

# SISTEM PAKAR DIAGNOSA KERUSAKAN PADA KOMPUTER BERBASIS *FRAME* MENGGUNAKAN METODE *BREADTH FIRST SEARCH*

**Shodik Sanjaya**

Teknik Informatika, STMIK Widya Cipta Dharma  
Jl. M. Yamin No.25, Samarinda, 75123  
E-mail : Shodik5@gmail.com

## ABSTRAK

Sistem Pakar Diagnosa Kerusakan Pada Komputer Berbasis Frame, merupakan sistem yang dibuat untuk memberikan pengetahuan dan informasi sekaligus diagnosa kerusakan komputer kepada teknisi junior tanpa harus berkonsultasi terlebih dahulu kepada teknisi senior atau pakarnya. Sistem pakar ini juga mampu memberikan solusi kerusakan atau cara penanganan kerusakan dalam bentuk gambar sehingga memudahkan teknisi junior untuk mengikuti langkah-langkah dalam menangani kerusakan pada komputer.

Pembuatan website sistem pakar ini dibuat dengan menggunakan dreamweaver dan MySql sebagai databasenya. Website system pakar ini dibentuk dengan menggunakan metode *Breadth First Search* sebagai metode penelusuran kerusakannya. Alat bantu pengembangan system yang digunakan yaitu *Flowchart* dan *Sitemap*.

Dengan menerapkan metode *Breadth First Search* maka dihasilkan sebuah sistem pakar untuk mendiagnosa kerusakan pada komputer yang dapat memberikan kemudahan kepada teknisi untuk mendapatkan informasi tentang gejala dan solusi kerusakan pada komputer. System pakar ini juga dapat membantu kinerja pakar yaitu dengan mudah menambah, mengganti dan menghapus data pengetahuannya.

***Kata Kunci:*** Sistem Pakar, Metode *Breadth First Search*(BFS), Diagnosa Kerusakan Pada Komputer Berbasis *Frame*

---

## 1. PENDAHULUAN

Sistem pakar dikembangkan sejalan dengan adanya teknologi informasi. Pembangunan sistem pakar bertujuan sebagai sarana bantu untuk memberikan solusi di dalam kehidupan kita. Salah satu contohnya adalah kasus kerusakan komputer merupakan kasus yang memerlukan bantuan seorang pakar (teknisi) dalam menyelesaikan masalah dengan mengandalkan pengetahuan yang dimilikinya.

CV.Rahayu Computer Loa Janan merupakan salah satu tempat *service* komputer dan *printer* yang ada di wilayah Loa Janan, yang berada tepat di jalan Soekarno Hatta No. 15 Samarinda. Tujuan dari CV.Rahayu Komputer adalah membantu masyarakat sekitar untuk menyelesaikan masalah kerusakan komputer, *printer* dan penjualan barang suku cadang.

Saat ini teknisi junior akan kesulitan dalam memperbaiki komputer apabila teknisi senior tidak ada ditempat dan teknisi junior harus menelpon teknisi senior untuk mendapatkan solusi dalam memperbaiki komputer dan membutuhkan waktu yang cukup lama untuk mendiagnosa kerusakan komputer, bahkan teknisi sering

sekali menunda pekerjaannya hanya untuk menghasilkan solusi dari kerusakan komputer. Pada skripsi ini dibuat perangkat lunak untuk mengatasi kasus kerusakan komputer dengan menghemat waktu.

Di aplikasi ini dapat membantu teknisi junior untuk mendapatkan solusi dengan cepat dan teknisi senior apabila ada jenis dan ciri-ciri yang di temukan lagi bisa menambahkannya di basis aturan.

Perangkat lunak yang dimaksud adalah sistem pakar untuk diagnosis kerusakan komputer yang di desain untuk memodelkan kemampuan seorang pakar dalam memecahkan suatu masalah yang berbasiskan pada pakar itu sendiri.

## 2. RUANG LINGKUP PENELITIAN

Permasalahan difokuskan pada:

1. Diagnosa kerusakan komputer.
2. Penelusuran diagnosa berdasarkan gejala-gejala kerusakan komputer.
3. Hasil diagnosa berupa solusi serta saran yang diberikan untuk memperbaiki komputer dan penyebab-penyebab rusaknya komputer.

### 3. BAHAN DAN METODE

Adapun bahan dan metode yang digunakan dalam membangun sistem pakar ini yaitu:

#### 3.1 Sistem Pakar

Menurut Martin dan Qxman dalam Kusri (2008), Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat diselesaikan oleh seorang pakar dalam bidang tertentu.

Menurut Turban (2005), Sistem pakar adalah paket perangkat lunak pengambilan keputusan atau pemecahan masalah yang dapat mencapai tingkat performan yang setara atau bahkan lebih dengan pakar manusia di beberapa bidang khusus dan biasanya mempersempit area masalah.

##### 3.1.1 Tahap Pengembangan Sistem Pakar

Sistem pakar pada dasarnya adalah perangkat lunak komputer, jadi pengembangannya mengikuti pengembangan perangkat lunak. Tujuan proses pengembangannya adalah memaksimalkan probabilitas pembangunan perangkat lunak yang dapat terus dan mampu bertahan dalam keterbatasan biaya dan perencanaan, sementara pengelolaan berubah. Untuk mengembangkan sistem pakar melibatkan enam tahap pokok :

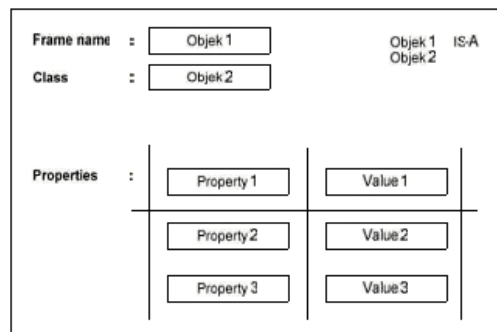
1. Inisialisasi proyek yaitu langkah pertama dalam pengembangan sistem pakar. Tujuan utamanya adalah mengidentifikasi masalah dan mempersiapkan aksi selanjutnya.
2. Analisis dan desain sistem yaitu sebuah konsep proyek disetujui, analisis sistem detail harus dijalankan untuk memperkirakan fungsionalis sistem.
3. *Prototyping* cepat yaitu proses *prototyping* sebenarnya bukan suatu fase, tetapi lebih merupakan siklus fase. Oleh karena cara pengetahuan didapatkan dan digabungkan ke dalam suatu sistem pakar, kita mendeskripsikannya sebagai suatu fase, *prototyping* telah menjadi sangat penting bagi pengembangan dan kesuksesan banyak sistem.
4. Pengembangan sistem yaitu proses yang panjang dan kompleks. Dalam fase ini, basis pengetahuan dikembangkan dan dilakukan pengujian, peninjauan dan perbaikan yang terus menerus.
5. Implementasi yaitu penyelesaian pengembangan sistem bukanlah akhir dari pengembangan sistem. Proses implementasi sistem pakar dapat menjadi lama dan kompleks.
6. Pasca implementasi yaitu beberapa aktivitas yang perlu adalah operasi, pemeliharaan, *upgrade* dan perluasan serta evaluasi sistem.

#### 3.2 Frame

Menurut Johar (2011), *frame* adalah struktur data yang menyertakan semua pengetahuan tentang objek tertentu. Pengetahuan ini diatur dalam struktur hierarki khusus yang mengizinkan diagnosis kemandirian pengetahuan. *Frame* berupa kumpulan-kumpulan *slot-slot* yang digunakan atau merupakan atribut untuk mendeskripsikan pengetahuan. Pengetahuan yang termuat dalam *slot* dapat berupa kejadian, lokasi, situasi

ataupun elemen-elemen lain. *Frame* digunakan untuk representasi pengetahuan deklaratif.

*Frame* dapat dipandang sebagai struktur data statik yang digunakan untuk merepresentasikan situasi-situasi yang telah dipahami dan *stereotype*. *Frame* berupa kumpulan-kumpulan *slot-slot* yang digunakan atau merupakan atribut untuk mendeskripsikan pengetahuan. Pengetahuan yang termuat dalam *slot* dapat berupa kejadian, lokasi, situasi ataupun elemen elemen lain. *Frame* digunakan untuk representasi pengetahuan deklaratif.



Gambar 1. Struktur *Frame*

Sumber : Johar, Asahar, 2011, *Implementasi Metode Frame Untuk Mendiagnosa Gangguan Kepribadian Dramatik Menggunakan Sistem Pakar*

*Frame* terdiri dua elemen dasar, yaitu slot dan *facet* (merupakan subslot). *Slot* merupakan kumpulan atribut atau properti yang menjelaskan objek yang direpresentasikan oleh *frame* dan *subslot* menjelaskan pengetahuan atau prosedur dari atribut pada *slot*. *Slot* dalam *frame* mungkin berisi informasi sebagai berikut :

1. Informasi identifikasi *frame*.
2. Hubungan *frame* dengan *frame* yang lain.
3. Penggambaran persyaratan yang dibutuhkan *frame*.
4. Informasi prosedural untuk menggunakan struktur yang digambarkan.
5. Informasi *default frame*.
6. Informasi baru.

Sedangkan, *subslot* terdiri dari beberapa bentuk antara lain :

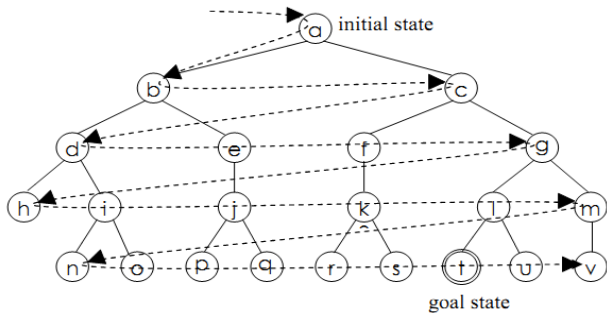
1. *Value* : nilai dari suatu atribut.
2. *Default* : nilai yang digunakan jika slot kosong atau tidak dideskripsikan pada instansiasi *frame*.
3. *Range* : jenis informasi yang muncul pada slot.
4. *If added* : berisi informasi tindakan yang akan dikerjakan jika nilai *slot* diisi.
5. *If needed* : *Facet (subslot)* ini digunakan pada kasus dimana tidak ada *Value* pada *slot*.
6. *Other* : Slot dapat berisi *frame*, *rule*, jaringan semantik ataupun tipe lain dari informasi.

Terdapat dua jenis properti yaitu properti statis dan dinamis. Properti statis merupakan fitur dari objek yang tidak dapat berubah, sedangkan properti dinamis merupakan fitur yang dapat berubah selama sistem berjalan.

#### 3.3 Breadth First Search

Menurut Rufaidah (2012), Prosedur *Breadth First*

Search merupakan pencarian yang dilakukan dengan mengamati setiap *node* di setiap tingkat graf sebelum bergerak menuju ruang yang lebih dalam. Atau dengan kata lain, penelusuran yang dilakukan dengan mengunjungi tiap-tiap *node* pada level yang sama hingga ditemukan *goal state*-nya. Untuk lebih jelasnya, perhatikan ilustrasi dari *Breadth First Search* pada gambar 2.



**Gambar 2** Teknik pencarian *Breadth First Search*  
**Sumber:** Rufaidah, Refi, 2012, *Perbandingan Algoritma Breadth First Search dan Depth First Search pada Aplikasi Game onet Menggunakan Platform Android.*

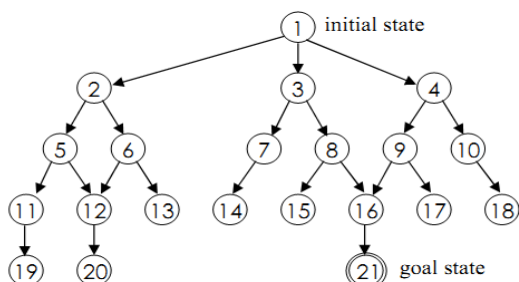
Pengimplementasian *Breadth First Search* dapat ditelusuri dengan menggunakan daftar (*list*), *open* dan *closed*, untuk menelusuri gerakan pencarian di dalam ruang keadaan. Prosedur untuk *Breadth First Search* dapat dituliskan sebagai berikut:

```

1 procedure BFS (G,v): [7]
2 create a queue Q
3 enqueue v onto Q
4 mark v
5 while Q is not empty:
6   t ← Q.dequeue()
7   if t is what we are looking for:
8     return t
9   for all edges e in G.incidentEdges(t) do
10    o ← G.opposite(t,e)
11    if o is not marked:
12      mark o
13      enqueue o onto Q
  
```

**Gambar 3.** Algoritma BFS  
**Sumber:** Rufaidah, Refi, 2012, *Perbandingan Algoritma Breadth First Search dan Depth First Search pada Aplikasi Game onet Menggunakan Platform Android.*

Untuk lebih jelasnya, simak contoh graf pada gambar 4.



**Gambar 4.** Contoh graf yang akan ditelusuri  
**Sumber:** Rufaidah, Refi, 2012, *Perbandingan Algoritma Breadth First Search dan Depth First Search pada Aplikasi Game onet Menggunakan Platform Android.*

Pada gambar 4, state 21 merupakan tujuannya (*goal*) sehingga bila ditelusuri menggunakan prosedur *Breadth First Search*, diperoleh:

1. Open = [1]; closed = [].
2. Open = [2, 3, 4]; closed = [1].
3. Open = [3, 4, 5, 6]; closed = [2, 1].
4. Open = [4, 5, 6, 7, 8]; closed = [3, 2, 1].
5. Open = [5, 6, 7, 8, 9, 10]; closed = [4, 3, 2, 1].
6. Open = [6, 7, 8, 9, 10, 11, 12]; closed = [5, 4, 3, 2, 1].
7. Open = [7, 8, 9, 10, 11, 12, 13] (karena 12 telah di-open); closed = [6, 5, 4, 3, 2, 1].
8. Open = [8, 9, 10, 11, 12, 13, 14]; closed = [7, 6, 5, 4, 3, 2, 1].
9. Dan seterusnya sampai state 21 diperoleh atau open = [].

Ada beberapa keuntungan menggunakan algoritma *Breadth First Search* ini, di antaranya adalah tidak akan menemui jalan buntu dan jika ada satu solusi maka *Breadth First Search* akan menemukannya, dan jika ada lebih dari satu solusi maka solusi optimal akan ditemukan. Namun ada tiga persoalan utama berkenaan dengan *Breadth First Search* ini yaitu:

1. Membutuhkan memori yang lebih besar, karena menyimpan semua node dalam satu pohon.
2. Membutuhkan sejumlah besar pekerjaan, khususnya jika lintasan solusi terpendek cukup panjang, karena jumlah node yang perlu diperiksa bertambah secara eksponensial terhadap panjang lintasan.
3. Tidak relevannya operator akan menambah jumlah node yang harus diperiksa.

Oleh karena proses *Breadth First Search* mengamati *node* di setiap level graf sebelum bergerak menuju ruang yang lebih dalam maka mula-mula semua keadaan akan dicapai lewat lintasan yang terpendek dari keadaan awal. Oleh sebab itu, proses ini menjamin ditemukannya lintasan terpendek dari keadaan awal ke keadaan tujuan (akhir). Lebih jauh karena mula-mula semua keadaan ditemukan melalui lintasan terpendek sehingga setiap keadaan yang ditemui pada kali kedua didapati pada sepanjang sebuah lintasan yang sama atau lebih panjang. Kemudian, jika tidak ada kesempatan ditemukannya keadaan yang identik pada sepanjang lintasan yang lebih baik maka algoritma akan menghapusnya.

#### 4. RANCANGAN SISTEM

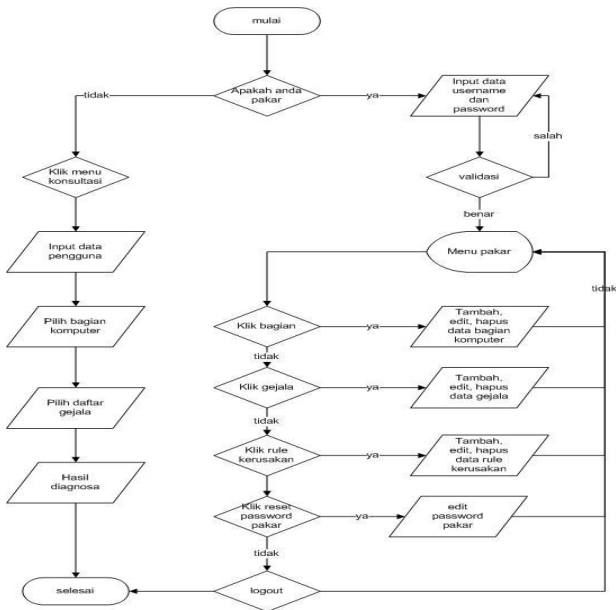
Berikut ini merupakan *flowchart* rancangan menu yang akan dibangun pada sistem pakar dan perancangan *database*:

##### 4.1 Flowchart system

*Flowchart* ini menjelaskan jalan sistem pada aplikasi sistem pakar ini. Pengguna dihadapkan pada dua pilihan yaitu sebagai Pengguna atau sebagai Pakar. Apabila sebagai pengguna, maka pengguna dapat menginput identitas diri dan di proses ke pemilihan bagian komputer. Jika sudah dipilih oleh pengguna, maka data direkam dan

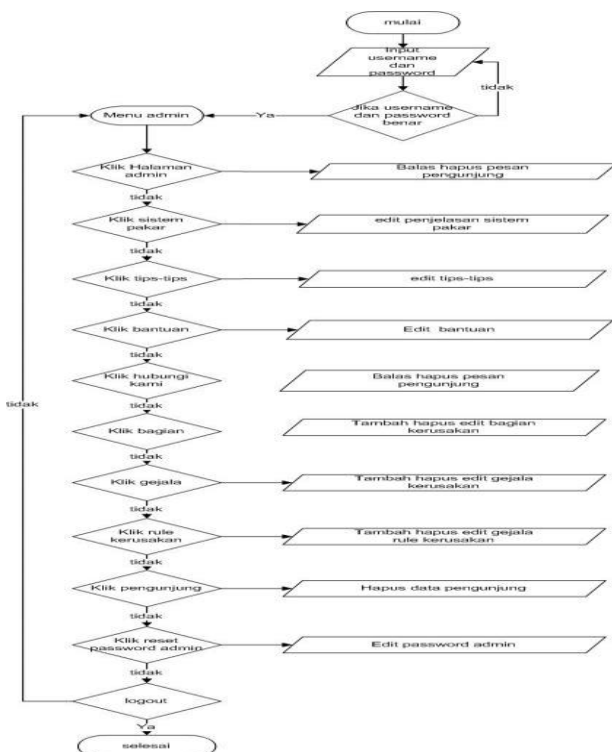
selanjutnya akan diproses sistem untuk menghasilkan hasil dari diagnosa dan menampilkan detail dari diagnosa sebagai hasil dari konsultasi.

Untuk seorang pakar agar dapat masuk ke sistem pakar, maka pakar harus menginput *login* dan *password* yang muncul pada saat memilih login sebagai pakar. Disini seorang pakar dapat menambah, mengubah, *menghapus* dan membuat aturan untuk bagian mesin, gejala kerusakan dan rule kerusakan beserta solusi untuk suku cadang yang rusak.



Gambar 5. flowchart sistem

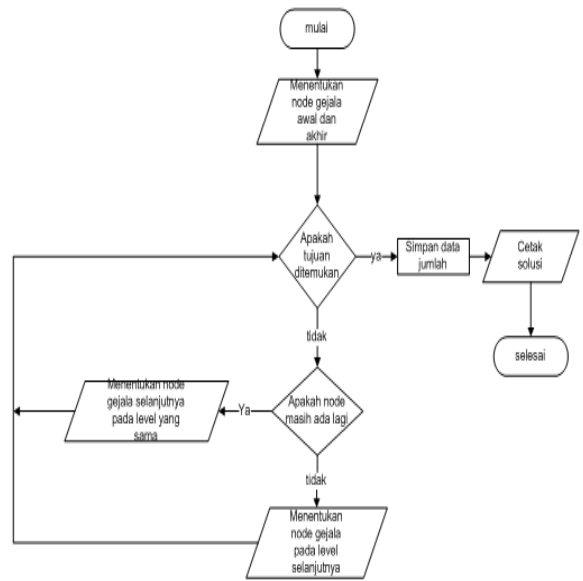
#### 4.2 Flowchart System Admin



#### Gambar 6. flowchart admin

Flowchart ini menjelaskan untuk seorang admin agar dapat masuk ke sistem pakar, maka admin harus menginput *login* dan *password* yang muncul pada saat memilih login sebagai admin. Disini seorang admin dapat memanejemen halaman utama, slide, kontak person, sistem pakar, tips-tips, bantuan, hubungi kami, bagian, gejala, rule kerusakan, pengunjung, reset password admin.

#### 4.3 Flowchart Breadth First Search



Gambar 7. flowchart breadth first search

Flowchart ini menjelaskan cara kerja metode *breadth first search*. Pencarian dimulai dari menentukan node awal dan akhir kemudian diproses bfs dan jika tujuan ditemukan maka cetak solusi jika *node* masih ada maka pencarian dilanjutkan pada level 1 *node* gejala yang sama atau bertetangga sampai ditemukan tujuannya.

Kemudian jika pada level 1 *node* gejala yang sama tidak ditemukan solusi maka pencarian akan dilanjutkan ke level 2 *node* gejala yang selanjutnya sampai ditemukan tujuannya kemudian melakukan proses simpan data jumlah *node* yang dipilih dan cetak solusinya.

#### 4.4 Tabel Aturan

Nama tabel : aturan  
 Primary Key : kd\_aturan  
 ForeignKey : -  
 Keterangan : Merupakan tabel untuk menampung aturan.

Tabel 1. aturan

NAMA	TIPE	PANJANG	KETERANGAN
Kd_aturan	int	11	Nama kode
Kode_kerusakan	varchar	10	Nama kode
Nama_kerusakan	varchar	50	Nama

Penjelasan	text		Penjelasan tentang
Solusi	text		Solusi dari kerusakan
G_kerusakan	varchar	255	Gambar solusi
Kd_ciri	varchar	10	Kode gejala

#### 4.5 Tabel Bagian

Nama tabel : bagian  
 Primary Key : kd\_bagian  
 ForeignKey : -  
 Keterangan : Merupakan tabel untuk bagian kerusakan.

Tabel 2. bagian

NAMA	TIPE	PANJANG	KETERANGAN
Kd_bagian	Varchar	50	Kode bagian kerusakan
Bagian	Varchar	50	Nama bagian

#### 4.6 Tabel Gejala

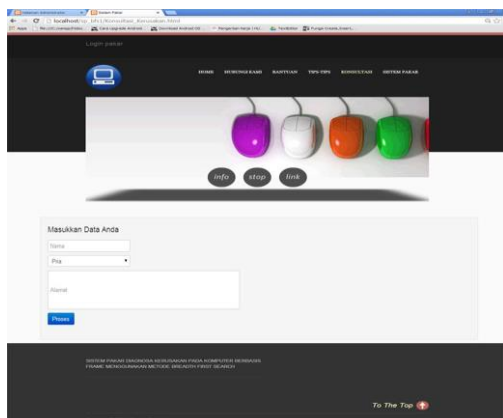
Nama tabel : gejala  
 Primary Key : kd\_ciri  
 ForeignKey : -  
 Keterangan : Merupakan tabel untuk menampung data gejala

Tabel 3. Gejala

NAMA	TIPE	PANJANG	KETERANGAN
Kd_ciri	varchar	10	Kode gejala
Ciri	varchar	255	Ciri gejala kerusakan
Kd_bagian	varchar	10	Kode bagian kerusakan

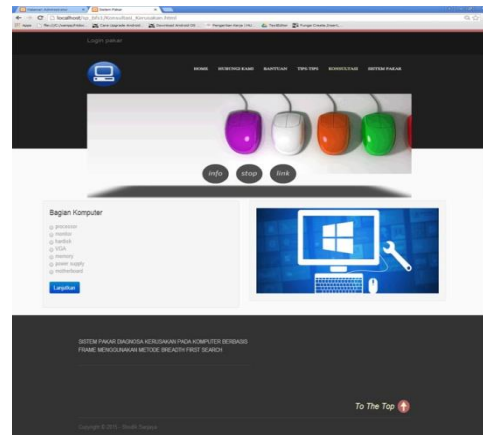
### 5. IMPLEMENTASI

Tampilan konsultasi disini digunakan untuk teknisi junior mencari solusi dari kerusakan komputer. Pertama teknisi junior memasukkan data diri pada website ini. Dimana teknisi junior menginputkan nama, jenis kelamin dan alamat kemudian diproses untuk disimpan dan dilanjutkan ke pemilihan bagian kerusakan komputer lihat di gambar 8.



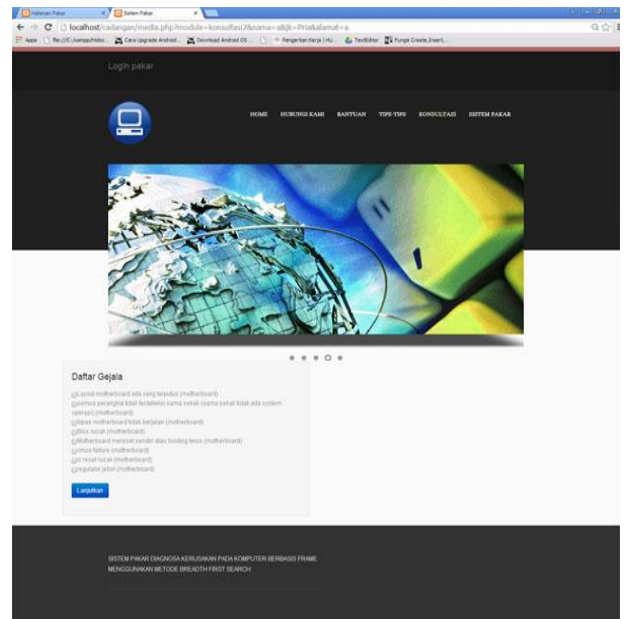
Gambar 8. tampilan masukkan data diri

Disini teknisi junior masuk ke form pilih bagian pada website ini. Dimana teknisi junior memilih bagian kerusakan komputer yang dirasa rusak komputer kemudian dilanjutkan ke daftar gejala untuk langsung memilih gejala sesuai gejala yang dirasakan pada komputer lihat pada gambar 9.



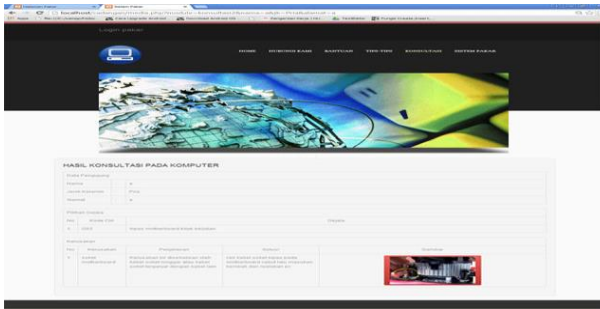
Gambar 9. tampilan pilih bagian komputer

Tampilan daftar gejala pada website ini berfungsi menentukan gejala kerusakan komputer yang dirasakan teknisi junior kemudian lanjut ke hasil diagnosa.



Gambar 10. tampilan daftar gejala.

tampilan hasil diagnosa ini menampilkan hasil diagnosa yang sebelumnya teknisi junior memilih gejala kerusakan yang akan di diagnosa. Halaman ini menampilkan bagian komputer, gejala yang dipilih, serta penjelasan solusi kerusakan, halaman ini juga menampilkan data teknisi junior yang melakukan konsultasi. Halaman ini juga menampilkan penyebab-penyebab terjadinya kerusakan komputer yang dialami teknisi junior.



**Gambar 11. tampilan hasil diagnosa kerusakan komputer**

## 6. KESIMPULAN

Berdasarkan hasil dari pembahasan dan penjelasan pada bab-bab sebelumnya, maka dapat dibuat kesimpulan yaitu :

1. Sistem Pakar Diagnosa Kerusakan Pada Komputer Berbasis *Frame* dengan menggunakan metode *Breadth First search* ini dibuat sebagai alat bantu untuk dapat mengetahui solusi kerusakan komputer dengan pengetahuan seorang pakar. Sistem Pakar Diagnosa Kerusakan Pada Komputer ini juga dapat dengan mudah untuk menambahkan atau *update* data yang ada tentang kerusakan komputer ini.

## 7. SARAN

Berdasarkan hasil dari penelitian ini akan beberapa saran, yaitu sebagai berikut :

1. sistem pakar diagnosa kerusakan pada komputer ini masih bisa di kembangkan menggunakan sms gateway dalam melakukan konsultasinya sehingga lebih cepat dalam mengetahui kerusakan pada komputer.
2. Untuk pengembangan selanjutnya sistem pakar diagnosa kerusakan pada komputer ini bisa ditambahkan video langkah-langkah cara memperbaiki komputer agar lebih memahami cara memperbaikinya.
3. Diharapkan agar sistem pakar diagnosa kerusakan pada komputer ini dapat dijadikan sebagai bahan acuan atau referensi bagi mahasiswa bidang informatika.
4. Untuk menjaga dan memelihara keakuratan data maka perlu dilakukan proses *update* data dan diharapkan dapat menemukan serta menambah gejala-gejala kerusakan pada komputer yang baru.
5. Dengan adanya sistem pakar diagnosa kerusakan pada komputer yang telah dibuat, diharapkan dapat disosialisasikan kepada teknisi junior agar dapat digunakan sesuai dengan fungsinya.

## 8. DAFTAR PUSTAKA

- Amanda, Berry, 2012, *Sistem Pakar Diagnosa Penyakit Osteoporosis*, Sekolah Tinggi Manajemen Informatika dan Komputer Widya Cipta Dharma, Samarinda
- Arhami, Muhammad, 2005, *Konsep Dasar Sistem Pakar*, Andi Offset : Yogyakarta.

- Febrian, Jack, 2004, *Menggunakan Internet*, Bandung : Informatika.
- Hartono, Jogiyanto, 2004, *Pengenalan Komputer*. Yogyakarta: andi offset.
- Jogianto, HM, 2005, *Analisis dan desain sistem informasi*, andi offset, Yogyakarta
- Johar, Asahar, 2011, *Implementasi Metode Frame Untuk Mendiagnosa Gangguan Kepribadian Dramatik Menggunakan Sistem Pakar*, Bengkulu : Universitas Bengkulu Fakultas Teknik.
- kamus besar bahasa Indonesia, 2005, balai pustaka, Jakarta.
- Kristanto, Andri, 2004, *Perancangan Sistem Informasi dan Aplikasinya*, Gava Media, Yogyakarta.
- Kusrini, 2008, *Sistem Pakar Teori dan Aplikasi*, Andi Offset, Yogyakarta.
- Madcoms, 2011, *memaksimalkan rumusan dengan fungsi Microsoft Access 2010*, Yogyakarta: Andi.
- Marlyaningrum, Arini, 2013, *Aplikasi Sistem Pakar Diagnosa pada Sistem Komputer*, Fakultas Teknik Universitas Widyatama, Bandung.
- Nugroho, Bunafit. 2004. *Aplikasi Pemrograman Web Dinamis dengan PHP dan MySQL*. Penerbit Gava Media: Yogyakarta
- Nugroho, Bunafit, 2004, *Php dan MySQL Dengan Editor Dreamweaver MX*, Yogyakarta: Andi.
- Nugroho, Bunafit. 2005. *Database Relasional Dengan MySQL*. Penerbit Andi: Yogyakarta.
- Pademmul, 2006, *Torch, Akibat dan Solusinya*, Solo : PT Wangsa Jastra Lestari.
- Permatasari, Niken Indah, 2013, *Rancang Bangun Sistem Pakar Diagnosis Kerusakan Komputer Dan penanganannya Berbasis Mobile Web*, Sekolah Tinggi Manajemen Informatika & Komputer, Surabaya.
- Rufaidah, Refi, 2012, *Perbandingan Algoritma Breadth First Search dan Depth First Search pada Aplikasi Game onet Menggunakan Platform Android*, Bandung, Fakultas Teknik Dan Ilmu Komputer Universitas Komputer Indonesia Bandung.
- Simarmata, Janner, 2010, *Rekayasa Perangkat Lunak, Yogyakarta : Andi Offset*.
- Sutisna, Dadan. 2007. *7 Langkah Mudah Menjadi Webmaster*. Penerbit Mediakita: Jakarta.
- Turban, Efrain, 2005, *Dicision Support System and Intelligent System*, Yogyakarta : Andi.
- Tutang. 2005. *Praktikum HTML (Hypertext Markup Language) untuk Pelajar, Mahasiswa, Umum*, Penerbit Datakom Lintas Buana: Jakarta.
- Yuhfizar, 2008, *10 JAM menguasai Internet: Teknologi dan Aplikasinya*. Penerbit Elex Media Komputindo: Jakarta.