# BAB 4

# IMPLEMENTASI DAN PENGUJIAN

## 4.1 Implementasi

Implementasi adalah hasil rancangan yang menjadi sebuah program aplikasi yang dapat dioperasikan dan mencapai hasil yang sesuai dengan rancangan. Setelah melakukan tahapan analisis dan perancangan maka tindakan selanjutnya adalah pencapaian hasil perangkat lunak yang dibuat.

## 4.2 Proses Dialog

Penggunaan program aplikasi ini tergolong sangat mudah, hanya dengan menjawab pertanyaan mengenai gejala kerusakan Yamaha P 900 yang diajukan oleh program, jawabannya berupa ‘y’ untuk ya atau ‘t’ untuk tidak, maka sistem akan mendiagnosis kerusakan. Kemudian sistem akan menampilkan kerusakan dan solusi. Jika gejala-gejala yang dimasukan pengguna tidak ada yang cocok dengan aturan suatu jenis kerusakan, maka akan muncul pesan bahwa kerusakan tidak dapat didiagnosis.

## 4.3 Kebutuhan Perangkat Keras dan Perangkat Lunak

Untuk menentukan layak tidaknya suatu program atau aplikasi yang akan digunakan oleh orang lain, maka dilakukan pengujian sekiranya dapat dijadikan acuan dalam program ini.

Spesifikasi minimal untuk menjalankan program ini adalah sebagai berikut:

Perangkat keras, yaitu:

1. Prosessor : Intel Pentium 4 ~1,86 GHz
2. Hardisk space : 10 Mb
3. Memori : 128 MB
4. VGA : 32 MB
5. Monitor : 800 x 600 pixel
6. Mouse : Standar PS/2
7. Keyboard : Standar PS/2

Perangkat lunak, yaitu:

1. Sistem Operasi Windows XP1
2. Turbo Prolog 2.0
3. Delphi 7
4. WinMysqlAdmin 1.4

### 4.4 Implementasi Struktur Data

Pembuatan struktur data dengan menggunakan database MySQL. Implementasi struktur data dalam bahasa SQL adalah sebagai berikut:

**Tabel Gejala**

```
CREATE TABLE `gejala` (
  `kode_gejala` varchar(6) NOT NULL,
  `nama_gejala` varchar(255) NOT NULL,
```

```
  `pertanyaan` varchar(255) NOT NULL,
  PRIMARY KEY  (`kode_gejala`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**Tabel Diagnosis**

```
CREATE TABLE `diagnosis` (
  `kode_diagnosis` varchar(6) NOT NULL,
  `nama_diagnosis` varchar(255) NOT NULL,
  PRIMARY KEY  (`kode_diagnosis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**Tabel Solusi**

```
CREATE TABLE `solusi` (
  `kode_solusi` varchar(6) NOT NULL,
  `nama_solusi` varchar(255) NOT NULL,
  PRIMARY KEY  (`kode_solusi`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

**Tabel Urut Diagnosis - Gejala**

```
CREATE TABLE `urut_diagnosis_gejala` (
  `id_udg` int(3) NOT NULL auto_increment,
  `kode_diagnosis` varchar(6) NOT NULL,
  `kode_gejala` varchar(6) NOT NULL,
  PRIMARY KEY  (`id_udg`),
  KEY `kode_diagnosis` (`kode_diagnosis`),
  KEY `kode_gejala` (`kode_gejala`),
  CONSTRAINT  `urut_diagnosis_gejala_ibfk_2`  FOREIGN  KEY
(`kode_gejala`) REFERENCES `gejala` (`kode_gejala`),
  CONSTRAINT  `urut_diagnosis_gejala_ibfk_1`  FOREIGN  KEY
(`kode_diagnosis`) REFERENCES `diagnosis` (`kode_diagnosis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

**Tabel Urut Diagnosis - Solusi**

```
CREATE TABLE `urut_diagnosis_solusi` (
  `id_uds` int(3) NOT NULL auto_increment,
  `kode_diagnosis` varchar(6) NOT NULL,
  `kode_solusi` varchar(6) NOT NULL,
  PRIMARY KEY  (`id_uds`),
  KEY `kode_diagnosis` (`kode_diagnosis`),
  KEY `kode_solusi` (`kode_solusi`),
  CONSTRAINT  `urut_diagnosis_solusi_ibfk_2`  FOREIGN  KEY
(`kode_solusi`) REFERENCES `solusi` (`kode_solusi`),
  CONSTRAINT  `urut_diagnosis_solusi_ibfk_1`  FOREIGN  KEY
(`kode_diagnosis`) REFERENCES `diagnosis` (`kode_diagnosis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;
```

### 4.5 Implementasi Antarmuka

Implementasi antarmuka dilakukan dengan setiap halaman program yang dibuat. Berikut ini adalah tampilan-tampilan pada aplikasi yang dibuat:

1. Tampilan Awal

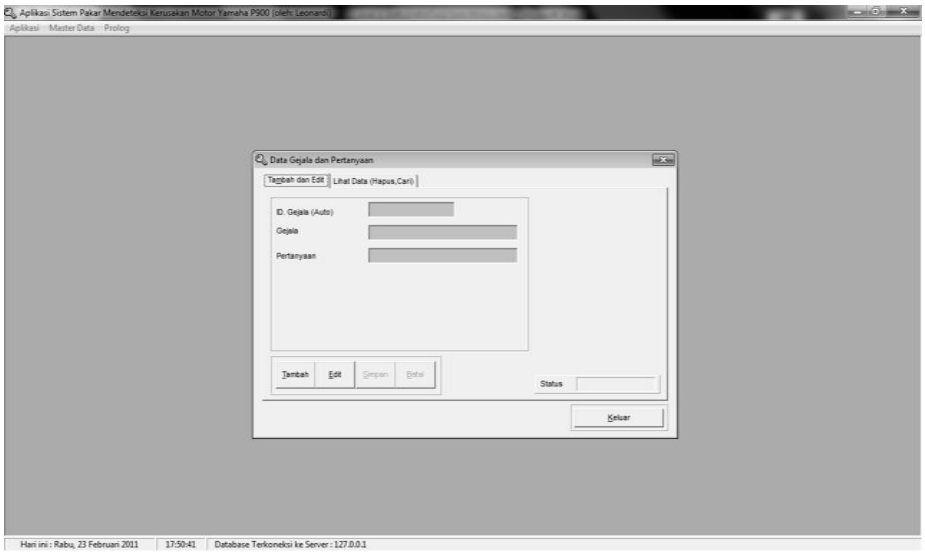
   Tampilan awal dari aplikasi dengan tiga akses menu yaitu Aplikasi, Master Data dan Prolog. Di mana prolog adalah menu untuk mengakses aplikasi yang dibangun melalui turbo prolog 2.0.



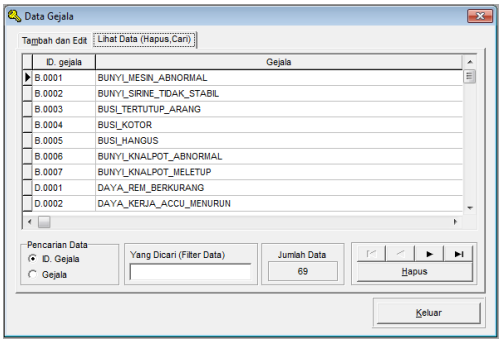

Gambar 4.1 Tampilan Awal

2. Tampilan Update Data Gejala

Gejala dan pertanyaan dapat diinputkan, dengan maksud melakukan pembaharuan rule-based pada query prolog.



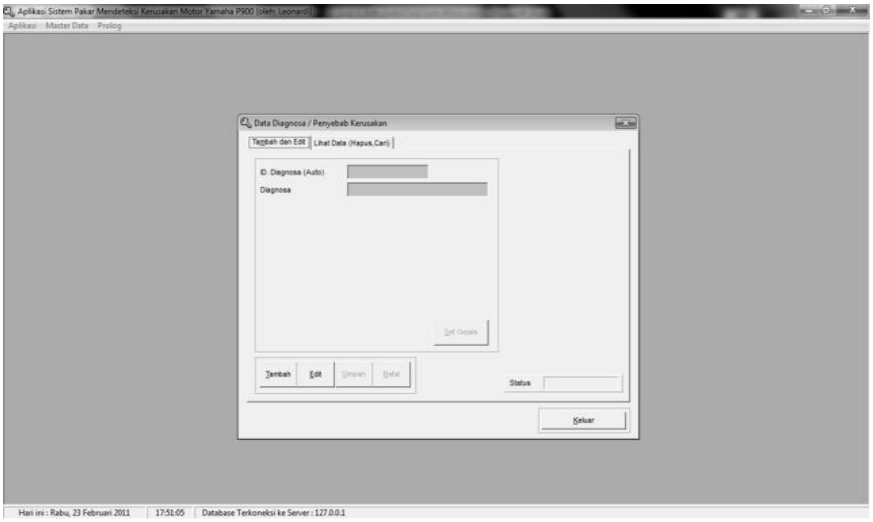

Gambar 4.2 Tampilan Update Data Gejala



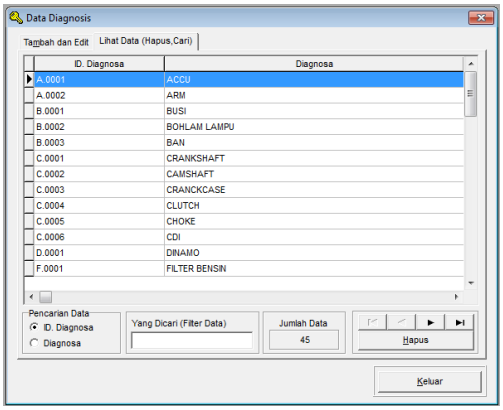

Gambar 4.3 Tampilan Lihat Data Gejala

3. Tampilan Update Data Diagnosis

Diagnosis kerusakan dapat diinputkan dengan maksud melakukan pembaharuan rule-based diagnosis kerusakan pada query prolog.



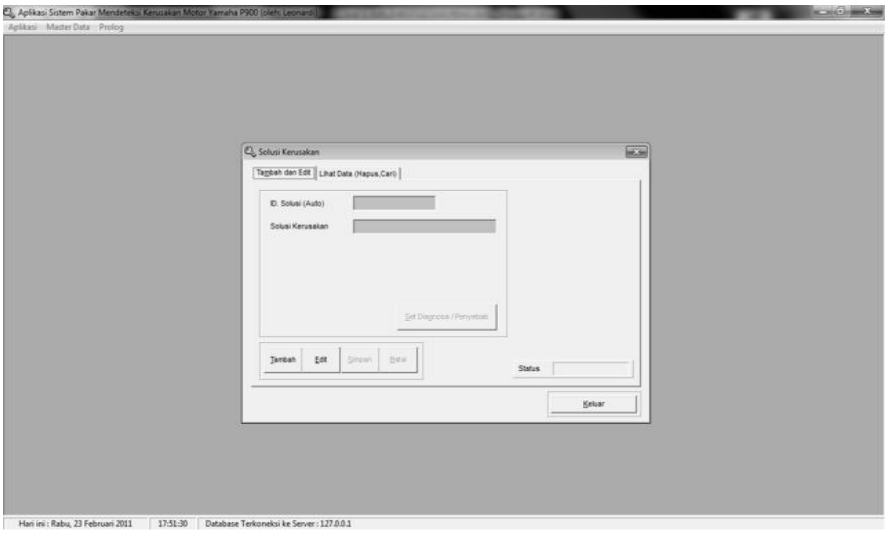

Gambar 4.4 Tampilan Update Data Diagnosis



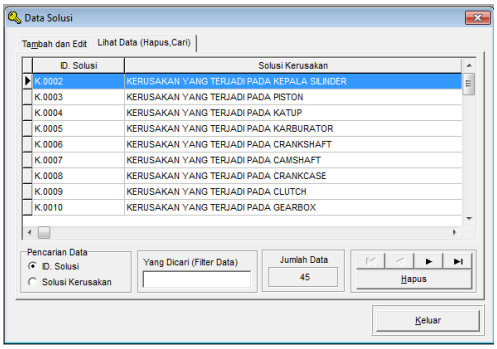

Gambar 4.5 Tampilan Lihat Data Diagnosis

4. Tampilan Update Data Solusi

Gejala dan kerusakan yang sudah diinputkan, kemudian ditetapkan menjadi solusi pada menu update data solusi.

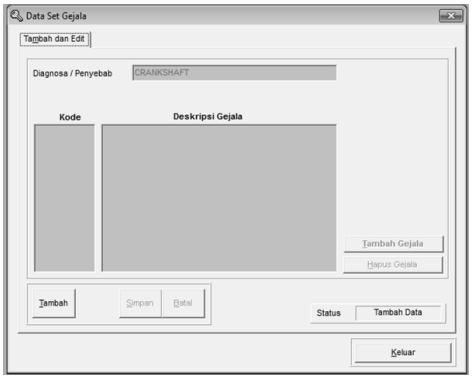



Gambar 4.6 Tampilan Update Data Solusi

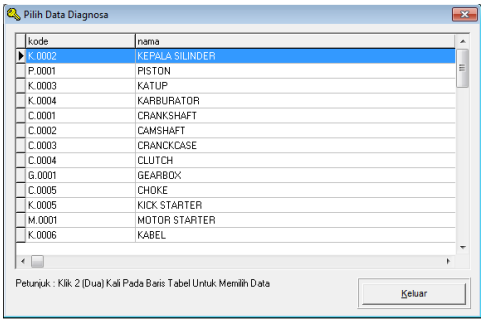



Gambar 4.7 Tampilan Lihat Data Solusi

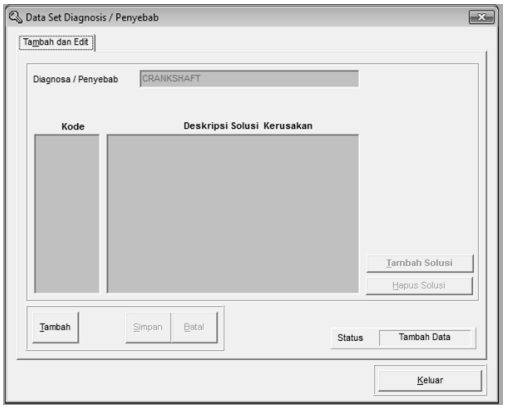
5. Tampilan Relasi Data



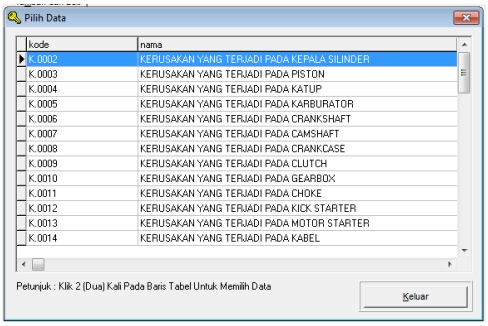

Gambar 4.8 Tampilan Set Diagnosis Tambah Relasi Gejala




Gambar 4.9 Tampilan Tambah Set Relasi Data Gejala



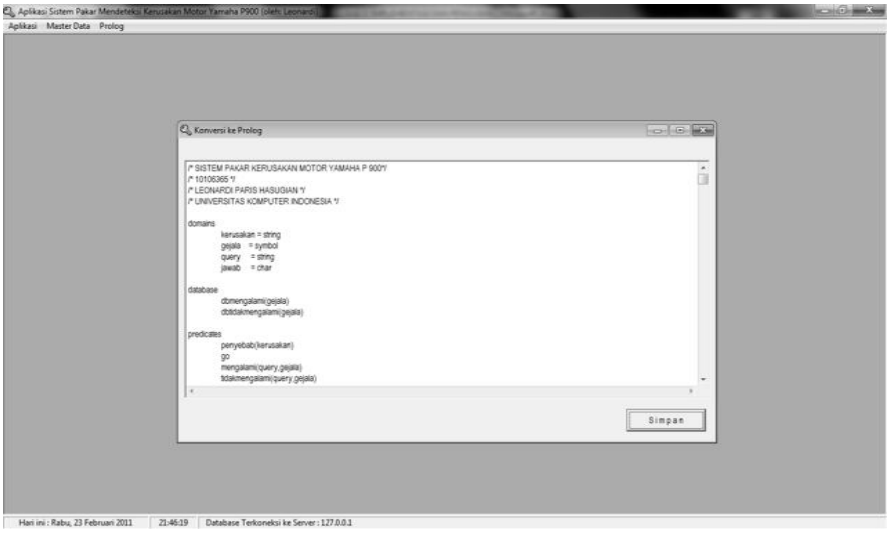

Gambar 4.10 Tampilan Set Diagnosis Tambah Relasi Solusi




Gambar 4.11 Tampilan Tambah Set Relasi Data Solusi

6. Tampilan Konversi Prolog

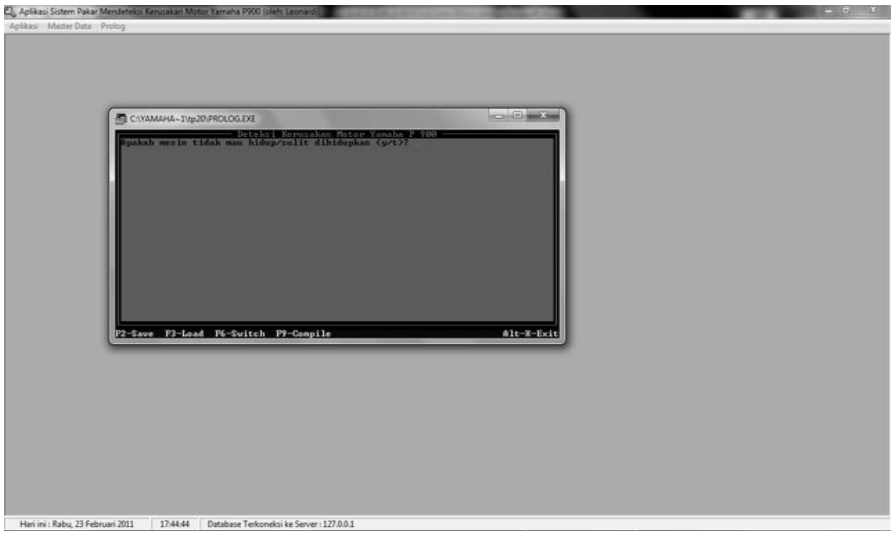
   Merupakan file .PRO yaitu file prolog yang akan dicompile oleh Turbo Prolog 2.0.




Gambar 4.12 Tampilan Konversi Prolog

7. Tampilan Eksekusi Aplikasi Turbo Prolog

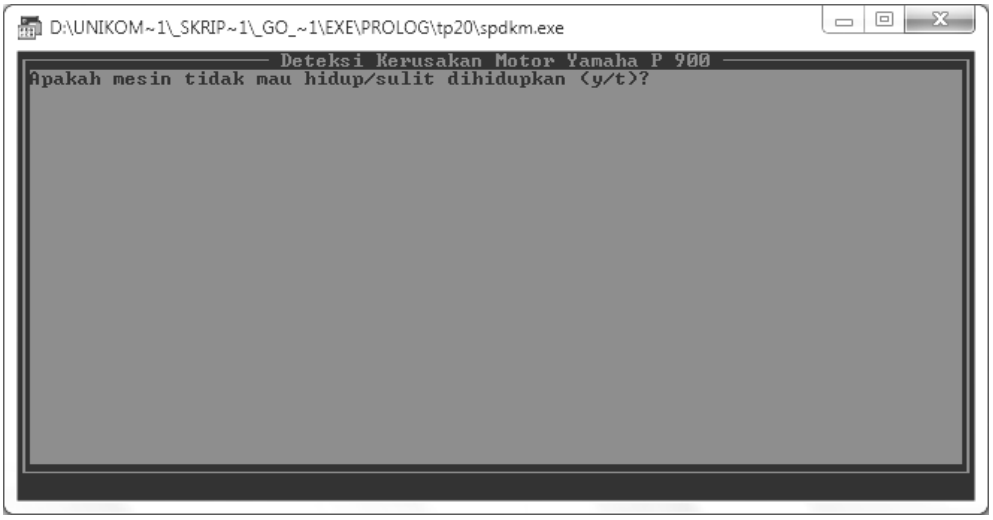
Jika mengeksekusi eksekusi prolog maka akan ditampilkan compiler turbo prolog 2.0.




Gambar 4.13 Tampilan Eksekusi Aplikasi Turbo Prolog

8. Tampilan Konsultasi Kerusakan

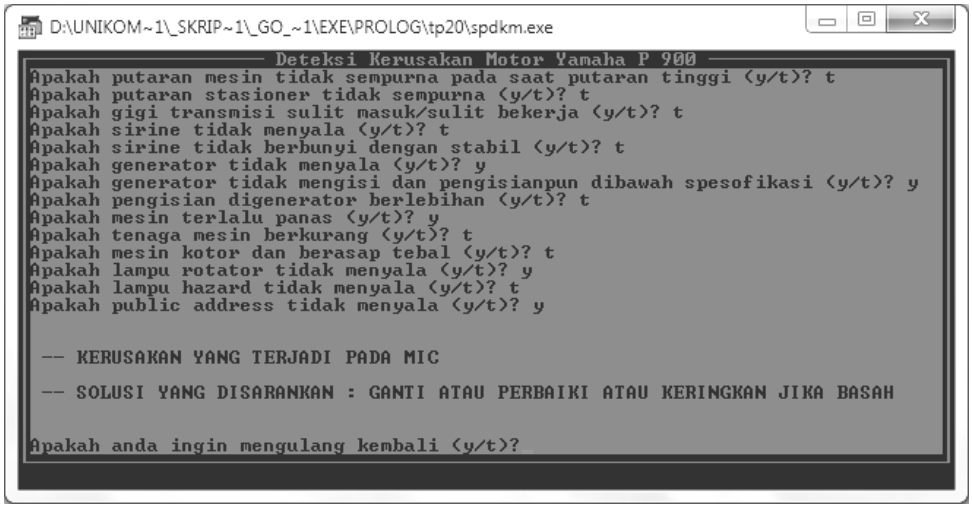
Pada gambar 4.14 dibawah ini menggunakan urutan pertanyaan yang diajukan oleh program, urutan pertanyaan didasarkan pada *ruled-based reasoning* yang terdapat pada tabel 3.2 dan gambar 3.2.




Gambar 4.14 Tampilan Konsultasi Kerusakan

9. Tampilan Kesimpulan

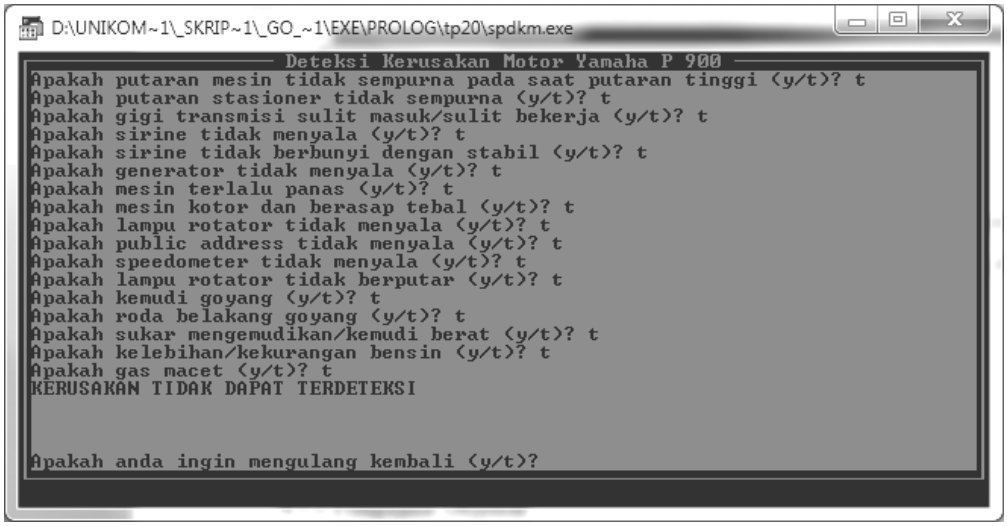
Pada gambar 4.15 dibawah ini menunjukan kesimpulan berupa jenis kerusakan dan solusi. Penarikan kesimpulan didasarkan pada jawaban dari pemakai terhadap pertanyaan yang diajukan oleh program, kemudian jawaban tersebut dianalisis oleh sistem untuk dicocokan kedalam jenis kerusakan yang memiliki gejala-gejala yang sesuai dengan data yang dimasukan oleh pemakai.




Gambar 4.15 Tampilan Kesimpulan

10. Tampilan Pesan Kegagalan

Pada gambar 4.16 dibawah ini menunjukan pesan kegagalan program dalam mendiagnosis kerusakan dikarenakan gejala-gejala yang dimasukan pemakai tidak ada yang sesuai dengan aturan kerusakan yang ada didalam struktur program.




Gambar 4.16 Tampilan Pesan Kegagalan

### 4.6 Pengujian Alpha

Pengujian perangkat lunak adalah elemen kritis dari jaminan kualitas perangkat lunak dan merepresentasikan kajian pokok dari spesifikasi, perancangan dan pengkodean. Pengujian yang digunakan untuk menguji sistem ini adalah metode pengujian *black-box*. Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak.

#### 4.6.1 Rencana Pengujian

Rencana pengujian akan dilakukan dengan menguji sistem secara *alpha* dan *beta*. Pengujian perangkat lunak aplikasi sistem pakar berikut menggunakan data uji berdasarkan data yang terdapat pada Sat Patwal Polda Metro Jaya. Rencana pengujian selengkapnya dapat dilihat pada Tabel dibawah ini:

Tabel 4.1 Rencana Pengujian Aplikasi Sistem Pakar Untuk Mendiagnosis Kerusakan Yamaha P 900

| Kelas Uji | Butir Uji | Jenis Pengujian |
|---|---|---|
| Tampilan Aplikasi | Verifikasi Background | *Black-box* |
| | Verifikasi Latar Form | *Black-box* |
| | Verifikasi Huruf Latar Form | *Black-box* |
| | | |
| Reset Data | Reset Data Gejala | *Black-box* |
| | Reset Data Kerusakan | *Black-box* |
| | Reset Data Solusi | *Black-box* |
| | Reset Relasi Gejala - Diagnosis | *Black-box* |
| | Reset Relasi Solusi - Diagnosis | *Black-box* |
| | | |
| Koneksi Database | Verifikasi IP Server | *Black-box* |
| | Verifikasi Nama User | *Black-box* |

| Kelas Uji | Butir Uji | Jenis Pengujian |
|---|---|---|
| | Verifikasi Password | *Black-box* |
| | Verifikasi Nama Database | *Black-box* |
| | | |
| Backup Database | Verifikasi Direktori Backup Database | *Black-box* |
| | Verifikasi Nama Database | *Black-box* |
| | | |
| Restore Database | Verifikasi Direktori Sumber Database | *Black-box* |
| | Verifikasi Nama Database | *Black-box* |
| | | |
| Tambah Gejala | Simpan Gejala | *Black-box* |
| | Batal Simpan Gejala | *Black-box* |
| | | |
| Tambah Kerusakan | Simpan Kerusakan | *Black-box* |
| | Batal Simpan Kerusakan | *Black-box* |
| | | |
| Tambah Solusi | Simpan Solusi | *Black-box* |
| | Batal Simpan Solusi | *Black-box* |
| | | |
| Edit Gejala | Cari Kode Gejala | *Black-box* |
| | Simpan Data Yang Diedit | *Black-box* |
| | Batal Edit Data Gejala | *Black-box* |
| | | |
| Edit Kerusakan | Cari Kode Kerusakan | *Black-box* |
| | Simpan Data Yang Diedit | *Black-box* |
| | Batal Edit Data Kerusakan | *Black-box* |
| | | |
| Edit Solusi | Cari Kode Solusi | *Black-box* |
| | Simpan Data Yang Diedit | *Black-box* |
| | Batal Edit Data Solusi | *Black-box* |
| | | |
| Hapus Gejala | Cari Kode Solusi | *Black-box* |
| | Hapus Data Solusi | *Black-box* |
| | Batal Hapus Data Solusi | *Black-box* |
| | | |
| Hapus Kerusakan | Cari Kode Kerusakan | *Black-box* |
| | Hapus Data Kerusakan | *Black-box* |

| Kelas Uji | Butir Uji | Jenis Pengujian |
|---|---|---|
| | Batal Hapus Data Kerusakan | *Black-box* |
| | | |
| Hapus Solusi | Cari Kode Solusi | *Black-box* |
| | Hapus Data Solusi | *Black-box* |
| | Batal Hapus Data Solusi | *Black-box* |
| | | |
| Cari Gejala | Cari Data Gejala | *Black-box* |
| | | |
| Cari Kerusakan | Cari Data Kerusakan | *Black-box* |
| | | |
| Cari Solusi | Cari Data Solusi | *Black-box* |

### 4.6.2 Kasus dan Hasil Pengujian

Pengujian yang dilakukan pada implementasi aplikasi diuji untuk mengetahui hasil pengamatan dari data yang dimasukan ke dalam aplikasi.

#### 4.6.2.1 Pengujian Konfigurasi Aplikasi

1. Pengujian Tampilan Aplikasi

Tabel 4.2 Pengujian Tampilan Aplikasi

| **Kasus dan Hasil Uji (Data Normal)** | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Gambar Latar Aplikasi | Gambar Latar Aplikasi berubah | Latar Gambar berubah sesuai inputan | Diterima |
| Gambar Latar Form | Gambar Latar Form berubah | Latar Form berubah sesuai inputan | Diterima |
| Warna Huruf Latar Form | Warna Huruf Latar Form berubah (hitam atau putih) | Warna Huruf latar form berubah sesuai inputan | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |

| | | | |
|---|---|---|---|
| Gambar Latar Aplikasi (salah) | Tidak dapat melakukan Update Gambar Latar Aplikasi | Kegagalan Update Tampilan Aplikasi | Diterima |
| Gambar Latar Form (salah) | Tidak dapat melakukan Update Latar Form | Kegagalan Update Tampilan Aplikasi | Diterima |

2. Pengujian Reset Data

Tabel 4.3 Pengujian Reset Data

| **Kasus dan Hasil Uji (Data Normal)** | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Pilih Check Box Reset Data | Dapat melakukan Reset data | Hasil reset berhasil | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| Checkbox Reset Data (tidak dipilih) | Kegagalan reset data | Tidak dapat mereset data | Diterima |

3. Pengujian Seting Database

Tabel 4.4 Pengujian Seting Database

| **Kasus dan Hasil Uji (Data Normal)** | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Seting database, Nama/IP server | Dapat melakukan Setting Database | Setting Database berhasil | Diterima |
| Seting database, Nama user | Dapat melakukan Setting Database | Setting Database berhasil | Diterima |
| Seting database, Kata sandi | Dapat melakukan Setting Database | Setting Database berhasil | Diterima |
| Seting database, Nama database | Dapat melakukan Setting Database | Setting Database berhasil | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |

| | | | |
|---|---|---|---|
| Seting database, Nama/IP server (kosong) | Kegagalan koneksi kedalam database | Database Gagal Terkoneksi | Diterima |
| Seting database, Nama user (kosong) | Kegagalan koneksi kedalam database | Database Gagal Terkoneksi | Diterima |
| Seting database, Kata sandi (salah/kosong) | Kegagalan koneksi kedalam database | Database Gagal Terkoneksi | Diterima |
| Seting database, Nama database (kosong) | Kegagalan koneksi kedalam database | Database Gagal Terkoneksi | Diterima |

4. Pengujian Backup Database

Tabel 4.5 Pengujian Backup Database

| **Kasus dan Hasil Uji (Data Normal)** | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Backup database, direktori backup database | Dapat melakukan Backup Database | Backup Database berhasil | Diterima |
| Backup database, nama database | Dapat melakukan Backup Database | Backup Database berhasil | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| Backup database, direktori backup database (kosong) | Kegagalan Backup Database | Muncul pesan Direktori Backup Database tidak boleh kosong | Diterima |
| Backup database, nama database (kosong) | Kegagalan Backup Database | Backup Database Gagal | Diterima |

5. Pengujian Restore Database

Tabel 4.6 Pengujian Restore Database

| Kasus dan Hasil Uji (Data Normal) | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Restore database, direktori sumber database | Dapat melakukan Restore Database | Restore Database berhasil | Diterima |
| Restore database, nama database | Dapat melakukan Restore Database | Restore Database berhasil | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| Restore database, direktori sumber database (kosong) | Kegagalan Restore Database | Muncul pesan Direktori Sumber Database tidak boleh kosong | Diterima |

### 4.6.2.2 Pengujian Master Data

1. Pengujian Data Gejala

4.7 Pengujian Data Gejala

| Kasus dan Hasil Uji (Data Normal) | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Gejala | Dapat menyimpan data gejala | Data gejala tersimpan | Diterima |
| Pertanyaan | Dapat menyimpan pertanyaan | Pertanyaan tersimpan | Diterima |
| Lihat data | Dapat memfilter dan melihat data gejala | Data gejala dapat difilter dan dilihat | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| Gejala (kosong) | Data Gejala tidak tersimpan | Kegagalan menyimpan Data Gejala | Diterima |
| Pertanyaan (kosong) | Pertanyaan tidak tersimpan | Kegagalan menyimpan | Diterima |

| | | | |
|---|---|---|---|
| | | pertanyaan | |

2. Pengujian Data Diagnosis

Tabel 4.8 Pengujian Data Diagnosis

| **Kasus dan Hasil Uji (Data Normal)** | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Diagnosis | Dapat menyimpan data diagnosis | Data diagnosis tersimpan | Diterima |
| Lihat data | Dapat memfilter dan melihat data diagnosis | Data diagnosis dapat difilter dan dilihat | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| Diagnosis (kosong) | Data Diagnosis tidak tersimpan | Proses simpan data diagnosis gagal dilakukan | Diterima |

3. Pengujian Data Solusi

Tabel 4.9 Pengujian Data Solusi

| **Kasus dan Hasil Uji (Data Normal)** | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Solusi | Dapat menyimpan data solusi | Data solusi tersimpan | Diterima |
| Lihat data | Dapat memfilter dan melihat data solusi | Data solusi dapat difilter dan dilihat | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| Solusi (kosong) | Data Solusi tidak tersimpan | Proses simpan data solusi gagal dilakukan | Diterima |

4. Pengujian Set Relasi Data Diagnosis - Gejala

Tabel 4.10 Pengujian Set Relasi Data Diagnosis - Gejala

| Kasus dan Hasil Uji (Data Normal) | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Memilih diagnosis pada baris diagnosis | Menampilkan form set relasi data diagnosis - gejala | Form set relasi data diagnosis - gejala dapat ditampilkan | Diterima |
| Memilih gejala pada baris gejala | Dapat melakukan tambah set relasi diagnosis - gejala | Set relasi data diagnosis - gejala berhasil | Diterima |

5. Pengujian Set Relasi Data Diagnosis - Solusi

Tabel 4.11 Pengujian Set Relasi Data Diagnosis - Solusi

| Kasus dan Hasil Uji (Data Normal) | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Memilih diagnosis pada baris diagnosis | Menampilkan form set relasi data diagnosis - solusi | Form set relasi data diagnosis - solusi dapat ditampilkan | Diterima |
| Memilih solusi pada baris solusi | Dapat melakukan tambah set relasi diagnosis - solusi | Set relasi data diagnosis - solusi berhasil | Diterima |

### 4.6.2.3 Pengujian Prolog

1. Pengujian Konversi Prolog

Tabel 4.12 Pengujian Konversi Prolog

| Kasus dan Hasil Uji | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Konversi Prolog | Dapat menyimpan | File Prolog .PRO | Diterima |

| | | | |
|---|---|---|---|
| | file Prolog .PRO | dapat tersimpan | |

2. Pengujian Buka Aplikasi Turbo Prolog

Tabel 4.13 Pengujian Buka Aplikasi Turbo Prolog

| **Kasus dan Hasil Uji** | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| Buka aplikasi turbo prolog | Menampilkan compiler turbo prolog 2.0 | Compiler dapat ditampilkan | Diterima |

### 4.6.2.4 Pengujian File Prolog

1. Pengujian File Prolog

Tabel 4.14 Pengujian File Prolog

| **Kasus dan Hasil Uji (Data Normal)** | | | |
|---|---|---|---|
| **Data Masukan** | **Hasil yang diharapkan** | **Hasil Pengamatan** | **Kesimpulan** |
| ‘y’ (ya) | Kesimpulan jenis kerusakan, solusi perbaikan | Kesimpulan jenis kerusakan, solusi perbaikan | Diterima |
| ‘t’ (tidak) | Kerusakan tidak dapat didiagnosis | Kerusakan tidak dapat didiagnosis | Diterima |
| Acak (jawaban sesuai dengan aturan kerusakan) | Kesimpulan jenis kerusakan, solusi perbaikan | Kesimpulan jenis kerusakan, solusi perbaikan | Diterima |
| Acak (jawaban tidak sesuai dengan aturan kerusakan) | Kerusakan tidak dapat didiagnosis | Kerusakan tidak dapat didiagnosis | Diterima |
| ‘y’/’t’ untuk mengulang diagnosis | Kembali ke pertanyaan awal | Kembali ke pertanyaan awal | Diterima |
| **Kasus dan Hasil Uji (Data Salah)** | | | |
| Inputan Selain | Pertanyaan akan | Pertanyaan akan | Diterima |

| | | | |
|---|---|---|---|
| huruf ‘y’/’t’ | diulang secara random | diulang | |

### 4.7 Pengujian Beta

Pengujian Beta merupakan pengujian yang dilakukan secara objekif dimana diuji secara langsung ke lapangan yaitu masyarakat luas dengan membuat kuesioner untuk mengetahui pendapat responden terhadap program aplikasi sistem pakar untuk mendiagnosis kerusakan Yamaha P 900, dan selanjutnya dibagikan kepada beberapa user dengan mengambil *sample* sebanyak 5 orang. Dari hasil kuesioner tersebut akan dilakukan perhitungan untuk dapat diambil kesimpulan dari penilaian terhadap pembuatan aplikasi sistem pakar ini. Adapun hasil pengujian beta adalah sebagai berikut:

HASIL PENGUJIAN PILIHAN KATEGORI JAWABAN

1. Apakah aplikasi sistem pakar yang dibuat mudah digunakan ?
   □ Sangat mudah digunakan
   □ Mudah digunakan
   □ Cukup mudah digunakan
   □ Kurang mudah digunakan
   □ Tidak mudah digunakan

| Kategori Jawaban | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Frekuensi Jawaban | 1 | 3 | 1 | 0 | 0 |

| Jumlah Frekuensi | 1 | 3 | 1 | 0 | 0 |
|---|---|---|---|---|---|
| Jumlah Populasi Sample | 5 | 5 | 5 | 5 | 5 |
| Jumlah Presentase | 20% | 60% | 20% | 0% | 0% |

2. Apakah tampilan aplikasi sistem pakar yang dibuat menarik ?
   □ Sangat menarik
   □ Menarik
   □ Cukup menarik
   □ Kurang menarik
   □ Tidak menarik

| Kategori Jawaban | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Frekuensi Jawaban | 0 | 2 | 3 | 0 | 0 |

| Jumlah Frekuensi | 0 | 2 | 3 | 0 | 0 |
|---|---|---|---|---|---|
| Jumlah Populasi Sample | 5 | 5 | 5 | 5 | 5 |
| Jumlah Presentase | 0% | 40% | 60% | 0% | 0% |

3. Apakah aplikasi sistem pakar yang dibuat bermanfaat dalam mendiagnosis kerusakan Yamaha P 900 ?

□ Sangat bermanfaat □ Kurang bermanfaat
□ Bermanfaat □ Tidak bermanfaat
□ Cukup bermanfaat

| Kategori Jawaban | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Frekuensi Jawaban | 2 | 2 | 1 | 0 | 0 |

| Jumlah Frekuensi | 2 | 2 | 1 | 0 | 0 |
|---|---|---|---|---|---|
| Jumlah Populasi Sample | 5 | 5 | 5 | 5 | 5 |
| Jumlah Presentase | 40% | 40% | 20% | 0% | 0% |

4. Apakah aplikasi sistem pakar yang dibuat memberikan informasi tambahan selain dari teknisi ?

□ Sangat memberikan Informasi □ Kurang memberikan Informasi
□ Memberikan Informasi □ Tidak memberikan Informasi
□ Cukup memberikan Informasi

| Kategori Jawaban | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Frekuensi Jawaban | 1 | 3 | 1 | 0 | 0 |

| Jumlah Frekuensi | 1 | 3 | 1 | 0 | 0 |
|---|---|---|---|---|---|
| Jumlah Populasi Sample | 5 | 5 | 5 | 5 | 5 |
| Jumlah Presentase | 20% | 60% | 20% | 0% | 0% |

5. Apakah aplikasi sistem pakar yang dibuat dapat membantu mengefesiensikan waktu dan biaya dalam mengidentifikasi kerusakan Yamaha P 900 ?

□ Sangat membantu □ Kurang membantu
□ Membantu □ Tidak membantu
□ Cukup membantu

| Kategori Jawaban | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| Frekuensi Jawaban | 0 | 2 | 2 | 0 | 1 |

| Jumlah Frekuensi | 0 | 2 | 2 | 0 | 1 |
|---|---|---|---|---|---|
| Jumlah Populasi Sample | 5 | 5 | 5 | 5 | 5 |
| Jumlah Presentase | 0% | 40% | 40% | 0 | 20% |

### 4.7.1 Evaluasi Hasil Pengujian

Dari pengujian yang telah dilakukan dapat disimpulkan bahwa aplikasi yang dibangun mudah untuk digunakan dengan tampilan yang cukup menarik. Aplikasi ini juga dirasa bermanfaat karena memberi informasi tambahan yang cukup. Kemudian kaitannya dengan efisiensi waktu dan biaya, aplikasi ini cukup membantu. Hasil pengujian ini baru secara persentase statistik biasa, dan mungkin saja hasil ini kurang akurat. Untuk itu, apabila ingin mengetahui jawaban dengan kedalaman akurasi, maka perlu adanya suatu pengolahan data statistik dengan acuan dari data hasil kuesioner yang sama.