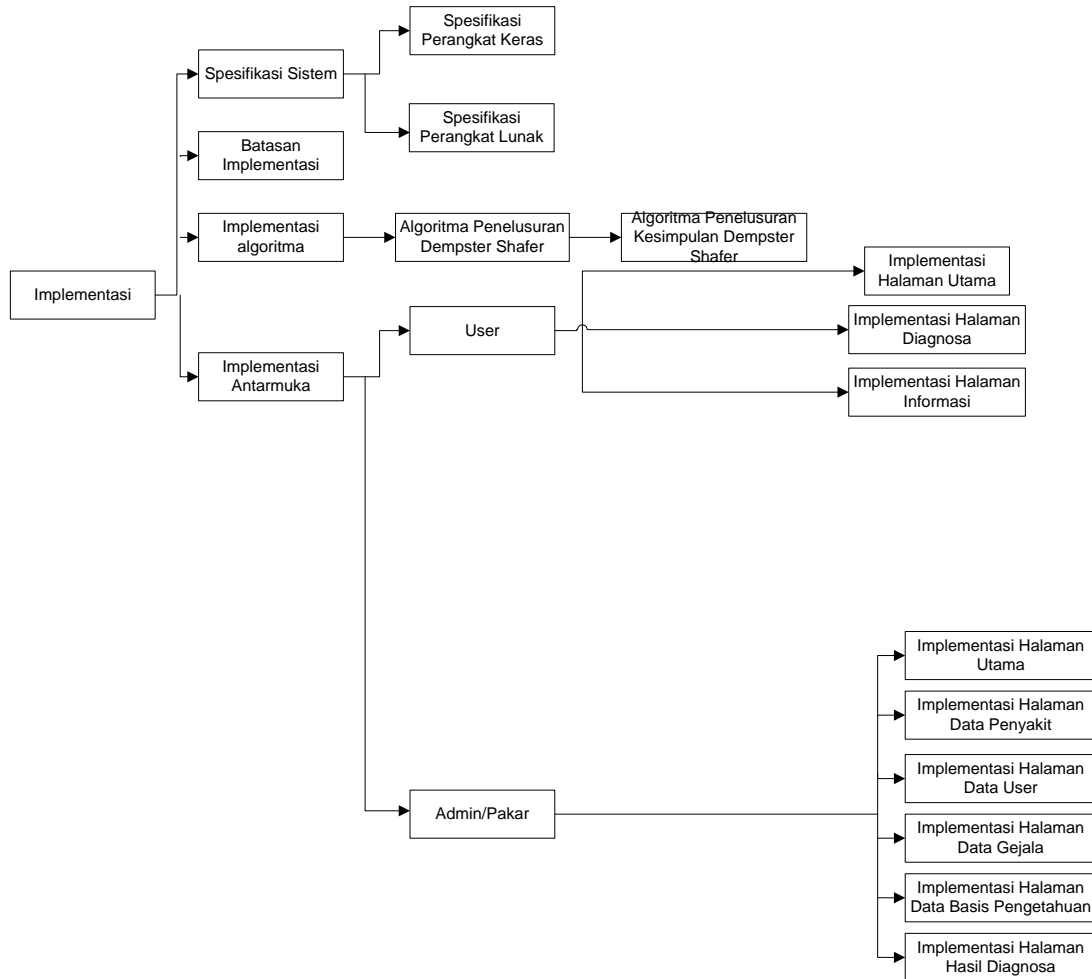


## BAB 5 IMPLEMENTASI

Bab ini membahas mengenai implementasi perangkat lunak berdasarkan hasil yang telah diperoleh dari analisis kebutuhan dan proses perancangan perangkat lunak yang telah dibuat. Pembahasan terdiri dari penjelasan tentang spesifikasi sistem, batasan-batasan dalam implementasi, implementasi algoritma, implementasi algoritma pada program, dan implementasi antarmuka. Pohon implementasi sistem pakar dapat dilihat pada Gambar 5.1.



Gambar 5.1 Pohon Implementasi

### 5.1 Spesifikasi Sistem

Hasil analisis kebutuhan dan perancangan perangkat lunak yang telah diuraikan pada Bab 4 menjadi acuan untuk melakukan implementasi menjadi sistem yang dapat berfungsi sesuai dengan kebutuhan. Spesifikasi sistem diimplementasikan pada spesifikasi perangkat keras dan perangkat lunak.

### 5.1.1 Spesifikasi Perangkat Keras

Aplikasi Pemodelan Sistem Pakar Diagnosis Penyakit Tanaman Jeruk menggunakan sebuah Komputer dengan spesifikasi perangkat keras yang dijelaskan pada Tabel 5.1.

**Tabel 5.1** Spesifikasi Perangkat Keras

Nama Komponen	Spesifikasi
Processor	Intel(R) Core i5-3210M CPU @2.50Ghz
Memory (RAM)	4 GB
Harddisk	1000 GB HDD

### 5.1.2 Spesifikasi perangkat Lunak

Aplikasi Pemodelan Sistem Pakar Diagnosis Penyakit Tanaman Jeruk menggunakan perangkat lunak dengan spesifikasi yang dijelaskan pada Tabel 5.2.

**Tabel 5.2** Spesifikasi Perangkat Lunak

Sistem Operasi	Microsoft Windows 7 (64-bit)
Bahasa Pemrograman	PHPdesigner
Tools Pemrograman	Notepad ++
<i>Server Localhost</i>	XAMPP 3.1.10
DBMS	PHPMYAdmin

## 5.2 Batasan-batasan Implementasi

Beberapa batasan dalam mengimplementasikan metode Dempster shafer pada Sistem Pakar Diagnosis penyakit tanaman jeruk adalah sebagai berikut:

1. Masukan yang diterima oleh sistem adalah berupa inputan dari gejala-gejala yang dipilih oleh pengguna.
2. Keluaran yang diterima oleh pengguna berupa hasil diagnosis jenis penyakit yang diderita tanaman jeruk.
3. Aplikasi yang digunakan berbasis web dengan basis data penyimpanan menggunakan *PHPMYAdmin*.
4. Metode yang digunakan yaitu *Dempster shafer*.
5. Sistem ini bersifat dinamis, yaitu dapat melakukan penambahan dan perubahan data aturan, jenis penyakit serta gejalanya.

## 5.3 Implementasi Algoritma

Implementasi yang dibahas menggunakan bahasa pemrograman PHP dan menggunakan database *PHPMYAdmin*. Bahasa pemrograman PHP digunakan untuk membangun aplikasi. Database *PHPMYAdmin* digunakan untuk menyimpan data. Pada penulisan implementasi ini yang dicantumkan hanya algoritma proses perhitungan dempster shafer dan proses pengambilan kesimpulan.

### 5.3.1 Proses Perhitungan Dempster shafer

Dalam *dempster shafer* gejala dari masing-masing penyakit memiliki bobot. Dalam aplikasi pemodelan sistem pakar ini, pengguna harus memilih gejala yang dialami pada tanaman jeruk. Sistem akan mendapatkan bobot untuk setiap gejala dari inputan pengguna. Masing-masing gejala memiliki bobot yang berbeda pada masing-masing penyakit. Implementasi algoritma proses penelusuran nilai bobot hingga perhitungannya dapat dilihat pada source code Implementasi algoritma perhitungan bobot.

```
1 /Fungsi Perhitungan
2 function get_diagnosis($gejala =
3 "'G001', 'G002', 'G004', 'G003'"){
4     $densitas = array();
5     $m_hasil = array();    $inc = 1;
6     $hasil_akhir = array();
7     $diag_hasil = array();
8     //query get data densitas dan kerusakan
9     $data = mysql_query("SELECT m.gejala,
10    GROUP_CONCAT(m.kerusakan) kom, MAX(m.mb) nilai, 1-MAX(m.mb)
11    tetha, GROUP_CONCAT(m.mb) densitas
12    FROM (SELECT * FROM basis_pengetahuan ORDER BY gejala, mb
13    DESC, kerusakan) m WHERE m.gejala in (".$gejala.") GROUP
14    BY m.gejala;");
15     while($row = mysql_fetch_row($data)){
16         $m_hasil[$inc] =
17         array($row[1]=>$row[2], 'th'=>$row[3]); // simpan nilai
18         densitas dan theta ke variable
19         $densitas[$row[0]] =
20         get_densitas($row[1], $row[4]); //
21         $inc++;
22         $diag_hasil = array(array(0,0,0));
23         if($inc>2){
24             $m_hasil[$inc] = hitung_data(
25             $m_hasil[$inc-1], $m_hasil[$inc-2]); // fungsi perhitungan
26             DS
27             $diag_hasil =
28             get_max_nilai($diag_hasil, $m_hasil[$inc], $row[0]); // Get
29             Nilai Tertinggi
30             $inc++;
31         } else $diag_hasil[] =
32         array($row[1], $row[2], $row[0]);
33     }
34     //hitung hasil AKhir
35     foreach($diag_hasil as $row){
36         if($row[0] == 'k' || $row[0] == 'th')
37         $hasil_akhir[] = array($row[0], $row[1], $row[1]);
38     }
```

```

33         else {
34             $kom = explode(',', $row[0]);
35             if(count($kom)>1) { //jika
36 kombinasi kerusakan lebih dari 1
37                 $dens =
38 getDensitasHasil($kom,$densitas,$row[2]);
39                 if(count($dens)>1){
40 //menentukan kerusakan yang diminan dari kombinasi
41 kerusakan yang ada
42                     foreach($kom as $row1){
43                         if($inc<=2)
44 $hasil_akhir[] = array($row1,$dens[$row1],$row[1]);
45                         else
46 $hasil_akhir[] = array($row1,$dens[$row1]*$row[1],$row[1]);
47                     }
48                 } else $hasil_akhir[] =
49 array($row[0],$row[1],$row[1]);
50                 } else $hasil_akhir[] =
51 array($row[0],$row[1],$row[1]); //kerusakan hanya 1
52 kombinasi
53             }
54         }
55         return
56 array('hitung'=>$m_hasil,'hasil'=>$diag_hasil,'densitas'=>$
57 densitas,'hasil_akhir'=>$hasil_akhir,'inc'=>$inc);
58     }
59
60 //fungsi pencarian nilai tertinggi
61     function get_max_nilai($max_nil,$data,$gejala){
62         foreach($data as $kom=>$value){
63             if($max_nil[0][1]<$value) { $max_nil =
64 array(); $max_nil[] = array($kom,$value,$gejala); }
65             else if($max_nil[0][1]==$value)
66 $max_nil[] = array($kom,$value,$gejala);
67         }
68         return $max_nil;
69     }
70 //mendapatkan densitas
71     function get_densitas($kom,$den){
72         $komb = explode(',',$kom);
73         $dens = explode(',',$den);
74         $hasil = array();
75         for($i=0; $i<count($komb); $i++){
76 $hasil[$komb[$i]] = $dens[$i]; }
77         return $hasil;

```

73	}
74	
75	

Source Code Implementasi algoritma perhitungan Bobot

## 5.4 Implementasi Antarmuka

Antarmuka Sistem Pakar Diagnosis penyakit tanaman jeruk ini digunakan oleh pengguna untuk berinteraksi dengan sistem perangkat lunak. Pada implementasi antarmuka perangkat lunak ini tidak semua halaman pada sistem ditampilkan, tetapi hanya antarmuka halaman tertentu saja. Beberapa antarmuka yang ditampilkan antara lain halaman diagnosis, implementasi halaman hasil diagnosis dan implementasi halaman basis pengetahuan.

### 5.4.1 Implementasi Halaman Data Basis Pengetahuan

Halaman data basis pengetahuan merupakan halaman untuk mengubah data yang ada pada tabel basis pengetahuan. Menu yang ditampilkan nama penyakit, gejala, bobot dan Menu yang disediakan adalah tambah basis pengetahuan, edit basis pengetahuan dan hapus basis pengetahuan. Berikut tampilan halaman data gejala dilihat pada Gambar 5.2.

No	Penyakit	Gejala	MB	Aksi
1	Busuk Buah	Permukaan daun tumbuh bintik hitam	0.8	<a href="#">Edit</a> <a href="#">Delete</a>
2	Kanker Jeruk	Daun timbul warna coklat	0.6	<a href="#">Edit</a> <a href="#">Delete</a>
3	Busuk Buah	Daun timbul warna coklat	0.7	<a href="#">Edit</a> <a href="#">Delete</a>
4	Tristeza	Sisi daun melengkung ke atas	0.7	<a href="#">Edit</a> <a href="#">Delete</a>
5	Tristeza	Pertumbuhan daun tidak teratur	0.6	<a href="#">Edit</a> <a href="#">Delete</a>
6	Tristeza	Pemucatan tulang daun berupa garis-garis putus / memanjang	0.8	<a href="#">Edit</a> <a href="#">Delete</a>
7	Embun Tepung	Pada daun atas tampak putih	0.8	<a href="#">Edit</a> <a href="#">Delete</a>
8	Embun Tepung	Tunas tidak normal	0.7	<a href="#">Edit</a> <a href="#">Delete</a>
9	Embun Tepung	Berkutil coklat	0.6	<a href="#">Edit</a> <a href="#">Delete</a>

Gambar 5.2 Implementasi halaman Data Basis Pengetahuan

### 5.4.2 Implementasi Halaman Diagnosis

Menu Diagnosis merupakan menu inti dari sistem pakar ini. Melalui halaman ini user dapat melakukan proses diagnosis penyakit tanaman jeruk. Pertama user akan disajikan berbagai gejala-gejala yang disebabkan penyakit tanaman jeruk berdasarkan bagian tanaman, yakni batang, buah dan daun. Kemudian user memilih gejala sesuai dengan diinginkan. Tampilan halaman diagnosis dapat dilihat pada Gambar 5.3.

## Diagnosa

Pilih Berdasarkan Gejala		
No	Gejala penyakit	Pilih
1	Permukaan daun tumbuh bintik hitam	<input type="checkbox"/>
2	Daun timbul warna coklat	<input checked="" type="checkbox"/>
3	Pemucatan tulang daun berupa garis-garis putus / memanjang	<input checked="" type="checkbox"/>
4	Sisi daun melengkung ke atas	<input type="checkbox"/>
5	Pertumbuhan daun tidak teratur	<input type="checkbox"/>
6	Pada daun atas tampak putih	<input type="checkbox"/>
7	Tunas tidak normal	<input type="checkbox"/>
8	Berkutil coklat	<input checked="" type="checkbox"/>

**Gambar 5.14** Implementasi halaman Diagnosa

Setelah pengguna melakukan diagnosis, maka akan muncul hasil diagnosis penyakit tanaman jeruk. Halaman hasil diagnosis menampilkan hasil diagnosis dan jenis penyakit. Adapun tampilan hasil diagnosis dapat dilihat pada Gambar 5.4.

## Diagnosa

Pilih Berdasarkan Gejala		Diagnosa Lagi
Pengguna	Joni Saputra	
Asumsi Penyakit yang dialami tanaman jeruk	Tristeza	
Deskripsi penyakit	Penyakit ini kerap menyerang pada bagian batang dan daun, yang juga dapat menghambat pertumbuhan dari pohon jeruk yang ditanam.	

**Gambar 5.15** Implementasi halaman Hasil Diagnosa