Voice* Pada Handphone Nokia E-90

Keterangan:

Setelah di lakukan diagnosa, di temukan masalah pada bagian *IC MIC* (Patah), kemudian di lakukan perbaikan sesuai dengan solusi, handphone kembali dalam keadaan normal.

Hasil dari pengujian dapat dilihat pada tabel 4.1 di bawah ini :

Tabel 4.1 Hasil Pengujian

No	Tipe Handphone	Pengujian	Keterangan
1	Nokia E-51	No Signal	Berhasil
2	Nokia E-51	No Charger	Berhasil
3	Nokia E-51	No Voice	Berhasil
4	Nokia E-51	No Radio	Berhasil
5	Nokia E-60	No Signal	Berhasil
6	Nokia E-60	No Charger	Berhasil
7	Nokia E-60	No Radio	Berhasil
8	Nokia E-60	Kamera Fault	Gagal
9	Nokia E-61	No Charger	Gagal
10	Nokia E-71	Kamera Fault	Berhasil
11	Nokia E-90	No Voice	Berhasil

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari Perancangan dan Implementasi Aplikasi Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series Menggunakan Metode *Forward Chaining* yang telah di lakukan, dapat di ambil kesimpulan sebagai berikut :

1. Aplikasi Sistem Pakar yang di buat telah di uji cobakan pada beberapa *counter* handphone dan dalam menganalisa kerusakan pada Nokia E-Series aplikasi dapat menganalisa secara baik, yaitu sebesar 81.8 % keberhasilan.
2. Dalam proses diagnosa kerusakan penentuan *rule* dari diagnosa sangat menentukan keakuratan diagnosa.
3. Dari hasil pengujian ada kegagalan dalam menganalisa kerusakan Handphone Nokia E-Series (tidak dapat menemukan kerusakan lebih khusus), hal ini dikarenakan kurangnya *database*.

5.2 Saran

Untuk memaksimalkan pemakaian program ini agar informasi dapat lebih akurat, disarankan agar :

1. Untuk pengembangan lebih lanjutnya aplikasi bisa ditambahkan gambar sesuai dengan solusi dari kerusakan yang di diagnosa.

Daftar Pustaka

- [1] Armin Irawan, "*Cara Praktis Jadi Teknisi Handphone : Software maupun Hardware*", CV Bintang Sakti, Surabaya. 2008.
- [2] Chandraleka Happy, "*Pemrograman Delphi 7.0*", Penerbit PT Elex Media Komputindo, Jakarta. 2003.
- [3] Dwi Budicahyono, "*Aplikasi handphone dengan FBUS dan Delphi 6.0*", Penerbit Andi Yogyakarta. 2006.
- [4] <http://cesstone.blogspot.com/2009/05/10-jenis-kerusakan-hp-yang-sering.html>
- [5] <http://europe.nokia.com/support/product-support/nokia-e61>
- [6] http://id.wikipedia.org/wiki/Nokia_Eseries
- [7] <http://nokiaeseriesphones.blogspot.com/>
- [8] <http://supriyan.staff.gunadarma.ac.id/Downloads/files/.../8-Sistem-Pakar.pdf>
- [9] <http://www.ponseljakarta.com/speknokia-eseries.htm>
- [10] <http://www.lintasberita.com/go/956924>
- [11] McLeod, "*Kerangka Dasar Sistem Informasi Manajemen Bagian I Pengantar*", Pustaka Binaman Pressindo, Jakarta, Indonesi. 2000
- [12] Pujiyanto, "*Praktis Belajar Borland Delphi 8.0 Bagi Pemula*", Penerbit Graha Ilmu, Yogyakarta. 2007.
- [13] Sonny Daniswara dan Riyan, "*Mencari dan Memperbaiki Kerusakan pada Handphone*", Kawan Pustaka.2007.
- [14] Yahya Yanuar Lukmanual Hakim, "*Pemrograman Delphi dengan Database Microsoft SQL Server*", Penerbit PT Elex Media Komputindo Kelompok Gramedia, Jakarta. 2006.



LAMPIRAN



**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

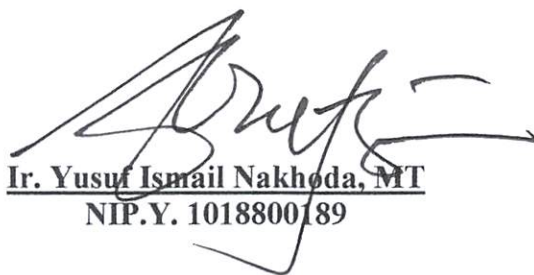
Nama : Harry Aulia Rahman
Nim : 05.12.507
Jurusan/konsentrasi : Teknik Elektro S-1 / Komputer dan Informatika
Judul Skripsi : Sistem Pakar Diagnosa Kerusakan Handphone Nokia E-Series
Menggunakan Metode *Forward Chaining*

Dipertahankan dihadapan Tim Penguji Ujian Skripsi jenjang program Strata Satu (S-1)

Pada Hari : Sabtu
Tanggal : 21 Agustus 2010
Dengan nilai : A (84,5) *84*


PANITIA UJIAN SKRIPSI

KETUA


Ir. Yusuf Ismail Nakhoda, MT
NIP.Y. 1018800189

ANGGOTA PENGUJI

Dosen Penguji I



M. Ibrahim Ashari, ST, MT.
NIP.P.1030100358

Dosen Penguji II



Sandy Nataly Mantja, SKom.



FORMULIR PERBAIKAN SKRIPSI

Nama : **HARRY AULIA RAHMAN**
Nim : **05.12.507**
Jurusan : **T. Elektro S-1**
Konsentrasi : **T. Komputer dan Informatika**
Masa Bimbingan : **30 Mei s/d 30 November 2010**
Judul Skripsi : **SISTEM PAKAR DIAGNOSA KERUSAKAN HANDPHONE
NOKIA E-SERIES MENGGUNAKAN METODE *FORWARD
CHAINING***

Tanggal	Penguji	Uraian	Paraf
21 Agustus 2010	I	1. Tentukan kalimat pengantar pada gambar dan tabel. 2. Daftar pustakaurut abjad.	
	II	1. Tambahkan penyelesain masalah pada abstrak. 2. Sistem informasi pada BAB II. 3. Perbaiki DFD, ERD dan Flowchart. 4. Tabel 3.12.2 ganti type BLOB. 5. Tambahkan tabel hasil pengujian. 6. Kesimpulan dan saran.	

Disetujui,

Dosen Penguji I

M. Ibrahim Ashari, ST, MT.
NIP.P.1030100358

Dosen Penguji II

Sandy Nataly M, Skom

Mengetahui,

Dosen Pembimbing I

Ir. Yusuf Ismail Nakhoda, MT.
NIP. Y. 1018800189

Dosen Pembimbing II

Dr.Eng.Aryuanto Soetedjo,ST.,MT.
NIP. Y. 1030800417



FORMULIR BIMBINGAN SKRIPSI

Nama : HARRY AULIA RAHMAN
NIM : 05.12.507
Masa Bimbingan : 30 MEI 2010s/d 30 NOVEMBER 2010 *24*
Judul Skripsi : SISTEM PAKAR DIAGNOSA KERUSAKAN HANDPHONE NOKIA
E – SERIES MENGGUNAKAN METODE FORWARD CHAINING

No	Tanggal	Uraian	Paraf Pembimbing
1	Selasa 29/06/2010	Revisi bab I margin dari latar belakang, rumusan masalah, tujuan dan batasan masalah.	<i>[Signature]</i>
2	Rabu 30/06/2010	Revisi bab I dan II, penulisan bahasa inggris harus di miringkan. Revisi bab II, teori harus diberi indeks keterangan daftar pustaka.	<i>[Signature]</i>
3	Kamis 01/072010	Revisi bab III, spasi text dalam tabel harus 1 spasi.	<i>[Signature]</i>
4	Senin 05/07/2010	Revisi bab II,III,IV, keterangan gambar dan tabel tidak di bold. Revisi bab III,IV, keterangan tabel harus di atas dan center.	<i>[Signature]</i>
5	Jumat 09/07/2010	Revisi bab IV, rapikan margin pada tiap sub bab. Revisi bab V, kata – kata dari kesimpulan dibuat lebih detail lagi.	<i>[Signature]</i>
6	Selasa 13/07/2010	Revisi bab IV, kata – kata di penulisan pengujian sistem kurang tepat.	<i>[Signature]</i>
7	Sabtu 17/07/2010	Revisi bab V, Kesimpulan harus lebih detail,dari hasil pengujian. Revisi Daftar pustaka, format penulisan nomor.	<i>[Signature]</i>
8	Jum'at 06/08/2010	Revisi makalah hasil, rapikan margin dan format penulisan abstraksi.	<i>[Signature]</i>
9	Sabtu 07/08/2010	Acc makalah Seminar Hasil	<i>[Signature]</i>
10	Kamis 19/08/2010	Acc Laporan Skripsi	<i>[Signature]</i>

Malang,
Dosen Pembimbing I

[Signature]
Ir. Yusuf Ismail Nakhoda,MT
NIP.Y.1018800189



FORMULIR BIMBINGAN SKRIPSI

Nama : HARRY AULIA RAHMAN
NIM : 05.12.507
Masa Bimbingan : 30 MEI 2010s/d 30 NOVEMBER 2010 *30*
Judul Skripsi : SISTEM PAKAR DIAGNOSA KERUSAKAN HANDPHONE NOKIA
E – SERIES MENGGUNAKAN METODE FORWARD CHAINING

No	Tanggal	Uraian	Paraf Pembimbing
1	Selasa 29/06/2010	Revisi bab I, tujuan dan batasan masalah.	
2	Kamis 01/07/2010	Acc Bab I,II	
3	Senin 05/07/2010	Revisi Bab III, penulisan keterangan tabel di atas.	
4	Jumat 09/07/2010	Acc bab III Revisi Bab IV, pengujian sistem	
5	Senin 02/08/2010	Revisi Bab IV, penulisan kata – kata pengujian sistem kurang tepat	
6	Selasa 03/08/2010	Acc Bab IV Revisi Bab V, kesimpulan kurang lengkap	
7	Rabu 04/08/2010	Acc Bab V Revisi makalah hasil, kesimpulan kurang lengkap	
8	Kamis 05/08/2010	Acc Makalah Seminar Hasil	
9	Selasa 17/08/2010	Acc Laporan Skripsi	
10			

Malang,
Dosen Pembimbing II

Dr. Eng. Aryuanto Soetedjo, ST, MT
NIP. Y.1030800417

Listing Program

Program Menu Utama

```
procedure TFmain.Gejala1Click(Sender: TObject);
begin
    Application.CreateForm(TFgejala, Fgejala);
    FGejala.ShowModal;
end;

procedure TFmain.Rule1Click(Sender: TObject);
begin
    Application.CreateForm(TFrule, Frule);
    Frule.ShowModal;
end;

procedure TFmain.Kerusakan1Click(Sender: TObject);
begin
    Application.CreateForm(TFkerusakan, Fkerusakan);
    Fkerusakan.ShowModal;
end;

procedure TFmain.FormShow(Sender: TObject);
begin
    Image1.Align:=AlClient;
end;

procedure TFmain.LogOut1Click(Sender: TObject);
begin
    Application.Terminate;
end;

procedure TFmain.Search2Click(Sender: TObject);
begin
    Application.CreateForm(TFTree, FTree);
    FTree.ShowModal;
```

```

end;

procedure TFmain.FormClose(Sender: TObject; var Action:
TCloseAction);
begin
    Application.Terminate;
    Action:=caFree;
end;

procedure TFmain.PencarianGejala1Click(Sender: TObject);
begin
    Application.CreateForm(TFCari, FCari);
    FCari.ShowModal;
end;

```

Program Gejala

```

procedure TFgejala.Button1Click(Sender: TObject);
begin
    If Edit1.text<>' ' Then
        Begin
            If Fdm.Gejala.Findkey([Edit1.text]) Then
                Begin
                    Fdm.Gejala.Edit;
                End Else
                Begin
                    Fdm.Gejala.Insert;
                End
            Fdm.Gejala.FieldName('Id_Gejala').AsString:=Edit1.Text;
            End;
            Fdm.Gejala.FieldName('Gejala').AsString:=Edit2.Text;
            Fdm.Gejala.Post;
        End
    End;

```

```

Fdm.Gejala.ApplyUpdates;

Fdm.Gejala.CommitUpdates;

Fdm.Gejala.Refresh;

Edit1.text:='';

Edit2.text:='';

Edit1.SetFocus;

End;

end;

Procedure TFgejala.Edit1Exit(Sender: TObject);

begin
  If Fdm.Gejala.Findkey([Edit1.text]) Then
    Begin
      Edit2.Text:=Fdm.Gejala.FieldByName('Gejala').AsString;
    End Else
    Begin
      Edit2.Text:='';
    End;
end;

procedure TFgejala.Button2Click(Sender: TObject);

begin
  If Fdm.Gejala.Findkey([DbGrid1.Fields[0].AsString]) Then
    Begin
      Fdm.Gejala.Delete;
      Fdm.Gejala.ApplyUpdates;
      Fdm.Gejala.CommitUpdates;
      Fdm.Gejala.Refresh;
    End;
end;

Procedure TFgejala.FormClose(Sender: TObject; var Action:
TCloseAction);

begin

```

```
Action:=caFree;  
  
end;  
end.
```

Program Kerusakan

```
Procedure TFkerusakan.FormShow (Sender: TObject);  
  
begin  
  
    C1.Clear;  
  
    C1.Refresh;  
  
    Fdm.Gejala.First;  
  
    Repeat  
  
C1.Items.Add (Fdm.Gejala.FieldName ('Id_Gejala').AsString+#3  
2+#32+Fdm.Gejala.FieldName ('Gejala').AsString);  
  
        Fdm.Gejala.Next;  
  
    Until Fdm.Gejala.Eof;  
  
    C1.ItemIndex:=0;  
  
    C1Change (Sender);  
  
end;  
  
procedure TFkerusakan.C1Change (Sender: TObject);  
  
begin  
  
    If Fdm.Kerusakan.Findkey ([Fmain.HIntku (C1.Text)]) Then  
  
        Begin  
  
M1.Text:=Fdm.Kerusakan.FieldName ('Keterangan').AsString;  
  
M2.Text:=Fdm.Kerusakan.FieldName ('Pengecekan').AsString;  
  
        M3.Text:=Fdm.Kerusakan.FieldName ('Solusi').AsString;  
  
        End Else  
  
        Begin
```

```

        M1.Text:='';

        M2.Text:='';

        M3.Text:='';

    End;

end;

Procedure TFkerusakan.Button1Click(Sender: TObject);
begin
    If Fdm.Kerusakan.Findkey([Fmain.HIntku(C1.Text)]) Then
        Begin
            Fdm.Kerusakan.Edit;
        End Else
        Begin
            Fdm.Kerusakan.Insert;
            Fdm.Kerusakan.FieldName('Id_Gejala').AsString:=Fmain.HIntku
            u(C1.Text);
        End;

        Fdm.Kerusakan.FieldName('Keterangan').AsString:=M1.Text;

        Fdm.Kerusakan.FieldName('Pengecekan').AsString:=M2.Text;
            Fdm.Kerusakan.FieldName('Solusi').AsString:=M3.Text;
            Fdm.Kerusakan.Post;
            Fdm.Kerusakan.ApplyUpdates;
            Fdm.Kerusakan.CommitUpdates;
            Fdm.Kerusakan.Refresh;

            M1.text:='';
            M2.text:='';
            M3.text:='';

            C1.ItemIndex:= C1.ItemIndex+1;
            C1.SetFocus;

```

```
end;

Procedure TFkerusakan.FormClose(Sender: TObject; var Action:
TCloseAction);

begin

    Action:=caFree;

end;
```

Program Rule

```
procedure TFrule.FormCreate(Sender: TObject);

begin

    ComboBox1.Clear;

    ComboBox1.Refresh;

    ComboBox1.Items.Add('None');

    Fdm.Gejala.First;

    Repeat

ComboBox1.Items.Add(Fdm.Gejala.FieldName('Id_Gejala').
AsString);

        Fdm.Gejala.Next;

    Until Fdm.Gejala.Eof;

    Combobox1.ItemIndex:=0;

    End;

    ComboBox1.Clear;

    ComboBox1.Refresh;

    ComboBox1.Items.Add('None');

    Fdm.Gejala.First;

    Repeat

ComboBox1.Items.Add(Fdm.Gejala.FieldName('Id_Gejala').
AsString);

        Fdm.Gejala.Next;
```

```

Until Fdm.Gejala.Eof;

Combobox1.ItemIndex:=0;

  If Fdm.Gejala.Findkey([Combobox1.text]) Then
    Begin
      Memo2.Text:=Fdm.Gejala.FieldByName('Pertanyaan').
AsString;
      End;
end;

procedure TFrule.FormShow(Sender: TObject);
begin
  ComboBox1.Clear;
  ComboBox1.Refresh;
  ComboBox1.Items.Add('None');
  Fdm.Gejala.First;
  Repeat
  ComboBox1.Items.Add(Fdm.Gejala.FieldByName('Id_Gejala').
AsString+#32+#32+Fdm.Gejala.FieldByName('gejala').
AsString);
    Fdm.Gejala.Next;
  Until Fdm.Gejala.Eof;
  Combobox1.ItemIndex:=0;
  Combobox1Change(Sender);

  ComboBox3.Clear;
  ComboBox3.Refresh;
  ComboBox3.Items.Add('Non');
  Fdm.Gejala.First;

```

```

Repeat
ComboBox3.Items.Add(Fdm.Gejala.FieldByName('Id_Gejala').AsSt
ring+#32+#32+Fdm.Gejala.FieldByName('gejala').AsString);
    Fdm.Gejala.Next;
Until Fdm.Gejala.Eof;
Combobox3.ItemIndex:=0;
Combobox3Change(Sender);
end;
procedure TFrule.ComboBox1Change(Sender: TObject);
begin
    ComboBox2.Clear;
    ComboBox2.Refresh;
    Fdm.Gejala.First;
    Repeat
        If Fdm.Gejala.FieldByName('Id_Gejala').AsString<>
Fmain.Hintku(Combobox1.Text) Then
ComboBox2.Items.Add(Fdm.Gejala.FieldByName('Id_Gejala').
AsString+#32+#32+Fdm.Gejala.FieldByName('gejala').AsString);
        Fdm.Gejala.Next;
    Until Fdm.Gejala.Eof;
    Combobox2.ItemIndex:=0;
end;
procedure TFrule.Button1Click(Sender: TObject);
begin
If Combobox1.text<>' ' Then
    Begin
        If Fdm.Rule.Findkey([Fmain.Hintku(Combobox1.text)+Fmain.
Hintku(Combobox2.text)]) Then
            Begin
                Fdm.Rule.Edit;
            end;
        end;
    end;
end;

```

```

    End Else

    Begin

        Fdm.Rule.Insert;

Fdm.Rule.FieldName('Kode_unik').AsString:=Fmain.
Hintku(Combobox1.text)+Fmain.Hintku(Combobox2.text);

        End;

Fdm.Rule.FieldName('Nodes').AsString:=Fmain.Hintku
(Combobox1.Text);

Fdm.Rule.FieldName('Tree').AsString:=Fmain.Hintku
(Combobox2.Text);

Fdm.Rule.FieldName('Nama_Tree').AsString:=Fmain.
Hintmu(Combobox2.Text);

        Fdm.Rule.Post;

        Fdm.Rule.ApplyUpdates;

        Fdm.Rule.CommitUpdates;

        Fdm.Rule.Refresh;

        Combobox1.SetFocus;

    End;

end;

Procedure TFrule.ComboBox3Change(Sender: TObject);
begin
    Q1.Close;

    Q1.Sql.Clear;

    Q1.Sql.Add('Select * from Rule Where Kode_unik<>""');

    If Combobox3.text<>'Non' Then

        Begin

            Q1.Sql.add('And Nodes=:A');

Q1.ParamByName('A').AsString:=Fmain.Hintku(Combobox3.text);

```

```

End Else

Begin
    Q1.Sql.add('And Nodes=:A');
    Q1.ParamByName('A').AsString:='Non';
End;;

Q1.Prepare;

Q1.Open;

end;

Procedure TFrule.PageControl1Change(Sender: TObject);

begin
If PageControl1.ActivePageIndex=1 Then
ComboBox3Change(Sender);
end;

procedure TFrule.Button2Click(Sender: TObject);
begin
    If Fdm.Rule.FindKey([DbGrid1.Fields[0].AsString]) Then
        Begin
            Fdm.Rule.Delete;
            Fdm.Rule.ApplyUpdates;
            Fdm.Rule.CommitUpdates;
            Fdm.Rule.Refresh;
        End;
        ComboBox3Change(Sender);
end;

Procedure TFrule.FormClose(Sender: TObject; var Action:
TCloseAction);
begin
    Action:=caFree;
end;

```

```
end.
```

Program form diagnosa

```
Procedure TFTree.FormShow(Sender: TObject);  
  
Var  
  I : Integer;  
  
begin  
  Pengexcekan.ActivePageIndex:=0;  
  
  S1.Cells[0,0]:='No';  
  S1.Cells[1,0]:='Id_Gejala';  
  S1.Cells[2,0]:='Gejala';  
  
  S1.ColWidths[0]:=30;  
  S1.ColWidths[1]:=70;  
  S1.ColWidths[2]:=300;  
  
  S2.Cells[0,0]:='No';  
  S2.Cells[1,0]:='Id_Gejala';  
  S2.Cells[2,0]:='Gejala';  
  
  Alamat:=1;  
  
  S2.ROWCount:=Alamat+1;  
  
  S2.Cells[0,Alamat]:=Inttostr(Alamat);  
  S2.Cells[1,Alamat]:='Non';  
  S2.Cells[2,Alamat]:='Root Sistem pakar';  
  
  S2.ColWidths[0]:=30;  
  S2.ColWidths[1]:=0;  
  S2.ColWidths[2]:=300;  
  
  Q1.Close;  
  
  Q1.Sql.Clear;  
  
  Q1.Sql.Add('Select * from RULe Where Nodes="Non"');  
  
  Q1.Prepare;
```

```

Q1.Open;

If Q1.Recordcount>0 Then
  Begin
    S1.RowCount:=Q1.RecordCount+1;

    Q1.First;

    I:=1;

    Repeat
      S1.Cells[0,I]:=IntToStr(I);

      S1.Cells[1,I]:=Q1.FieldName('Tree').AsString;

      S1.Cells[2,I]:=Q1.FieldName('Nama_Tree').AsString;

      Q1.Next;

      I:=I+1;

    Until Q1.Eof;

  ENd;

  S1Click(Sender);

end;

Procedure TFTree.NextClick(Sender: TObject);

Var

I: Integer;

begin

  M1.Text:='';

  M2.Text:='';

  M3.Text:='';

  P1.Caption:=S1.Cells[1, S1.Row];

  P2.Caption:=S1.Cells[2, S1.Row];

  { Alamat:=Alamat+1;

  S2.ROWCount:=Alamat+1;

  S2.Cells[0,Alamat]:=Inttostr(Alamat);

  S2.Cells[1,Alamat]:=S1.Cells[1, S1.Row];

  S2.Cells[2,Alamat]:=S1.Cells[2, S1.Row];  }

```

```

Q1.Close;

Q1.Sql.CLEAR;

Q1.Sql.Add('Select * from Rule Where Nodes=:A');

Q1.ParamByName('A').AsString:=S1.Cells[1, S1.Row];

Q1.Prepare;

Q1.Open;

If Q1.Recordcount>0 Then
  Begin
    Alamat:=Alamat+1;
    S2.ROWCOUNT:=Alamat+1;
    S2.Cells[0,Alamat]:=Inttostr(Alamat);
    S2.Cells[1,Alamat]:=S1.Cells[1, S1.Row];
    S2.Cells[2,Alamat]:=S1.Cells[2, S1.Row];
    S1.RowCount:=Q1.RecordCount+1;

    Q1.First;

    I:=1;

    Repeat
      S1.Cells[0,I]:=IntTostr(I);
      S1.Cells[1,I]:=Q1.FieldByName('Tree').AsString;
      S1.Cells[2,I]:=Q1.FieldByName('Nama_Tree').AsString;
      Q1.Next;
      I:=I+1;
    Until Q1.Eof;

  ENd;
end;

Procedure TFTree.Button1Click(Sender: TObject);

Var
  i : Integer;

begin
  If Alamat>1 Then

```

```

Begin
  Alamat:=Alamat-1;

  S2.Cells[0,Alamat]:=Inttostr(Alamat);

  P1.Caption:=S2.Cells[1, Alamat];
  P2.Caption:=S2.Cells[2, Alamat];

  Q1.Close;

  Q1.Sql.CLEAR;

  Q1.Sql.Add('Select * from Rule Where Nodes=:A');
  Q1.ParamByName('A').AsString:=P1.Caption;

  Q1.Prepare;

  Q1.Open;

  If Q1.Recordcount>0 Then
    Begin
      S1.RowCount:=Q1.RecordCount+1;

      Q1.First;

      I:=1;

      Repeat
        S1.Cells[0,I]:=IntTostr(I);
        S1.Cells[1,I]:=Q1.FieldName('Tree').AsString;
        S1.Cells[2,I]:=Q1.FieldName('Nama_Tree').
AsString;

        Q1.Next;

        I:=I+1;

      Until Q1.Eof;

    END;

  Alamat:=Alamat+1;

  S2.Cells[0,Alamat]:=Inttostr(Alamat);

  S2.Cells[1,Alamat]:='';
  S2.Cells[2,Alamat]:='';

  Alamat:=Alamat-1;

```

```

S2.RowCount:=Alamat+1;

  If Alamat<=1 Then P2.Caption:='Pilihkan Kerusakan Yang
Terjadi Pada Handphone Anda';

End;

end;

Procedure TFTree.S1Click(Sender: TObject);

begin

  P3.Caption:=#32+#32+#32+S1.Cells[2,S1.Row];
  P4.Caption:=#32+#32+#32+S1.Cells[2,S1.Row];
  P5.Caption:=#32+#32+#32+S1.Cells[2,S1.Row];

  If Fdm.Kerusakan.Findkey([S1.Cells[1,S1.Row]]) Then

    Begin

      M1.Text:=Fdm.Kerusakan.FieldByName('Keterangan').
AsString;

      M2.Text:=Fdm.Kerusakan.FieldByName('Pengecekan').
AsString;

      M3.Text:=Fdm.Kerusakan.FieldByName('Solusi').AsString;

    End Else

    Begin

      M1.Text:='';
      M2.Text:='';
      M3.Text:='';

    End;

end;

Procedure TFTree.FormClose(Sender: TObject; var Action:
TCloseAction);

begin

  Action:=caFree;

end;

procedure TFTree.PengexcekanChange(Sender: TObject);

```

```
begin
```

```
    S1Click(Sender);
```

```
end;
```

```
end.
```