

Sistem Pakar Diagnosa Kerusakan Sepeda Motor Bebek Berbasis Android

Yaman Khaeruzzaman

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Cirebon
yaman.khaeruzzaman@umc.ac.id

Abstrak

Sistem pakar adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang dilakukan oleh para ahli. Pada tulisan ini, sistem pakar dapat diterapkan untuk mengetahui kerusakan pada sepeda motor. Karena pada umumnya ada beberapa pengendara sepeda motor kurang mengerti dengan gangguan atau kerusakan yang terjadi pada sepeda motornya. Mereka cenderung menyerahkannya kepada teknisi, tanpa peduli kerusakan itu sederhana atau terlalu rumit untuk diperbaiki. Pada penelitian dengan judul “Aplikasi Sistem Pakar Diagnosa Kerusakan Sepeda Motor Berbasis Android”, dirancang sebuah sistem pakar yang menyediakan kemudahan bagi pemakainya untuk menemukan solusi yang tersedia bagi penanganan kerusakan sepeda motor mereka.

Aplikasi ini dibangun dengan teknologi komputer berbasis *Android*, bahasa pemrograman *PHP* dan menggunakan *MySQL* sebagai basis data. Sedangkan metode pengembangannya menggunakan model *waterfall*.

Kata Kunci: Teknologi Komputer, *Web*, *PHP*, *MySQL*, *Waterfall*.

PENDAHULUAN

A. Latar Belakang

Sudah kita ketahui bahwa perkembangan Teknologi seluler saat ini semakin pesat. Hanya dengan sebuah telepon seluler kini kita dapat melakukan banyak hal dan juga yang sekarang sering digunakan adalah untuk menjalankan aplikasi-aplikasi *mobile* sebagai sarana hiburan, jejaring sosial ataupun sebagai media untuk mendapatkan dan mengolah data informasi. Salah satu sistem operasi yang saat ini semakin berkembang adalah *Android*. Bisa dipastikan, jumlah perangkat berbasis *Android* yang berada di tangan pengguna di Indonesia bakal bertambah secara

signifikan.

Dengan berkembangnya perangkat *mobile* serta teknologi yang menyertainya akan sangat berpengaruh pada perkembangan aplikasi *mobile*. Perkembangan aplikasi *mobile* tersebut akhirnya memberikan dampak pada berbagai bidang kehidupan kita. Salah satunya adalah bidang otomotif.

Pesatnya pertumbuhan sepeda motor tentunya juga harus didukung oleh kesiapan mekaniknya, sebab semakin banyak jumlah sepeda motor yang digunakan orang akan semakin banyak pula timbulnya kerusakan mesin. Dalam prakteknya pabrik telah melatih mekaniknya untuk menyelesaikan masalah kerusakan sepeda motor yang muncul, karena banyaknya pengguna

sepeda motor maka jumlah itu tidak cukup untuk menyelesaikan masalah kerusakan sepeda motor, untuk itu banyak bengkel yang berdiri untuk dapat membantu menyelesaikan masalah kerusakan sepeda motor.

Terkadang penulis melihat beberapa orang yang mengalami masalah sepeda motornya di jalan dan dia tidak tahu penyebabnya apa cara perbaikannya, dan yang lebih kasihan jika yang mengalami masalah seorang wanita, kita tahu sekarang banyak wanita karir yang pulang pergi ke kantor menggunakan sepeda motor sendirian.

Penulisan ini berisi tentang pembuatan Sistem Pakar dibangun untuk menemukan penyebab kerusakan mesin motor berbasis Android. Alasan penulis membuat judul tersebut karena pada saat ini teknologi semakin maju, Hampir setiap orang saat ini memiliki Smartphone Berbasis Android dan juga Sepeda Motor, Dapat kita lihat pada jaman sekarang ini hampir setiap orang memiliki sepeda motor dan bahkan dalam satu keluarga memiliki lebih dari satu. Dan tidak semuanya mengerti tentang mesin.

Sistem pakar adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar dapat menyelesaikan masalah seperti yang bisa dilakukan oleh para ahli. Sistem pakar sebagai salah satu cabang dari kecerdasan buatan dapat digunakan untuk kehidupan manusia termasuk dalam bidang otomotif

Saat ini hampir semua bidang yang dikerjakan manusia memanfaatkan teknologi komputer, diantaranya dalam bidang otomotif merupakan salah satu bidang yang juga menggunakan dan memanfaatkan teknologi komputer. Salah satunya adalah untuk mendiagnosa kerusakan yang terjadi pada sepeda motor, Khususnya Motor bebek, Maka dari itu dibutuhkan sebuah aplikasi untuk

mendiagnosa gejala-gejala dan keluhan yang dirasakan oleh pemilik sepeda motor. Alasan dibuatnya aplikasi ini adalah dengan alasan tidak semua orang mengerti tentang Mesin sepeda motor, dan juga berdasarkan sedikit permasalahan yang saya lihat di lingkungan sehari-hari seperti kasus yang terjadi seperti cerita diatas tadi.

Maka dari itu penulis membuat aplikasi ini dengan berbasis Android agar lebih membantu, memudahkan, dan mengetahui penyebab kerusakan sepeda motornya itu, sebelum ia bawa ke mekanik para pemilik sepeda motor.

Dari latar belakang yang telah penulis jabarkan diatas, penulis mengambil judul “**SISTEM PAKAR DIAGNOSA KERUSAKAN SEPEDA MOTOR BEBEK BERBASIS ANDROID**” Sebagai bahan penelitian.

B. Identifikasi Masalah

Permasalahan yang berhasil penulis identifikasi, yaitu:

- a. Tidak semua orang yang memiliki motor tau akan penanganannya jika terjadi kerusakan, terutama mengenai manfaat system pakar dengan aplikasi berbasis android.
- b. Kecenderungan masyarakat awam tidak perduli akan kerusakan kecil yang terjadi, dan apabila dibiarkan akan berakibat fatal.

C. Rumusan Masalah

Dari latar belakang diatas, maka permasalahan yang dapat dirumuskan adalah sebagai berikut:

- a. Bagaimana merancang dan membuat aplikasi ponsel berbasis android dengan sistem pakar diagnosa kerusakan sepeda motor bebek agar lebih praktis dan bermanfaat terutama untuk yang masih awam.

- b. Bagaimana cara pengguna kendaraan sepeda motor dapat memperoleh informasi tentang jenis kerusakan, dan solusi kerusakan dengan mudah dengan menggunakan aplikasi ini.

D. Batasan Masalah

Dalam pembuatan penulis menentukan batasan sebagai berikut:

- a. Aplikasi ini berbasis android yang nantinya dapat digunakan pada telepon genggam (hp).
- b. Aplikasi ini hanya terbatas mendiagnosa kerusakan sepeda motor bebek dengan menggunakan metode certainty factor.
- c. Platform yang digunakan mulai dari Android versi 2.3 ke atas.
- d. Tujuan Implementasi aplikasi berupa aplikasi sistem pakar diagnosa kerusakan sepeda motor bebek.

E. Maksud dan Tujuan Penelitian

1. Maksud

Adapun maksud dari pembuatan aplikasi ini adalah untuk memudahkan masyarakat untuk mendiagnosa kerusakan sepeda motor agar tidak berakibat fatal, terutama untuk yang masih awam tentang mesin.

2. Tujuan

Adapun tujuan dari penelitian ini adalah:

- a. Membuat perangkat lunak sistem pakar yang dapat menganalisa gangguan kerusakan pada sepeda motor.
- b. Menjadikan pengetahuan yang mudah didapatkan untuk mencari solusi kerusakan pada sepeda motor.

- c. Untuk mengetahui diagnosa gejala kerusakan pada kendaraan sepeda motor tanpa bantuan seorang pakar.

F. Metode dan Teknik Penelitian

1. Metode Penelitian

Metode penelitian dalam pembangunan aplikasi sistem pakar diagnose kerusakan sepeda motor bebek ini menggunakan metode penelitian R&D (*Research & Development*). Menurut Sujadi (2003:164) penelitian dan pengembangan atau (R&D) *Research and Development* adalah suatu proses atau langkah-langkah untuk mengembangkan suatu produk baru, atau menyempurnakan produk yang telah ada, yang dapat dipertanggung jawabkan. Produk tersebut tidak selalu berbentuk benda atau perangkat keras (*hardware*), seperti buku, modul, alat bantu pembelajaran di kelas atau di laboratorium, tetapi bisa juga perangkat lunak (*software*), seperti program komputer untuk pengolahan data, pembelajaran dikelas, perpustakaan atau laboratorium, ataupun model-model pendidikan, pembelajaran, pelatihan, bimbingan, evaluasi, manajemen, dan lain-lain [2].

Dalam penelitian ini, penulis menggunakan metode penelitian dan pengembangan atau R&D (*Research and Development*) karena penulis melakukan pengembangan suatu aplikasi baru dan melakukan pengumpulan data dengan melihat langsung kondisi di lapangan sehingga dapat dijadikan pertimbangan dalam pengambilan keputusan.

2. Teknik Penelitian

a) Studi Kepustakaan

Yaitu penelitian yang dilaksanakan berdasarkan data yang di peroleh dari teori-teori yang didapat dari buku penunjang, literatur, jurnal, dan paper yang berhubungan dengan topik yang di ambil sebagai bahan pembanding atau dasar pembahasan lanjut, serta untuk memperoleh landasan-landasan teori dari sistem yang di kembangkan.

b) Observasi

Teknik pengumpulan data dengan mengadakan pengamatan secara langsung dan mengindra terhadap objek penelitian.

c) Wawancara

Teknik pengumpulan data dengan jalan melakukan tanya jawab dengan responden serta teknisi sepeda motor.

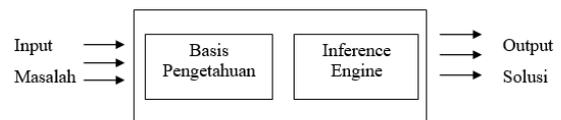
LANDASAN TEORI

A. Kecerdasan Buatan

Kecerdasan Buatan adalah salah satu cabang Ilmu pengetahuan berhubungan dengan pemanfaatan mesin untuk memecahkan persoalan yang rumit dengan cara yang lebih manusiawi[6]. Menurut Sri Kusumadewi, Kecerdasan buatan atau *artificial intelligence* merupakan salah satu bagian komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti dan sebaik yang dilakukan oleh manusia[7].

Kecerdasan buatan bertujuan untuk mengembangkan suatu program komputer yang dapat memecahkan suatu masalah tertentu dengan cara yang

dianggap cerdas seperti layaknya manusia. Sehingga dengan perancangan kecerdasan buatan yang baik, diharapkan peran manusia dapat meminimaliskan dan dapat meringankan beban kerja manusia. Kecerdasan buatan atau *Artificial Intelligence* bekerja dengan menerima *input*, diproses dan kemudian menghasilkan *output*, yang berupa solusi dari suatu masalah berdasarkan kumpulan pengetahuan yang ada. Seperti dilihat pada gambar 2.1 dibawah ini.



Gambar 2.1 Konsep Kecerdasan Buatan di Komputer

Lingkup utama implementasi dalam kecerdasan buatan saat ini ditemui pada bidang-bidang berikut:

- Sistem pakar (*Expert System*). Disini komputer digunakan sebagai saran untuk menyimpan pengetahuan para pakar. Dengan demikian komputer akan memiliki keahlian untuk menyelesaikan masalah dengan meniru keahlian yang dimiliki para pakar.
- Pengolahan bahasa alami (*Natural Language Processing*). Dengan pengolahan bahasa alami ini diharapkan user mampu berkomunikasi dengan komputer dengan menggunakan bahasa sehari-hari.
- Pengenalan ucapan (*Speech Recognition*). Melalui pengenalan ucapan diharapkan manusia mampu berkomunikasi dengan komputer dengan menggunakan suara.
- Robotika dan Sistem sensor.
- Computer vision*, mencoba untuk dapat mengintrepetasikan gambar atau objek-objek tampak melalui komputer.
- Intelligent Computer aid Instruction*. Komputer dapat digunakan sebagai tutor yang dapat melatih dan mengajar.

- g. *Game Playing*. Yaitu metode *artificial intelligence* yang meniru cara berfikir manusia dan *game*. Contohnya adalah program *perfect chessmate* yang mampu berfikir setara dengan *grandmaster* catur.

B. Sistem Pakar

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. Ketika hendak membuat suatu keputusan yang kompleks atau memecahkan masalah untuk mengeluarkan solusi, sering kali kita meminta nasihat dengan berkonsultasi dengan seorang pakar atau ahli.

1. Pengertian Sistem Pakar

Sistem pakar merupakan cabang dari AI (*Artificial Intelligent*) yang membuat ekstensi khusus untuk spesialisasi pengetahuan guna memecahkan suatu permasalahan pada *Human Expert*. *Human Expert* merupakan seseorang yang ahli dalam suatu bidang ilmu pengetahuan tertentu, ini berarti bahwa *expert* memiliki suatu pengetahuan atau *skill* khusus yang dimiliki oleh orang lain. *Expert* dapat memecahkan suatu permasalahan yang tidak dapat dipecahkan oleh orang lain dengan cara efisien.

Pengetahuan di dalam *Expert system* berasal orang atau *knowledge* yang berasal dari buku-buku referensi, surat kabar atau karya ilmiah orang lain. Pengetahuan manusia ke dalam komputer, agar komputer dapat menyelesaikan masalah seperti yang

biasa dilakukan oleh para ahli (Kusumadewi, 2003). Atau dengan kata lain sistem pakar adalah sistem yang didesain dan diimplementasikan dengan bantuan bahasa pemrograman tertentu untuk dapat menyelesaikan masalah seperti yang dilakukan oleh para pakar dalam hal ini adalah dokter.

Sistem pakar mempunyai banyak definisi, tetapi pada dasarnya sistem pakar diterapkan untuk mendukung pemecahan masalah. Berikut ini beberapa definisi sistem pakar, antara lain:

- a. Sistem pakar adalah suatu sistem yang bisa melayani atau meniru kemampuan seorang pakar (Giarratano dan Riley, 1994).
- b. Sistem pakar merupakan suatu model dan prosedur yang berkaitan dalam suatu daerah tertentu, yang mana tingkat keahliannya dapat dibandingkan dengan keahlian seorang pakar (Ignizio, 1991)
- c. Sistem pakar adalah program komputer yang didesain untuk meniru kemampuan memecahkan masalah dari seorang pakar, Pakar adalah orang yang memiliki kemampuan atau mengerti dalam menghadapi suatu masalah lewat pengalaman, seorang pakar mengembangkan kemampuan yang membuatnya dapat memecahkan permasalahan dengan hasil yang baik dan efisien (Jhon Durkin, 1994)
- d. Sistem pakar merupakan sistem yang berbasis komputer yang menggunakan pengetahuan, fakta dan teknik penalaran dalam memecahkan suatu masalah, biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tertentu (Marlin dan Ozman, 1998)

2. Konsep Dasar 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 (Kusumadewi, 2003). Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para ahli.

Menurut Turban (1995) konsep dasar sistem pakar mengandung Keahlian (*expertise*), pakar (*expert*), pengalihan keahlian (*transferring expertise*), inferensi (*inferencing*), aturan (*rules*) dan kemampuan menjelaskan (*explanation capability*). Keahlian (*expertise*) adalah suatu kelebihan penguasaan pengetahuan di bidang tertentu yang diperoleh dari pelatihan, membaca atau pengalaman. Pengetahuan tersebut memungkinkan para ahli untuk dapat mengambil keputusan lebih cepat dan lebih baik daripada seseorang yang bukan ahli.

Pakar (*Expert*) 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.

Pengalihan keahlian (*transferring expertise*) dari para ahli ke komputer untuk kemudian dialihkan lagi ke orang lain yang bukan ahli, hal inilah yang merupakan tujuan utama dari sistem pakar. Proses ini membutuhkan 4 aktivitas yaitu :

- a. Tambah pengetahuan (dari para ahli atau sumber-sumber lainnya)
- b. Representasi pengetahuan (ke komputer)
- c. Inferensi pengetahuan

- d. dan pengalihan pengetahuan ke *user*.

3. Ciri-ciri Sistem Pakar

Suatu sistem dikatakan sistem pakar apabila memiliki ciri-ciri sebagai berikut (Sri Kusumadewi, 2003):

- a. Dapat memberikan penalaran untuk data-data yang tidak pasti.
- b. Terbatas pada *domain* keahlian tertentu.
- c. Berdasarkan pada kaidah atau *rule* tertentu.
- d. Dirancang untuk dikembangkan secara bertahap.
- e. Keluarannya atau *output* bersifat anjuran.
- f. Dapat mengemukakan rangkaian alasan-alasan yang diberikannya dengan cara yang dapat dipahami.

4. Keunggulan Sistem Pakar

Sistem pakar sebagai program memiliki keunggulan bila dibandingkan dengan sistem konvensional, diantaranya adalah:

- a. Menghimpun data dalam jumlah yang sangat banyak.
- b. Menyimpan data tersebut untuk jangka waktu yang panjang dalam suatu bentuk tertentu.
- c. Mengerjakan perhitungan secara cepat dan tepat juga tanpa jemu mencari kembali data yang tersimpan dengan kecepatan tinggi.

Seperti terlihat pada tabel 2.1 dibawah ini, perbandingan sistem konvensional dengan sistem pakar.

Tabel 2.1 Perbandingan sistem konvensional dengan sistem pakar

Sistem Konvensional	Sistem Pakar
Informasi dan pemrosesannya biasanya jadi satu dengan program	Basis pengetahuan merupakan bagian terpisah dari mekanisme inferensi
Pengubahan program cukup sulit dan membosankan	Pengubahan aturan dapat dilakukan dengan mudah
Biasanya tidak bisa menjelaskan mengapa suatu input data itu dibutuhkan, atau bagaimana output itu diperoleh.	Penjelasan adalah bagian terpenting dari sistem pakar
Sistem hanya akan beroperasi jika sistem tersebut sudah lengkap.	Sistem dapat beroperasi hanya dengan beberapa aturan.
Eksekusi dilakukan langkah demi langkah	Eksekusi dilakukan pada keseluruhan basis pengetahuan.
Menggunakan data.	Menggunakan pengetahuan.
Tujuan utamanya adalah efisiensi.	Tujuan utamanya adalah efektivitas.

Sumber: foto Sri Kusumadewi (2003:112)

5. Keuntungan Sistem Pakar

Berikut ini merupakan keuntungan bila menggunakan sistem pakar, diantaranya ialah:

1. Menjadikan pengetahuan dan nasihat lebih mudah didapat.
2. Meningkatkan output dan produktivitas.
3. Menyimpan kemampuan dan keahlian seorang pakar.
4. Meningkatkan penyelesaian masalah yang khusus.
5. Meningkatkan reliabilitas.
6. Memberikan respons (jawaban) yang cepat.
7. Merupakan panduan yang cerdas.
8. Dapat bekerja dengan informasi yang kurang lengkap dan mengandung ketidakpastian.

6. Kelemahan Sistem Pakar

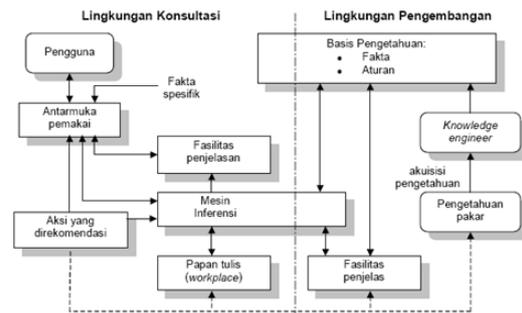
Disamping memiliki beberapa keuntungan, sistem pakar juga memiliki beberapa kelemahan menurut Kusumadewi(2003:111), diantaranya:

1. Biaya yang diperlukan untuk membuat dan memeliharanya sangat mahal.

2. Sulit dikembangkan, hal ini erat kaitannya dengan ketersediaan pakar dibidangnya.
3. Sistem pakar tidak 100 % bernilai benar.

7. Struktur Sistem Pakar

Sistem pakar terdiri dari dua bagian pokok (Sri Kusumadewi 2003:113), yaitu: lingkungan pengembangan (*Development Environment*) dan lingkungan konsultasi (*Consultation Environment*). Lingkungan pengembangan digunakan sebagai pembangun sistem pakar. Lingkungan konsultasi digunakan oleh seseorang yang bukan ahli. Seperti dapat dilihat pada gambar 2.2 berikut:



Gambar 2.2 Struktur Sistem Pakar (Turban 1995)

Komponen-komponen yang terdapat dalam sistem pakar adalah seperti yang terdapat pada Gambar 2.2, yaitu User Interface (antar muka pengguna), basis pengetahuan, akuisisi pengetahuan, mesin inferensi, workplace, fasilitas penjelasan dan perbaikan pengetahuan.

1. Basis pengetahuan

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

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

Basis pengetahuan direpresentasikan dengan menggunakan aturan berbentuk *IF-THEN*. Bentuk ini digunakan apabila kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu.

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

Basis pengetahuan akan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang terjadi sekarang.

2. Mesin Inferensi

Mesin inferensi berfungsi untuk melakukan penelusuran pengetahuan yang terdapat dalam basis pengetahuan untuk mencapai kesimpulan tertentu. Mesin Inferensi menyediakan arahan tentang bagaimana menggunakan pengetahuan sistem dalam membangun agenda yang mengorganisasikan dan mengontrol langkah yang diambil untuk memecahkan persoalan saat konsultasi berlangsung. Ada 3 elemen utama dalam mesin inferensi:

a. *Interpreter*

Mengeksekusi item-item agenda yang terpilih dengan menggunakan aturan-aturan dalam basis pengetahuan yang sesuai.

b. *Scheduler*

Akan mengontrol agenda.

c. *Consistency enforce*

Bertujuan memelihara konsistensi dalam mempresentasikan solusi yang bersifat darurat.

3. Akuisisi Pengetahuan

Adalah akumulasi, transfer dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam komputer. Dalam tahap ini *knowledge engineer* berusaha menyerap pengetahuan untuk selanjutnya ditransfer ke dalam basis pengetahuan. Pengetahuan diperoleh dari pakar, dilengkapi dengan buku, basis data, laporan penelitian dan pengalaman pemakai.

4. Antarmuka Pemakai

Antarmuka digunakan mempermudah komunikasi antar pemakai dengan sistem. Komunikasi tersebut berupa permintaan informasi yang diperlukan sistem untuk pencarian solusi, pembagian informasi dari pemakai, pemberian informasi dari pemakai kepada sistem, permintaan informasi penjelasan dari pemakai kepada sistem, permintaan informasi penjelasan oleh pemakai dan pemberian informasi oleh sistem.

5. Fasilitas Penjelasan

Fasilitas penjelasan membantu perancangan pengetahuan untuk memperbaiki dan meningkatkan pengetahuan, member kejelasan dan keyakinan kepada pemakai tentang proses atau hasil yang diberikan sistem pakar. Fasilitas ini digunakan untuk melacak respond dan

memberikan penjelasan tentang sistem pakar secara interaktif melalui pertanyaan:

- a. Mengapa suatu pertanyaan ditanyakan oleh sistem pakar
- b. Bagaimana konklusi dicapai
- c. Mengapa ada alternatif yang dibatalkan
- d. Rencana apa yang akan digunakan untuk mencapai suatu solusi

6. Fasilitas Perbaikan Pengetahuan

Pakar manusia dapat menganalisa performansinya sendiri, belajar darinya dan meningkatkannya untuk konsultasi berikut. Adanya evaluasi dengan sistem pakar ini akan menghasilkan basis pengetahuan yang lebih baik serta penalaran yang lebih efektif.

7. *Workplace* merupakan memori yang digunakan untuk merekam kejadian yang sedang berlangsung termasuk keputusan sementara. Ada tiga tipe keputusan yang dapat direkam, yaitu:

- a. Rencana : Bagaimana menghadapi masalah.
- b. Agenda : Aksi-aksi potensial yang sedang menunggu untuk dieksekusi.
- c. Solusi : Calon aksi yang akan dibangkitkan.

8. Metode Inferensi dalam Sistem Pakar

Inferensi merupakan proses untuk menghasilkan informasi dari fakta yang diketahui atau diasumsikan. Inferensi adalah konklusi logis

(*logical conclusion*) atau implikasi berdasarkan pada informasi yang tersedia. Dalam sistem pakar, proses inferensi dilakukan dalam suatu modul yang disebut *Inference engine* (mesin inferensi). Ketika representasi pengetahuan pada bagian *knowledge base* telah lengkap, atau paling tidak telah berada pada level cukup akurat, maka referensi pengetahuan tersebut telah siap digunakan. Sedangkan inferensi engine merupakan modul yang berisi program tentang bagaimana mengendalikan proses reasoning.

Sesuai dengan tujuan sistem pakar untuk mengembangkan dan memasyarakatkan serangkaian usulan jawaban dari suatu masalah, untuk itu sistem pakar memiliki suatu strategi penalaran (*inference*) dimana proses penalaran itu akan ditemukan berbagai macam jawaban.

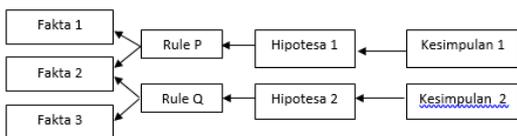
Terdapat dua pendekatan untuk mengontrol inferensi dalam sistem pakar berbasis aturan, yaitu pelacakan ke belakang (*backward chaining*) dan pelacakan ke depan (*forward chaining*).

a. Pelacakan ke belakang (*backward chaining*)

Backward chaining

adalah suatu alasan yang berkebalikan dengan hypothesis, potensial konklusinya mungkin akan terjadi atau terbukti, karena adanya fakta yang mendukung akan *hypothesis* tersebut (Giarratano dan Riley, 1994). Pencocokan fakta atau pernyataan dimulai dari bagian 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 (Sri Kusumadewi, 2003). Metode *backward chaining* dapat dilihat pada gambar 2.3.



Gambar 2.3 Proses pelacakan *backward chaining*

Sebagai contoh akan diuraikan sebagai berikut, jika suatu masalah mempunyai sederetan kaidah seperti tertulis dibawah ini:

- R1 : A and C, THEN E
- R2 : IF D and C, THEN F
- R3 : IF B and E, Then F
- R4: IF B THEN C
- R5 : IF F THEN G

Dimana sebagai acuan diketahui bahwa fakta A dan B adalah *true* (benar) dan G adalah *GOAL* (tujuan).

Berikut ini langkah-langkah yang digunakan dalam metode *backward chaining*:

1. Langkah 1: Mencari kebenaran dasar dari tujuan berdasarkan fakta yang ada, dimana sebagai acuannya kita sudah mengetahuinya.
2. Langkah 2: R5 menunjukkan bahwa jika F benar maka G benar.

Untuk itu, maka kita akan melihat R2 dan R3.

3. Langkah 3: R2 menunjukkan bahwa D belum tentu benar sebab D tidak termasuk dalam fakta acuan, sehingga R2 tidak bisa digunakan, maka kita akan melihat ke kaidah yang lainnya yaitu kaidah R3.
4. Langkah 4 : Pada kaidah R3, kita ketahui sesuai fakta acuan yang ada bahwa B adalah benar, selanjutnya kita akan melihat apakah E benar.
5. Langkah 5 : Pada kaidah R1 sangat tergantung dengan kebenaran A dan C
6. Langkah 6 : Karena A diketahui sebagai fakta acuan adalah benar, selanjutnya kita akan melihat apakah C benar, dengan melihat R4.
7. Langkah 7: R4 menunjukkan bahwa C adalah benar karena B adalah benar

Dari langkah diatas dapat diambil kesimpulan bahwa G adalah benar.

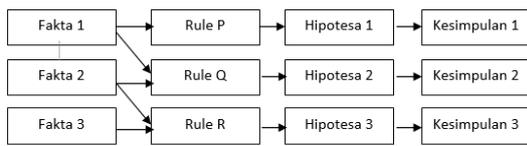
- b. Pelacakan ke depan (*forward chaining*)

Forward chaining merupakan fakta untuk mendapatkan kesimpulan (*conclusion*) dari fakta tersebut (Menurut Giarratano dan Riley, 1994). Penalaran ini berdasarkan fakta yang ada (*data driven*), metode ini adalah kebalikan dari metode *backward chaining*, dimana metode ini

dijalankan dengan mengumpulkan fakta-fakta yang ada untuk menarik kesimpulan. Dengan kata lain, prosesnya dimulai dari *facts* (fakta-fakta yang ada) melalui proses *inference fact* (penalaran fakta-fakta) menuju suatu *goal* (suatu tujuan). Metode ini bisa juga disebut menggunakan aturan IF-THEN dimana premise (IF) menuju *conclusion* (THEN) atau dapat juga dituliskan sebagai berikut.

THEN (konklusi)

Metode pelacakan *forward chaining* dapat dilihat pada gambar 2.4.



Gambar 2.4 Proses pelacakan *forward chaining*

Ada dua pendapat mengenai pelaksanaan metode ini. Pertama dengan cara membawa seluruh data yang didapat ke sistem pakar. Kedua dengan membawa bagian-bagian penting saja dari data yang didapat ke sistem. Pakar. Cara pertama lebih baik digunakan jika sistem pakar terhubung dengan proses otomatis dan menerima seluruh data dari database. Cara kedua menghemat waktu dan biaya dengan mengurangi data dan mengambil data yang dianggap perlu. Sebagai contoh, seperti kasus diatas maka berdasarkan metode ini langkah-langkah yang diambil :

R1 : IF A and C, THEN B

R2 : IF D and C, THEN F

R3: IF B and E, THEN F

R4 : IF B, THEN C

R5 : IF F, THEN G

Dan faktanya : A adalah benar, B benar.

Langkah 1: R4 menunjukkan bahwa C benar, karena A dan B adalah benar.

Langkah 2: Karena A dan C benar, maka E adalah benar.

Langkah 3: Karena B dan E benar, maka F adalah benar.

Langkah 4: Karena F adalah benar maka dengan demikian G adalah benar.

Kedua jenis strategi ini akan mengarah pada suatu kesimpulan. Namun efisiensinya tergantung dari kondisi masalah yang dihadapi, jika suatu masalah memiliki *premise* yang jumlahnya lebih sedikit dibanding *conclusion* maka strategi yang ditawarkan adalah *forward chaining*, sebaliknya jika jumlah *premise* lebih banyak jika dibandingkan dari *conclusion*, maka strategi yang ditawarkan *backward chaining* (ignizio,1991).

Kedua metode tersebut dipengaruhi oleh tiga macam penelusuran, yaitu *depth-first search*, *breadth-first search* dan *best-first search*. Pada *depth-first search* melakukan penelusuran kaidah secara mendalam dari simpul akar bergerak menurun ke tingkat dalam yang berurutan. *Breadth-first search* bergerak dari simpul akar, simpul yang ada pada setiap tingkat diuji sebelum pindah ke tingkat

selanjutnya. Dan pada *best-first search* bekerja berdasarkan kombinasi kedua metode sebelumnya.

9. Rekayasa Pengetahuan

Definisi menurut rekayasa pengetahuan (*knowledge engineering*) merupakan proses pembentukan suatu sistem pakar dengan mengambil data dari seorang ahli (*human expert*) atau dari nara-sumber lainnya yang kemudian diolah menjadi suatu sistem pakar (Giarratano dan Riley, 1994). Tujuan utama dalam rekayasa pengetahuan untuk membangun perangkat lunak modular sehingga perubahan dapat dibuat dalam suatu modul tanpa mempengaruhi kerja modul lainnya. Rekayasa pengetahuan membantu pakar mengekstraksi pengetahuan yang dimilikinya dan biasanya berperan sebagai pembangun *system*.

ANALISIS DAN PERANCANGAN SISTEM

A. Analisis Sistem

1. Analisis Kebutuhan Fungsional

Analisis kebutuhan fungsional dilakukan untuk memberikan gambaran mengenai permasalahan dan prosedur yang akan berjalan pada sistem. Berikut beberapa rencana kebutuhan fungsional aplikasi ini:

- Dapat menampilkan diagnosa kerusakan pada sepeda motor Bebek.
- Dapat mempermudah dan memberi pengetahuan bagi pengguna sepeda motor Bebek.

2. Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional dilakukan untuk mengetahui spesifikasi kebutuhan untuk sistem. Berdasarkan kebutuhan fungsional di atas fungsi untuk menampilkan informasi dan gambar sudah ada sejak API Level 1. Namun, penulis menggunakan minimum requirement API Level 9 atau Gingerbread. Hal ini dipertimbangkan karena pangsa pasar pengguna android mayoritas sudah menggunakan android versi Gingerbread keatas. Pemilihan minimum requirement ini juga memperhatikan dari segi user interface yang dimiliki android versi ini lebih bervariasi.

Untuk membangun aplikasi ini juga dibutuhkan *software* sebagai berikut:

1) Bahasa Pemrograman Java

Dalam hal ini digunakan Java Development Kit (JDK) dan Java Runtime Environment (JRE) versi 1.7.0.450.

2) Software Eclipse

Untuk memudahkan dalam pengembangan aplikasi, maka digunakan software eclipse karena memiliki beberapa fasilitas yang diperlukan dalam pembangunan aplikasi ini. Adapun dalam pengembangan ini digunakan Eclipse Indigo dikarenakan telah mendukung Android Development Tools.

3) Sistem Operasi

Untuk penggunaan sistem operasi dapat digunakan

Windows 7, 8 (32-bit) atau (64-bit), Windows XP (32-bit) atau Vista (32 atau 64 bit), Mac OS X 10.4.8 atau diatasnya, dan Linux.

- 4) Android Software Development Kit (Android SDK)

Android SDK menyediakan development environment dengan semua komponen yang diperlukan. Antara lain tools pengembangan, libraries ,dokumentasi, serta contoh aplikasi dan disertakan pula emulator untuk mensimulasikan aplikasi berjalan pada perangkat. Adapun SDK yang digunakan adalah SDK r11 windows.

- 5) Android Development Tools (ADT)

Android membuat kostum plugin untuk IDE Eclipse, sehingga dengan adanya ADT ini memberikan kemudahan dalam pengembangan aplikasi, membuat tampilan antarmuka aplikasi, menambahkan komponen yang diperlukan, mendebug aplikasi dengan menggunakan perangkat SDK Android, dan bahkan membungkus aplikasi yang telah dikembangkan untuk di distribusikan.

B. Perancangan Sistem

1. Perancangan Proses

Pada tahap ini dilakukan pembuatan model dari perangkat lunak. Maksud dari pembuatan model ini adalah untuk memperoleh pengertian lebih baik dari aliran data dan kontrol, proses-proses

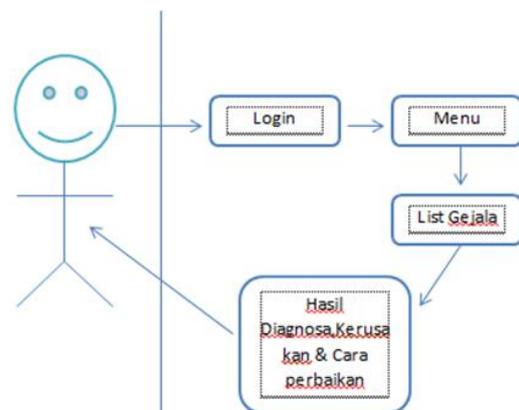
fungsional, tingkah laku operasi dan informasi yang terkandung di dalamnya.

a. Use Case Diagram

Use Case Diagram, digunakan untuk memodelkan bisnis proses berdasarkan perspektif pengguna system. Use case diagram terdiri atas diagram untuk use case , admin dan user.

User mempresentasikan orang yang akan mengoperasikan atau orang yang berinteraksi dengan sistem aplikasi.Sedangkan admin berlaku sebagai pengolah data.

Berikut ini adalah use case diagram yang sudah dispesifikasikan berdasarkan admin dan user yang berperan terhadap sistem pakar aplikasi ini.



Gambar 3.1 Use Case Diagram

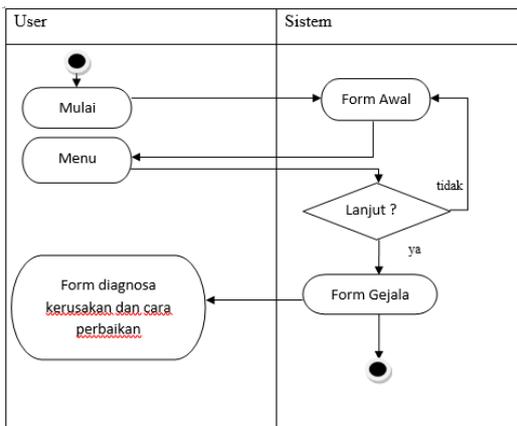
b. Activity Diagram

Activity diagram menggambarkan bagaimana alur proses dari sebuah sistem. Komponen utama dalam sebuah activity diagram adalah state dan message. Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam

suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti use case atau interaksi.

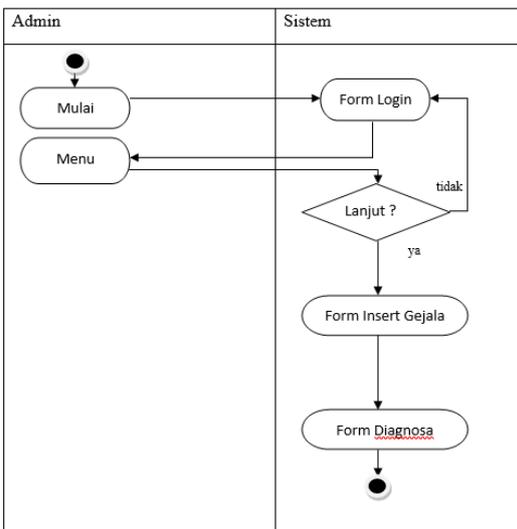
Pada tahap ini activity diagram digunakan untuk memodelkan perilaku use case object pada aplikasi sistem pakar ini, dibuat dua activity diagram untuk user dan admin.

a. Activity Diagram (User)



Gambar 3.2 Activity Diagram (User)

b. Activity Diagram (Admin)



Gambar 3.3 Activity Diagram (Admin)

C. Perancangan Basis Aturan

Dalam Perancangan data, akan dijelaskan bagaimana data-data yang terdapat dalam system sesuai

dengan fungsinya sebagai input ataupun output sistemnya.

a. Kerusakan Dan Gejala kerusakannya

1. Kerusakan pada system pelumas

- IF Jumlah Minyak Pelumas yang tersisa dalam mesin sedikit

- AND Minyak Pelumas Kotor / Pekat

- AND Tekanan Minyak Pelumas Rendah

- THEN Kerusakan pada system pelumas

2. Kerusakan pada system bahan bakar

- IF *Mesin Berputar Tetapi Tidak Mau Hidup*

- AND Susah dihidupkan Saat di starter

- AND Putaran Stasioner tidak rata

- AND Campuran Bahan Bakar Miskin

- AND Campuran Bahan Bakar Kaya

- THEN Kerusakan pada system bahan bakar

3. Kerusakan pada kepala silinder, silinder dan piston

- IF *Tekanan Kompresi Rendah*

- AND Tekanan Kompresi Terlalu Tinggi

- AND Suara Mesin Berisik

- AND Mesin Tidak Dapat stasioner

- AND Asap Knalpot (gas sisa pembakaran) Banyak

- THEN Kerusakan pada kepala silinder, silinder dan piston

4. Kerusakan pada Kopling

- IF Kopling slip pada saat akselerasi

- AND Handle kpling keras

- AND Motor Bergerak Maju Walaupun Kopling Ditekan
 - AND Sistem Kerja Kopling Terasa Kasar
 - THEN Kerusakan pada kopling
5. Kerusakan pada poros engkol dan gigi transmisi
- IF Suara Berisik dari Transmisi
 - AND Sukar Memindahkan Gigi
 - AND Suara Berisik dari Poros Engkol
 - THEN Kerusakan pada poros engkol dan gigi transmisi
6. Kerusakan Alternator dan system stater
- IF Sumber Arus Listrik Tidak Ada
 - AND Motor starter berputar, tetapi mesin tidak berputar
 - AND Motor starter berputar lambat
 - AND Suara rantai mesin berisik
 - THEN Kerusakan alternator dan system stater
7. Kerusakan pada Kemudi, Roda depan dan Suspensi
- IF Pengemudian berat
 - AND Kemudi tertarik kesalah-satu arah
 - AND Roda depan goyang
 - AND Suspensi terlalu lembut dan terdapat suara
 - AND Suspensi terasa keras
 - THEN Kerusakan pada Kemudi, Roda depan dan Suspensi
8. Kerusakan pada rem dan rantai roda
- IF Daya pengereman kurang (jenis tromol)
 - AND Daya pengereman kurang (jenis ckram)
- AND Handle rem terlalu keras
 - AND Rantai roda berisik
 - AND Rantai roda lepas
 - THEN Kerusakan pada rem dan rantai roda
9. Kerusakan pada pengapian, pengisian dan beban
- IF Bunga api lemah atau tidak ada
 - AND Tidak ada pengisian
 - AND Lampu sein mati
 - AND Arah lampu tidak berubah ketika dim dijalankan
 - THEN Kerusakan pada pengapian, pengisian dan beban

IMPLEMENTASI DAN PENGUJIAN SISTEM

A. Implementasi

Proses implementasi bertujuan untuk menuangkan/merealisasikan semua perancangan yang telah direncanakan ke dalam bentuk kode program atau dengan kata lain tahap implementasi ini merupakan tahapan lanjutan dari tahap perancangan yang telah dibahas pada bab sebelumnya, dalam tahap implementasi ini akan dijelaskan mengenai perangkat keras (hardware) dan perangkat lunak (software) yang diperlukan dalam membangun serta menjalankan *aplikasi diagnose kerusakan sepeda motor* ini, file-file yang diperlukan dalam membangun sistem, tampilan aplikasi beserta potongan-potongan script logika program maupun *script* untuk menampilkan *user interface* nya.

1. Batasan Implementasi

Dalam mengimplementasikan Aplikasi Diagnosa kerusakan sepeda motor Berbasis Android ini

ada beberapa hal yang menjadi batasan implementasi, yaitu :

- a. Aplikasi hanya sebatas mendiagnosa kerusakan dari gejala gejala yang dipilih oleh user.
- b. Aplikasi Ini hanya menampilkan informasi jenis kerusakan, penyebab dan cara perbaikan sepeda motor dari gejala yang telah di pilih oleh user.

2. Implementasi Tampilan

Implementasi tampilan atau *user interface* menggambarkan tampilan dari aplikasi yang dibangun yaitu implementasi tampilan system pakar kerusakan Sepeda motor berbasis Android. Berikut adalah implementasi tampilan yang akan dibangun.

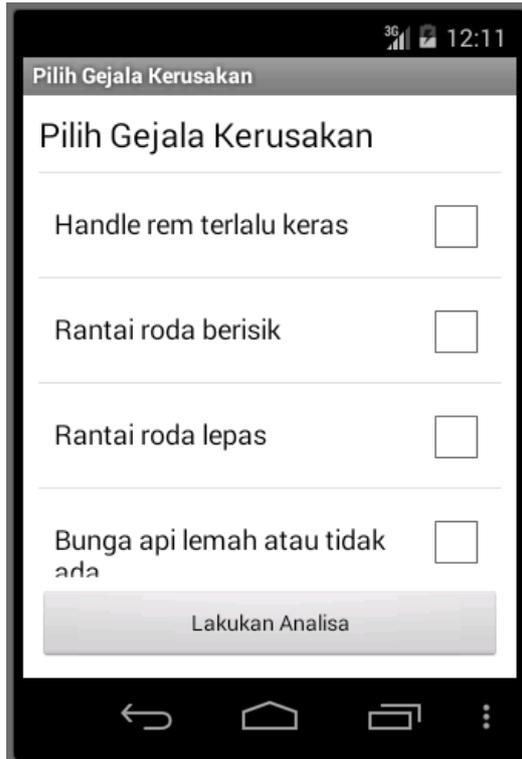
- a. Tampilan Layout Utama

Tampilan halaman utama yang menampilkan tiga tombol, masing masing tombol itu memiliki fungsi yang berbeda, diantaranya Tombol Diagnosa & Perbaikan, About, dan Exit.



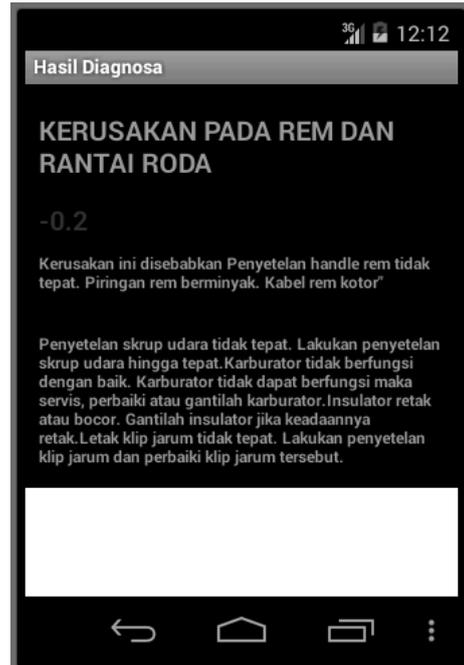
- b. Tampilan Layout Check Box diagnose

Pada layout ini menampilkan Segala gejala kerusakan yang ada pada sepeda motor, yang nantinya akan dicentang oleh user untuk mendapatkan informasi jenis kerusakan apa, penyebabnya apa dan cara untuk menyelesaikannya.



c. Tampilan Layout Hasil diagnose

Pada layout ini akan memunculkan informasi dari hasil gejala kerusakan yang di pilih oleh user pada layout check box sebelumnya, Menjelaskan tentang Penyebab kerusakan, Gejala, dan Penjelasan tentang Cara untuk perbaikan dari kerusakan itu.



3. Cara Menggunakan Aplikasi

1. Buka aplikasi, dengan cara sentuh *icon* dari program diagnose, maka akan muncul Layout utama yang terdiri dari tiga tombol.
2. Tekan tombol diagnose dan perbaikan untuk memilih gejala kerusakan yang ingin kita ketahui sebab dan cara penyelesaiannya dengan cara menekan salah satu list atau lebih.
3. Jika gejala telah kita pilih lalu sentuh tombol Lakukan analisa, maka akan muncul layout baru yang berisi tentang penyebab dan cara perbaikannya.

B. Pengujian Perangkat Lunak

Pengujian terhadap Aplikasi ini terbagi menjadi dua bagian, yaitu pengujian fungsional (alpha) dan pengujian beta. Untuk pengujian fungsional difokuskan menggunakan metode black box yang menitik beratkan

pada persyaratan fungsional dari perangkat lunak yang dibangun.

1. Rencana Pengujian

Berikut ini adalah tabel rencana pengujian Aplikasi Sistem pakar kerusakan sepeda motor berbasis android.

Tabel 4.1 Rencana Pengujian

Kelas Uji	Butir Uji	Jenis Pengujian
Layout Awal	Tombol Diagnosa dan perbaikan	Black Box
	Tombol About	Black Box
	Tombol Exit	Black Box
Layout Check box	Check box nama nama gejala	BlackBox
Layout hasil diagnosa	Informasi penyebab, dan cara perbaikan.	BlackBox

2. Kasus dan Hasil Pengujian Alpha

Berdasarkan rencana pengujian yang telah dilakukan maka didapat hasil pengujian alpha sebagai berikut:

a. Pengujian menu Awal

Tabel 4.2 Pengujian Halaman Awal

Kasus dan Hasil Uji					
Butir Uji	Data Masukan	Yang Diharapkan	Hasil tampilan	Pengamatan	Kesimpulan
Jalankan Aplikasi	Pertama kali membuka aplikasi	Aplikasi akan menampilkan Layout Menu Awal		Terdapat tiga tombol Sebagai pilihan menu yaitu diagnose & perbaikan	[√] Diterima [] Ditolak
Button Lanjut	Klik tombol diagnose & perbaikan	Aplikasi akan menampilkan check box gejala		Terdapat nama nama gejala kerusakan	[√] Diterima [] Ditolak
Hasil Dari check box yang dipilih	Klik tombol lakukan analisa	Aplikasi akan menampilkan tampilan layout hasil dari hasil salah satu atau lebih dari gejala		Terdapat sebab jenis kerusakan, dan cara perbaikan.	[√] Diterima [] Ditolak

Rangkuman Hasil Pengujian :

- Aplikasi dapat menganalisa jenis gejala, kerusakan dan cara perbaikan sesuai dengan yang diharapkan
- Aplikasi yang dapat menampilkan hasil diagnose dari gejala yang dipilih oleh user.

KESIMPULAN DAN SARAN

A. Kesimpulan

Setelah melakukan analisis, perancangan, dan pengujian, maka dapat diperoleh beberapa kesimpulan yaitu :

1. Aplikasi system pakar kerusakan sepeda motor berbasis android ini dapat mendiagnosa terhadap suatu kerusakan serta solusi untuk memperbaikinya.
2. Aplikasi system pakar kerusakan sepeda motor berbasis android ini cukup membantu masyarakat umum untuk memperoleh informasi dimana letak kerusakan – kerusakan yang terjadi pada sepeda motor serta solusi untuk menanganinya.

B. Saran

Agar system pakar kerusakan sepeda motor berbasis android ini kedepannya lebih baik, maka yang dapat disarankan adalah :

1. Memperbaiki dan memperindah tampilan pengguna untuk menyajikan kenyamanan oleh user.
 2. Menyajikan gejala dan solusi yang lebih lengkap lagi, lebih detail dan terperinci beserta tips perawatannya.
 3. Menyajikan keakuratan dari hasil gejala yang dipilih oleh user.
- Button yang terdapat pada halaman awal dapat berfungsi dengan benar

DAFTAR PUSTAKA

- [1] Maturidi, Ade Johar, *Metode Penelitian Teknik Informatika*, Deepublish, Yogyakarta, 2012.
- [2] Kusumadewi, Sri, *Artificial Intelligence*, Graha ilmu, Edisi Pertama, Yogyakarta, 2013.
- [3] Nazruddin, Syafaat, H (2012) *Pemrograman aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung. Informatika
- [4] Novianti, Dian, M.Kom. (2014), *Analisa dan Perancangan Sistem Informasi Berorientasi Objek*. Yogyakarta.
- [5] Suprianto, Dodit. (2010). *Membuat Aplikasi Dekstop Menggunakan MYSQL & VB.NET Secara Profesional*. Malang.
- [6] Murya, Yosef (2013). *Pemrograman Android Blackbox*. Jasakom.
- [7] Irawan. (2008). *Java Untuk Profesional*. Palembang