

**Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan  
Masalah Kerusakan Dan Perawatan Sepeda Motor**

**TUGAS AKHIR**

Oleh :

**Donny Alberta Ginting      3310701064**

Disusun untuk memenuhi syarat kelulusan Program Diploma III



**PROGRAM STUDI APLIKASI PERANGKAT LUNAK  
JURUSAN TEKNIK INFORMATIKA  
POLITEKNIK BATAM  
BATAM  
2010**

## **LEMBAR PENGESAHAN**

Batam, 2 Agustus 2010

**Pembimbing,**

**Afdhol Dzikri, SST**  
**NIK. 107048**

## LEMBAR PERNYATAAN

Dengan ini, saya:

NIM : 3310701064

Nama : Donny Alberta Ginting

adalah mahasiswa Teknik Informatika Politeknik Batam yang menyatakan bahwa tugas akhir dengan judul:

APLIKASI SISTEM PAKAR UNTUK MEMBANTU MENYELESAIKAN  
MASALAH KERUSAKAN DAN PERAWATAN MOTOR

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa ijin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, 2 Agustus 2010

**Donny Alberta Ginting**  
**NIM. 3310701064**

## KATA PENGANTAR

Puji dan syukur kehadiran Tuhan Yang Maha Esa, berkat rahmat, hidayat serta bimbinganNya, penyusun dapat menyelesaikan Tugas Akhir yang berjudul Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan Dan Perawatan Motor. Aplikasi ini bertujuan menjadikan pengetahuan yang mudah di dapatkan konsumen bengkel untuk mencari solusi kerusakan pada sepeda motor.

Dalam kesempatan ini, penyusun mengucapkan terima kasih kepada :

1. Ayahanda J.A.Ginting, Ibunda S.Br.Lingga, serta Kakanda Esra Elly dan Adinda Ira, Ingan, Samuel yang telah memberikan doa dan dukungannya.
2. Bapak Afdhol Dzikri, SST, selaku pembimbing yang telah memberikan pengarahan dan bimbingan selama penyusunan Tugas akhir ini.
3. Ferdesly Silaban dan Laura Oktami Manalu yang telah meluangkan waktu dan pemikirannya serta memberikan dukungan moral dan doanya.
4. Sahabatku Doli Indraja Raya Rajagukguk, Ferry Christian Sarungpaet, Teddy Alexander Situmorang, Arif Frans Christian Sibarani, Chandra Rojali Nainggolan yang telah memberikan motivasinya.
5. Bapak/Ibu Dosen jurusan Teknik Informatika atas bimbingannya.
6. Teman-teman seperjuangan Teknik Informatika angkatan 2007 yang memberikan semangat dan doa.

Akhirnya atas segala bimbingan dan dukungan yang diberikan dengan tulus dan ikhlas. Penyusun mengucapkan terima kasih. Semoga laporan ini dapat bermanfaat bagi pembaca yang ingin mengembangkan sebuah aplikasi yang serupa.

Batam, 30 Juli 2010

Penulis

## **ABSTRAK**

### **APLIKASI SISTEM PAKAR UNTUK MEMBANTU MENYELESAIKAN MASALAH KERUSAKAN DAN PERAWATAN MOTOR**

Aplikasi sistem pakar untuk menyelesaikan masalah kerusakan dan perawatan sepeda motor ini di buat bertujuan untuk menjadikan pengetahuan yang mudah di dapatkan konsumen bengkel, untuk mencari solusi kerusakan pada sepeda motor dan informasi tentang cara merawat sepeda motor.

Untuk mendapatkan solusi kerusakan sepeda motor user di arahkan untuk melakukan penelusuran berdasarkan masalah kerusakan yang dihadapi. Tahapan berawal dari macam kerusakan, jenis kerusakan dan ciri kerusakan.

Sedangkan mengenai informasi pemeliharaan rutin user dapat membaca langsung di aplikasi. Informasi ini berbentuk dokumen, yang beritanya selalu dapat di perbaharui oleh pakar pada aplikasi ini.

Kata Kunci: sistem pakar, macam kerusakan, jenis kerusakan, ciri kerusakan, solusi, pemeliharaan rutin.

## **ABSTRACT**

### **APPLICATION OF EXPERT SYSTEM FOR SOLVING THE PROBLEM OF DAMAGE AND MAINTENANCE OF MOTORCYCLES**

The application of an expert system to solve the problem of damage and maintenance of motorcycle is made the knowledge that aims to make consumers get easier in the garage, to look for solutions of damage on the motorcycle and information about how to care for motorcycles.

To obtain solutions in damage to motorcycle users are directed to conduct a search based on the problems faced destruction. Beginning stages of kinds of damage, type of damage and damage characteristics.

While on routine maintenance of user information can be read directly in the application. This information is in the form of documents, which story can always be revised by an expert in this application.

Keywords: expert system, types of damage, type of damage, damage characteristics, solutions, routine maintenance.

# **BAB 1 PENDAHULUAN**

## **1.1 Latar Belakang**

Dalam kehidupan sehari-hari, tak jarang terdengar keluhan - keluhan yang berhubungan dengan motor, terutama jika terjadi mogok ditengah jalan. Kebingungan tersebut sering terjadi dikarenakan oleh kurangnya pengetahuan tentang sifat-sifat dari motor itu sendiri. Dikarenakan hal tersebut maka tak jarang jika motor masih di kendarai walaupun kondisinya sedang rusak (rusak ringan), keadaan ini dapat menyebabkan kerusakan yang sangat vital bahkan dapat menyebabkan kecelakaan.

Terkadang mekanik juga sering lupa akan mekanisme kerja pada motor - motor tertentu. Sehingga dibutuhkan buku panduan tentang motor yang bersangkutan. Hal ini juga di nilai masih kurang efektif dan efisien dalam pelayanan konsumen, karena setiap bengkel pasti ingin memberikan yang terbaik kepada konsumennya. Oleh sebab itu sangat di butuhkan pelayanan yang cepat dan tepat sehingga mampu bersaing dengan bengkel-bengkel lainnya. Maka di buatlah aplikasi yang bisa membantu dalam menyelesaikan masalah kerusakan dan perawatan sepeda motor.

## **1.2 Rumusan Masalah**

Dari permasalahan yang telah dijelaskan di atas, dapat diambil rumusan masalah sebagai berikut :

1. Kurangnya pengetahuan konsumen bengkel mengenai masalah-masalah yang ada pada sepeda motor.
2. Besarnya kemungkinan mekanik lupa akan mekanisme kerja pada sepeda motor.

### **1.3 Batasan Masalah**

Agar pembahasan masalah tetap berada dalam batasan yang diinginkan dan tidak terlampaui jauh melewati batas apa yang akan dibahas dari permasalahan yang sebenarnya, maka dilakukan pembatasan masalah dengan ruang lingkup, sebagai berikut:

1. Tidak memberikan informasi mengenai kerusakan pada sepeda motor selain: Honda, Suzuki, Yamaha dan Kawasaki.
2. Tidak memberikan informasi mengenai kerusakan pada sistem bahan bakar yang tidak menggunakan karburator.

### **1.4 Tujuan Penelitian**

Adapun tujuan dari penelitian tugas akhir ini, adalah sebagai berikut :

1. Menjadikan pengetahuan yang mudah di dapatkan konsumen bengkel untuk mencari solusi kerusakan pada sepeda motor.
2. Sebagai pedoman bila mekanik lupa akan mekanisme kerja sepeda motor pada bagian tertentu.

### **1.5 Ikhtisar Buku**

Sistematika pembahasan laporan ini disusun dengan spesifikasi sebagai berikut:

- Bab 1 Pendahuluan berisi penjelasan mengenai latar belakang, tujuan pembuatan, batasan masalah, dan ikhtisar buku.
- Bab 2 Landasan Teori berisi penjelasan mengenai teori-teori yang berhubungan dengan penelitian.
- Bab 3 Analisis berisi tentang deskripsi fungsional yang mencakup konteks diagram pada aplikasi dan analisis kebutuhan data yang dikelola oleh aplikasi.
- Bab 4 Deskripsi Perancangan berisi tentang deskripsi data yang dikelola oleh aplikasi, dekomposisi fungsional modul, ketergantungan antar layar dan struktur menu.

- Bab 5 Implementasi dan Pengujian berisi tentang spesifikasi ketergantungan antar modul, struktur direktori dan deskripsi file yang merupakan struktur direktori dan pengumpulan fungsi menjadi file serta pengujian dan hasilnya.
- Bab 6 Kesimpulan dan Saran yang berisi tentang kesimpulan dari hasil pembangun aplikasi dan saran untuk perbaikan dan pengembangan.
- Lampiran A Berisi perancangan rinci tabel
- Lampiran B Perancangan Rinci Fungsional.  
Berisi tentang spesifikasi fungsi/proses dari sistem yang terdiri dari Spesifikasi layar utama, spesifikasi objek pada layar, spesifikasi layar pesan, spesifikasi algoritma dan spesifikasi report.
- Lampiran C Daftar rinci file dan data  
Berisi Tentang stuktur direktori pengembangan aplikasi
- Lampiran D Dokumen rinci dan testing  
Berisi tentang tabel pengujian fungsi-fungsi pada aplikasi.
- Lampiran E Manual Program penggunaan Aplikasi.

## **BAB 2 LANDASAN TEORI**

### **2.1 Sistem Pakar**

Sistem pakar adalah suatu program komputer yang dirancang untuk mengambil keputusan seperti keputusan yang diambil oleh seorang atau beberapa orang pakar. Dalam penyusunannya, sistem pakar mengkombinasikan kaidah-kaidah penarikan kesimpulan (*inference rules*) dengan basis pengetahuan tertentu yang diberikan oleh satu atau lebih pakar dalam bidang tertentu. Kombinasi dari kedua hal tersebut disimpan dalam komputer, yang selanjutnya digunakan dalam proses pengambilan keputusan untuk penyelesaian masalah tertentu.<sup>[1]</sup>

#### **2.1.1. Sistem Pakar Produksi (*production system*)**

Sistem pakar produksi adalah jenis sistem pakar yang menyimpan pengetahuan sebagai fakta-fakta dan kaidah-kaidah, dimana masing-masing kaidah tersebut disebut kaidah produksi. Sistem pakar produksi memiliki dua komponen dasar yaitu basis pengetahuan (*Knowledge Base*) dan mesin inferensi (*Inference Engine*). Basis pengetahuan adalah data atau pengetahuan yang dipergunakan untuk membuat suatu keputusan. Sedangkan mesin inferensi adalah bagian dari sistem pakar yang mereduksi fakta-fakta baru dari fakta-fakta yang telah ada dengan mempergunakan kaidah-kaidah yang ada.

Proses produksi ini menyangkut penjadwalan dan unifikasi. Disamping itu mesin inferensi juga mengontrol aliran dan tahapan inferensi. Mesin inferensi akan mengambil fakta yang ada dari basis kaidah atau basis data statis dan memori yang bekerja atau basis data dinamis kemudian menggunakannya untuk menguji kaidah-kaidah tersebut selama proses unifikasi dan ketika sukses, kaidah tersebut ditambahkan ke memori yang bekerja.<sup>[1]</sup>

### **2.1.2. Sistem Pakar Kerangka (*frame-based system*)**

Sistem pakar jenis ini dalam melakukan diagnosis dan memecahkan masalah memerlukan suatu kerangka (*frame*), Dimana semua masalah dan diagnosis kemungkinan pemecahan masalahnya telah didefinisikan terlebih dahulu. Cara pendefinisian dengan cara membuat pohon - pohon masalah dimana puncaknya berupa gejala awal diteruskan dengan pertanyaan-pertanyaan yang akan menanyakan gejala lanjutan dari gejala awal tadi, selanjutnya dengan melakukan pendeteksian maka akan diketahui pemecahan masalahnya. Pendefinisian yang lain yaitu dengan cara perancangan basis aturan yang akan dijelaskan pada bab berikutnya. Cara perancangan basis aturan inilah yang digunakan penulis sebagai basis dari sistem pakar untuk memecahkan masalah pada motor.<sup>[1]</sup>

### **2.2 Gambaran umum masalah yang biasa ditimbulkan oleh sepeda motor dan saran**

1. Mesin mendadak mati ketika dikendarai di jalan.

Saran : 1. Periksalah businya, apakah businya mati atau businya kotor.

2. Periksalah platina, apakah platina tidak bekerja dengan sempurna atau sama sekali tidak bekerja akibat baut platina longgar.
3. Periksalah kabel pengapian, apakah kabel pengapian lepas/longgar.
4. Periksalah karburator, apakah bensin tidak turun ke karburator.

2. Keluar asap putih dari knalpot.

Saran : 1. Periksalah Torak, cincin torak dan lubang silinder apakah sudah aus/rusak.

2. Periksalah tangki oli pelumas, apakah pengisian oli pelumas terlalu banyak.
3. Periksalah tekanan pompa oli apakah tekanan terlalu tinggi.
4. Penggunaan mutu oli yang kurang baik.

3. Mesin hidupnya tersendat-sendat setelah dipanaskan.

Saran : 1. Periksalah busi, apakah busi kadang kala mengeluarkan api kadang kala tidak.

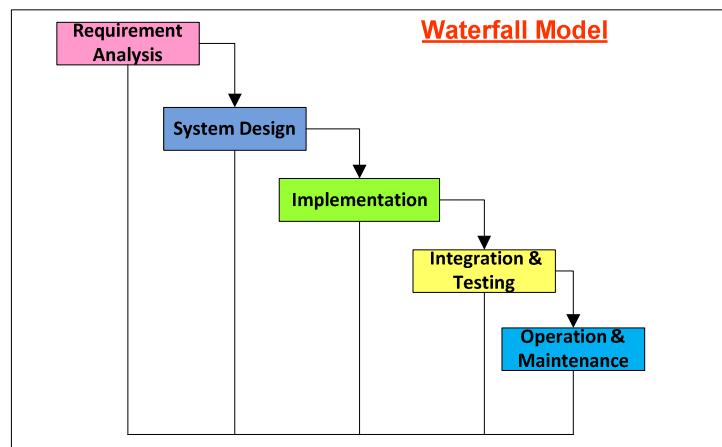
2. Periksa koil pengapian, apakah koil pengapian rusak.
3. Periksa plampung pada karburator, apakah sudah di setel dengan tepat.

Contoh onderdil sepeda motor : Ban, Karburator, Klakson, CDI, Knalpot, Rem, Starter, Speedometer, Piston, Ring Piston, Tangki, Sock Breaker, Busi, Koil, dll.<sup>[2]</sup>

### 2.3 Pengembangan Sistem dengan Metode Waterfall

Waterfall adalah sebuah model pengembangan perangkat lunak dilakukan secara sekuensial, dimana satu tahap dilakukan setelah tahap sebelumnya selesai dilaksanakan.

Ada lima tahap dalam model waterfall, yaitu: Requirement Analysis, System Design, Implementation, Integration & Testing, Operations & Maintenance. Sesuai dengan namanya waterfall (air terjun) maka tahapan dalam model ini disusun bertingkat, setiap tahap dalam model ini dilakukan berurutan, satu sebelum yang lainnya (lihat tanda anak panah).



Gambar 2.1 Tahapan Metode Waterfall

Berikut ini penjelasan tentang masing-masing tahap dalam model waterfall:

### 1. *Requirement Analysis*

Seluruh kebutuhan software harus bisa didapatkan dalam fase ini, termasuk didalamnya kegunaan software yang diharapkan pengguna dan batasan software. Informasi ini biasanya dapat diperoleh melalui wawancara, survey atau diskusi. Informasi tersebut dianalisis untuk mendapatkan dokumentasi kebutuhan pengguna untuk digunakan pada tahap selanjutnya.

### 2. *System Design*

Tahap ini dilakukan sebelum melakukan coding. Tahap ini bertujuan untuk memberikan gambaran apa yang seharusnya dikerjakan dan bagaimana tampilannya. Tahap ini membantu dalam menspesifikasikan kebutuhan hardware dan sistem serta mendefinisikan arsitektur sistem secara keseluruhan.

### 3. *Implementation*

Dalam tahap ini dilakukan pemrograman. Pembuatan software dipecah menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap berikutnya. Selain itu dalam tahap ini juga dilakukan pemeriksaan terhadap modul yang dibuat, apakah sudah memenuhi fungsi yang diinginkan atau belum.

### 4. *Integration & Testing*

Di tahap ini dilakukan penggabungan modul-modul yang sudah dibuat dan dilakukan pengujian ini dilakukan untuk mengetahui apakah software yang dibuat telah sesuai dengan desainnya dan masih terdapat kesalahan atau tidak.

### 5. *Operation & Maintenance*

Ini merupakan tahap terakhir dalam model waterfall. Software yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.<sup>[6]</sup>

## 2.4 Basis Data

Basis data atau sering pula disebut *database* adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis, sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat lunak yang digunakan untuk mengelola dan memanggil kueri (*query*) basis data disebut sistem manajemen basis data (*database management system*, DBMS). Sistem basis data dapat dipelajari dalam ilmu informasi.

Konsep dasar dari basis data adalah kumpulan dari catatan-catatan, atau potongan dari pengetahuan. Sebuah basis data memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya. Hal ini sering disebut skema. Skema menggambarkan obyek yang diwakili suatu basis data dan hubungan di antara obyek tersebut. Ada banyak cara untuk mengorganisasi skema atau memodelkan struktur basis data, ini dikenal sebagai model basis data atau model data. Model yang umum digunakan sekarang adalah model relasional, dimana pada model relational ini dapat mewakili semua informasi dalam bentuk tabel-tabel yang saling berhubungan, dimana setiap tabel terdiri dari baris dan kolom. Dalam model ini, hubungan antar tabel diwakili dengan menggunakan nilai yang sama antar tabel. Model yang lain seperti model hierarkis dan model jaringan menggunakan cara yang lebih eksplisit untuk mewakili hubungan antar tabel.

Dari penjelasan tersebut, maka dapat juga disimpulkan bahwa Basis Data merupakan Kumpulan file/tabel yang saling berelasi (berhubungan) yang disimpan dalam media penyimpanan elektronik. Dalam pengertian lain basis data adalah koleksi terpadu dari data yang saling berkaitan yang dirancang untuk memenuhi kebutuhan informasi suatu *enterprise* (dunia usaha). Dari pengertian tersebut dapat diambil kesimpulan pada masing – masing table/file didalam database berfungsi untuk menampung/menyimpan data-data, dimana masing – masing data yang ada pada table/file tersebut saling berhubungan dengan satu sama lainnya.

## **2.5 Microsoft Access**

Microsoft Access atau Microsoft Office Access adalah sebuah program aplikasi basis data komputer relasional yang ditujukan untuk kalangan rumahan dan perusahaan kecil hingga menengah. Aplikasi ini merupakan anggota dari beberapa aplikasi Microsoft Office, selain tentunya Microsoft Word, Microsoft Excel, dan Microsoft PowerPoint. Aplikasi ini menggunakan mesin basis data Microsoft Jet Database Engine, dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna.<sup>[4]</sup>

## **2.6 Visual Basic 2005**

Microsoft Visual Basic 2005 adalah sebuah alat untuk mengembangkan dan membangun aplikasi dengan menggunakan bahasa BASIC. Dengan menggunakan aplikasi ini, para programmer dapat membangun aplikasi Windows Forms dan juga aplikasi command-line. Alat ini dapat diperoleh secara terpisah dari beberapa produk lainnya (seperti Microsoft Visual C++, Visual C#, atau Visual J#), atau juga dapat diperoleh secara terpadu dalam Microsoft Visual Studio 2005. Bahasa Visual Basic 2005 sendiri menganut paradigma bahasa pemrograman berorientasi objek yang dapat dilihat sebagai evolusi dari Microsoft Visual Basic versi sebelumnya.

BASIC pertama kali dikembangkan pada tahun 1963 oleh John George Kemeny dan Thomas Eugene Kurtz yang berasal dari Dartmouth College, untuk mengizinkan akses terhadap komputer bagi para mahasiswa jurusan selain jurusan ilmu eksakta. Pada waktu itu, hampir semua komputer membutuhkan perangkat lunak, dan waktu itu belum ada perangkat lunak yang dijual secara bebas, sehingga hanya orang-orang tertentu yang dapat menggunakan komputer, yakni para matematikawan dan ilmuwan, karena mereka dapat membangun perangkat lunak sendiri. Bahasa BASIC, setelah diciptakan menjadi menjamur dan banyak dimodifikasi. Bahasa BASIC menjadi bahasa yang paling populer digunakan pada komputer mikro pada akhir tahun 1970-an dan komputer rumahan pada tahun 1980-an. Dan hingga saat ini, menjadi bahasa yang dialeknya beberapa kali berevolusi.<sup>[5]</sup>


## 2.7 Data Flow Diagram (DFD)

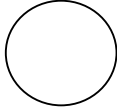

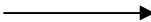
Sebelum mengimplementasi program, maka dilakukan pembuatan DFD atau Data Flow Diagram. DFD adalah suatu grafik yang menjelaskan sebuah sistem dengan menggunakan bentuk-bentuk atau simbol untuk menggambarkan aliran data dari proses-proses yang saling berhubungan. DFD juga menggambarkan *input*, *process*, dan *output* yang terjadi pada sebuah sistem. Dalam mendokumentasikan sebuah sistem, DFD mempunyai *level-level* mulai dari yang terkecil, yaitu level 0 atau sering disebut *context* diagram.

*Context* diagram ini merupakan gambaran paling umum dari sistem, yang hanya memiliki satu proses saja untuk mewakili seluruh sistem. Semakin bertambahnya level dalam DFD akan semakin *detail* digambarkannya proses-proses yang ada pada sistem, tetapi yang boleh bertambah hanya proses dan data *flow* saja. Sedangkan untuk data *source*, jumlahnya harus tetap dengan yang ada pada *context* diagram.

Khusus untuk data *store*, pada *context* diagram masih belum digambarkan, akan digambarkan pada level 1 (satu) dan konsisten jumlahnya sampai level berikutnya.

**Tabel 2.1**  
**Simbol Data Flow Diagram**

Simbol	Keterangan
	Simbol entitas eksternal atau terminador menggambarkan asal atau tujuan data dari luar sistem.

	Simbol lingkaran menggambarkan proses dimana aliran data masuk ditransformasikan ke aliran data keluar
	Tempat penyimpanan Data
	Simbol aliran data menggambarkan aliran data.

## 2.8 Entity Relation Diagram (ERD)

ERD merupakan suatu model untuk menjelaskan hubungan antar data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi.

ERD untuk memodelkan struktur data dan hubungan antar data, untuk menggambarkannya digunakan beberapa notasi dan simbol. Pada dasarnya ada tiga simbol yang digunakan, yaitu :

a. *Entity* / Entitas

Entity merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari *entity* ini biasanya digambarkan dengan persegi panjang.

b. Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasi isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol elips.

c. Hubungan / Relasi

Hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda.

Relasi dapat digambarkan sebagai berikut :

Relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dalam satu basis data yaitu :

1. Satu ke satu (*One to one*)

Hubungan relasi satu ke satu yaitu setiap entitas pada himpunan entitas A berhubungan paling banyak dengan satu entitas pada himpunan entitas B.

2. Satu ke banyak (*One to many*)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan satu entitas pada himpunan entitas A.





3. Banyak ke satu (*many to one*)

Banyak himpunan pada entitas A dapat berhubungan dengan satu entitas pada himpunan entitas B, tetapi setiap entitas pada entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

4. Banyak ke banyak (*Many to many*)

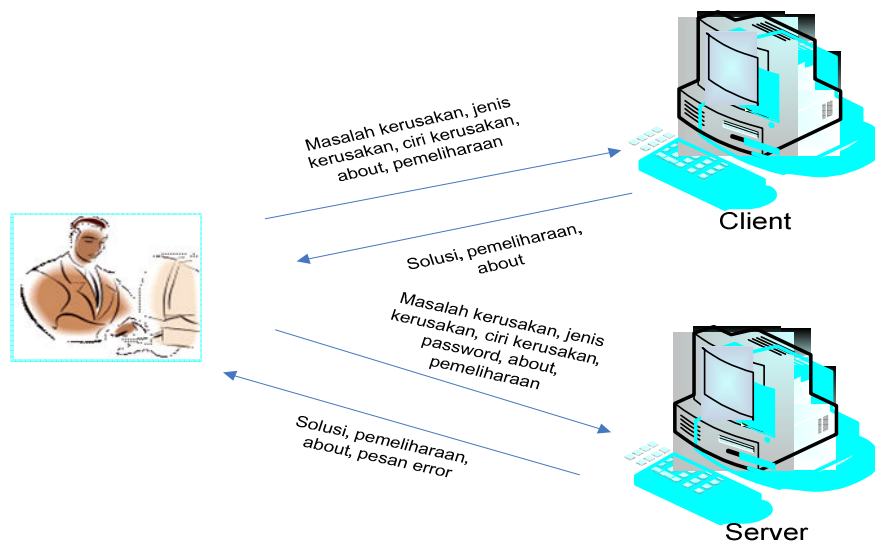
Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B.

**Tabel 2.2**  
**Simbol Entity Relation Diagram (ERD)**

Simbol	Keterangan
	<p><b>Entitas</b>, adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai.</p>
	<p><b>Relasi</b>, menunjukkan adanya hubungan di antara sejumlah entitas yang berbeda.</p>
	<p><b>Atribut</b>, berfungsi mendeskripsikan karakter entitas (atribut yang berfungsi sebagai <i>primary key</i> diberi garis bawah)</p>
	<p>Garis, sebagai penghubung antara relasi dengan entitas, relasi dan entitas dengan atribut.</p>

## BAB 3 ANALISIS

### 3.1 Deskripsi Umum Sistem



Gambar 3.1. Deskripsi umum sistem

Penjelasan Gambar :

Pada aplikasi ini hanya terdapat dua user yaitu konsumen bengkel dan mekanik senior. Konsumen bengkel memasukkan masalah ke Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan Dan Perawatan Sepeda Motor, kemudian aplikasi akan memberikan solusi kepada user sesuai dengan informasi yang dibutuhkan. Mekanik senior bertugas memasukkan masalah – masalah kerusakan yang ada pada sepeda motor beserta solusinya.

### 3.2 Karakteristik Pengguna

Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan Dan Perawatan Sepeda Motor ini terkait dengan beberapa pengguna yang dapat mengakses aplikasi. Untuk level user adalah konsumen bengkel dan level administrator adalah seorang mekanik senior sebagai pengelola aplikasi.

**Tabel 3.2.1 Kategori Pengguna Aplikasi ini**

<b>Kategori Pengguna</b>	<b>Tugas</b>	<b>Hak Akses Ke Aplikasi</b>	<b>Jabatan</b>
Admin	<ul style="list-style-type: none"><li>• Memasukkan informasi-informasi ke dalam aplikasi sistem pakar</li></ul>	<ul style="list-style-type: none"><li>- Menambah data.</li><li>- Mengedit data.</li><li>- Menghapus data.</li></ul>	Mekanik senior
User	<ul style="list-style-type: none"><li>• Memasukkan keyword</li></ul>	<ul style="list-style-type: none"><li>• Memasukkan masalah.</li><li>• Melihat solusi.</li></ul>	Konsumen bengkel

### **3.3 Batasan Sistem**

Batasan Sistem pada Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan Dan Perawatan Sepeda Motor adalah:

1. Aplikasi ini dikembangkan dengan pemograman Visual Basic 2005.
2. Aplikasi ini menggunakan database Mysql.
3. Aplikasi ini hanya di alokasikan untuk dua komputer (*client & server*).
4. Aplikasi ini hanya memiliki satu admin.

### **3.4 Lingkungan Operasional dan Pengembangan**

#### **3.4.1 Lingkungan Operasional**

- a. Perangkat Keras:
  - Prosesor : *Minimal pentium IV*
  - Kebutuhan memory utama: *512 Mb*
- b. Sistem Operasi: *Microsoft XP*
- c. DBMS: *Microsoft Access*
- d. Program atau utilities lain: *.Net Framework*

### **3.4.2 Lingkungan Pengembangan**

- a. Perangkat keras:
  - Prosesor: *Intel Dual-Core*
  - Kebutuhan memory utama minimal: *1 Gb*
- b. Sistem operasi: *Mirosoft XP*
- c. DBMS: *Microsoft Access*
- d. Program atau utilities lain: *Visual Basic 2005*.

### **3.5 Aturan Penomoran**

Aturan penamaan dan penomoran pada aplikasi ini adalah:

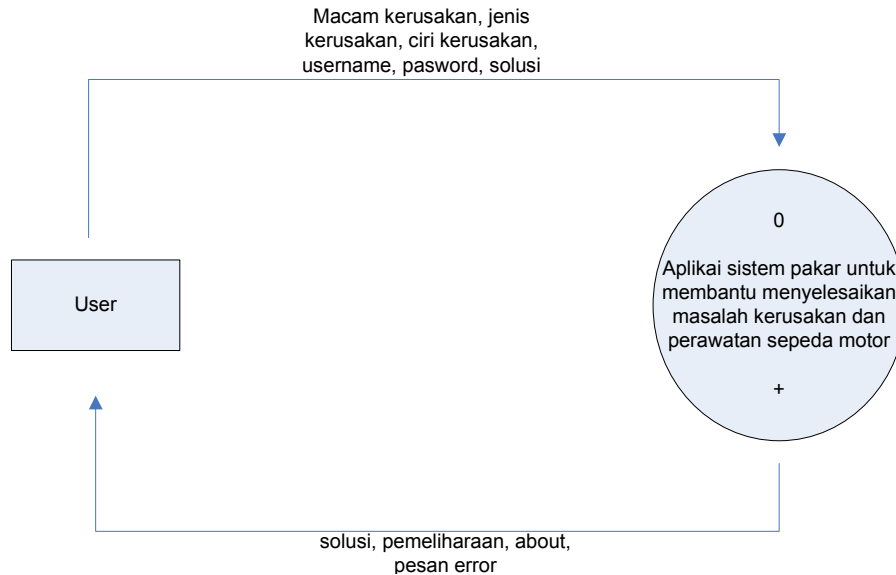
- 1 Penomoran fungsi di beri nomor sesuai dengan nomor fungsinya dan di awali huruf F, misalnya: F1
- 2 Tabel penyimpanan data di dalam database di beri nama sesuai dengan nama tabel dan di ikuti dengan huruf tb, misalnya: tbMacamKerusakan

### **3.6 Deskripsi Fungsional**

Aplikasi ini menangani beberapa proses yang akan dijelaskan pada *context* diagram dan DFD Level 1 diantaranya adalah sebagai berikut :

1. Penelusuran
2. Input pakar
3. Pemeliharaan
4. About

### 3.6.1 Context Diagram

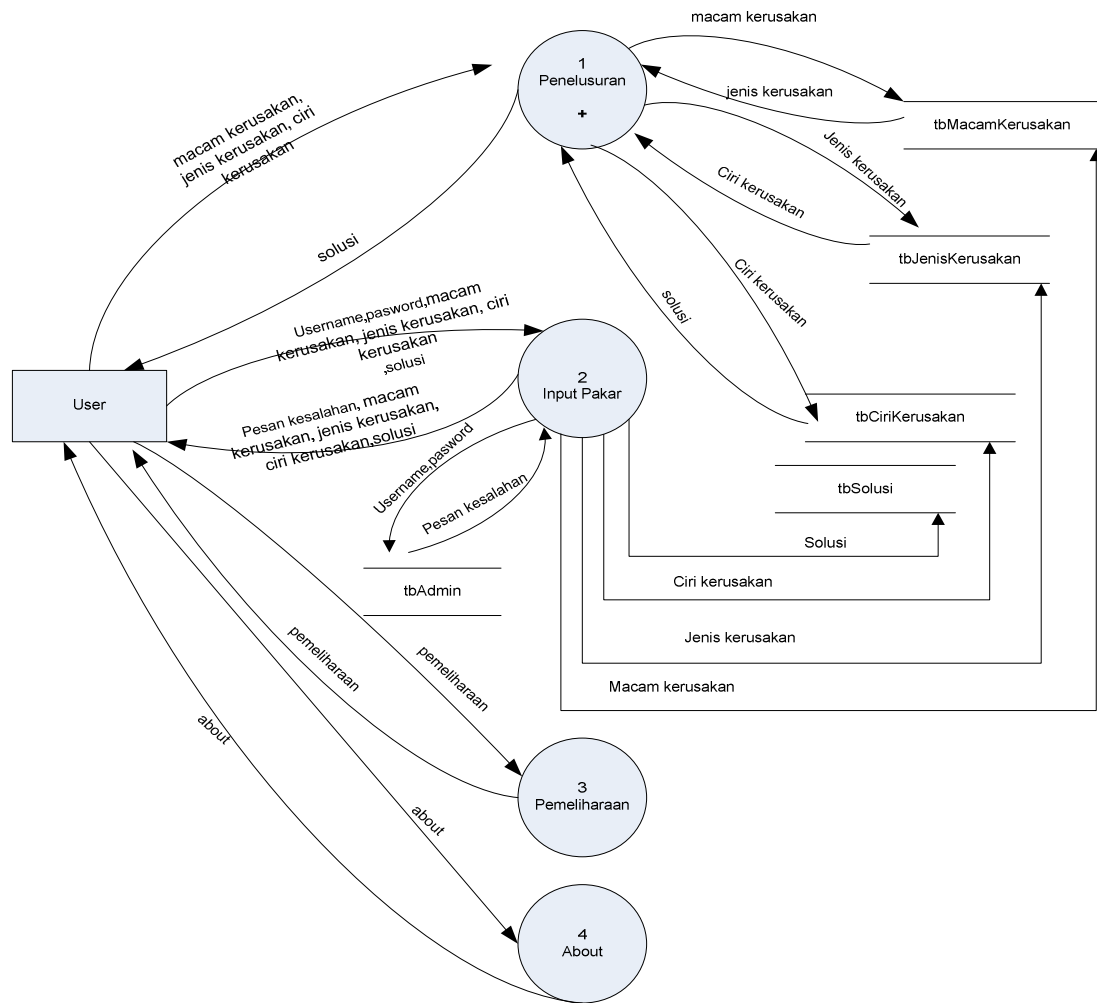


**Gambar 3.6.1. Context Diagram**

Penjelasan gambar :

User yang menggunakan aplikasi ini adalah konsumen bengkel dan mekanik bengkel. Untuk melihat solusi masalah kerusakan, user harus melakukan penelusuran ke aplikasi yang berawal dari macam kerusakan, jenis kerusakan, dan ciri kerusakan. Hasil penelusuran ini akan di proses aplikasi. Jika solusi yang tepat di temukan maka akan ditampilkan. Jika tidak ada solusi maka akan muncul pesan solusi tidak ditemukan.

### 3.6.2 DFD Level 1



Gambar 3.6.2 DFD Level 1

Penjelasan gambar:

1. Penelusuran

Pada proses ini user di arahkan untuk melakukan penelusuran berdasarkan masalah kerusakan yang dihadapi. Tahapan berawal dari macam kerusakan, jenis kerusakan dan ciri kerusakan. Hasil penelusuran ini akan di proses aplikasi. Jika solusi yang tepat di temukan maka akan ditampilkan. Jika tidak ada solusi maka akan muncul pesan solusi tidak ditemukan.

2. Input pakar

Proses penginputan informasi- informasi yang di lakukan oleh admin ke dalam aplikasi sistem pakar yang mencakup masalah- masalah kerusakan pada sepeda motor. Penginputan informasi berdasarkan tahapan yang telah di tentukan.

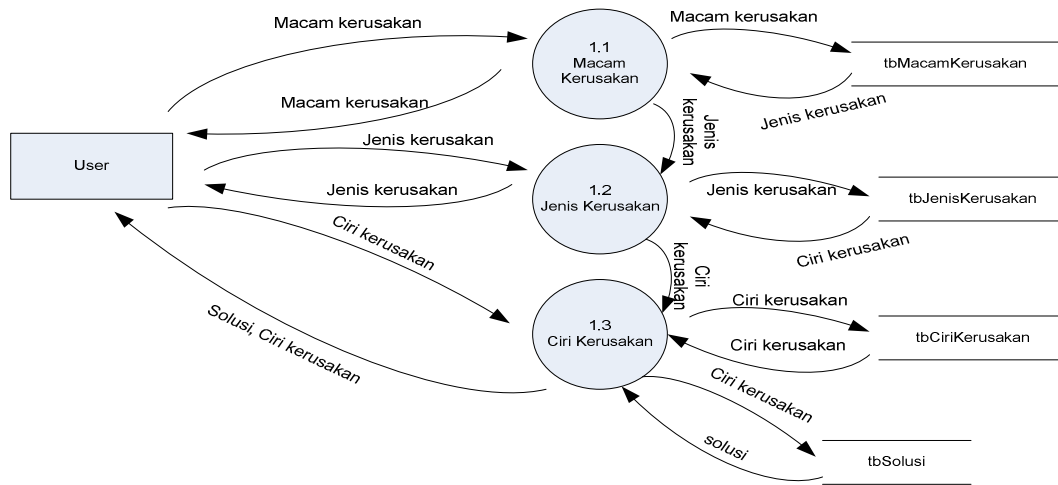
3. Pemeliharaan

Pada proses ini aplikasi menampilkan informasi pemeliharaan rutin untuk mencegah terjadinya masalah kerusakan pada sepeda motor. Informasi pemeliharaan berbentuk dokumen yang dapat di edit langsung dari aplikasi. Informasi pemeliharaan hanya dapat di edit oleh admin.

4. About

Proses menampilkan tentang profile aplikasi sistem pakar untuk membantu menyelesaikan masalah kerusakan dan perawatan sepeda motor.

### 3.6.3 DFD Level 2 Proses Penelusuran



Gambar 3.6.3 DFD Level 2 Proses Penelusuran

Penjelasan gambar:

#### 1.1. Macam Kerusakan

Pada proses ini user di arahkan untuk melakukan penelusuran awal berdasarkan macam kerusakan sepeda motor. Informasi macam kerusakan di tampilkan dari tabel tbMacamKerusakan.

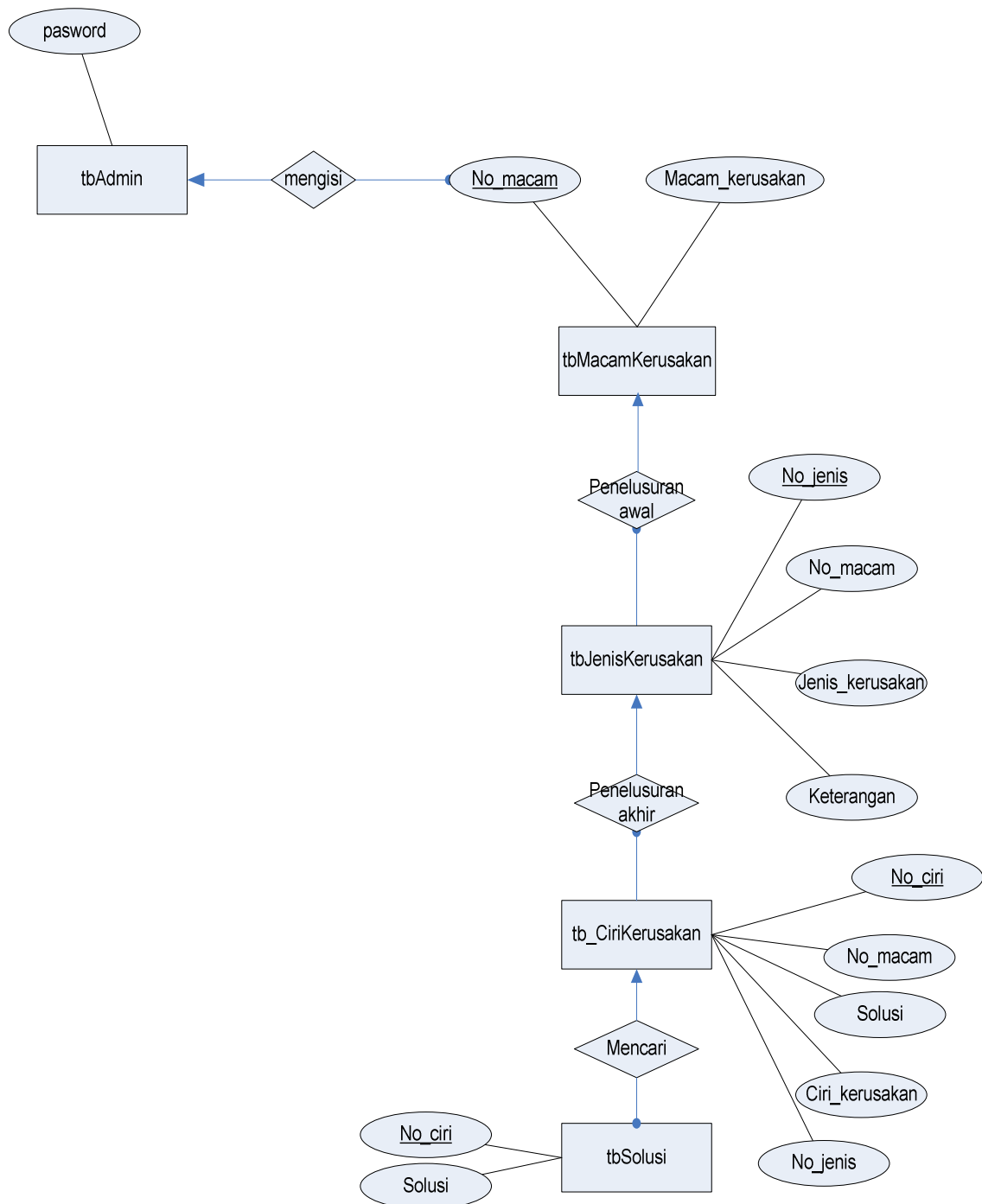
#### 1.2. Jenis Kerusakan

Pada proses ini user di arahkan untuk melakukan penelusuran lanjutan berdasarkan jenis kerusakan sepeda motor. Informasi macam kerusakan di tampilkan dari tabel tbJenisKerusakan.

#### 1.3. Ciri Kerusakan

Pada proses ini user di arahkan untuk melakukan penelusuran akhir berdasarkan ciri kerusakan sepeda motor. Informasi ciri kerusakan di tampilkan dari tabel tbCiriKerusakan dan tbSolusi. Hasil penelusuran dari proses 1.1. Macam Kerusakan sampai proses 1.3. Ciri Kerusakan akan menghasilkan solusi yang bisa membantu user dalam menyelesaikan kerusakan pada sepeda motor. Jika solusi tidak di temukan maka akan tampil pesan solusi tidak di temukan.

### 3.7 E.R. Diagram



Gambar 3.7. E.R Diagram

## BAB 4 PERANCANGAN

### 4.1 Deskripsi Data

Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan Dan Perawatan Sepeda Motor merupakan aplikasi yang menggunakan basis data.

**Tabel 4.1 Daftar Tabel Deskripsi Data Pada Aplikasi**

No	Nama tabel	Jenis	Primary key	Volume	Perkiraan laju	Deskripsi isi	Constraint integrity
1.	tbAdmin	Master	-	1 record	1 record/hari	Pasword admin	-
2.	tbMacam Kerusakan	Master	No_macam	± 200 record	± 20 record/hari	Berisi tentang data macam kerusakan motor.	-
3.	tbJenis Kerusakan	Master	No_jenis	± 200 record	± 20 record/hari	Data jenis kerusakan motor.	-
4.	tbCiriKerusakan	Master	No_ciri	± 200 record	± 20 record/hari	Berisi tentang data ciri kerusakan motor.	-
5.	tbSolusi	Master	-	± 200 record	± 20 record/hari	Berisi data solusi kerusakan	-

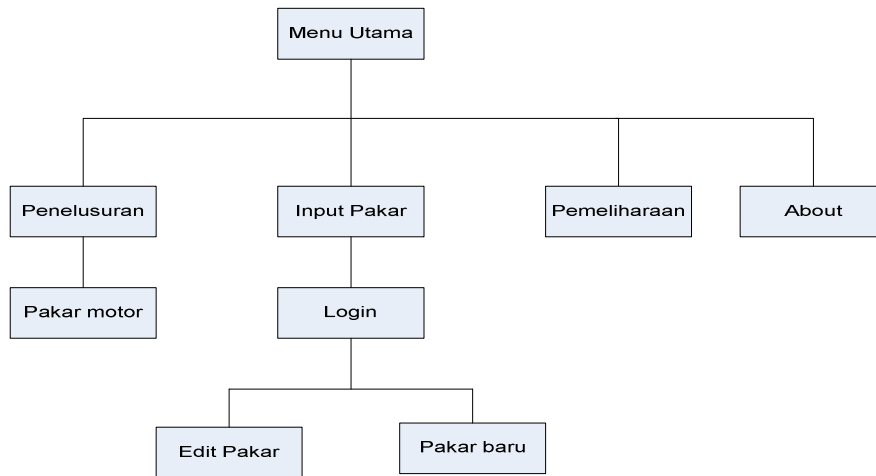
### 4.1.1 Daftar Tabel

Di bawah ini merupakan daftar tabel basis data yang terdapat pada Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan Dan Perawatan Sepeda Motor.

**Tabel 4.2 Daftar Tabel Basis Data Pada Aplikasi**

No	Nama Tabel	Primary key	Data Store	E/R	Deskripsi isi
1.	tbAdmin	-	tbAdmin		Pasword admin
2.	tbMacam Kerusakan	No_macam	tbMacamKerusakan		Berisi tentang data macam kerusakan motor.
3.	tbJenis Kerusakan	No_jenis	tbJenisKerusakan		Berisi tentang data jenis kerusakan motor.
4.	tbCiri Kerusakan	No_ciri	tbCiriKerusakan		Berisi tentang data ciri kerusakan motor.
5.	tbSolusi	-	tbSolusi		Berisi tentang data solusi kerusakan.

## 4.1.2 Spesifikasi Kebergantungan Antar Layar



**Gambar 4.2.1 Spesifikasi Kebergantungan Antar Layar Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan Dan Perawatan Sepeda Motor**

## 4.1.3 Struktur Menu

Aplikasi Sistem Pakar Untuk Membantu Menyelesaikan Masalah Kerusakan Dan Perawatan Sepeda Motor :

- Menu Penelusuran
  - Pakar Motor
- Menu Input Pakar
  - Menu Login
    - Edit Pakar
    - Pakar Baru
- Menu Pemeliharaan
- About

## BAB 5 IMPLEMENTASI DAN PENGUJIAN

Setelah dilakukan tahap perancangan aplikasi ini maka tahap selanjutnya adalah implementasi aplikasi tersebut. Lingkungan yang digunakan untuk mengimplementasikan aplikasi ini telah dijelaskan dalam bab III. Implementasi akan menghasilkan aplikasi yang dapat dijalankan di lingkungan operasional. Untuk mengetahui apakah aplikasi tersebut dapat melakukan fungsi sesuai deskripsi perencanaan maka perlu dilakukan pengujian.

### 5.1 Struktur Direktori dan Deskripsi File

**Tabel 5.2.1 Daftar Direktori dan file**  
**Aplikasi sistem pakar untuk membantu menyelesaikan masalah kerusakan dan perawatan pada motor.**

<b>Nama Direktori</b>	<b>Nama File</b>	<b>Nama Modul</b>	<b>Nama Fungsi</b>	<b>Keterangan</b>
PakarMotor	formLogin.vb	Tidak ada	F1	Form ini berfungsi untuk masuk ke Menu admi/ pakar.
	formGantipass.vb	Tidak ada	-	Form ini berfungsi untuk mengganti password admin.
	formPenelusuran.vb	Tidak ada	F2	Form ini berfungsi untuk melakukan penelusuran.
	formMenuUtama.vb	Tidak ada	-	Form ini berfungsi untuk menampilkan menu utama
	formPemeliharaan.vb	Tidak ada	F3	Form ini berfungsi untuk menampilkan info pemeliharaan.

Nama Direktori	Nama File	Nama Modul	Nama Fungsi	Keterangan
PakarMotor	formInputPakar.vb	Tidak ada	F4	Form ini berfungsi untuk meng- <i>edit</i> informasi pakar.
	formNewPakarMK.vb	Tidak Ada	F5	Form ini berfungsi untuk memasukkan macam kerusakan baru.
	formNewPakarJK.vb	Tidak ada	F6	Form ini berfungsi untuk memasukkan jenis kerusakan baru.
	formPakarBaru.vb	Tidak ada	F7	Form ini berfungsi untuk menyimpan pakar baru.
	formSolusi.vb	Tidak Ada	F8	Form ini berfungsi untuk menampilkan solusi.
	formLoading.vb	Tidak Ada	-	Form ini berfungsi untuk loading.

## 5.2 Pengujian dan Hasilnya

Implementasi dan Pengujian Aplikasi yang saya lakukan sudah sesuai dengan deskripsi perancangan. Untuk lebih jelasnya dapat dilihat dalam lampiran D.

## **BAB 6 KESIMPULAN DAN SARAN**

### **6.1 Kesimpulan**

Setelah melakukan pengembangan terhadap Aplikasi sistem pakar untuk membantu menyelesaikan masalah kerusakan dan perawatan motor, maka kesimpulan yang didapat adalah:

1. Aplikasi ini bisa menjadi pengetahuan yang mudah di dapatkan konsumen bengkel untuk mencari solusi kerusakan pada sepeda motor.
2. Aplikasi ini bisa menjadi pedoman bila mekanik lupa akan mekanisme kerja sepeda motor pada bagian tertentu.

### **6.2 Saran**

Saran yang dapat diberikan untuk penyempurnaan dari aplikasi ini adalah:

1. Aplikasi ini diharapkan nantinya bisa memberikan informasi mengenai kerusakan pada sepeda motor selain: Honda, Suzuki, Yamaha dan Kawasaki.
2. Solusi pada aplikasi ini nantinya dapat disertai gambar.

## LAMPIRAN A PERANCANGAN RINCI TABEL

### A.1 Spesifikasi Tabel Admin

Nama tabel : tbAdmin

Deskripsi isi : Data Admin

Primary Key : -

Volume : ±1 Record

Perkiraan Laju: ±1 Record/Hari

Daftar Field

<b>Nama field</b>	<b>Deskripsi Isi</b>	<b>Type &amp; Length</b>	<b>Boleh NULL</b>	<b>Default</b>	<b>Keterangan lain</b>
Pasword	Password	Varchar(10)	Tidak	-	-

### A.2 Spesifikasi Tabel Macam Kerusakan

Nama tabel : tbMacamKerusakan

Deskripsi isi : Data Macam Kerusakan sepeda motor

Primary Key : NoMacam

Volume : ±200 Record

Perkiraan Laju: ±20 Record/Hari

Daftar Field

<b>Nama field</b>	<b>Deskripsi Isi</b>	<b>Type &amp; Length</b>	<b>Boleh NULL</b>	<b>Default</b>	<b>Keterangan lain</b>
NoMacam	Nomor macam kerusakan	AutoNumber	Tidak	-	Primary key
MacamKerusakan	Macam- macam kerusakan	Varchar(80)	Tidak	-	-

### A.3 Spesifikasi Tabel Jenis Kerusakan

Nama tabel : tbJenisKerusakan

Deskripsi isi : Data Jenis Kerusakan sepeda motor

Primary Key : NoJenis

Volume : ±200 Record

Perkiraan Laju: ±20 Record/Hari

Daftar Field

Nama field	Deskripsi Isi	Type & Length	Boleh NULL	Default	Keterangan lain
NoJenis	Nomor jenis kerusakan	AutoNumber	Tidak	-	Primary key
NoMacam	Nomor macam kerusakan	Integer(6)	Tidak	-	-
JenisKerusakan	Jenis- jenis kerusakan	Varchar(100)	boleh	-	-
Keterangan	Keterangan	Varchar(100)	boleh	-	-

### A.4 Spesifikasi Tabel Ciri Kerusakan

Nama tabel : tbCiriKerusakan

Deskripsi isi : Data Ciri- ciri kerusakan

Primary Key : NoCiri

Volume : ±200 Record

Perkiraan Laju: ±20 Record/Hari

#### Daftar Field

<b>Nama field</b>	<b>Deskripsi Isi</b>	<b>Type &amp; Length</b>	<b>Boleh NULL</b>	<b>Default</b>	<b>Keterangan lain</b>
NoCiri	Nomor ciri kerusakan	AutoNumber	Tidak	-	Primary key
NoMacam	Nomor macam kerusakan	Integer(6)	Tidak	-	-
NoJenis	Nomor jenis kerusakan	Integer(6)	Tidak	-	-
CiriKerusakan	Ciri- ciri kerusakan	Varchar(100)	boleh	-	-
Solusi	Solusi	Varchar(500)	boleh	-	-

#### **A.5 Spesifikasi Tabel Solusi**

Nama tabel : tbSolusi

Deskripsi isi : Data Solusi Kerusakan sepeda motor

Primary Key : NoCiri

Volume : ±200 Record

Perkiraan Laju: ±20 Record/Hari

#### Daftar Field

<b>Nama field</b>	<b>Deskripsi Isi</b>	<b>Type &amp; Length</b>	<b>Boleh NULL</b>	<b>Default</b>	<b>Keterangan lain</b>
NoCiri	Nomor ciri kerusakan	Integer(6)	Tidak	-	Primary key
Solusi	Solusi kerusakan	Varchar(80)	Tidak	-	-

## LAMPIRAN B PERANCANGAN RINCI FUNGSIONAL

### B.1. Spesifikasi Fungsi / Proses FLoginAdmin

Identifikasi / Nama : Login Admin  
Deskripsi Isi : Tampilan Login Admin  
Jenis : Form Entry columnar

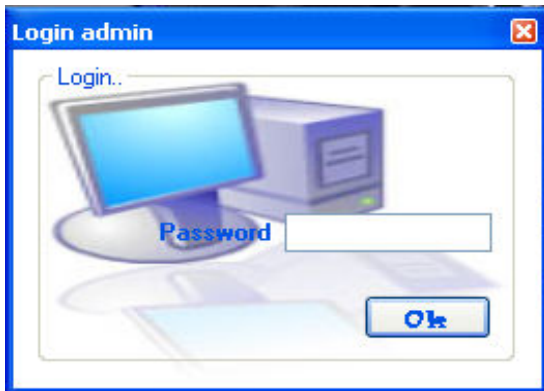
#### B.1.1. Spesifikasi Tabel Input

Nama Tabel : tbAdmin

#### B.1.2. Spesifikasi Tabel Output

Tidak ada.

#### B.1.3. Spesifikasi Layar Login Admin



#### B.1.4. Spesifikasi Query

Query1 = SELECT pasword FROM tbAdmin WHERE pasword="txtPasword";

#### B.1.5. Spesifikasi Field Pada Layar

Label	Field	Tabel	I/O	Format
Pasword	txtPasword	tbAdmin	I	Varchar(15)

### B.1.6. Spesifikasi Function Key/Objek-Objek pada layar

<b>Id Objek</b>	<b>Jenis</b>	<b>Keterangan</b>
1.	lblpassword	Label password
2.	txtPasword	Untuk inputan password
3.	btnOk	Untuk Login
4.	lblGantiPwd	Untuk ganti password

### B.1.7. Spesifikasi layar pesan

Tidak ada

### B.1.8. Spesifikasi proses/algoritma

**B.1.8.1. <F1 >** : Menu Login Admin

**Objek terkait** : Data Admin

**Event** : Klik

<b>Initial State (IS) :</b> TextField Kosong
<b>Final State (FS) :</b> Masuk ke menu Tambah Macam Kerusakan
<b>Spesifikasi Proses/algoritma:</b> Koneksi open; Input password; IF(password== true){ Tampilkan Menu Admin; }else{ Tampil messageBox(Password yang anda masukkan salah); } Koneksi close

### B.1.9. Spesifikasi Report

Tidak ada

### B.2. Spesifikasi Fungsi / Proses FInsertMacamKerusakan

Identifikasi / Nama : Insert Macam Kerusakan

Deskripsi Isi : Macam kerusakan

Jenis : Form Entry columnar

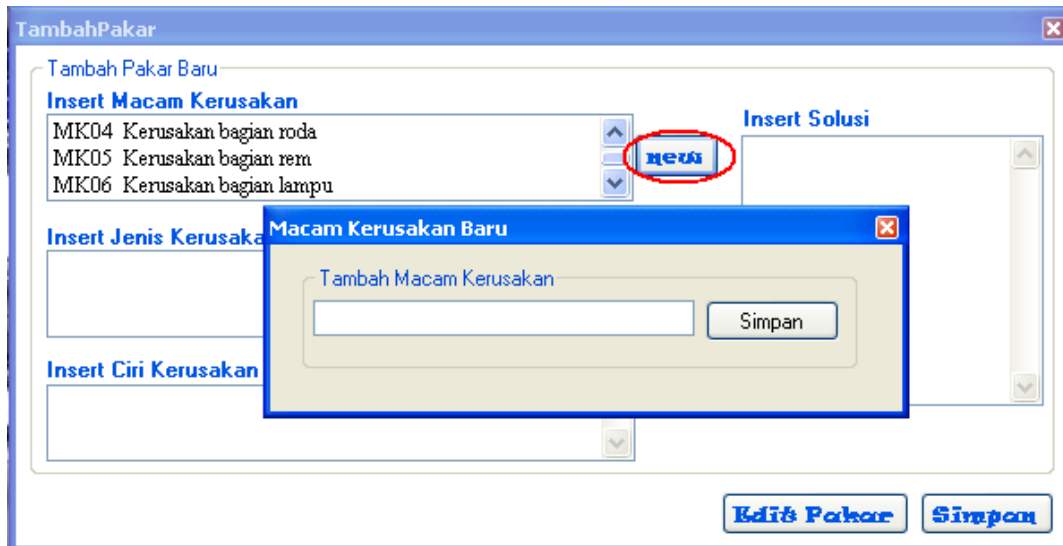
#### B.2.1. Spesifikasi Tabel Input

Nama Tabel : tbMacamKerusakan

### B.2.2. Spesifikasi Tabel Output

Nama Tabel : tbMacamKerusakan

### B.2.3. Spesifikasi Layar Insert Macam Kerusakan



### B.2.4. Spesifikasi Query

Query1 = INSERT INTO tbMacamKerusakan (NoMacam,'MacamKerusakan') values ('txtNewPakarMK');

### B.2.5. Spesifikasi Field Pada Layar

Label	Field	Tabel	I/O	Format
grpMK	txtNewPakarMK	tbMacamKerusakan	I/O	Varchar(80)

### B.2.6. Spesifikasi Function Key/Objek-Objek pada layar

Id Objek	Jenis	Keterangan
1.	grpNewPakarMK	Group box macam kerusakan
2.	txtNewPakarMK	Untuk inputan macam kerusakan baru
3.	btnNewPakarMK	Untuk menyimpan macam kerusakan baru

### B.2.7. Spesifikasi layar pesan

Tidak ada

## B.2.8. Spesifikasi proses/algoritma

**B.2.8.1. <F1 >** : Menu Tambah Macam Kerusakan

**Objek terkait** : Data Macam Kerusakan

**Event** : Klik

<b>Initial State (IS):</b> TextField password kosong
<b>Final State (FS):</b> TextField password berisi
<b>Spesifikasi Proses/algoritma:</b> Koneksi open; Input Macam Kerusakan; IF(txtNewPakarMK =" ") { Tampil messageBox(TextField Tidak Boleh Kosong); }else{ insert into tbMacamKerusakan (NoMacam,MacamKerusakan) values (txtNewPakarMK); } Koneksi close;

## B.2.9. Spesifikasi Report

Tidak ada

## B.3. Spesifikasi Fungsi / Proses FInsertJenisKerusakan

Identifikasi / Nama : Insert Jenis Kerusakan

Deskripsi Isi : Jenis Kerusakan

Jenis : Form Entry columnar

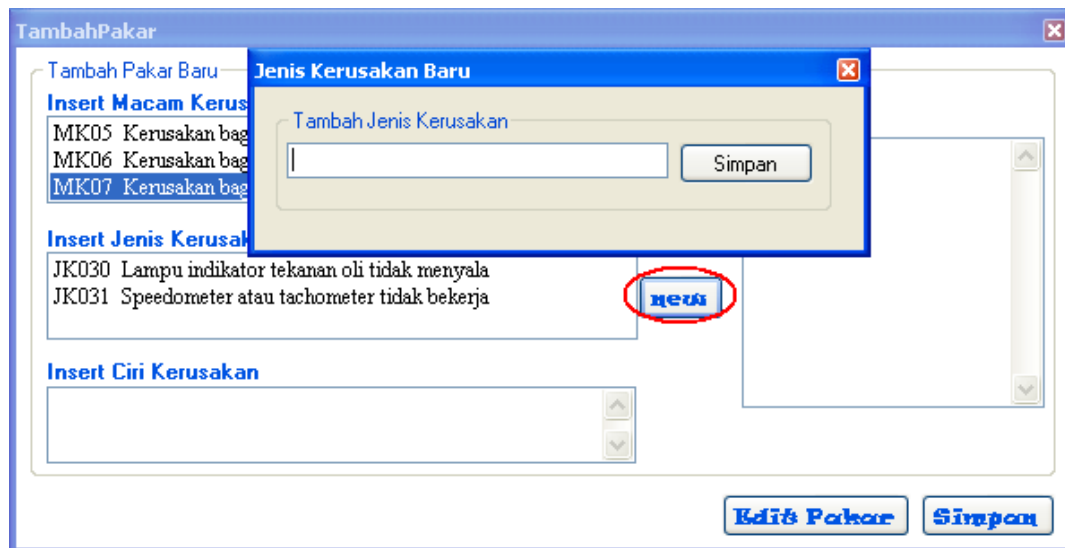
### B.3.1. Spesifikasi Tabel Input

Nama Tabel : tbJenisKerusakan

### B.3.2. Spesifikasi Tabel Output

Nama Tabel : tbJenisKerusakan

### B.3.3. Spesifikasi Layar Insert Jenis Kerusakan



### B.3.4. Spesifikasi Query

Query1 = INSERT INTO tbJenisKerusakan (NoJenis, NoMacam, 'JenisKerusakan', 'Keterangan') values (txtNoMK, 'txtNewPakarJK', 'txtKetJK' );

### B.3.5. Spesifikasi Field Pada Layar

Label	Field	Tabel	I/O	Format
grpJK	txtNewPakarJK	tbJenisKerusakan	I/O	Varchar(80)

### B.3.6. Spesifikasi Function Key/Objek-Objek pada layar

Id Objek	Jenis	Keterangan
1.	grpNewPakarJK	Group box jenis kerusakan
2.	txtNewPakarJK	Untuk inputan jenis kerusakan baru
3.	btnNewPakarJK	Untuk menyimpan jenis kerusakan baru

### B.3.7. Spesifikasi layar pesan

Tidak ada.

### B.3.8. Spesifikasi proses/algoritma

**B.3.8.1. <F1 >** : Menu Tambah Jenis Kerusakan

**Objek terkait** : Data Jenis Kerusakan

**Event** : Klik

<b>Initial State (IS) :</b> TextField Kosong
<b>Final State (FS) :</b> Masuk ke menu Tambah Ciri Kerusakan
<b>Spesifikasi Proses/algoritma:</b> Koneksi open; Input Jenis Kerusakan; IF(txtNewPakarJK = " ") { Tampil messageBox(TextField Tidak Boleh Kosong); }else{ insert into tbjeniskerusakan (NoJenis, NoMacam, JenisKerusakan, Keterangan) values (txtNoMK,txtNewPakarJK,txtKetJK); } Koneksi close;

### B.3.9. Spesifikasi Report

Tidak ada

### B.4. Spesifikasi Fungsi / Proses FInsertCiriKerusakan

Identifikasi / Nama : Insert Ciri Kerusakan

Deskripsi Isi : Ciri Kerusakan

Jenis : Form Entry columnar

#### B.4.1. Spesifikasi Tabel Input

Nama Tabel : tbCiriKerusakan

#### B.4.2. Spesifikasi Tabel Output

Tidak ada.

### B.4.3. Spesifikasi Layar Insert Ciri Kerusakan

The screenshot shows a window titled "TambahPakar" with a sub-header "Tambah Pakar Baru". It contains three main sections:
 

- Insert Macam Kerusakan:** A list box with items "MK05 Kerusakan bagian rem", "MK06 Kerusakan bagian lampu", and "MK07 Kerusakan bagian instrumen". A "New" button is to its right.
- Insert Jenis Kerusakan:** A list box with items "JK030 Lampu indikator tekanan oli tidak menyala" and "JK031 Speedometer atau tachometer tidak bekerja". A "New" button is to its right.
- Insert Ciri Kerusakan:** An empty text input field, which is circled in red in the image.

 To the right of these sections is a large text area labeled "Insert Solusi". At the bottom right, there are "Edit Pakar" and "Simpan" buttons.

### B.4.4. Spesifikasi Query

Query1 = INSERT INTO tbCiriKerusakan (NoCiri,NoJenis, NoMacam, 'CiriKerusakan', 'Solusi') values (txtNoJK, txtNoMK, ' txtNewPakarCK', 'txtSolusi' );

### B.4.5. Spesifikasi Field Pada Layar

Label	Field	Tabel	I/O	Format
grpCK	txtNewPakarMK	tbJenisKerusakan	I	Varchar(80)
lblSolusiTP	txtSolusiTP	tbJenisKerusakan	I	Varchar(1000)

### B.4.6. Spesifikasi Function Key/Objek-Objek pada layar

Id Objek	Jenis	Keterangan
1.	Label2	Label insert ciri kerusakan
2.	Label5	Label insert solusi
3.	txtTpCK	Untuk inputan ciri kerusakan baru
4.	txtTpSolusiCK	Untuk inputan solusi kerusakan

### B.4.7. Spesifikasi layar pesan

Tidak ada

#### B.4.8. Spesifikasi proses/algoritma

**B.4.8.1. <F1 >** : Menu Tambah Ciri Kerusakan

**Objek terkait** : Data Ciri Kerusakan

**Event** : Klik

<b>Initial State (IS) :</b> TextField Kosong
<b>Final State (FS) :</b> Kembali ke Menu Tambah Macam Kerusakan
<b>Spesifikasi Proses/algoritma:</b> Koneksi open; Input Ciri Kerusakan; IF(txtNewPakarCK =" "){ Tampil messageBox(TextField Tidak Boleh Kosong); }else{ INSERT INTO tbCiriKerusakan (NoCiri,NoJenis, NoMacam, 'CiriKerusakan','Solusi') values (txtNoJK, txtNoMK,' txtNewPakarCK','txtSolusi' ); } Koneksi close;

#### B.4.9. Spesifikasi Report

Tidak ada

#### B.5. Spesifikasi Fungsi / Proses FSelectMacamKerusakan

Identifikasi / Nama : Select Macam Kerusakan

Deskripsi Isi : Macam Kerusakan

Jenis : Form Entry columnar

##### B.5.1. Spesifikasi Tabel Input

Nama Tabel : tbMacamKerusakan

##### B.5.2. Spesifikasi Tabel Output

Nama Tabel : tbMacamKerusakan

### B.5.3. Spesifikasi Layar Select Macam Kerusakan

### B.5.4. Spesifikasi Query

Query1 = SELECT MacamKerusakan FROM tbMacamKerusakan WHERE NoMacam="IstMacamMasalah";

### B.5.5. Spesifikasi Field Pada Layar

Label	Field	Tabel	I/O	Format
grpMKPenelusuran	LstPMacamMasalah	tbMacamKerusakan	I/O	Varchar(80)

### B.5.6. Spesifikasi Function Key/Objek-Objek pada layar

<b>Id Objek</b>	<b>Jenis</b>	<b>Keterangan</b>
1.	LstMacamMasalah	List macam masalah
2.	grpPMMK	Grup macam masalah
3.	btnPMRefresh	Untuk merefresh macam masalah
4.	btnPMOkMK	Untuk lanjut ke proses berikutnya

### B.5.7. Spesifikasi layar pesan

Tidak ada

### B.5.8. Spesifikasi proses/algoritma

**B.5.8.1. <F1 >** : Menu Pilih Macam Kerusakan

**Objek terkait** : Data Macam Kerusakan

**Event** : Klik

<b>Initial State (IS) :</b> Macam kerusakan belum dipilih
<b>Final State (FS) :</b> Tampil Jenis kerusakan
<b>Spesifikasi Proses/algoritma:</b> Koneksi open; Input Macam Kerusakan; IF (LstMacamMasalah = “”) { Tampil messageBox(Anda Belum Memilih Macam Masalah); }else{ select MacamKerusakan from tbMacamKerusakan where NoMacam= LstMacamMasalah; } Koneksi close;

### B.5.9. Spesifikasi Report

Tidak ada

## B.6. Spesifikasi Fungsi / Proses FSelectJenisKerusakan

Identifikasi / Nama : Select Jenis Kerusakan

Deskripsi Isi : Jenis Kerusakan

Jenis : Form Entry columnar

### B.6.1. Spesifikasi Tabel Input

Nama Tabel : tbJenisKerusakan

### B.6.2. Spesifikasi Tabel Output

Nama Tabel : tbJenisKerusakan

### B.6.3. Spesifikasi Layar Select Jenis Kerusakan

The screenshot shows a software application window titled "Penelusuran" with a blue header. The main content area is divided into several sections:

- Left Panel:** Contains the title "PENELUSURAN MASALAH" in large blue letters. Below it is a section "Pilih Macam Kerusakan" with a list of items: MK03 Kerusakan bagian sistem suspensi, MK04 Kerusakan bagian roda, MK05 Kerusakan bagian rem, MK06 Kerusakan bagian lampu (highlighted in blue), MK07 Kerusakan bagian instrumen, and MK08 Kerusakan klakson. There are "Refresh" and "Ok" buttons below the list.
- Top Right Panel:** Titled "Pilih Jenis Kerusakan", it contains a list of items: JK026 Lampu utama yang rusak, JK027 Lampu rem tidak dapat menyala, JK028 Sekering mengarah putus tems, and JK029 Sinyal belok yang rusak (highlighted with a red oval). Below this list is a "Keterangan" text area and an "Ok" button.
- Bottom Left Panel:** Contains a cartoon illustration of two people sitting at computers.
- Bottom Right Panel:** Titled "Pilih Ciri Kerusakan", it is currently empty and has a "SOLUSI" button below it.

### B.6.4. Spesifikasi Query

```
Query1 = SELECT NoJenis, JenisKerusakan FROM tbJenisKerusakan WHERE NoMacam= IstMacamMasalah;
```

### B.6.5. Spesifikasi Field Pada Layar

Label	Field	Tabel	I/O	Format
grpJKPenelusuran	LstPJenisMasalah	tbJenisKerusakan	I/O	Varchar(80)

### B.6.6. Spesifikasi Function Key/Objek-Objek pada layar

<b>Id Objek</b>	<b>Jenis</b>	<b>Keterangan</b>
1.	LstJenisMasalah	List jenis masalah
2.	grpPMJK	Grup jenis masalah
3.	btnPMOkJK	Untuk lanjut ke proses berikutnya

### B.6.7. Spesifikasi layar pesan

Tidak ada

### B.6.8. Spesifikasi proses/algoritma

**B.6.8.1. <F1 >** : Menu Pilih Jenis Kerusakan

**Objek terkait** : Data Jenis Kerusakan

**Event** : Klik

<b>Initial State (IS) :</b> Jenis kerusakan belum dipilih
<b>Final State (FS) :</b> Tampil ciri kerusakan
<b>Spesifikasi Proses/algoritma:</b> Koneksi open; Input Jenis Kerusakan; IF (LstJenisMasalah = “”) { Tampil messageBox(Anda Belum Memilih Jenis Masalah); }else{  SELECT NoJenis, JenisKerusakan FROM tbJenisKerusakan WHERE NoMacam= lstMacamMasalah;  } Koneksi close;

### B.6.9. Spesifikasi Report

Tidak ada

### B.7. Spesifikasi Fungsi / Proses FSelectCiriKerusakan

Identifikasi / Nama : Select Ciri Kerusakan

Deskripsi Isi : Ciri Kerusakan

Jenis : Form Entry columnar

### B.7.1. Spesifikasi Tabel Input

Nama Tabel : tbCiriKerusakan

### B.7.2. Spesifikasi Tabel Output

Nama Tabel : tbCiriKerusakan

### B.7.3. Spesifikasi Layar Select Ciri Kerusakan

**Penelusuran**

**PENELUSURAN MASALAH**

Pilih Macam Kerusakan

- MK03 Kerusakan bagian sistem suspensi
- MK04 Kerusakan bagian roda
- MK05 Kerusakan bagian rem
- MK06 Kerusakan bagian lampu**
- MK07 Kerusakan bagian instrumen
- MK08 Kerusakan klakson

Refresh OK

Pilih Jenis Kerusakan

- JK026 Lampu utama yang rusak
- JK027 Lampu rem tidak dapat menyala
- JK028 Sekering mengarah putus terus
- JK029 Sinyal belok yang rusak**

Keterangan

-

OK

Pilih Ciri Kerusakan

- CK060 Bulb lampu yang putus**
- CK061 Bulb lampu yang salah
- CK062 Kontak yang buruk antara "bulb" dan "socket"
- CK063 Kabel aliran yang longgar
- CK064 "Relay" sinyal belok yang rusak

SOLUSI

### B.7.4. Spesifikasi Query

```
Query1 = SELECT NoCiri, CiriKerusakan FROM tbCiriKerusakan WHERE  
NoJenis= IstJenisMasalah ;
```

### B.7.5. Spesifikasi Field Pada Layar

Label	Field	Tabel	I/O	Format
grpCKPenelusuran	LstPCiriMasalah	tbCiriKerusakan	I/O	Varchar(80)

### B.7.6. Spesifikasi Function Key/Objek-Objek pada layar

Id Objek	Jenis	Keterangan
1.	LstCiriMasalah	List ciri masalah
2.	grpPMCK	Grup ciri masalah
3.	btnPMOkCK	Untuk lanjut ke proses selanjutnya

### B.7.7. Spesifikasi layar pesan

Tidak ada

### B.7.8. Spesifikasi proses/algoritma

**B.7.8.1. <F1 >** : Menu Pilih Ciri Kerusakan

**Objek terkait** : Data Ciri Kerusakan

**Event** : Klik

<b>Initial State (IS):</b> Ciri kerusakan belum dipilih
<b>Final State (FS):</b> Tampil menu report
<b>Spesifikasi Proses/algoritma:</b> Koneksi open; Input Ciri Kerusakan; IF (LstCiriMasalah = "") { Tampil messageBox(Anda Belum Memilih Ciri Masalah); }else{  Tampil report; } Koneksi close;

### B.7.9. Spesifikasi Report

Tidak ada

## B.8. Spesifikasi Fungsi / Proses FCariSolusi

Identifikasi / Nama : Cari Solusi  
Deskripsi Isi : Solusi  
Jenis : Master/Detail

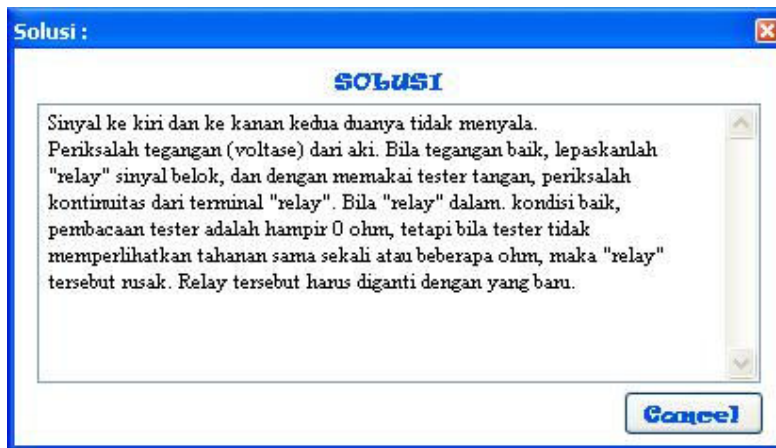
### B.8.1. Spesifikasi Tabel Input

Tidak ada.

### B.8.2. Spesifikasi Tabel Output

Nama Tabel : tbCiriKerusakan

### B.8.3. Spesifikasi Layar Select Ciri Kerusakan



### B.8.4. Spesifikasi Query

```
Query1 = SELECT Solusi FROM tbCiriKerusakan WHERE NoMacam= txtNomacam  
AND  
NoJenis= txtNoJenis AND NoCiri= txtNoCiri;
```

### B.8.5. Spesifikasi Field Pada Layar

Label	Field	Tabel	I/O	Format
lblSsolusi	txtSsolusi	tbJenisKerusakan	O	Varchar(1000)

### B.8.6. Spesifikasi Function Key/Objek-Objek pada layar

<b>Id Objek</b>	<b>Jenis</b>	<b>Keterangan</b>
1.	txtSolusi	Solusi masalah
2.	btnCancel	Untuk kembali ke penelusuran awal

### B.8.7. Spesifikasi layar pesan

<b>Kasus</b>	<b>Pesan</b>
Jika Solusi tidak ditemukan	Solusi tidak di temukan

### B.8.8. Spesifikasi proses/algorithm

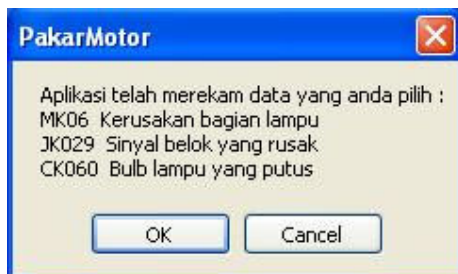
**B.8.8.1. <F1 >** : Menu Tampil Solusi

**Objek terkait** : Data Ciri Kerusakan

**Event** : Display

<b>Initial State (IS) :</b> Ciri kerusakan belum dipilih
<b>Final State (FS) :</b> Tampil menu report
<b>Spesifikasi Proses/algorithm:</b> Koneksi open; Input Ciri Kerusakan; IF (LstCiriMasalah = “”) { Tampil messageBox(Anda Belum Memilih Ciri Masalah); }else{  select Solusi from tbCiriKerusakan where NoMacam= lblMacamMasalah and NoJenis= lbJenisMasalah and NoCiri= lblCiriMasalah; } Koneksi close;

### B.8.9. Spesifikasi Report



## LAMPIRAN C DAFTAR RINCI FILE DAN DATA

### C.1. Struktur direktori

#### C.1.2. Direktori pengembangan

Direktori yang digunakan pada pengembangan aplikasi ini:

- PakarMotor, berisi data base, source code dan aplikasi untuk komputer server & client.

#### C.1.3. Direktori operasional

Direktori yang digunakan pada pengembangan aplikasi ini:

- PakarMotor, berisi data base, source code dan aplikasi untuk komputer server & client.

### C.2. Isi Direktori PakarMotor

#### C.2.1 Directory of Directory of C:\Documents and Settings\doni doank\Desktop\PakarMotor

```
06/18/2010 11:23 PM <DIR>      .
06/18/2010 11:23 PM <DIR>      ..
06/18/2010 11:23 PM <DIR>      PakarMotor
02/22/2010 10:24 PM          919 PakarMotor.sln
          1 File(s)      919 bytes
          3 Dir(s) 30,482,874,368 bytes free
```

#### C.2.1.1 Directory of C:\Documents and Settings\doni doank\Desktop\PakarMotor\PakarMotor

```
06/18/2010 11:23 PM <DIR>      .
06/18/2010 11:23 PM <DIR>      ..
06/18/2010 11:23 PM <DIR>      bin
06/17/2010 04:23 PM          6,961 formAbout.Designer.vb
06/17/2010 04:23 PM          33,801 formAbout.resx
06/17/2010 04:23 PM          178 formAbout.vb
06/17/2010 04:51 PM          5,945 formGantipass.Designer.vb
06/17/2010 04:51 PM          27,903 formGantipass.resx
06/17/2010 04:51 PM          1,494 formGantipass.vb
06/16/2010 10:42 AM          26,747 formInputPakar.Designer.vb
06/16/2010 10:42 AM          111,685 formInputPakar.resx
```

06/17/2010 04:18 PM 13,958 formInputPakar.vb  
 06/17/2010 04:51 PM 4,513 formLogin.Designer.vb  
 06/17/2010 04:51 PM 27,903 formLogin.resx  
 06/17/2010 04:53 PM 4,512 formLogin.vb  
 06/17/2010 04:32 PM 10,869 formMenuUtama.Designer.vb  
 06/17/2010 04:32 PM 338,982 formMenuUtama.resx  
 06/18/2010 07:42 PM 3,140 formMenuUtama.vb  
 06/18/2010 07:41 PM 3,102 formNewPakarJK.Designer.vb  
 06/18/2010 07:41 PM 5,814 formNewPakarJK.resx  
 06/18/2010 07:41 PM 1,561 formNewPakarJK.vb  
 06/15/2010 12:20 AM 3,102 formNewPakarMK.Designer.vb  
 06/15/2010 12:20 AM 5,814 formNewPakarMK.resx  
 06/17/2010 04:18 PM 1,062 formNewPakarMK.vb  
 06/16/2010 11:22 AM 11,116 formPakarBaru.Designer.vb  
 06/16/2010 11:22 AM 5,814 formPakarBaru.resx  
 06/17/2010 04:18 PM 3,261 formPakarBaru.vb  
 06/17/2010 04:00 PM 3,648 formPemeliharaan.Designer.vb  
 06/17/2010 04:00 PM 5,814 formPemeliharaan.resx  
 06/17/2010 04:18 PM 1,051 formPemeliharaan.vb  
 06/18/2010 06:12 PM 14,816 formPenelusuran.Designer.vb  
 06/18/2010 06:12 PM 23,170 formPenelusuran.resx  
 06/18/2010 06:12 PM 4,671 formPenelusuran.vb  
 06/17/2010 04:39 PM 3,651 formSolusi.Designer.vb  
 06/17/2010 04:39 PM 5,814 formSolusi.resx  
 06/17/2010 04:39 PM 188 formSolusi.vb  
 06/17/2010 04:18 PM 8,849 ModUmum.vb  
 06/18/2010 11:23 PM <DIR> My Project  
 06/18/2010 11:23 PM <DIR> obj  
 06/15/2010 01:49 AM 8,268 PakarMotor.vbproj  
 35 File(s) 739,177 bytes  
 5 Dir(s) 30,482,931,712 bytes free

**C.2.1.1.1 Directory of C:\Documents and Settings\doni doank  
 \Desktop\PakarMotor\PakarMotor\My Project**

06/18/2010 11:23 PM <DIR> .  
 06/18/2010 11:23 PM <DIR> ..  
 06/15/2010 01:18 AM 1,518 Application.Designer.vb  
 06/15/2010 01:18 AM 475 Application.myapp  
 02/22/2010 10:24 PM 1,173 AssemblyInfo.vb  
 06/14/2010 11:51 PM 2,776 Resources.Designer.vb  
 09/23/2005 03:27 AM 5,612 Resources.resx  
 06/14/2010 11:51 PM 2,977 Settings.Designer.vb  
 09/23/2005 03:27 AM 279 Settings.settings  
 7 File(s) 14,810 bytes  
 2 Dir(s) 30,482,857,984 bytes free

**C.2.1.1.2 Directory of C:\Documents and Settings\doni doank\Desktop  
\PakarMotor\PakarMotor\obj**

06/18/2010 11:23 PM <DIR> .  
06/18/2010 11:23 PM <DIR> ..  
06/18/2010 11:23 PM <DIR> Debug  
06/18/2010 07:55 PM 785 PakarMotor.vbproj.FileList.txt  
1 File(s) 785 bytes  
3 Dir(s) 30,482,857,984 bytes free

**C.2.1.1.2.1 Directory of C:\Documents and Settings\doni doank  
\Desktop\PakarMotor\PakarMotor\obj\Debug**

06/18/2010 11:23 PM <DIR> .  
06/18/2010 11:23 PM <DIR> ..  
06/18/2010 07:42 PM 462,848 PakarMotor.exe  
06/17/2010 04:23 PM 18,957 PakarMotor.frmAbout.resources  
06/17/2010 04:51 PM 15,021 PakarMotor.frmGantipass.resources  
06/16/2010 10:42 AM 70,980 PakarMotor.frmInputPakar.resources  
06/17/2010 04:51 PM 15,021 PakarMotor.frmLogin.resources  
06/17/2010 04:32 PM 221,872 PakarMotor.frmMenuUtama.resources  
06/18/2010 07:41 PM 180 PakarMotor.frmNewPakarJK.resources  
06/15/2010 12:20 AM 180 PakarMotor.frmNewPakarMK.resources  
06/18/2010 06:12 PM 11,854 PakarMotor.frmPakarMesin.resources  
06/17/2010 04:00 PM 180 PakarMotor.frmPemeliharaan.resources  
06/17/2010 04:39 PM 180 PakarMotor.frmSoLusi.resources  
06/16/2010 11:22 AM 180 PakarMotor.frmTambahPakar.resources  
06/18/2010 07:42 PM 210,432 PakarMotor.pdb  
02/22/2010 10:31 PM 180 PakarMotor.Resources.resources  
06/18/2010 07:41 PM 2,094 PakarMotor.vbproj.GenerateResource.Cache  
06/18/2010 07:42 PM 672 PakarMotor.xml  
06/18/2010 11:23 PM <DIR> TempPE  
16 File(s) 1,030,831 bytes  
3 Dir(s) 30,482,857,984 bytes free

**C.2.1.1.2.2 Directory of C:\Documents and Settings\doni doank\Desktop  
\PakarMotor\PakarMotor\obj\Debug\TempPE**

06/18/2010 11:23 PM <DIR> .  
06/18/2010 11:23 PM <DIR> ..  
06/14/2010 11:51 PM 7,680 My Project.Resources.Designer.vb.dll  
1 File(s) 7,680 bytes  
2 Dir(s) 30,482,857,984 bytes free

**C.2.1.1.2.3 Directory of C:\Documents and Settings\doni doank\Desktop  
PakarMotor\PakarMotor\bin\Debug**

```
06/18/2010 11:23 PM <DIR>      .
06/18/2010 11:23 PM <DIR>      ..
06/18/2010 07:38 PM      1,355,776 dPakar.mdb
06/18/2010 07:42 PM      462,848 PakarMotor.exe
06/18/2010 07:42 PM      210,432 PakarMotor.pdb
09/23/2005 06:56 AM        5,632 PakarMotor.vshost.exe
06/18/2010 07:42 PM         672 PakarMotor.xml
06/18/2010 06:03 PM         4,837 PEMELIHARAAN.RTF
    6 File(s)    2,040,197 bytes
    2 Dir(s) 30,482,857,984 bytes free
```

## LAMPIRAN D DOKUMEN RINCI TESTING

### D.1. Tim penguji

1. Afdhol Dziki, SST

2.

### D.2. Hasil Rinci Pengujian

No	Nomor Fungsi	Deskripsi Fungsional	Kelompok Uji	Prosedur dan Kasus Uji	Hasil yang diharapkan	Hasil Test	Tester	Tanggal Testing	Keterangan
1	F1 (P1)	Login admin	Normal	Login sebagai admin	Masuk ke menu admin	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
			Data salah	Login sebagai admin	Tampil pesan kesalahan "".	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
2	F2 (P2.1)	Insert macam kerusakan	Normal	Mengisi data macam kerusakan	Data macam kerusakan bertambah	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
			Data salah	Mengisi data macam kerusakan	Tampil pesan kesalahan "".	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-

3	F3 (P2.2)	Insert jenis kerusakan	Normal	Mengisi data jenis kerusakan	Data jenis kerusakan bertambah	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
			Data salah	Mengisi data jenis kerusakan	Tampil pesan kesalahan ""	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
4	F4 (P3.1)	Insert ciri kerusakan	Normal	Mengisi data ciri kerusakan	Data ciri kerusakan bertambah	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
			Data Salah	Mengisi data ciri kerusakan	Tampil pesan kesalahan ""	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
5	F5 (P3.2)	Select macam kerusakan	Normal	Melihat data macam kerusakan	Tampil data macam kerusakan	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
			Data salah	Melihat data macam kerusakan	Tampil pesan kesalahan ""	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
6	F6 (P3.3)	Select jenis kerusakan	Normal	Melihat data jenis kerusakan	Tampil data jenis kerusakan	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
			Data salah	Melihat data jenis kerusakan	Tampil pesan kesalahan ""	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
7	F7 (P4)	Select ciri kerusakan	Normal	Melihat data ciri kerusakan	Tampil data ciri kerusakan	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
			Data Salah	Melihat data ciri kerusakan	Tampil pesan kesalahan ""	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
8	F8 (P5)	Cari Solusi	Normal	Melihat data solusi	Tampil data solusi	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-

			Data Salah	Melihat data solusi	Tampil pesan kesalahan ""	<b>Diterima</b>	<b>AZ</b>	20 Juni 2010	-
--	--	--	------------	---------------------	---------------------------	-----------------	-----------	--------------	---