

**BAB I: PENDAHULUAN**

Membahas tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat dari penelitian, metodologi penelitian dan sistematika penulisan laporan yang digunakan dalam pembuatan penelitian ini.

**BAB II: LANDASAN TEORI**

Membahas dasar-dasar teori yang digunakan dalam merancang dan membangun aplikasi sistem pakar untuk diagnosa komputer pada ponsel dengan J2ME. Teori ini meliputi teknologi Java, J2ME, sistem pakar, dan tentang identifikasi kerusakan komputer.

**BAB III: ANALISIS KEBUTUHAN PERANGKAT LUNAK**

Memuat uraian tentang metode analisis, analisis masalah, analisis sistem dan hasil analisis mengenai semua perangkat lunak yang diperlukan untuk membangun sistem serta menghasilkan pemenuhan kebutuhan sistem yang diwujudkan dari perangkat lunak, meliputi kebutuhan masukan dan keluaran sistem, fungsi-fungsi yang dibutuhkan dan antar muka yang diinginkan.

**BAB IV: PERANCANGAN PERANGKAT LUNAK**

Memuat uraian tentang metode perancangan perangkat lunak yang digunakan. Hasil perancangan merupakan terjemahan kebutuhan perangkat lunak yang berupa gambaran kasar mengenai perancangan sistem dan perancangan penyimpanan data yang akan dibangun.

## BAB V: IMPLEMENTASI PERANGKAT LUNAK

Menjelaskan tentang implementasi perangkat lunak yang dibangun pada sistem nyata yang akan diimplementasikan sesuai kebutuhan pengguna.

## BAB VI: ANALISIS KINERJA PERANGKAT LUNAK

Memuat dokumentasi hasil pengujian terhadap perangkat lunak yang dibandingkan kesesuaiannya dengan analisis dan perancangan yang telah dilakukan sebelumnya apakah sesuai dengan kebutuhan pengguna dan memiliki ketahanan sebagai suatu sistem yang baik.

## BAB VII: KESIMPULAN DAN SARAN

Memuat kesimpulan-kesimpulan dari seluruh rangkaian proses pengembangan perangkat lunak, baik pada tahap analisis, perancangan, implementasi terutama pada analisis kinerja perangkat lunak. Bab ini juga membahas saran yang dapat dipergunakan oleh pihak yang berkepentingan maupun untuk peneliti terhadap keterbatasan-keterbatasan yang ditemukan dalam pembuatan penelitian ini.

1. Java 2 Platform, Standard Edition (J2SE™)

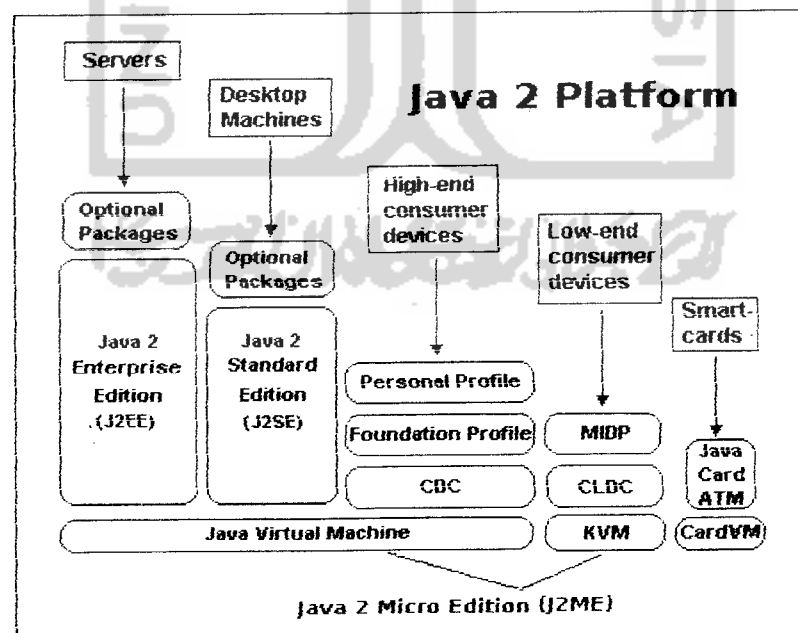
*Platform* digunakan untuk menjalankan dan mengembangkan aplikasi Java pada *level Personal Computer (PC)*. *Platform* ini berisi *class-class* inti pada Java dan *Graphical User Interface (GUI)*.

2. Java 2 Platform, Micro Edition (J2ME™)

*Platform* ini digunakan untuk menjalankan dan mengembangkan aplikasi-aplikasi Java pada *handheld devices* atau perangkat-perangkat semacam *handphone*, *Personal Digital Assistance (PDA)* dan *PocketPC*.

3. Java 2 Platform, Enterprise Edition (J2EE™)

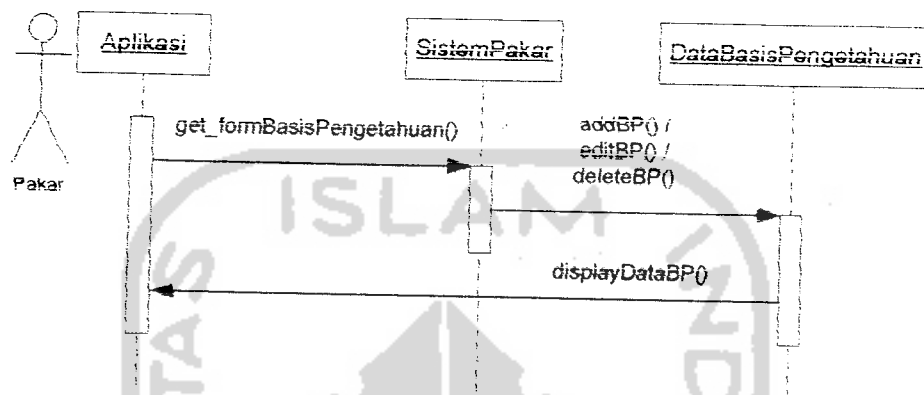
*Platform* ini berupa paket yang berisi *class-class* dan *interface-interface* yang digunakan untuk menjalankan dan mengembangkan aplikasi Java berbasis *web*, seperti *class-class* Servlet, *Java Server Pages (JSP)* dan *Enterprise JavaBeans (EJB)* serta Java CORBA.



**Gambar 2.1** Lingkungan kerja Teknologi Java

### 3. Olah Data Basis Pengetahuan

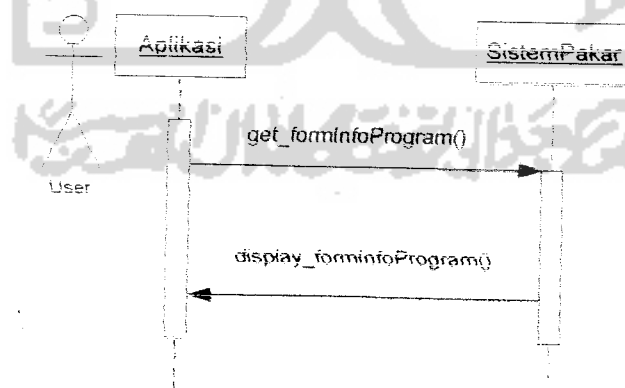
*Sequence Diagram* ini menunjukkan proses untuk olah data basis pengetahuan.



**Gambar 4.6** *Sequence Diagram* untuk Olah Data Basis Pengetahuan

### 4. Lihat Info Program

*Sequence Diagram* ini menunjukkan proses untuk melihat informasi tentang program.



**Gambar 4.8** *Sequence Diagram* untuk Lihat Info Program

:: Halaman Konsultasi ::

Pilih Piranti Komputer Yang Rusak :

- \_1. Piranti Pemrosesan
- \_2. Piranti Input
- \_3. Piranti Output

Kembali

Gambar 4.12 Halama Konsultasi

:: Pilih Gejala Kerusakan ::

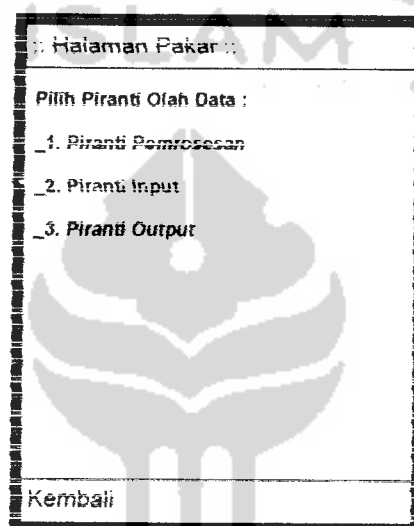
- ☒ Gejala 1
- ☒ Gejala 2
- ☒ Gejala 3

Batal Diagnosa

Gambar 4.13 Halama Pilih Gejala Kerusakan

menu untuk olah data, yaitu simpan data, ubah data, hapus data. Setelah memilih menu untuk olah data maka akan muncul Halaman Olah Data.

- d. Halaman Olah Data ada tiga macam, yaitu Olah Data Gejala (lihat gambar 4.18), Olah Data Kerusakan (lihat gambar 4.20), Olah Data Basis Pengetahuan (lihat gambar 4.22). Tombol “Ok” untuk proses olah data, sedangkan tombol “Batal” untuk membatalkannya.



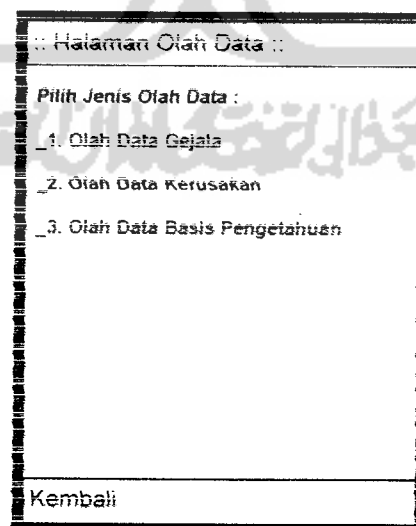
Halaman Pakar ::

Pilih Piranti Olah Data :

- \_1. Piranti Pemrosesan
- \_2. Piranti Input
- \_3. Piranti Output

Kembali

Gambar 4.15 Halaman Pakar



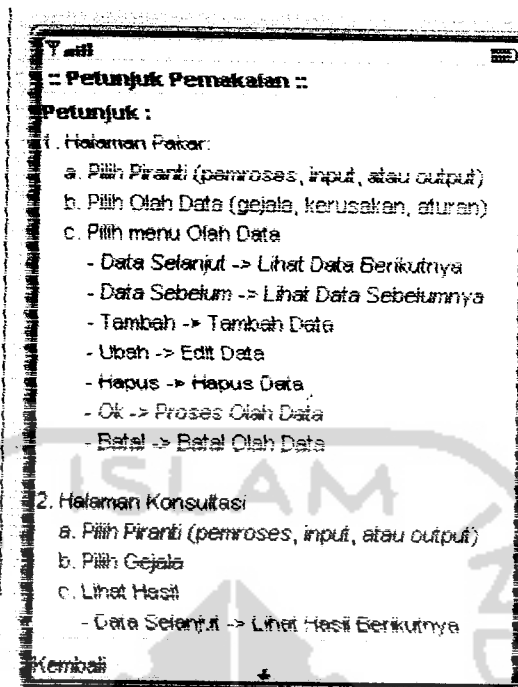
Halaman Olah Data ::

Pilih Jenis Olah Data :

- \_1. Olah Data Gejala
- \_2. Olah Data Kerusakan
- \_3. Olah Data Basis Pengetahuan

Kembali

Gambar 4.16 Halaman Pilihan Olah Data



Gambar 5.15 Tampilan halaman bantuan

### 5.3 Implementasi Prosedural

Implementasi prosedural pada aplikasi ini dikelompokkan kedalam kelas-kelas seperti yang telah dijelaskan pada bagian perancangan, yaitu:

#### 5.3.1 Class DataGejala

Prosedur-prosedur yang terdapat pada *class* DataGejala adalah sebagai berikut :

1. Prosedur untuk membuat/membuka RecordStore RSGejala

Nama prosedur : bukaRS

Jenis : public

2. Prosedur untuk menutup RecordStore RSGejala

Nama prosedur : tutupRS

Y. M. H. ABC

**:: Olah Data Basis Pengetahuan ::**

**Id Aturan :**  
1

**Id Gejala :**  
1

**Id Kerusakan :**  
1

**Nilai MB :**  
0.8

**Nilai MD :**  
0.1

Batal OK

Gambar 6.13 Tampilan halaman olah data basis pengetahuan

Y. M. H. ABC

**:: Data Aturan Piriati Penroses ::**

**Id Record : 2**

**Id Aturan : 2**

**Id Kerusakan : 1**

**Id Gejala : 2**

**Nilai MB : 0.9**

**Nilai MD : 0.1**

**Jumlah Record : 5**

Kembali Menu

Gambar 6.14 Tampilan hasil proses hapus data



5. Penggunaan NetBeans IDE 5.0 untuk mendesain dan menulis kode program membuat tampilan aplikasi ini lebih menarik dan lebih mudah dalam penulisan kode program.

## 7.2 Saran

Beberapa saran yang dapat diberikan untuk pengembangan penelitian selanjutnya adalah sebagai berikut:

1. Antarmuka aplikasi ini masih sangat terbatas. Hal ini karena kemampuan ponsel memang terbatas. Untuk pengembangan selanjutnya, antarmuka mungkin bisa dibuat lebih baik, menarik, dan mudah bagi *user* untuk menggunakannya.
2. Mengatur alokasi memori yang digunakan oleh aplikasi J2ME, karena semakin kecil kebutuhan memori akan semakin baik aplikasi berjalan.
3. Membangun aplikasi yang bersifat *client-server* sehingga proses *update* data pakar bisa dilakukan oleh pengelola aplikasi. Hal ini akan sangat memudahkan bagi *user*, karena data kerusakan yang diakses selalu *up to date* dan *user* tidak perlu lagi berurusan dengan input data pakar.