

**DETEKSI KERUSAKAN PADA *HARDWARE* KOMPUTER
MENGUNAKAN METODE *DEMPSTER SHAFER***

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

Aris Sugiarto

NIM: 0910963070

UNIVERSITAS BRAWIJAYA



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

DETEKSI KERUSAKAN PADA *HARDWARE* KOMPUTER MENGGUNAKAN METODE
DEMPSTER SHAFER

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Aris Sugiarto
NIM: 0910963070

Skripsi ini telah diuji dan dinyatakan lulus pada
26 Agustus 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Randy Cahya W, S.ST.,M.kom
NIP: 201405 880206 1

Lailil Muflikhah, S.Kom, M.Sc
NIP: 19741113 200501 2 001

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T. M.T, Ph.D
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 10 Agustus 2016



Aris Sugiarto

NIM: 0910963070

KATA PENGANTAR

Puji syukur kehadiran Allah SWT karena atas segala rahmat dan karunia-Nya, penulis dapat menyelesaikan penyusunan skripsi sebagai syarat untuk memperoleh gelar sarjana dengan judul “Deteksi Kerusakan Pada *Hardware* Komputer Menggunakan Metode *Dempster Shafer*”.

Penulis menyadari bahwa skripsi ini tidak akan dapat diselesaikan dengan baik tanpa keterlibatan dari berbagai pihak. Untuk itu, penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Randy Cahya W, S.ST.,M.kom selaku dosen pembimbing pertama yang telah memberikan bimbingan dan masukan dalam penyusunan skripsi ini.
2. Lailil Muflikhah, S.Kom, M.Sc selaku dosen pembimbing kedua yang telah memberikan pengarahan tentang penulisan skripsi ini.
3. Kedua orang tua atas segala doa, nasihat, dan perhatiannya dalam mendampingi dan memberikan dukungan moral kepada penulis.
4. Segenap Bapak dan Ibu dosen atas kesediaan membagi ilmu kepada penulis selama menempuh pendidikan di Program Studi Teknik Informatika dan Ilmu Komputer Universitas Brawijaya.
5. Bapak Hanifan Muklish, S.Kom selaku teknisi service komputer dan elektronik PT. Elekompetindo Solution yang telah memberikan pengetahuan seputar komputer terutama memberikan data dan penjelasan mengenai kerusakan pada *hardware* komputer untuk menyelesaikan skripsi ini.
6. Teman-teman seperjuangan, Muid, Hendra, Okky, Bisma, Angga, Wahyu, Hexa, lif, Lintang, Arya, Cholid, Vivin dan Rosi yang tak henti-hentinya selalu berbagi semangatnya dan saling berbagi ilmu untuk menyelesaikan skripsi.

Semoga segala pertolongan dan kebaikan semuanya mendapatkan berkah dan balasan dari Allah SWT. Penulis menyadari bahwa skripsi ini masih jauh dari kata sempurna, oleh karena itu kritik dan saran yang sifatnya membangun sangat diperlukan untuk memperbaiki mutu penulisan selanjutnya dan juga kebaikan penulis secara pribadi. Akhir kata, penulis berharap semoga skripsi ini dapat bermanfaat bagi semua pihak.

Malang, 10 Agustus 2016

Penulis

unyeliciouz41@gmail.com

ABSTRAK

Kerusakan pada *hardware* komputer sering terjadi, contohnya kerusakan *processor*, kerusakan *VGA*, maupun *power supply*. Kerusakan-kerusakan tersebut memiliki gejala yang berbeda-beda. Untuk dapat mengetahui kerusakan yang terdapat pada *hardware* komputer dilakukan implementasi metode *Dempster Shafer* dengan cara memasukkan gejala-gejala yang terjadi pada *hardware* komputer dan dihitung nilai kemungkinan kerusakan yang tertinggi sehingga hasil yang diperoleh menjadi lebih akurat. Dari hasil pengujian yang dilakukan, aplikasi mampu menghasilkan akurasi dengan data pakar sebesar 83% dari 12 data yang diuji.

Kata Kunci: *Hardware* Komputer, *Dempster Shafer*

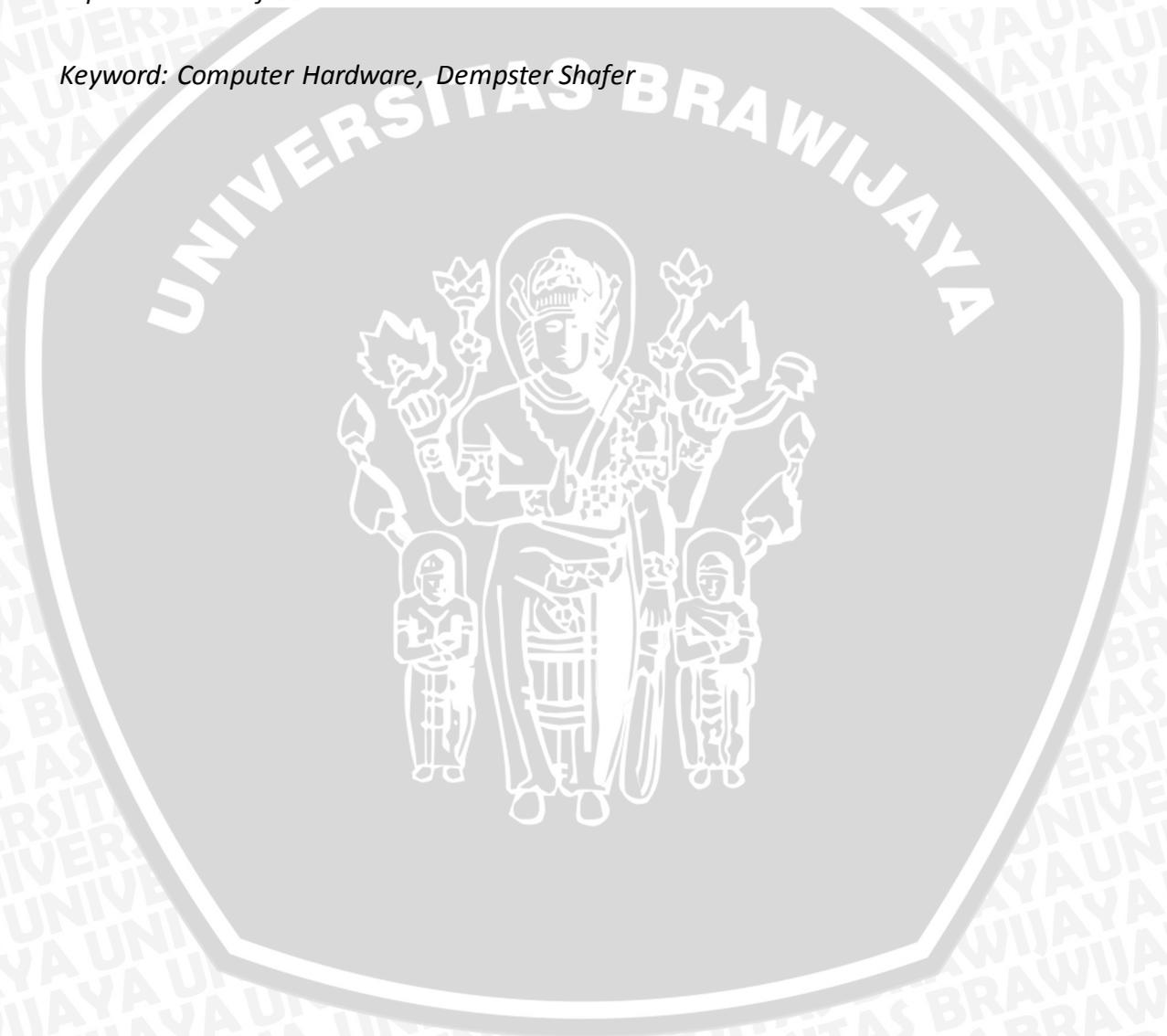
UNIVERSITAS BRAWIJAYA



ABSTRACT

Damaged into computer hardware is often happened, for example, damage to the processor, VGA damage, as well as power supply. The damages that have different symptoms. To be able to know that there is damage to computer hardware implementation methods Dempster Shafer conducted by inserting the symptoms that occur in the computer hardware and the calculated value of the highest possible damage so that the results become more accurate. From the results of tests performed, the application is able to produce the data accuracy by experts at 83% of 12 test data.

Keyword: Computer Hardware, Dempster Shafer



DAFTAR ISI

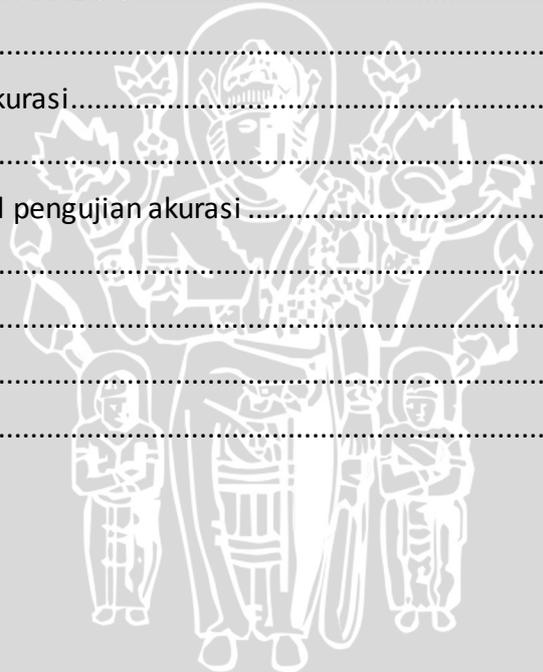
COVER.....	i
PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR	iv
ABSTRAK.....	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR	xi
DAFTAR PERSAMAAN	xii
DAFTAR <i>SOURCE CODE</i>	xiii
DAFTAR LAMPIRAN	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	2
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah.....	3
1.6 Sistematika pembahasan.....	3
BAB 2 LANDASAN KEPUSTAKAAN	5
2.1 Kajian pustaka.....	5
2.2 Landasan teori	6
2.2.1 Pengertian deteksi	6
2.2.2 Pengertian <i>hardware</i>	6
2.2.3 Pengertian komputer.....	6
2.2.4 Jenis-jenis <i>hardware</i> pada komputer	6
2.2.4.1 <i>Processor (CPU) Central Processing Unit</i>	6
2.2.4.2 <i>Hard disk</i>	6
2.2.4.3 <i>Video Graphic Adapter (VGA)</i>	7
2.2.4.4 <i>Random Access Memory (RAM)</i>	7
2.2.4.5 <i>Power supply</i>	8



2.2.4.6 Monitor	8
2.2.4.7 Keyboard dan mouse	8
2.2.4.8 Port atau (USB) Universal Serial Bus.....	9
2.2.4.9 Sound card	9
2.2.5 Pengertian metode	9
2.2.6 Metode Dempster Shafer	10
BAB 3 METODOLOGI	14
3.1 Studi literatur	15
3.2 Pengumpulan data	15
3.3 Analisis kebutuhan.....	15
3.4 Perancangan	15
3.5 Implementasi	16
3.5.1 Identifikasi <i>input</i>	16
3.5.2 Identifikasi <i>output</i>	17
3.6 Pengujian dan analisis.....	17
3.7 Pengambilan kesimpulan.....	17
BAB 4 PERANCANGAN.....	18
4.1 Analisis kebutuhan.....	18
4.1.1 Analisis perangkat lunak.....	18
4.1.2 Analisis perangkat keras	18
4.1.3 Analisis pengguna	18
4.1.4 Analisis <i>input</i>	19
4.1.5 Analisis proses.....	19
4.1.6 Analisis <i>output</i>	19
4.2 Perancangan algoritma	19
4.3 Perancangan Dempster Shafer	19
4.4 Perhitungan manual	27
4.4.1 Perhitungan 1 gejala (kasus 1).....	27
4.4.2 Perhitungan 2 gejala (kasus 2).....	27
4.4.3 Perhitungan 3 gejala (kasus 3).....	29
4.5 Antarmuka aplikasi	32
4.5.1 Antarmuka halaman utama	32



4.5.2 Antarmuka halaman deteksi kerusakan	33
4.5.3 Antarmuka halaman hasil deteksi kerusakan.....	33
BAB 5 IMPLEMENTASI	35
5.1 Implementasi spesifikasi.....	35
5.1.1 Spesifikasi perangkat keras.....	35
5.1.2 Spesifikasi perangkat lunak	35
5.2 Implementasi algoritma <i>Dempster Shafer</i>	36
5.3 Implementasi antarmuka.....	41
5.3.1 Implementasi antarmuka halaman utama	42
5.3.2 Implementasi antarmuka halaman deteksi kerusakan	42
5.3.3 Implementasi antarmuka halaman hasil deteksi kerusakan.....	43
BAB 6 PENGUJIAN DAN ANALISIS.....	44
6.1 Pengujian	44
6.2 Pengujian akurasi.....	44
6.3 Analisis	46
6.4 Analisis hasil pengujian akurasi	46
BAB 7 PENUTUP.....	47
7.1 Kesimpulan.....	47
7.2 Saran	47
DAFTAR PUSTAKA	48



DAFTAR TABEL

Tabel 2.1 Aturan kombinasi untuk m3 contoh 1.....	11
Tabel 2.2 Aturan kombinasi untuk m5 contoh 1.....	12
Tabel 4.1 Deskripsi pengguna	18
Tabel 4.2 Data komputer.....	21
Tabel 4.3 Densitas gejala terhadap kerusakan	22
Tabel 4.4 Aturan kombinasi m3	28
Tabel 4.5 Aturan kombinasi m3	30
Tabel 4.6 Aturan kombinasi m5	31
Tabel 5.1 Spesifikasi perangkat keras	35
Tabel 5.2 Spesifikasi perangkat lunak.....	35
Tabel 6.1 Hasil pengujian akurasi deteksi kerusakan.....	44



DAFTAR GAMBAR

Gambar 2.1 <i>Processor</i>	6
Gambar 2.2 <i>Hard disk</i>	7
Gambar 2.3 <i>VGA</i>	7
Gambar 2.4 <i>RAM</i>	7
Gambar 2.5 <i>Power supply</i>	8
Gambar 2.6 <i>Monitor</i>	8
Gambar 2.7 <i>Keyboard dan mouse</i>	9
Gambar 2.8 <i>Port USB</i>	9
Gambar 2.9 <i>Sound card</i>	9
Gambar 3.1 Alur metodologi penelitian	14
Gambar 3.2 Desain umum perancangan	16
Gambar 4.1 <i>Flowchart Dempster Shafer</i>	20
Gambar 4.2 Antarmuka halaman utama	32
Gambar 4.3 Antarmuka halaman deteksi kerusakan.....	33
Gambar 4.4 Antarmuka halaman hasil deteksi kerusakan	33
Gambar 5.1 Implementasi antarmuka halaman uta ma	42
Gambar 5.2 Implementasi antarmuka deteksi kerusakan.....	42
Gambar 5.3 Implementasi antarmuka hasil deteksi kerusakan.....	43

DAFTAR PERSAMAAN

Persamaan 2.1.....	10
Persamaan 2.2.....	11
Persamaan 2.3.....	46



DAFTAR SOURCE CODE

<i>Source Code 5.1 Input</i>	39
<i>Source Code 5.2 Output</i>	41



DAFTAR LAMPIRAN

LAMPIRAN A Data gejala kerusakan pada *hardware* komputer..... 50





BAB 1 PENDAHULUAN

Pada bab ini berisi pembahasan tentang hal yang menjadi latar belakang topik, permasalahan yang akan dibahas dari topik, batasan dari analisis, tujuan dan manfaat dari topik yang akan dibahas, serta sistematika dalam penulisan skripsi.

1.1 Latar belakang

Komputer adalah sebuah teknologi yang dikembangkan untuk memudahkan manusia pada suatu bentuk nyata yang berfungsi untuk membantu dalam kehidupan sehari-hari. Kerusakan komputer sering terjadi salah satu contohnya adalah kasus kerusakan pada *hardware* komputer merupakan kasus yang memerlukan bantuan seorang pakar (teknisi) dalam menyelesaikan masalah dengan mengandalkan pengetahuan yang dimilikinya. Beberapa pengguna komputer baik itu pekerja maupun seorang *businessman* membutuhkan waktu yang cukup lama untuk mendeteksi kerusakan pada *hardware* komputer karena mengalami kesulitan untuk mendeteksi kerusakan pada *hardware* komputer, bahkan teknisi sangat sering menunda pekerjaannya hanya untuk menghasilkan solusi akibat timbulnya permasalahan rusaknya *hardware* pada komputer (Rizal, 2011).

Pada penelitian ini dilakukan proses pendeteksian kerusakan pada *hardware* komputer menggunakan metode *Dempster Shafer* dimana pengguna akan menginputkan gejala-gejala yang dialami, kemudian komputer akan mengolah data tersebut sehingga menghasilkan pendeteksian kerusakan yang terdapat pada komputer tersebut. Teori *Dempster Shafer* adalah suatu teori matematika untuk pembuktian berdasarkan *belief functions & plausible reasoning* (fungsi kepercayaan dan pemikiran yang masuk akal), yang digunakan untuk mengkombinasikan potongan informasi yang terpisah (bukti) untuk mengkalkulasi kemungkinan dari suatu peristiwa. Teori ini dikembangkan oleh Arthur P. Dempster dan Glenn Shafer (Dewi, 2014).

Metode ini pernah digunakan pada penelitian yang berjudul "Pemodelan Sistem Pakar Diagnosa Penyakit Malaria dengan Metode *Dempster Shafer*". Dalam penelitian tersebut didapatkan hasil pengujian akurasi antara hasil perhitungan *Dempster Shafer* dengan hasil sistem pakar berdasarkan 20 data yang diuji adalah 90% yang menunjukkan bahwa sistem pakar ini dapat bekerja dengan baik (Fadhillah, 2016).

Latar belakang ini juga diangkat dari salah satu contoh penelitian lain dengan judul "Sistem Pakar Diagnosa Kerusakan Komputer dengan metode *Forward Chaining* dan *Certainty Factor* di Universitas Kanjuruhan Malang" dari *survey* yang dilakukan pada beberapa toko komputer dan membandingkan waktu antara mendiagnosa kerusakan komputer secara manual dengan mendiagnosa menggunakan aplikasi, didapatkan bahwa waktu yang dibutuhkan untuk mendiagnosa kerusakan komputer adalah 70% lebih efisien menggunakan aplikasi dari waktu yang dibutuhkan apabila mendiagnosa secara manual (Rizal, 2011), latar belakang dari penelitian tersebut diangkat dari banyaknya pemilik komputer

yang mengalami kesulitan untuk mengetahui sebab kerusakan komputer dan cara penanganannya menggunakan penerapan metode *Forward Chaining* dan *Certainty Factor* untuk pembangunan sistem pakarnya (Rizal, 2011). Keunggulan dari metode *Dempster Shafer* yang diterapkan dalam penelitian "Pemodelan Sistem Pakar Diagnosa Penyakit Malaria dengan Metode *Dempster Shafer*" terbukti dapat berjalan dengan baik dan memiliki presentase sebesar 90% daripada metode *Forward Chaining* dan *Certainty Factor* yaitu sebesar 70% dalam kasus mendiagnosa penyakit atau kerusakan pada sebuah objek tertentu.

Beranjak dari pentingnya permasalahan diatas terlihat perolehan tingkat akurasi menggunakan metode *Dempster Shafer* lebih tinggi daripada menggunakan metode *Forward Chaining* dan *Certainty Factor* maka untuk melakukan penelitian yang terfokus pada pendeteksian kerusakan *hardware* komputer dengan metode *Dempster Shafer* yang akan dituangkan dalam sebuah skripsi dengan judul "Deteksi kerusakan pada *hardware* komputer menggunakan metode *Dempster Shafer*".

1.2 Rumusan masalah

Sebagaimana telah dikemukakan pada latar belakang masalah, maka dapat mengidentifikasi masalah sebagai berikut:

1. Bagaimana menerapkan metode *Dempster Shafer* untuk mendeteksi kerusakan pada *hardware* komputer.
2. Bagaimana hasil akurasi untuk mendeteksi kerusakan pada *hardware* komputer dengan metode *Dempster Shafer*.

1.3 Tujuan

Tujuan penelitian yang dilakukan adalah sebagai berikut:

1. Mengimplementasikan metode *Dempster Shafer* untuk mendeteksi kerusakan pada *hardware* komputer.
2. Menguji tingkat akurasi dalam mendeteksi kerusakan pada *hardware* komputer menggunakan metode *Dempster Shafer*.

1.4 Manfaat

Harapan penulis terhadap masalah yang diangkat di latar belakang jika penelitian ini berhasil, yaitu:

1. Mempersingkat waktu teknisi untuk mendeteksi kerusakan pada *hardware* komputer.
2. Mempermudah pengguna umum komputer dengan lebih praktis untuk mengetahui penyebab dan kerusakan pada *hardware* komputer.
3. Menambah pengetahuan pengguna umum mengenai *hardware* komputer.

1.5 Batasan masalah

Dalam penyusunan skripsi ini, dapat membatasi masalah sebagai berikut:

- Data penelitian ini diambil langsung secara wawancara dari *service* komputer dan elektronik PT. Elekompetindo Solution Jl. Pahlawan Mujamil No. 36 Malang.
- Data komputer berupa form gejala dan form kerusakan.

1.6 Sistematika pembahasan

Agar penyusunan skripsi dapat tersusun secara sistematis maka sistematika penulisan skripsi ini sebagai berikut:

BAB I PENDAHULUAN

Bab I berisi penjelasan tentang latar belakang, tujuan penelitian, rumusan masalah, batasan masalah, manfaat penelitian, dan sistematika penulisan. Bab I menjadi dasar dari keseluruhan pelaksanaan penelitian mengenai deteksi kerusakan pada *hardware* komputer menggunakan *Dempster Shafer*.

BAB II LANDASAN KEPUSTAKAAN

Bab II menguraikan tentang kajian pustaka yang menjadi dasar acuan dilakukannya penelitian mengenai deteksi kerusakan pada *hardware* komputer menggunakan *Dempster Shafer* serta membahas teori-teori yang berkaitan dan menunjang dalam penyelesaian penelitian. Dasar teori ini yang diambil berasal dari jurnal, buku dan sumber referensi lainnya yang berhubungan dengan topik yang diteliti.

BAB III METODOLOGI

Bab III menjelaskan metode yang digunakan dan langkah kerja yang dilakukan dalam penulisan yang terdiri dari studi literatur, pengumpulan data, analisis kebutuhan, perancangan, implementasi, pengujian, analisis dan pengambilan kesimpulan.

BAB IV PERANCANGAN

Bab IV menjelaskan mengenai perancangan desain aplikasi secara detail tentang deteksi kerusakan pada *hardware* komputer menggunakan *Dempster Shafer*.

BAB V IMPLEMENTASI

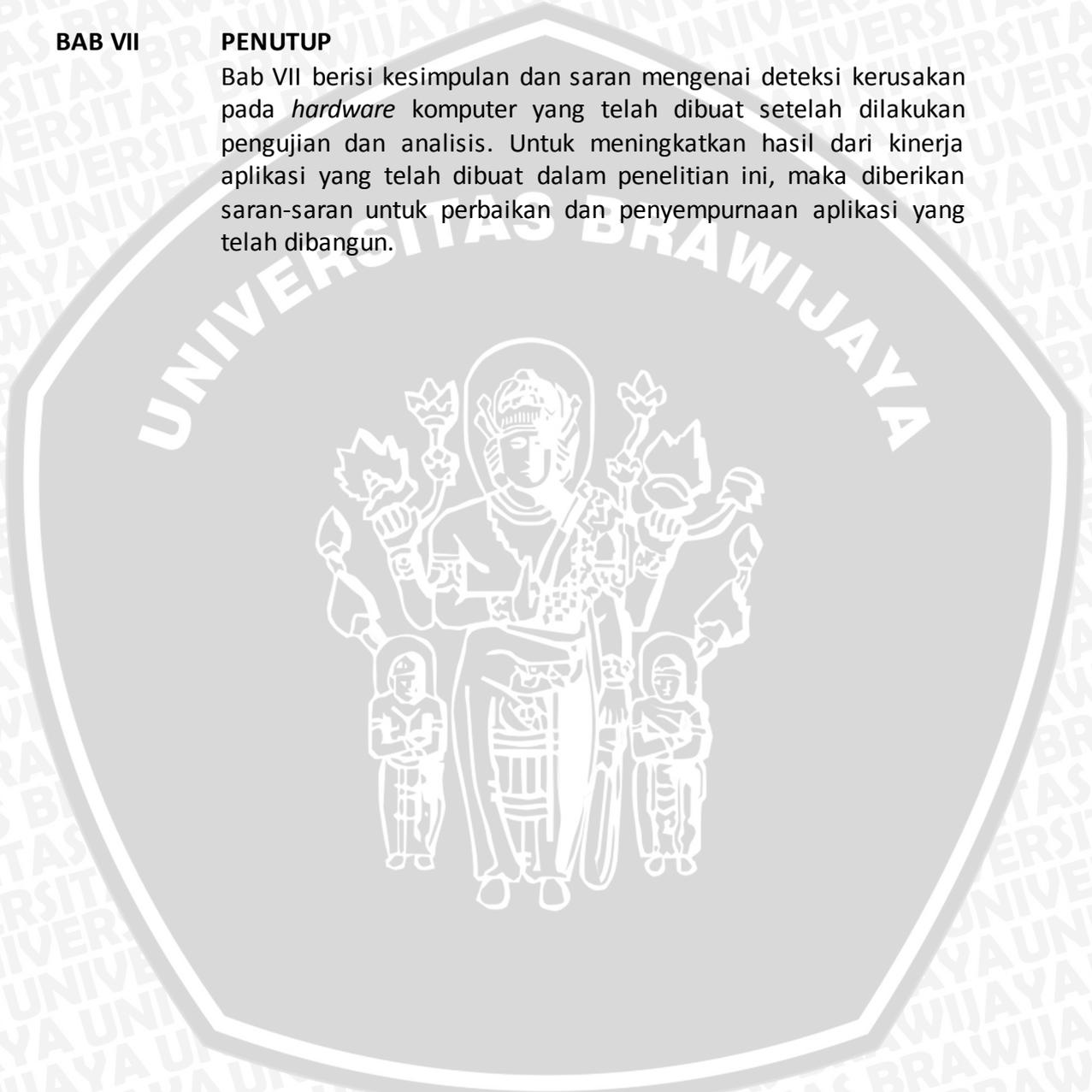
Bab V mengemukakan hasil dari penerapan dari analisis dan perancangan aplikasi yang telah dilakukan serta akan dibahas bagaimana jalannya aplikasi maupun perangkat yang dibutuhkan untuk aplikasi yang telah dibuat untuk deteksi kerusakan pada *hardware* komputer menggunakan *Dempster Shafer*.

BAB VI PENGUJIAN DAN ANALISIS

Bab VI membahas tentang pengujian pada aplikasi yang telah dibangun. Pengujian ini dilakukan secara bertahap sesuai dengan penjelasan pada Bab III. Pada bab ini juga dilakukan analisis atas hasil pengujian terhadap aplikasi yang telah direalisasikan sehingga menjawab permasalahan pada sub bab rumusan masalah.

BAB VII PENUTUP

Bab VII berisi kesimpulan dan saran mengenai deteksi kerusakan pada *hardware* komputer yang telah dibuat setelah dilakukan pengujian dan analisis. Untuk meningkatkan hasil dari kinerja aplikasi yang telah dibuat dalam penelitian ini, maka diberikan saran-saran untuk perbaikan dan penyempurnaan aplikasi yang telah dibangun.



BAB 2 LANDASAN KEPUSTAKAAN

Bab ini berisi kajian pustaka dan pembahasan tentang dasar teori yang diperlukan untuk penelitian. Kajian pustaka membahas penelitian yang telah ada sebelumnya. Landasan teori membahas teori yang diperlukan untuk menyusun penelitian yang diusulkan.

2.1 Kajian pustaka

Penelitian mengenai deteksi kerusakan pada *hardware* komputer ini bukanlah yang pertama kali dilakukan. Sebelumnya sudah ada penelitian terdahulu tentang kerusakan komputer. Didalam penelitian Rizal, (2011) yang berjudul “Sistem Pakar Diagnosa Kerusakan Komputer dengan Metode *Forward Chaining dan Certainty Factor* di Universitas Kanjuruhan Malang” dari *survey* yang dilakukan pada beberapa toko komputer dan membandingkan waktu antara mendiagnosa kerusakan komputer secara manual dengan mendiagnosa menggunakan aplikasi, latar belakang dari penelitian tersebut diangkat dari banyaknya pemilik komputer yang mengalami kesulitan untuk mengetahui sebab kerusakan komputer dan cara penanganannya menggunakan penerapan metode *Forward Chaining dan Certainty Factor* untuk pembangunan sistem pakarnya. Metode *Forward Chaining dan Certainty Factor* yang diterapkan dalam penelitian tersebut dapat memberikan solusi untuk sebab dan cara penanganan kerusakan komputer dengan waktu yang lebih efisien sebesar 70% dari waktu yang didapat bila mendiagnosa kerusakan komputer secara manual. Dan juga penelitian “Pemodelan Sistem Pakar Diagnosa Penyakit Malaria dengan Metode *Dempster Shafer*”. Dalam penelitian ini didapatkan hasil pengujian akurasi antara hasil perhitungan *Dempster Shafer* dengan hasil sistem pakar berdasarkan 20 data yang diuji adalah 90% yang menunjukkan bahwa sistem pakar ini dapat bekerja dengan baik (Fadhillah, 2016).

Dilihat dari hasil uji akurasi antara metode *Dempster Shafer* dan metode *Forward Chaining dan Certainty Factor*, terlihat bahwa akurasi metode *Dempster Shafer* memiliki presentase yang lebih tinggi dibandingkan metode *Forward Chaining dan Certainty Factor* dalam kasus mendiagnosa penyakit atau kerusakan pada sebuah objek tertentu.

Terkait dengan kedua penelitian tersebut agar dapat menjadi penguat informasi dan acuan bagi penelitian saat ini, maka untuk melakukan penelitian yang terfokus pada pendeteksian kerusakan *hardware* komputer dengan metode *Dempster Shafer* maka penulis menuangkannya dengan judul penelitian “Deteksi kerusakan pada *hardware* komputer menggunakan metode *Dempster Shafer*”. Pada penelitian ini akan lebih terfokus pada deteksi jenis kerusakan *processor*, *hard disk*, *VGA*, *RAM*, *power supply*, *monitor*, *keyboard* dan *mouse*, *port* atau *USB*, dan *sound card* pada *hardware* komputer dengan memasukkan gejala-gejala pada data form yang telah didapatkan dari pakar komputer.

2.2 Landasan teori

Landasan teori berisi tentang teori-teori yang mendukung usulan penelitian saat ini. Seperti pengertian deteksi, *hardware*, komputer, jenis-jenis *hardware* pada komputer, pengertian metode, metode *Dempster Shafer*.

2.2.1 Pengertian deteksi

Deteksi merupakan usaha menemukan dan menentukan keberadaan, anggapan, atau kenyataan dari hasil yang didapat (Gunawan, 2015).

2.2.2 Pengertian *hardware*

Hardware atau perangkat keras adalah salah satu komponen dari sebuah komputer yang sifat alatnya bisa dilihat dan diraba secara langsung atau yang berbentuk nyata, yang berfungsi untuk mendukung proses komputerisasi (Kristianto, 2013).

2.2.3 Pengertian komputer

Komputer adalah alat elektronik yang mampu melaksanakan tugas antara lain menerima *input*, memproses *input* sesuai dengan programnya, menyimpan perintah-perintah dan hasil pengolahan, menyediakan *output* dalam bentuk informasi (Rizal, 2011).

2.2.4 Jenis-jenis *hardware* pada komputer

Pada tahap ini, peneliti menjabarkan beberapa jenis-jenis *hardware* pada komputer secara umum beserta fungsinya.

2.2.4.1 *Processor (CPU) Central Processing Unit*

CPU atau *processor* bertugas melakukan proses dan mengelola informasi, *processor* biasa disebut sebagai otak dari sebuah komputer (Raka, 2011).



Gambar 2.1 *Processor*

Sumber: Packerz (2015)

2.2.4.2 *Hard disk*

Hard disk drive terdiri dari kepingan piring dengan permukaan magnetik berfungsi menyimpan informasi yang sedang tidak diproses. Sifat

hard disk adalah *non-volatile*, artinya menyimpan informasi secara permanen dan informasi tidak hilang pada saat kegalan daya (Raka, 2011).



Gambar 2.2 Hard disk

Sumber: Mapta (2004)

2.2.4.3 Video Graphic Adapter (VGA)

Video graphic card atau kartu grafis berfungsi memberikan informasi ke layar, dan digunakan untuk membuat grafik pada komputer. Sebuah *monitor CRT* atau *LCD*, menampilkan gambar dan *video* dari komputer dengan memanipulasi warna *pixel* pada layar (Raka, 2011).

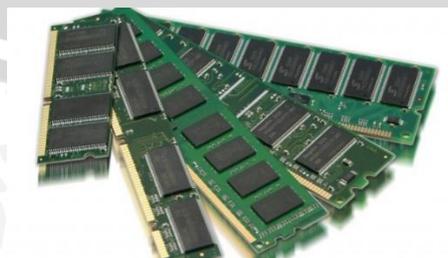


Gambar 2.3 VGA

Sumber: Anon. (2008)

2.2.4.4 Random Access Memory (RAM)

RAM atau *Random Access Memory* (memori akses acak) berfungsi menyimpan dan memproses informasi ketika komputer diperintahkan untuk bekerja. *RAM* lebih cepat daripada *hard disk*, tapi tidak stabil, hal ini berarti *RAM* hanya menyimpan informasi sementara dan semua informasi akan hilang ketika *power* komputer dimatikan (Raka, 2011).



Gambar 2.4 RAM

Sumber: Atlanta (2016)

2.2.4.5 **Power supply**

Power supply berfungsi sebagai penyuplai tegangan listrik langsung kepada komponen-komponen yang berada di dalam *casing* komputer. *Power supply* juga berfungsi untuk mengubah tegangan AC menjadi DC, karena perangkat keras komputer hanya dapat beroperasi dengan arus DC (Cholis, 2015).



Gambar 2.5 Power Supply

Sumber: Bagus (2016)

2.2.4.6 **Monitor**

Monitor adalah perangkat keras yang digunakan sebagai alat *output* data secara grafis pada sebuah *CPU*, *monitor* juga kerap disebut sebagai layar tampilan komputer (Cholis, 2015).



Gambar 2.6 Monitor

Sumber: Raqwe (2015)

2.2.4.7 **Keyboard dan mouse**

Pengertian dari *keyboard* adalah perangkat keras pada komputer yang berbentuk papan dengan berbagai macam fungsi perintah yang selanjutnya dikirim ke perangkat *CPU*. *Keyboard* terdiri dari banyak tombol ketik dengan simbol masing-masing, sedangkan *mouse* adalah sebuah perangkat keras (*hardware*) yang terhubung ke komputer baik melalui kabel atau nirkabel. *Mouse* juga berfungsi sebagai penggerak *pointer* untuk menunjukkan lokasi tertentu di layar *monitor* (Cholis, 2015).



Gambar 2.7 Keyboard dan mouse

Sumber: Anon. (2016)

2.2.4.8 Port atau (USB) Universal Serial Bus

Fungsi Port USB (Universal Serial Bus) adalah standar *bus serial* untuk perangkat penghubung, biasanya kepada komputer namun juga digunakan di peralatan lainnya seperti konsol permainan, ponsel dan *PDA* (Cholis, 2015).



Gambar 2.8 Port USB

Sumber: Rasyid (2014)

2.2.4.9 Sound card

Sound card berfungsi untuk memainkan file suara, menghasilkan suara untuk *speaker*, dan menerima suara sebagai masukan dari mikrofon. *Output audio speaker* seperti musik dan efek suara (Cholis, 2015).



Gambar 2.9 Sound card

Sumber: Babin (2005)

2.2.5 Pengertian metode

Di dalam buku yang berjudul “Analisis dan Pemodelan Sistem Pakar” menyatakan bahwa metode adalah cara, pendekatan, atau proses untuk menyampaikan informasi (Dewi, 2014).

2.2.6 Metode Dempster Shafer

Metode *Dempster-Shafer* adalah suatu teori matematika untuk pembuktian berdasarkan *belief functions* and *plausible reasoning* (fungsi kepercayaan dan pemikiran yang masuk akal), yang digunakan untuk mengkombinasikan potongan informasi yang terpisah (bukti) untuk mengkalkulasi kemungkinan dari suatu peristiwa. Teori ini dikembangkan oleh Arthur P. Dempster dan Glenn Shafer. Secara umum Teori *Dempster-Shafer* ditulis dalam suatu interval:

Belief

Belief (Bel) adalah ukuran kekuatan *evidence* (gejala) dalam mendukung suatu himpunan bagian. Jika bernilai 0 maka mengindikasikan bahwa tidak ada *evidence*, dan *Plausibility* (P1) jika bernilai 1 menunjukkan adanya kepastian yang direpresentasikan dalam bentuk Persamaan 2.1.

Plausibility

Plausibility (Pl) dinotasikan sebagai:

$$Pl(s) = 1 - Bel(\neg s) \quad (2.1)$$

Plausibility juga bernilai 0 sampai 1, maka dapat dikatakan bahwa $Bel(\neg s) = 1$, dan $Pl(\neg s) = 0$. *Plausibility* akan mengurangi tingkat kepercayaan dari *evidence*. Pada teori *Dempster-Shafer* dikenal adanya *frame of discrement* yang dinotasikan dengan θ (theta). *Frame* ini merupakan semesta pembicaraan dari sekumpulan hipotesis (Dewi, 2014).

1. Contoh kasus penyakit hewan:

Misalkan = $\theta \{A, B, C, D\}$

Dengan:

A = Alergi

B = Flu

C = Demam

D = Bronchitis

Tujuannya adalah untuk mengkaitkan ukuran kepercayaan elemen-elemen dari θ . Tidak semua *evidence* secara langsung mendukung tiap-tiap elemen. Sebagai contoh, panas mungkin hanya mendukung $\{B, C, D\}$. Untuk itu perlu adanya probabilitas fungsi densitas (m). Nilai m tidak hanya mendefenisikan elemen-elemen θ saja, tetapi juga semua himpunan bagiannya (sub-set). Sehingga jika θ berisi n elemen, maka sub-set dari θ berjumlah 2^n . Sub-set merupakan himpunan bagian dari kombinasi elemen-elemen θ , sedangkan n elemen adalah jumlah dari elemen semesta pembicaraan dari sekumpulan hipotesa pada θ . Sehingga pada contoh diatas sub-set yang bisa dihasilkan berjumlah 2. Selanjutnya harus ditunjukkan bahwa jumlah semua densitas (m) dalam sub-set θ sama dengan 1. Apabila tidak ada informasi apapun untuk memilih hipotesis, maka:

$$m(\theta) = 1,0$$

Jika kemudian diketahui bahwa panas merupakan gejala dari flu, demam, dan bronchitis dengan $m = 0,8$, maka:

$$m\{B,C,D\} = 0,8$$

$$m\{\theta\} M_Q = 1 - 0,8 = 0,2$$

Apabila diketahui X adalah sub-set dari θ , dengan m_1 sebagai fungsi densitasnya. Dengan Y juga merupakan sub-set dari θ dengan m_2 sebagai fungsi densitasnya, maka dapat di bentuk fungsi kombinasi m_1 dan m_2 sebagai m_3 dapat dilihat pada Persamaan 2.2.

$$m_3(z) = \frac{\sum_{x \cap y = z} m_1(x).m_2(y)}{1 - \sum_{x \cap y = \theta} m_1(x).m_2(y)} \quad (2.2)$$

Keterangan:

m : probabilitas densitas

xny : penyakit X irisan penyakit Y

θ : *frame of discrement*

Contoh:

Salah satu ternak pak Amri mengalami gejala demam tinggi dan lemah, mengeluarkan saliva, serta limpa besar dan rapuh. Dari diagnosa dokter penyakit yang diderita oleh sapi pak Amri adalah antraks, penyakit ngorok, penyakit mulut dan kuku dan demam tiga hari.

a. Gejala 1: Demam tinggi dan lemah

Diketahui nilai kepercayaan setelah dilakukan observasi demam tinggi dan lemah sebagai gejala penyakit antraks, penyakit ngorok dan demam tiap hari.

$$m_1 \{A,N,D\} = 0,6$$

$$m_1 \{\theta\} M_Q = 1 - 0,6 = 0,4$$

Keterangan:

A : Antraks

N : Penyakit ngorok

D : Demam tiga hari

b. Gejala 2: Mengeluarkan saliva

Kemudian diketahui juga nilai kepercayaan setelah dilakukan observasi terhadap mengeluarkan saliva sebagai gejala dari penyakit ngorok, demam tiga hari dan penyakit mulut dan kuku.

$$m_2 \{N,D,P\} = 0,7$$

$$m_2 \{\theta\} M_Q = 1 - 0,7 = 0,3$$

Maka dihitung kombinasi nilai densitas baru untuk menentukan nilai densitas m_3 dapat dilihat pada Tabel 2.1.

Tabel 2.1 Aturan kombinasi untuk m_3 contoh 1

	$\{N,D,P\} (0,7)$	$\theta M_Q (0,3)$
$\{A,N,D\} (0,6)$	$\{N,D\} (0,42)$	$\{A,N,D\} (0,18)$
$\theta M_Q (0,4)$	$\{N,D,P\} (0,28)$	$\theta M_Q (0,12)$

Sumber: Diadaptasi dari Harry (2015)

Keterangan:

A : Antraks

N : Penyakit ngorok

D : Demam tiga hari

P : Penyakit mulut dan kuku

$$m3 \{N,D\} = \frac{0,42}{1-0} = 0,42$$

$$m3 \{A,N,D\} = \frac{0,18}{1-0} = 0,18$$

$$m3 \{N,D,P\} = \frac{0,28}{1-0} = 0,28$$

$$m3 \{\theta\}M_Q = \frac{0,12}{1-0} = 0,12$$

c. Gejala 3: Limpa besar dan rapuh

Kemudian diketahui juga nilai kepercayaan setelah dilakukan observasi terhadap limpa besar dan rapuh sebagai gejala dari penyakit antraks.

$$m4 \{A\} = \frac{0,9}{1-0}$$

$$m4 \{\theta\}M_Q = 1 - 0,9 = 0,1$$

Maka dihitung kombinasi nilai densitas baru untuk menentukan nilai densitas m3 dapat dilihat pada Tabel 2.2.

Tabel 2.2 Aturan kombinasi untuk m5 contoh 1

	{A} (0,9)	θ M_Q (0,1)
{ N,D} (0,42)	θ(M_Q2) (0,378)	{ N,D} (0,042)
{A,N,D} (0,18)	{A} (0,162)	{A,N,D} (0,018)
{N,D,P} (0,28)	θ(M_Q2) (0,252)	{N,D,P} (0,028)
θ M_Q (0,12)	{A} (0,108)	θ M_Q(0,012)

Sumber: Diadaptasi dari Harry (2015)

Keterangan:

A : Antraks

N : Penyakit ngorok

D : Demam tiga hari

P : Penyakit mulut dan kuku

θ : Tidak beririsan (M_Q2)

Sehingga dapat dihitung:

$$m5 \{N,D\} = \frac{0,042}{1-(0,378 + 0,252)} = 0,113$$

$$m5 \{A,N,D\} = \frac{0,018}{1-(0,378 + 0,252)} = 0,048$$

$$m5 \{A\} = \frac{0,162 + 0,108}{1-(0,378 + 0,252)} = 0,729$$

$$m5 \{N,D,P\} = \frac{0,028}{1-(0,378 + 0,252)} = 0,075$$

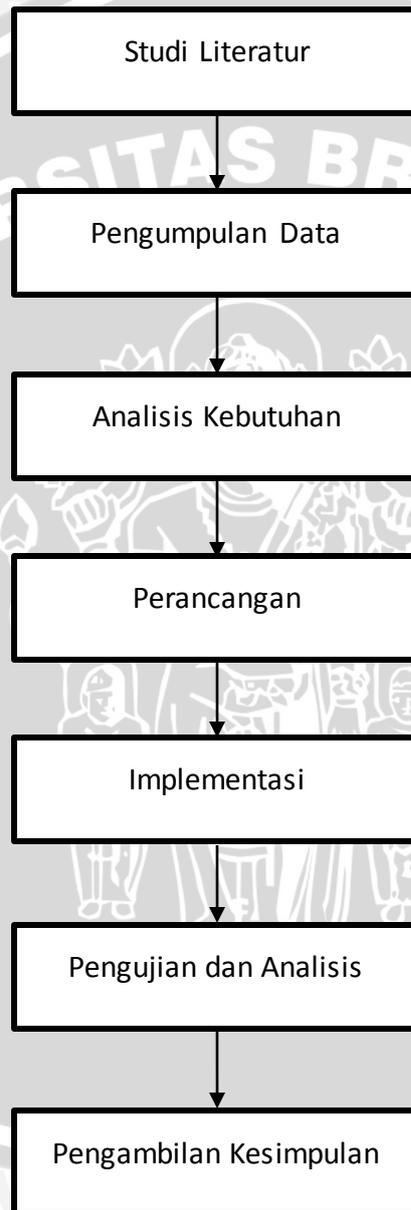
$$m5 \{\theta\}M_Q = \frac{0,012}{1-(0,378 + 0,252)} = 0,032$$

Sehingga dapat disimpulkan bahwa probabilitas densitasnya terbesar penyakit yang dialami sapi pak Amri adalah antraks dengan probabilitas densitas sebesar $0,729 \times 100\% = 72,9\%$ (Harry, 2015).



BAB 3 METODOLOGI

Bab metodologi ini membahas langkah-langkah yang dilakukan penulis dalam pembuatan aplikasi untuk mendeteksi kerusakan pada *hardware* komputer dengan, metode *Dempster Shafer*. Langkah-langkah tersebut antara lain studi literatur, pengumpulan data, analisis kebutuhan, perancangan, implementasi, pengujian dan analisis, dan pengambilan kesimpulan. Langkah-langkah dalam penelitian digambarkan dalam diagram pada Gambar 3.1.



Gambar 3.1 Alur metodologi penelitian

3.1 Studi literatur

Mempelajari literatur dari beberapa bidang ilmu yang berhubungan dengan aplikasi untuk mendeteksi kerusakan *hardware* pada komputer, diantaranya:

- > Aplikasi
- > Metode *Dempster Shafer*
- > Beragam kerusakan pada *hardware* komputer

Literatur tersebut diperoleh dari buku, jurnal, pakar dan penelitian sebelumnya.

3.2 Pengumpulan data

Pada tahap pengumpulan data penelitian yang diperlukan berupa definisi kerusakan dan gejala-gejala kerusakan pada *hardware* komputer. Sumber data diperoleh dari hasil wawancara yang dilakukan dengan seorang pakar yaitu teknisi *service* komputer PT. Elekompetindo Solution. Dari hasil wawancara dengan pakar, penulis mendapatkan data pengetahuan tentang gejala-gejala kerusakan pada *hardware* komputer, data kerusakan komputer juga didapat dari komputer yang sedang dalam proses pengerjaan PT. Elekompetindo Solution, serta penulis meminta nilai densitas pada tiap gejala yang nantinya akan digunakan untuk dilakukan perhitungan menggunakan metode *Dempster Shafer*.

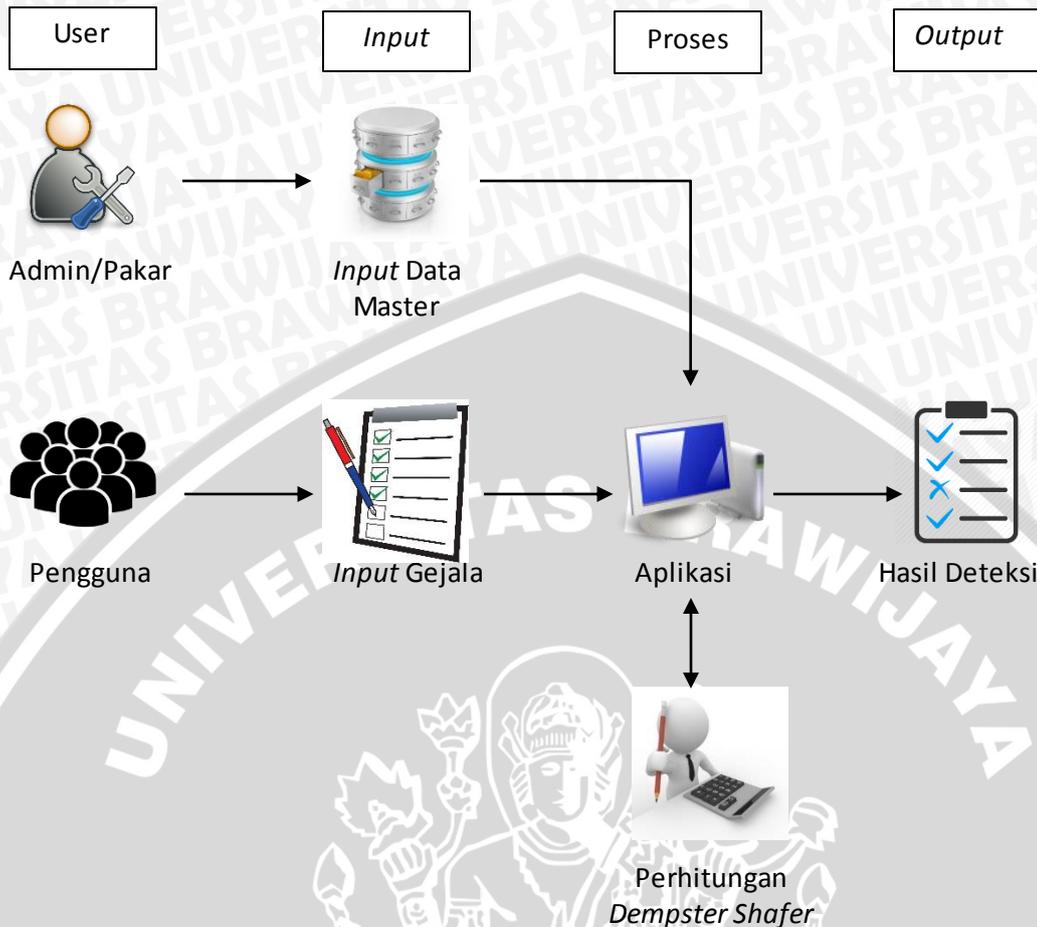
3.3 Analisis kebutuhan

Analisis kebutuhan berguna untuk mendaftar macam-macam kebutuhan yang diperlukan dalam pembuatan aplikasi. Berikut ini daftar kebutuhan dalam pembuatan aplikasi:

- > Spesifikasi kebutuhan *hardware*, meliputi:
 - Komputer dengan prosesor core i3
 - Memory 2 GB
 - *Hard disk* 320 GB
- > Spesifikasi kebutuhan *software*, meliputi:
 - Windows 7 64 bit sebagai sistem operasi.
 - XAMPP sebagai server localhost, MySQL sebagai database management sistem (DBMS).
- > Data yang diperlukan, meliputi:
 - Berbagai macam data mengenai kerusakan pada *hardware* komputer.
 - Data gejala pada masing-masing kerusakan yang terjadi pada *hardware* komputer.
 - Data komputer
 - Nilai densitas data gejala kerusakan *hardware* komputer.

3.4 Perancangan

Perancangan dibutuhkan agar penulis dapat dengan mudah melakukan proses deteksi kerusakan pada *hardware* komputer. Oleh karena itu diperoleh suatu konsep gambaran mengenai proses berjalannya aplikasi yang akan dibuat dalam penelitian ini. Pada Gambar 3.2 dapat dilihat desain aplikasi yang akan digunakan oleh penulis.



Gambar 3.2 Desain umum perancangan

Pada gambar tersebut dijelaskan bagaimana cara aplikasi bekerja. Pertama *admin* *menginputkan* data gejala dan data kerusakan beserta nilai densitasnya. Setelah data disimpan pada *database* maka data tersebut akan dijadikan acuan dari perhitungan deteksi kerusakan menggunakan metode *Dempster Shafer* pada aplikasi. Pengguna dapat melakukan deteksi kerusakan dengan *menginput* gejala yang terjadi pada *hardware* komputer kedalam aplikasi. Aplikasi kemudian memproses nilai densitas berdasarkan gejala-gejala kerusakan yang dimasukkan oleh pengguna melalui proses perhitungan densitas *Dempster Shafer*. Hasil deteksi kerusakan didapat melalui perhitungan nilai densitas tertinggi.

3.5 Implementasi

3.5.1 Identifikasi *input*

Pada proses identifikasi *input*, yang diperlukan adalah melakukan pengumpulan data atau informasi yang mendukung dalam pembangunan aplikasi pakar untuk mendeteksi dan memecahkan masalah *hardware*. Aplikasi mengajukan beberapa pertanyaan kepada pengguna, dimana pertanyaan ini adalah salah satu cara aplikasi dalam mengumpulkan informasi tentang suatu masalah yang hendak dipecahkan.

Untuk menjawab pertanyaan yang ditampilkan pada layar *monitor*, pengguna cukup memilih beberapa gejala yang dialami dengan memberi tanda *check* pada *check box*.

3.5.2 Identifikasi *output*

Aplikasi menerima masukan dari pengguna melalui berbagai pertanyaan yang diajukan oleh aplikasi, maka aplikasi memberikan kesimpulan dari jawaban pertanyaan tersebut lalu aplikasi akan mengakumulasi berbagai jawaban dari pengguna, dimana masing-masing jawaban itu akan sangat mempengaruhi kesimpulan yang didapat. Dimana aplikasi akan memberikan informasi tentang letak kerusakan yang terjadi pada *hardware* beserta penjelasan tentang kerusakan tersebut dan solusi penanganan kerusakan *hardware* pada komputer.

3.6 Pengujian dan analisis

Pada tahap ini dilakukan uji coba deteksi kerusakan pada *hardware* komputer yang bertujuan untuk menunjukkan bahwa aplikasi dapat menghasilkan deteksi kerusakan sesuai dengan yang diharapkan.

- > Pengujian dilakukan dengan cara memeriksa apakah sudah tidak terdapat *error* yang terjadi dan aplikasi telah berjalan dengan baik.
- > Pengujian dilakukan dengan cara membandingkan hasil dari pakar dengan hasil deteksi dari penerapan metode *Dempster Shafer* apakah aplikasi sudah memenuhi kesesuaian dengan hasil dari pakar atau yang diinginkan pakar.

3.7 Pengambilan kesimpulan

Kesimpulan dibuat setelah semua tahapan perancangan, pengolahan data, implementasi dan pengujian aplikasi terhadap metode yang digunakan selesai diterapkan. Isi dari kesimpulan diharapkan dapat menjadi acuan pada penelitian lain untuk mengembangkan aplikasi deteksi kerusakan pada *hardware* komputer. Selain itu, pada akhir penulisan tersedia saran yang bersifat untuk membangun, mengembangkan, menyempurnakan penelitian ini pada lain kesempatan.

BAB 4 PERANCANGAN

Bab ini membahas tentang perancangan “Deteksi kerusakan pada *hardware* komputer menggunakan metode *Dempster Shafer*”. Analisis kebutuhan aplikasi terdiri dari 6 tahapan yaitu analisis perangkat lunak, analisis perangkat keras, analisis pengguna, analisis *input*, analisis proses, analisis *output*. Perancangan algoritma, perancangan *Dempster Shafer*, dan perancangan aplikasi.

4.1 Analisis kebutuhan

4.1.1 Analisis perangkat lunak

Analisis kebutuhan perangkat lunak yang digunakan adalah sistem operasi *Microsoft Windows XP Home Edition*, *Windows Vista*, *Microsoft Windows 7*, dan *Microsoft Windows 8* dengan bahasa pemrogramannya menggunakan PHP dan databasenya menggunakan *MySQL*.

4.1.2 Analisis perangkat keras

Analisis kebutuhan perangkat keras (*hardware*) yang direkomendasikan untuk menjalankan aplikasi ini, antara lain:

1. *Processor* dual core
2. *Memory (RAM)* 1 GB
3. *VGA Card* 64 Mb
4. *Hard disk* 1 GB

4.1.3 Analisis pengguna

Aktor-aktor yang berinteraksi dengan sistem beserta deskripsinya dapat dilihat pada Tabel 4.1.

Tabel 4.1 Deskripsi pengguna

Aktor	Deskripsi Pengguna
Pengguna	Pengguna yang dapat menggunakan perancangan aplikasi untuk melihat gejala kerusakan pada <i>hardware</i> komputer.
Pakar (Teknisi)	Pengguna yang berpengalaman dan memiliki pengetahuan di bidang komputer. Terdapat macam-macam data gejala, data kerusakan, serta nilai densitas dari masing-masing gejala kerusakan pada <i>hardware</i> komputer.
Knowledge Engineer	Pengguna yang menyerap informasi dari pakar yang kemudian menerapkannya pada aplikasi.

4.1.4 Analisis input

Pakar memberikan data *input* berupa:

- > Data gejala. Data ini berupa kode beserta jenis gejala-gejala yang terjadi sebelum teridentifikasi kerusakan pada masing-masing gejala yang dialami dan nilai densitasnya.
- > Data kerusakan. Data ini berupa kode beserta jenis spesifikasi kerusakan yang diperkirakan.
- > Data aturan (*rule*) ditambahkan menurut keterkaitan gejala dan jenis kerusakan yang ditimbulkan.

4.1.5 Analisis proses

Proses dari aplikasi ini adalah proses penalaran. Aplikasi akan melakukan penalaran untuk mendeteksi jenis kerusakan yang terjadi pada *hardware* komputer sesuai dengan gejala-gejala yang dimasukkan oleh pengguna. Pada aplikasi telah diterapkan aturan (*rule*) basis pengetahuan untuk penelusuran jenis kerusakan yang terjadi pada *hardware* komputer.

4.1.6 Analisis output

Data *output* dari aplikasi ini adalah hasil proses diagnosis dari *input* menggunakan perhitungan dari metode *Dempster Shafer*. Hasil diagnosis berdasarkan fakta gejala kerusakan pada *hardware* komputer yang pengguna masukkan ketika melakukan diagnosis. Hasil *output* dari aplikasi terdiri dari gejala yang terpilih, jenis kerusakan, dan nilai kepercayaan.

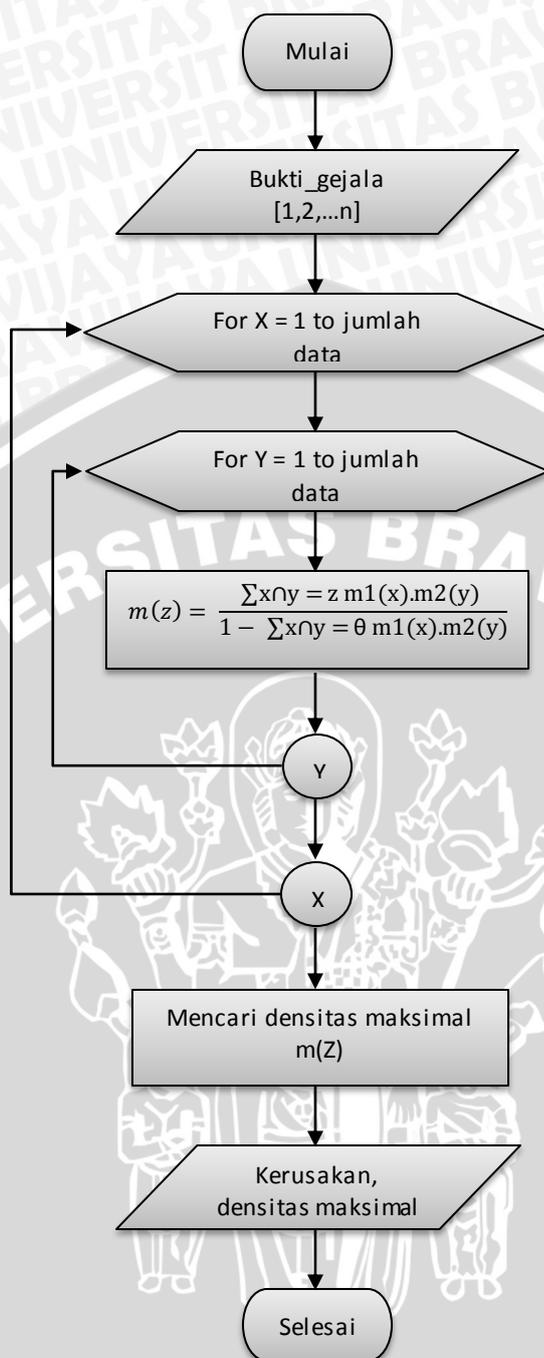
4.2 Perancangan algoritma

Perancangan algoritma digunakan untuk mendeteksi kerusakan pada *hardware* komputer. Metode *Dempster Shafer* digunakan untuk proses pengambilan kesimpulan. Tahap yang secara umum dilakukan baik oleh orang awam maupun seorang pakar dalam bidang kerusakan untuk melakukan proses deteksi adalah dengan melihat gejala kerusakan yang timbul pada *hardware* komputer tersebut, semakin detail dalam spesifik gejala yang terlihat maka semakin besar pula tingkat kepercayaannya.

Pada perancangan algoritma dilakukan perhitungan manual menggunakan *Microsoft Excel* untuk menentukan jenis kerusakan pada *hardware* komputer menggunakan metode *Dempster Shafer*.

4.3 Perancangan *Dempster Shafer*

Perancangan *Dempster Shafer* ini membahas tentang perancangan deteksi kerusakan pada *hardware* komputer menggunakan metode *Dempster Shafer*. *Flowchart* pada perancangan *Dempster Shafer* dapat dilihat pada Gambar 4.1.



Gambar 4.1 Flowchart Dempster Shafer

Proses deteksi kerusakan pada *hardware* komputer diawali dengan memasukkan data gejala kemudian dilakukan proses perhitungan menggunakan rumus kombinasi *Dempster Shafer* dilanjutkan dengan mencari nilai densitas terbesar hingga proses selesai. Akan diambil 5 data set yang dihitung perhitungan manualnya. Tabel 4.2 menjelaskan tentang data yang diambil. Tabel 4.3 menjelaskan tentang nilai densitas gejala pada komputer terhadap kerusakan yang telah ditentukan oleh pakar.

Tabel 4.2 Data komputer

Kom 1	Gejala
G13	Flashdisk tidak terdeteksi

Kom 2	Gejala
G03	Komputer sering mengalami <i>hang</i>
G09	<i>Booting</i> terlalu lama

Kom 3	Gejala
G16	Komputer tidak bisa dinyalakan
G15	Tercium seperti bau hangus pada komputer
G05	Komputer tiba-tiba mati

Kom 4	Gejala
G04	Komputer tiba-tiba mati saat bermain <i>game</i>
G19	Komputer berbunyi beep 3 kali kemudian bunyi menghilang
G20	<i>Monitor blue screen</i>

Kom 5	Gejala
G11	Penggunaan <i>cursor mouse</i> patah-patah
G18	Komputer berbunyi beep secara terus menerus
G21	Komputer tiba-tiba restart

Keterangan:

Kom 1 : Komputer Segaran

Kom 2 : Komputer Tasikmadu

Kom 3 : Komputer Kademangan

Kom 4 : Komputer Banjararum Estate

Kom 5 : Komputer Tumapel



Tabel 4.3 Densitas gejala terhadap kerusakan

Kode	Gejala	Kerusakan								
		K01	K02	K03	K04	K05	K06	K07	K08	K09
		<i>Processor</i>	<i>Hard disk</i>	<i>VGA</i>	<i>RAM</i>	<i>Power Supply</i>	<i>Monitor</i>	<i>Keyboard + Mouse</i>	<i>Port atau USB</i>	<i>Sound Card</i>
G01	CD/DVD ROM tidak dapat membaca CD/DVD	0,2				0,5				
G02	Keyboard tidak responsif							0,7	0,6	
G03	Komputer sering mengalami <i>hang</i>	0,7	0,6	0,3	0,5					
G04	Komputer tiba-tiba mati saat bermain <i>game</i>	0,6		0,7	0,5					
G05	Komputer tiba-tiba mati	0,6	0,5		0,4	0,7				

Kode	Gejala	Kerusakan								
		K01	K02	K03	K04	K05	K06	K07	K08	K09
		<i>Processor</i>	<i>Hard disk</i>	<i>VGA</i>	<i>RAM</i>	<i>Power Supply</i>	<i>Monitor</i>	<i>Keyboard + Mouse</i>	<i>Port atau USB</i>	<i>Sound Card</i>
G06	LCD mengeluarkan flek bergaris			0,5			0,6			
G07	Suara <i>speaker</i> pecah								0,4	0,5
G08	Suhu komputer tidak normal/cepat panas	0,5	0,4	0,3	0,3	0,6				
G09	<i>Booting</i> terlalu lama	0,5	0,2		0,4					
G10	Suara komputer terlalu berisik	0,3	0,5							

Kode	Gejala	Kerusakan								
		K01	K02	K03	K04	K05	K06	K07	K08	K09
		<i>Processor</i>	<i>Hard disk</i>	<i>VGA</i>	<i>RAM</i>	<i>Power Supply</i>	<i>Monitor</i>	<i>Keyboard + Mouse</i>	<i>Port atau USB</i>	<i>Sound Card</i>
G11	Penggunaan <i>cursor mouse</i> patah-patah	0,3			0,6			0,4	0,2	
G12	<i>Mouse</i> sering <i>on/off</i> dengan sendirinya ketika digunakan							0,6	0,1	
G13	<i>Flashdisk</i> tidak terdeteksi								0,5	
G14	<i>Sound off/on</i> secara tiba-tiba								0,6	0,4
G15	Tercium seperti bau hangus pada komputer	0,7	0,4		0,2	0,6				

Kode	Gejala	Kerusakan								
		K01	K02	K03	K04	K05	K06	K07	K08	K09
		<i>Processor</i>	<i>Hard disk</i>	<i>VGA</i>	<i>RAM</i>	<i>Power Supply</i>	<i>Monitor</i>	<i>Keyboard + Mouse</i>	<i>Port atau USB</i>	<i>Sound Card</i>
G16	Komputer tidak bisa dinyalakan	0,6	0,5		0,4	0,8				
G17	Komputer tidak responsif/lelet	0,6	0,4		0,2					
G18	Komputer berbunyi beep secara terus menerus				0,7					0,4
G19	Komputer berbunyi beep 3 kali kemudian bunyi menghilang			0,6						0,2
G20	<i>Monitor blue screen</i>	0,4		0,5						

Kode	Gejala	Kerusakan								
		K01	K02	K03	K04	K05	K06	K07	K08	K09
		<i>Processor</i>	<i>Hard disk</i>	<i>VGA</i>	<i>RAM</i>	<i>Power Supply</i>	<i>Monitor</i>	<i>Keyboard + Mouse</i>	<i>Port atau USB</i>	<i>Sound Card</i>
G21	Komputer tiba-tiba restart	0,3		0,1	0,4					
G22	Monitor buram/warna tidak terang			0,6			0,4		0,1	

4.4 Perhitungan manual

Implementasi metode *Dempster Shafer* dalam proses perhitungan manual deteksi kerusakan pada *hardware* komputer dibagi menjadi 3 kasus, yaitu kasus 1 dengan perhitungan 1 gejala, kasus 2 dengan perhitungan 2 gejala dan kasus 3 dengan perhitungan 3 gejala. Pada Tabel 4.2 dapat dilihat ada 5 *unit* komputer, dan setiap *unit* komputer memiliki setiap gejalanya masing-masing. Gejala yang terdapat pada tiap-tiap komputer masing-masing memiliki 1 hingga 3 gejala.

4.4.1 Perhitungan 1 gejala (kasus 1)

Untuk perhitungan kasus 1 gejala pada metode *Dempster Shafer* dapat dihitung:

➤ Kom 1

Pada kom 1 memiliki 1 gejala yaitu flashdisk tidak terdeteksi dengan kode gejala G13, dari Tabel 4.3 nilai densitas gejala terhadap kerusakan diketahui nilai densitas hanya terdapat pada kode kerusakan K08 sebesar 0,5. *Flashdisk* tidak terdeteksi adalah gejala dari kerusakan *port* atau *USB*, maka:

$$m1 \{K08\} = 0,5$$

$$m1 \{\emptyset\} = 1 - m1 \{K08\}$$

$$= 1 - 0,5$$

$$= 0,5$$

4.4.2 Perhitungan 2 gejala (kasus 2)

➤ Kom 2

• Gejala 1: Komputer sering mengalami *hang*

Pada kom 2 komputer sering mengalami *hang* dengan kode gejala G03 adalah gejala dari kerusakan *processor*, *hard disk*, *VGA*, dan *RAM*. Dari Tabel 4.3 nilai densitas tertinggi sebesar 0,7 terdapat pada gejala kerusakan *processor*, maka:

$$m1 \{K01, K02, K03, K04\} = m1 \{K01\}$$

$$m1 \{K01\} = 0,7$$

$$m1 \{\emptyset\} = 1 - m1 \{K01\}$$

$$= 1 - 0,7$$

$$= 0,3$$

• **Gejala 2: Booting terlalu lama**

Pada kom 2 memiliki gejala ke 2 yaitu *Booting* terlalu lama dengan kode gejala G09 adalah gejala dari kerusakan *processor*, *hard disk* dan *RAM*. Dari Tabel 4.3 nilai densitas tertinggi sebesar 0,5 terdapat pada gejala kerusakan *processor*, maka:

$$m_2 \{K01, K02, K04\} = m_2 \{K01\}$$

$$m_2 \{K01\} = 0,5$$

$$m_2 \{\emptyset\} = 1 - m_2 \{K01\}$$

$$= 1 - 0,5$$

$$= 0,5$$

Maka dihitung kombinasi nilai densitas baru untuk menentukan nilai densitas m_3 dapat dilihat pada Tabel 4.4.

Tabel 4.4 Aturan kombinasi m_3

m1	m2	
		$\{(K01, K02, K04)\} = 0,5$
$\{(K01, K02, K03, K04)\} = 0,7$	$\{(K01, K02, K04)\} = 0,35$	$\{(K01, K02, K03, K04)\} = 0,35$
$\{\emptyset\} = 0,3$	$\{(K01, K02, K04)\} = 0,15$	$\{\emptyset\} = 0,15$

Uraian:

a. $m_1 \{(K01, K02, K03, K04)\} \cap m_2 \{(K01, K02, K04)\} = \{(K01, K02, K04)\}$

$$0,7 * 0,5 = 0,35$$

$$\{(K01, K02, K04)\} = 0,35$$

b. $m_1 \{\emptyset\} \cap m_2 \{(K01, K02, K04)\} = \{(K01, K02, K04)\}$

$$0,3 * 0,5 = 0,15$$

$$\{(K01, K02, K04)\} = 0,15$$

c. $m_1 \{(K01, K02, K03, K04)\} \cap m_2 \{\emptyset\} = \{(K01, K02, K03, K04)\}$

$$0,7 * 0,5 = 0,35$$

$$\{(K01, K02, K03, K04)\} = 0,35$$

d. $m_1 \{\emptyset\} \cap m_2 \{\emptyset\} = \{\emptyset\}$

$$0,3 * 0,5 = 0,15$$

$$\{\emptyset\} = 0,15$$



maka, dapat dihitung:

$$m_3 \{(K01, K02, K04)\} = \frac{\{(K01, K02, K04)\} + \{(K01, K02, K04)\}}{1-0}$$

$$= \frac{0,35 + 0,15}{1-0} = 0,5$$

$$m_3 \{(K01, K02, K03, K04)\} = \frac{\{(K01, K02, K03, K04)\}}{1-0}$$

$$= \frac{0,35}{1-0} = 0,35$$

$$m_3 \{\emptyset\} = \frac{0,15}{1-0} = 0,15$$

Dari hasil perhitungan menggunakan *Dempster Shafer*, maka dapat disimpulkan komputer pengguna mengalami kerusakan pada *processor* dengan nilai kepercayaan sebesar 0,5, dan nilai ketidaktungkinannya sebesar 0,15.

4.4.3 Perhitungan 3 gejala (kasus 3)

➤ Kom 3

• Gejala 1: Komputer tidak bisa dinyalakan

Pada kom 3 komputer tidak bisa dinyalakan dengan kode gejala G16 adalah gejala dari kerusakan *processor*, *hard disk*, *RAM*, dan *power supply*. Dari Tabel 4.3 nilai densitas tertinggi sebesar 0,7 terdapat pada gejala kerusakan *power supply*, maka:

$$m_1 \{K01, K02, K04, K05\} = m_1 \{K05\}$$

$$m_1 \{K05\} = 0,8$$

$$m_1 \{\emptyset\} = 1 - m_1 \{K05\}$$

$$= 1 - 0,8$$

$$= 0,2$$

• Gejala 2: Tercium seperti bau hangus pada komputer

Pada kom 3 memiliki gejala ke 2 yaitu tercium seperti bau hangus pada komputer dengan kode gejala G15 adalah gejala dari kerusakan *processor*, *hard disk*, *RAM*, dan *power supply*. Namun dari Tabel 4.3 nilai densitas tertinggi sebesar 0,7 terdapat pada gejala kerusakan *processor*, maka:

$$m_2 \{K01, K02, K04, K05\} = m_2 \{K01\}$$

$$m_2 \{K01\} = 0,7$$

$$m_2 \{\emptyset\} = 1 - m_2 \{K01\}$$

$$= 1 - 0,7$$

$$= 0,3$$

Maka dihitung kombinasi nilai densitas baru untuk menentukan nilai densitas m3 dapat dilihat pada Tabel 4.5.

Tabel 4.5 Aturan kombinasi m3

m1	m2	
		$\{(K01, K02, K04, K05)\} = 0,7$
$\{(K01, K02, K04, K05)\} = 0,8$	$\{(K01, K02, K04, K05)\} = 0,56$	$\{(K01, K02, K04, K05)\} = 0,24$
$\{\emptyset\} = 0,2$	$\{(K01, K02, K04, K05)\} = 0,14$	$\{\emptyset\} = 0,06$

Uraian:

a. $m1 \{(K01, K02, K04, K05)\} \cap m2 \{(K01, K02, K04, K05)\} = \{(K01, K02, K04, K05)\}$

$$0,8 * 0,7 = 0,56$$

$$\{(K01, K02, K04, K05)\} = 0,56$$

b. $m1 \{\emptyset\} \cap m2 \{(K01, K02, K04, K05)\} = \{(K01, K02, K04, K05)\}$

$$0,2 * 0,7 = 0,14$$

$$\{(K01, K02, K04, K05)\} = 0,14$$

c. $m1 \{(K01, K02, K04, K05)\} \cap m2 \{\emptyset\} = \{(K01, K02, K04, K05)\}$

$$0,8 * 0,3 = 0,24$$

$$\{(K01, K02, K04, K05)\} = 0,24$$

d. $m1 \{\emptyset\} \cap m2 \{\emptyset\} = \{\emptyset\}$

$$0,2 * 0,3 = 0,06$$

$$\{\emptyset\} = 0,06$$

maka, dapat dihitung:

$$m3 \{(K01, K02, K04, K05)\} = \frac{\{(K01, K02, K04, K05)\} + \{(K01, K02, K04, K05)\} + \{(K01, K02, K04, K05)\}}{1-0}$$

$$= \frac{0,56 + 0,1 + 0,24}{1-0} = 0,94$$

$$m3 \{\emptyset\} = \frac{0,06}{1-0} = 0,06$$

• **Gejala 3 : Komputer tiba-tiba mati**

Pada kom 3 memiliki gejala ke 3 yaitu komputer tiba-tiba mati dengan kode gejala G05 adalah gejala dari kerusakan *processor, hard disk, RAM, dan power*



supply. Dari Tabel 4.3 nilai densitas tertinggi sebesar 0,7 terdapat pada gejala kerusakan *power supply*, maka:

$$m_4 \{K01, K02, K04, K05\} = m_4 \{K05\}$$

$$m_4 \{K05\} = 0,7$$

$$m_4 \{\emptyset\} = 1 - m_4 \{K05\}$$

$$= 1 - 0,7$$

$$= 0,3$$

Maka dihitung kombinasi nilai densitas baru untuk menentukan nilai densitas m_3 dapat dilihat pada Tabel 4.6.

Tabel 4.6 Aturan kombinasi m_5

m3	m4	
		$\{(K01, K02, K04, K05)\} = 0,7$
$\{(K01, K02, K04, K05)\} = 0,94$	$\{(K01, K02, K04, K05)\} = 0,658$	$\{(K01, K02, K04, K05)\} = 0,282$
$\{\emptyset\} = 0,06$	$\{(K01, K02, K04, K05)\} = 0,042$	$\{\emptyset\} = 0,018$

Uraian:

a. $m_3 \{(K01, K02, K04, K05)\} \cap m_4 \{(K01, K02, K04, K05)\} = \{(K01, K02, K04, K05)\}$

$$0,94 * 0,7 = 0,658$$

$$\{(K01, K02, K04, K05)\} = 0,658$$

b. $m_3 \{\emptyset\} \cap m_4 \{(K01, K02, K04, K05)\} = \{(K01, K02, K04, K05)\}$

$$0,06 * 0,7 = 0,042$$

$$\{(K01, K02, K04, K05)\} = 0,042$$

c. $m_3 \{(K01, K02, K04, K05)\} \cap m_4 \{\emptyset\} = \{(K01, K02, K04, K05)\}$

$$0,94 * 0,3 = 0,282$$

$$\{(K01, K02, K04, K05)\} = 0,282$$

d. $m_3 \{\emptyset\} \cap m_4 \{\emptyset\} = \{\emptyset\}$

$$0,06 * 0,3 = 0,018$$

$$\{\emptyset\} = 0,018$$

maka, dapat dihitung:

$$m5 \{(K01, K02, K04, K05)\} = \frac{0,658 + 0,282 + 0,042}{1-0} = 0,982$$

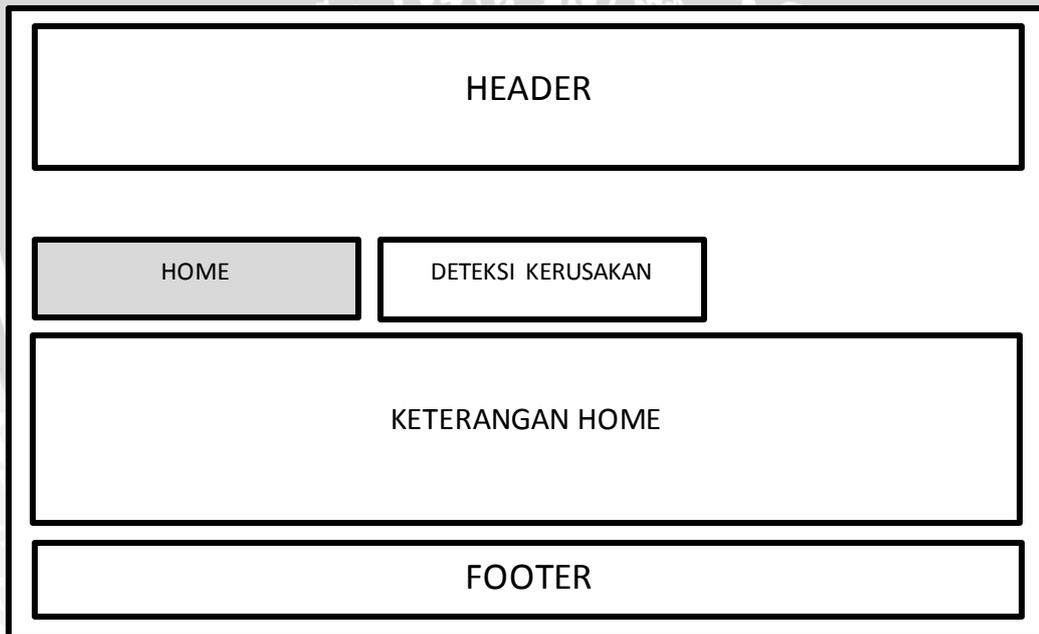
$$m5 \{\emptyset\} = \frac{0,018}{1-0} = 0,018$$

Dari hasil perhitungan menggunakan *Dempster Shafer*, maka dapat disimpulkan komputer pengguna mengalami kerusakan pada *power supply* dengan nilai kepercayaan sebesar 0,982, dan nilai ketidakmungkinannya sebesar 0,018.

4.5 Antarmuka aplikasi

Antarmuka aplikasi yang berfungsi untuk proses interaksi antara pengguna dengan aplikasi deteksi kerusakan *hardware* pada komputer, dimana aplikasi menampilkan daftar gejala-gejala yang dapat diberi *check* oleh pengguna lalu program akan melakukan proses deteksi kerusakan *hardware* pada komputer sesuai dengan gejala yang telah dipilih.

4.5.1 Antarmuka halaman utama



Gambar 4.2 Antarmuka halaman utama

Antarmuka halaman utama terdiri dari *header*, menu *home*, menu deteksi kerusakan, keterangan *home* dan *footer*. Pengguna dapat langsung melakukan deteksi kerusakan pada *hardware* komputer dengan melakukan klik pada menu deteksi kerusakan. Perancangan antarmuka halaman utama dapat dilihat pada Gambar 4.2.

4.5.2 Antarmuka halaman deteksi kerusakan

The wireframe shows a page layout with the following elements:

- HEADER:** A rectangular box at the top.
- Section Header:** A box containing the text "Deteksi Kerusakan".
- Form Area:** A large box containing:
 - Text: "Masukkan Gejala:"
 - List of symptoms:
 - > G01 with a checked checkbox
 - > G02 with an unchecked checkbox
 - > G03 with an unchecked checkbox
 - ↓
 - > G22 with an unchecked checkbox
- Submit Button:** A box containing the text "Submit".
- FOOTER:** A rectangular box at the bottom.

Gambar 4.3 Antarmuka halaman deteksi kerusakan

Pada Gambar 4.3 adalah perancangan antarmuka halaman deteksi kerusakan untuk pengguna. Pengguna dapat memilih gejala-gejala yang sudah tersedia dalam daftar kode gejala G01 hingga G22 beserta keterangan gejalanya dengan memberi tanda *check* dengan melakukan klik pada *check box* yang telah tersedia di masing-masing gejala, kemudian klik tombol *Submit* untuk mulai melakukan proses deteksi kerusakan *hardware* pada komputer.

4.5.3 Antarmuka halaman hasil deteksi kerusakan

The wireframe shows a page layout with the following elements:

- HEADER:** A rectangular box at the top.
- Section Header:** A box containing the text "Hasil Deteksi Kerusakan".
- Content Area:** A large box containing the text "KETERANGAN".
- Back Button:** A box containing the text "Back".
- FOOTER:** A rectangular box at the bottom.

Gambar 4.4 Antarmuka halaman hasil deteksi kerusakan

Pada halaman antarmuka hasil deteksi kerusakan, pengguna dapat melihat hasil dari proses deteksi sesuai dengan gejala yang telah dimasukkan, serta terdapat tombol *Back* yang berfungsi untuk kembali ke halaman deteksi kerusakan. Perancangan antarmuka halaman hasil deteksi kerusakan dapat dilihat pada Gambar 4.4.



BAB 5 IMPLEMENTASI

Pada tahap implementasi aplikasi ini dibahas bagaimana penerapan dari analisis dan perancangan aplikasi yang telah dilakukan serta dibahas bagaimana jalannya aplikasi maupun perangkat yang dibutuhkan untuk aplikasi yang telah dibuat untuk deteksi kerusakan pada *hardware* komputer menggunakan *Dempster Shafer*.

5.1 Implementasi spesifikasi

Pada tahap implementasi spesifikasi diterapkan hasil kebutuhan dan perancangan perangkat lunak yang telah dipaparkan pada bab sebelumnya untuk menjadi acuan melakukan implementasi aplikasi yang berfungsi sesuai kebutuhan. Spesifikasi aplikasi diimplementasi pada spesifikasi perangkat keras dan perangkat lunak.

5.1.1 Spesifikasi perangkat keras

Proses pembuatan aplikasi deteksi kerusakan komputer menggunakan metode *Dempster Shafer* menggunakan sebuah laptop dengan spesifikasi perangkat keras pada Tabel 5.1.

Tabel 5.1 Spesifikasi perangkat keras

Jenis perangkat keras	Spesifikasi
<i>Processor</i>	<i>Intel(R) Core(TM) i3 2.13GHz</i>
<i>Memory (RAM)</i>	<i>3072 MB</i>
<i>VGA Card</i>	<i>NVIDIA GeForce 310M 1510 MB</i>
<i>Hard disk</i>	<i>116 GB</i>

5.1.2 Spesifikasi perangkat lunak

Proses pembuatan aplikasi deteksi kerusakan komputer menggunakan metode *Dempster Shafer* menggunakan sebuah laptop dengan spesifikasi perangkat lunak pada Tabel 5.2.

Tabel 5.2 Spesifikasi perangkat lunak

Nama perangkat lunak	Spesifikasi
Sistem operasi	<i>Microsoft Windows 8 (64-bit)</i>
Bahasa pemrograman	<i>HTML dan PHP</i>
<i>Tools</i> pemrograman	<i>Dreamweaver CS 6</i>
<i>Server localhost</i>	<i>XAMPP</i>
<i>DBMS</i>	<i>MySQL</i>

5.2 Implementasi algoritma *Dempster Shafer*

Proses deteksi ini diterapkan dengan cara menginputkan fakta-fakta gejala yang terjadi pada komputer pengguna, lalu dilakukan proses deteksi kerusakan. *Input Dempster Shafer* dapat dilihat pada *Source Code* 5.1. Hasil *output* dari gejala kerusakan dapat dilihat pada *Source Code* 5.2.

No.	Source Code
1	<code><table border="1"></code>
2	<code> <tr></code>
3	<code> <td>CD/DVD ROM tidak dapat membaca CD/DVD</td></code>
4	<code> <td><input type="checkbox" value="1" name="G1"</code>
5	<code>id="G1"></code>
6	<code> <label for="G1"></label></td></code>
7	<code> </tr></code>
8	<code> <tr></code>
9	<code> <td>Keyboard tidak responsif</td></code>
10	<code> <td><input type="checkbox" value="2" name="G1"</code>
11	<code>id="G1"></code>
12	<code> <label for="G1"></label></td></code>
13	<code> </tr></code>
14	<code> <tr></code>
15	<code> <td>Komputer sering mengalami hang</td></code>
16	<code> <td><input type="checkbox" value="3" name="G1"</code>
17	<code>id="G1"></code>
18	<code> <label for="G1"></label></td></code>
19	<code> </tr></code>
20	<code> <tr></code>
21	<code> <td>Komputer tiba-tiba mati saat bermain</code>
22	<code>game</td></code>
23	<code> <td><input type="checkbox" value="4" name="G1"</code>
24	<code>id="G1"></code>
25	<code> <label for="G1"></label></td></code>
26	<code> </tr></code>
27	<code> <tr></code>
28	<code> <td>Komputer tiba-tiba mati</td></code>
29	<code> <td><input type="checkbox" value="5" name="G1"</code>
30	<code>id="G1"></code>
31	<code> <label for="G1"></label></td></code>
32	<code> </tr></code>
33	<code> <tr></code>
34	<code> <td>LCD mengeluarkan flek bergaris</td></code>
35	<code> <td><input type="checkbox" value="6" name="G1"</code>
36	<code>id="G1"></code>
37	<code> <label for="G1"></label></td></code>
38	<code> </tr></code>
39	<code></tr></code>

```
40      <td>Suara speaker pecah</td>
41      <td><input type="checkbox" value="7" name="G1"
42 id="G1">
43      <label for="G1"></label></td>
44  </tr>
45  <tr>
46      <td>Suhu komputer tidak normal/cepat panas</td>
47      <td><input type="checkbox" value="8" name="G1"
48 id="G1">
49      <label for="G1"></label></td>
50  </tr>
51  <tr>
52      <td>Booting terlalu lama</td>
53      <td><input type="checkbox" value="9" name="G1"
54 id="G1">
55      <label for="G1"></label></td>
56  </tr>
57  <tr>
58      <td>Suara komputer terlalu berisik</td>
59      <td><input type="checkbox" value="10" name="G1"
60 id="G1">
61      <label for="G1"></label></td>
62  </tr>
63  <tr>
64      <td>Penggunaan cursor mouse patah-patah</td>
65      <td><input type="checkbox" value="11" name="G1"
66 id="G1">
67      <label for="G1"></label></td>
68  </tr>
69  <tr>
70      <td>Mouse sering on/off dengan sendirinya ketika
71 digunakan</td>
72      <td><input type="checkbox" value="12" name="G1"
73 id="G1">
74      <label for="G1"></label></td>
75  </tr>
76  <tr>
77      <td>Flashdisk tidak terdeteksi</td>
78      <td><input type="checkbox" value="13" name="G1"
79 id="G1">
80      <label for="G1"></label></td>
81  </tr>
82  <tr>
83      <td>Sound off/on secara tiba-tiba</td>
84      <td><input type="checkbox" value="14" name="G1"
85 id="G1">
86      <label for="G1"></label></td>
```

```
87     </tr>
88     <tr>
89         <td>Tercium seperti bau hangus pada komputer</td>
90         <td><input type="checkbox" value="15" name="G1"
91 id="G1">
92         <label for="G1"></label></td>
93     </tr>
94     <tr>
95         <td>Komputer tidak bisa dinyalakan</td>
96         <td><input type="checkbox" value="16" name="G1"
97 id="G1">
98         <label for="G1"></label></td>
99     </tr>
100    <tr>
101        <td>Komputer tidak responsif/lelet</td>
102        <td><input type="checkbox" value="17" name="G1"
103 id="G1">
104        <label for="G1"></label></td>
105    </tr>
106    <tr>
107        <td>Komputer berbunyi beep secara terus
108 menerus</td>
109        <td><input type="checkbox" value="18" name="G1"
110 id="G1">
111        <label for="G1"></label></td>
112    </tr>
113    <tr>
114        <td>Komputer berbunyi beep 3 kali kemudian bunyi
115 menghilang</td>
116        <td><input type="checkbox" value="19" name="G1"
117 id="G1">
118        <label for="G1"></label></td>
119    </tr>
120    <tr>
121        <td>Monitor blue screen</td>
122        <td><input type="checkbox" value="20" name="G1"
123 id="G1">
124        <label for="G1"></label></td>
125    </tr>
126    <tr>
127        <td>Komputer tiba-tiba restart</td>
128        <td><input type="checkbox" value="21" name="G1"
129 id="G1">
130        <label for="G1"></label></td>
131    </tr>
132    <tr>
133        <td>Monitor buram/warna tidak terang</td>
```

```

134     <td><input type="checkbox" value="22" name="G1"
135 id="G1">
136     <label for="G1"></label></td>
137 </tr>
138 </table>
139 <input name="Submit" class="btn btn-succes"
140 type="submit" value="Submit">
141 </form>
142 </body>

```

Source Code 5.1 Input

Penjelasan:

1. Baris 1 membuat tabel *input* gejala.
2. Baris 2-138 membuat jenis *input* gejala.
3. Baris 139-142 membuat tombol *submit* gejala.

No.	Source Code
1	if (\$_POST['G1']=="1")
2	{
3	\$gab1=array(\$processor[0],\$power[0]);
4	echo "Kerusakan terjadi pada Power supply";
5	}
6	else if (\$_POST['G1']=="2")
7	{
8	\$gab2=array(\$keymouse[1],\$port[1]);
9	echo "Kerusakan terjadi pada Keyboard dan Mouse";
10	}
11	else if (\$_POST['G1']=="3")
12	{
13	\$gab3=array(\$processor[2],\$harddisk[2],\$vga[2],\$ram[
14	2]);
15	echo "Kerusakan terjadi pada Processor";
16	}
17	else if (\$_POST['G1']=="4")
18	{
19	\$gab4=array(\$processor[3],\$harddisk[3],\$ram[3]);
20	echo "Kerusakan terjadi pada VGA";
21	}
22	else if (\$_POST['G1']=="5")
23	{
24	

```
25 $gab5=array($processor[4],$harddisk[4],$ram[4],$power[4]);
26 r[4]);
27 echo "Kerusakan terjadi pada Power supply";
28 }
29
30 else if ($_POST['G1']=="6")
31 {
32 echo "Kerusakan terjadi pada Monitor";
33 }
34 else if ($_POST['G1']=="7")
35 {
36 echo "Kerusakan terjadi pada Sound Card";
37 }
38 else if ($_POST['G1']=="8")
39 {
40 echo "Kerusakan terjadi pada Power supply";
41 }
42 else if ($_POST['G1']=="9")
43 {
44 echo "Kerusakan terjadi pada Processor";
45 }
46 else if ($_POST['G1']=="10")
47 {
48 echo "Kerusakan terjadi pada Hard Disk";
49 }
50 else if ($_POST['G1']=="11")
51 {
52 echo "Kerusakan terjadi pada RAM";
53 }
54 else if ($_POST['G1']=="12")
55 {
56 echo "Kerusakan terjadi pada Keyboard dan Mouse";
57 }
58 else if ($_POST['G1']=="13")
59 {
60 echo "Kerusakan terjadi pada Port USB";
61 }
62 else if ($_POST['G1']=="14")
63 {
64 echo "Kerusakan terjadi pada Port USB";
65 }
66 else if ($_POST['G1']=="15")
67 {
68 echo "Kerusakan terjadi pada Processor";
69 }
70 else if ($_POST['G1']=="16")
71 {
```

```
72 echo "Kerusakan terjadi pada Power supply";
73 }
74 else if ($_POST['G1']=="17")
75 {
76 echo "Kerusakan terjadi pada Processor";
77 }
78 else if ($_POST['G1']=="18")
79 {
80 echo "Kerusakan terjadi pada RAM";
81 }
82 else if ($_POST['G1']=="19")
83 {
84 echo "Kerusakan terjadi pada VGA";
85 }
86 else if ($_POST['G1']=="20")
87 {
88 echo "Kerusakan terjadi pada VGA";
89 }
90 else if ($_POST['G1']=="21")
91 {
92 echo "Kerusakan terjadi pada RAM";
93 }
94 else if ($_POST['G1']=="22")
95 {
96 echo "Kerusakan terjadi pada VGA";
97 }
98 else
99 {
100 }
```

Source code 5.2 Output

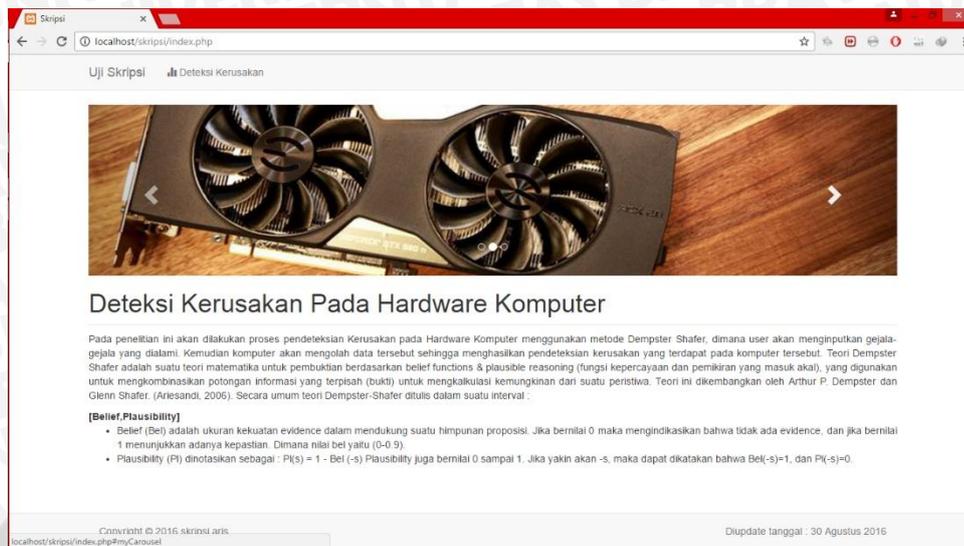
Penjelasan:

Baris 1-100 pemilihan untuk *output* hasil deteksi kerusakan dari *input* yang telah dipilih.

5.3 Implementasi antarmuka

Antarmuka aplikasi berfungsi untuk memudahkan pengguna agar dapat berinteraksi dengan aplikasi deteksi kerusakan pada *hardware* komputer. Implementasi antarmuka sesuai pada tahap perancangan yang telah dibuat pada bab sebelumnya terdiri dari tiga halaman yaitu, antarmuka halaman utama, antarmuka halaman deteksi kerusakan dan antarmuka halaman hasil deteksi kerusakan.

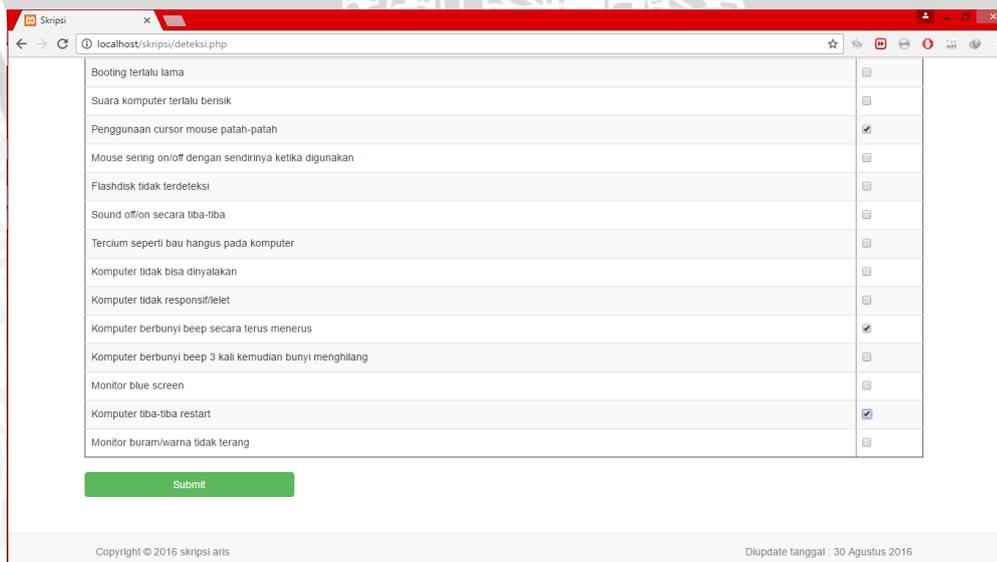
5.3.1 Implementasi antarmuka halaman utama



Gambar 5.1 Implementasi antarmuka halaman utama

Antarmuka halaman utama terdiri dari menu uji skripsi atau *home* pada bab perancangan, menu deteksi kerusakan dan menu uji skripsi merupakan *header* pada bab perancangan yang telah dibuat, keterangan *home* yang berisi ringkasan mengenai deteksi kerusakan pada *hardware* komputer dan *footer* yang berupa tahun pembuatan dan tanggal diperbaruinya aplikasi. Pengguna dapat langsung melakukan deteksi kerusakan pada *hardware* komputer dengan melakukan klik pada menu deteksi kerusakan yang akan mengarahkan pengguna ke halaman deteksi kerusakan. Implementasi antarmuka halaman utama dapat dilihat pada Gambar 5.1.

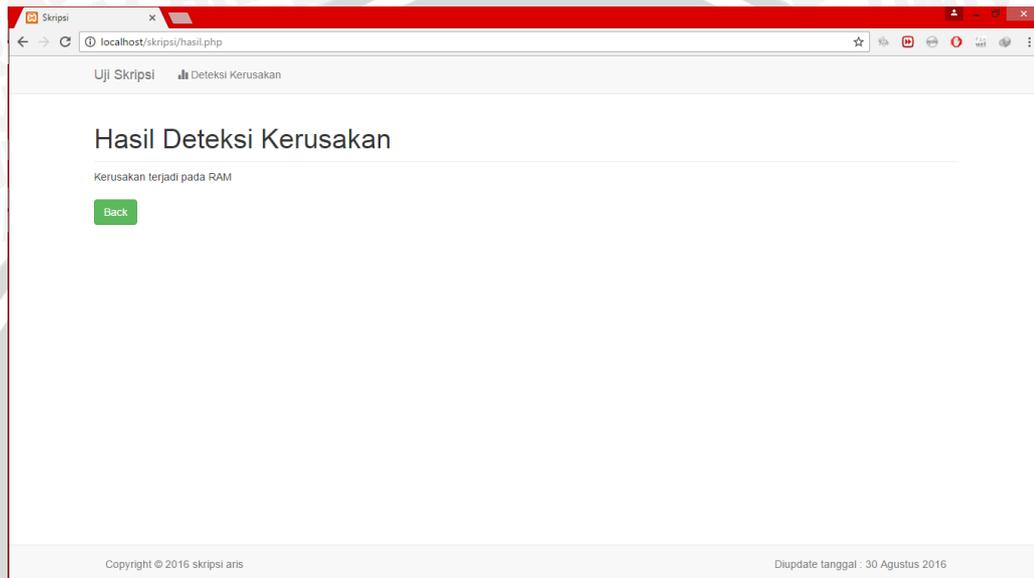
5.3.2 Implementasi antarmuka halaman deteksi kerusakan



Gambar 5.2 Implementasi antarmuka halaman deteksi kerusakan

Pada Gambar 5.2 adalah implementasi antarmuka halaman deteksi kerusakan untuk pengguna. Pengguna dapat memilih gejala-gejala yang sudah tersedia dalam daftar kode gejala G01 hingga G22 beserta keterangan gejalanya sesuai pada perancangan yang telah dibuat pada bab sebelumnya dengan memberi tanda *check* lalu melakukan klik pada *check box* yang telah tersedia di masing-masing gejala, kemudian klik tombol *Submit* untuk mulai melakukan proses deteksi kerusakan *hardware* pada komputer.

5.3.3 Implementasi antarmuka halaman hasil deteksi kerusakan



Gambar 5.3 Implementasi antarmuka halaman hasil deteksi kerusakan

Pada implementasi antarmuka halaman hasil deteksi kerusakan, pengguna dapat melihat hasil dari proses deteksi sesuai dengan gejala yang telah dimasukkan, serta terdapat tombol *Back* yang berfungsi untuk kembali ke halaman deteksi kerusakan. Implementasi antarmuka halaman hasil deteksi kerusakan dapat dilihat pada Gambar 5.3.

BAB 6 PENGUJIAN DAN ANALISIS

Pada bab ini membahas tentang pengujian pada aplikasi yang telah dibangun. Pengujian ini dilakukan secara bertahap sesuai dengan penjelasan pada Bab III dan juga dilakukan analisis atas hasil pengujian terhadap aplikasi yang telah direalisasikan sehingga menjawab permasalahan pada sub bab rumusan masalah.

6.1 Pengujian

Pada tahap ini dilakukan proses pengujian akurasi untuk membuktikan seberapa besar tingkat akurasi aplikasi deteksi kerusakan pada *hardware* komputer menggunakan *dempster shafer*.

6.2 Pengujian akurasi

Pengujian akurasi diterapkan untuk membuktikan kesesuaian aplikasi dalam memberikan *output* yang berupa jenis kerusakan dari gejala yang telah dipilih. Data yang diuji berjumlah 12 sampel data analisis pakar. Hasil deteksi yang diperoleh dari data yang telah diuji aplikasi dibandingkan dengan hasil analisis pakar. Proses pengujian akurasi dilakukan dengan cara memasukkan data fakta gejala ke dalam aplikasi yang telah dibuat, kemudian aplikasi akan menghitung menggunakan metode *dempster shafer* untuk menghasilkan jenis kerusakannya. Hasil akurasi dari data yang telah diuji dari 12 sampel data dapat dilihat pada Tabel 6.1.

Tabel 6.1 Hasil pengujian akurasi deteksi kerusakan

No.	Data Komputer	Fakta Gejala	Data Kerusakan Sebenarnya	Data Kerusakan Hasil Perhitungan	Akurasi
1	Kom 1	Flashdisk tidak terdeteksi	Port atau USB	Port atau USB	Ya
2	Kom 2	Komputer sering mengalami <i>hang</i> dan <i>booting</i> terlalu lama	Processor	Processor	Ya
3	Kom 3	Komputer tidak bisa dinyalakan, tercium seperti bau hangus pada komputer, dan komputer tiba-tiba mati	Processor	Power Supply	Tidak

No.	Data Komputer	Fakta Gejala	Data Kerusakan Sebenarnya	Data Kerusakan Hasil Perhitungan	Akurasi
4	Kom 4	Komputer tiba-tiba mati saat bermain <i>game</i> , komputer berbunyi beep 3 kali kemudian bunyi menghilang, dan <i>monitor blue screen</i>	VGA	VGA	Ya
5	Kom 5	Penggunaan <i>cursor mouse</i> patah-patah, komputer berbunyi beep secara terus menerus, dan komputer tiba-tiba restart	RAM	RAM	Ya
6	Kom 6	Suara komputer terlalu berisik	Hard disk	Hard disk	Ya
7	Kom 7	LCD mengeluarkan flek bergaris dan <i>monitor</i> buram/warna tidak terang	Monitor	VGA	Tidak
8	Kom 8	Keyboard tidak responsif, <i>mouse</i> sering on/off dengan sendirinya ketika digunakan	Keyboard dan mouse	Keyboard dan mouse	Ya
9	Kom 9	Suara <i>speaker</i> pecah	Sound card	Sound card	Ya
10	Kom 10	Sound off/on secara tiba-tiba	Port atau USB	Port atau USB	Ya
11	Kom 11	Komputer tidak responsif/lelet	Processor	Processor	Ya
12	Kom 12	CD/DVD ROM tidak dapat membaca CD/DVD dan suhu komputer tidak normal/cepat panas	Power Supply	Power Supply	Ya

Hasil akurasi yang mengindikasikan keterangan Tidak adalah *output* dari data kerusakan hasil perhitungan *dempster shafer* yang hasilnya tidak sesuai dengan data kerusakan sebenarnya dari pakar, namun hasil akurasi yang mengindikasikan keterangan Ya yang artinya *output* dari data kerusakan hasil

perhitungan *dempster shafer* sama atau sesuai dengan data kerusakan sebenarnya dari pakar. Berdasarkan Tabel 6.1 dapat dihitung nilai hasil akurasi dari metode *dempster shafer* untuk mendeteksi kerusakan pada *hardware* komputer. Proses perhitungan untuk mencari nilai akurasi direpresentasikan dalam bentuk Persamaan 2.3.

$$\text{Nilai akurasi} = \frac{\text{data akurat}}{\text{jumlah data}} \times 100\% \quad (2.3)$$

$$\text{Nilai akurasi} = \frac{10}{12} \times 100\% = 83\%$$

Berdasarkan 12 data yang telah diuji dapat disimpulkan bahwa akurasi aplikasi sebesar 83% yang menunjukkan bahwa aplikasi ini dapat berfungsi dengan baik sesuai dengan data kerusakan sebenarnya dari pakar. Nilai persentase 83% diperoleh dari pembagian data akurat 10 dari 12 jumlah data.

6.3 Analisis

Analisis bertujuan untuk mendapatkan kesimpulan dari hasil pengujian aplikasi deteksi kerusakan pada *hardware* komputer menggunakan metode *dempster shafer* yang telah dilakukan. Proses analisis yang dilakukan adalah analisis hasil pengujian akurasi.

6.4 Analisis hasil pengujian akurasi

Hasil pengujian akurasi pada aplikasi menunjukkan adanya kesesuaian dan ketidaksesuaian hasil deteksi kerusakan pada *hardware* komputer. Proses analisis terhadap pengujian akurasi deteksi kerusakan pada *hardware* komputer menggunakan metode *dempster shafer*, ketidaksesuaian dari hasil deteksi kerusakan perhitungan *dempster shafer* dengan data kerusakan sebenarnya disebabkan oleh keunikan jenis data yang dialami pada kom 3 yang jika dihitung nilai densitasnya dari kombinasi perhitungan 3 gejala menggunakan metode *dempster shafer* menghasilkan kerusakan terjadi pada *power supply* dan data kerusakan yang diproses dalam perhitungan *dempster shafer* bernilai angka yang hasilnya bernilai pasti, karena jika diteliti lagi dari jenis gejala kerusakan yang dialami pada kom 3 terdapat 2 jenis kombinasi kerusakan yaitu *processor* dan *power supply* dari 3 gejala kerusakan. Ketidakesuaian hasil deteksi kerusakan perhitungan *dempster shafer* dengan data kerusakan sebenarnya dari pakar juga dipengaruhi atas keputusan pakar dalam meneliti kerusakan utama pada *hardware* komputer berdasarkan pengalaman pakar selama menjadi pakar komputer.

BAB 7 PENUTUP

Pada bab ini berisi kesimpulan dan saran mengenai deteksi kerusakan pada *hardware* komputer yang telah dibuat setelah dilakukan pengujian dan analisis. Untuk meningkatkan hasil dari kinerja aplikasi yang telah dibuat dalam penelitian ini, maka diberikan saran-saran untuk perbaikan dan penyempurnaan aplikasi yang telah dibangun.

7.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, penulis dapat memberikan kesimpulan sebagai berikut:

1. Metode *Dempster Shafer* dapat menghasilkan *output* berupa nilai kerusakan yang terjadi pada *hardware* komputer, menggunakan *inputan* berupa banyaknya gejala dan besar densitas dari masing-masing gejala.
2. Tingkat akurasi metode *Dempster Shafer* untuk menghitung nilai kerusakan pada *hardware* komputer sebesar 83% dengan *inputan* jumlah data sebanyak 12.

7.2 Saran

Berdasarkan kesimpulan dari penelitian yang telah diperoleh, penulis mendapatkan saran yang bersifat membangun untuk perkembangan penelitian, antara lain:

1. Penelitian lebih lanjut dapat dilakukan dalam menentukan nilai densitas dari tiap gejala sehingga akurasi menjadi lebih optimal.
2. Aplikasi dapat dikembangkan seiring bertambahnya waktu jika ditemukan adanya gejala dan kerusakan baru pada *hardware* komputer.

DAFTAR PUSTAKA

- Anonym, 2016. *Keyboard dan mouse*. [image online] Tersedia di: <<http://plugable.com/products/usb-kmb1>> [Diakses 17 Juni 2016]
- Anonym, 2008. *VGA*. [image online] Tersedia di: <<http://www.pcstats.com/articleview.cfm?articleid=2256&page=2>> [Diakses 25 Mei 2016]
- Atlanta, 2016. *RAM*. [image online] Tersedia di: <<atlanta2016.medicaldosimetry.org>> [Diakses 25 Mei 2016]
- Babin, P., 2005. *Sound card* [image online] Tersedia di: <<http://www.bcot1.com/soundcard.html>> [Diakses 17 Juni 2016]
- Bagus, 2016. *Power supply*. [image online] Tersedia di: <<http://tech.dbagus.com/tips-memilih-power-supply-komputer-yang-tepat>> [Diakses 17 Juni 2016]
- Cholis, 2015. Jenis-jenis perangkat keras komputer. Tersedia di: <<http://perangkatkeraskomputer.net>> [Diakses 27 Maret 2016]
- Dewi, 2014. Implementasi Metode *Dempster Shafer* pada Sistem Pakar untuk Diagnosa Jenis-jenis Penyakit *Diabetes Melitus*. Tersedia di: <<eprints.dinus.ac.id>> [Diakses 31 Januari 2016]
- Fadhillah, 2016. Pemodelan Sistem Pakar Diagnosa Penyakit *Malaria* dengan Metode *Dempster Shafer*. Universitas Brawijaya Malang. hal: 4-7. Tersedia melalui: Ruang baca FILKOM.
- Gunawan, 2015. Deteksi Kata pada Kalimat dalam bentuk Ucapan dengan Algoritma *Hashing N-Gram*. hal: 37-38.
- Harry, 2015. Rancang Bangun Sistem Pakar Diagnosa Penyakit Sapi dengan Metode *Dempster Shafer* Berbasis Web. hal: 12-16.
- Kristianto, 2013. *Hardware dan Software*. Universitas Narotama Surabaya. Tersedia di: <<http://santirianingrum.dosen.narotama.ac.id>> [Diakses 8 Februari 2016]
- Mapta, 2004. *Harddisk*. [image online] Tersedia di: <<http://www.pcstats.com/articleview.cfm?articleid=2256&page=2>> [Diakses 25 Mei 2016]
- Packerz, 2015. *Processor*. [image online] Tersedia di: <<http://packerz.uk/computing-terminology-explained-what-is-a-processor>> [Diakses 25 Mei 2016]
- Raka, 2011. *Macam-macam hardware komputer* <<http://jaringankomputer.org/perangkat-keras-komputer-dan-macam-macam-hardware-komputer>> [Diakses 27 Maret 2016]

Raqwe, 2015. *Monitor*. [image online] Tersedia di: <<http://www.raqwe.com/acer-introduced-large-format-computer-monitor>> [Diakses 17 Juni 2016]

Rasyid, F., 2014. *Port USB*. [image online] Tersedia di: <<http://segiempat.com/tips-dan-cara/teknologi/komputer/port-standar-komputer>> [Diakses 17 Juni 2016]

Rizal, 2011. Sistem Pakar Diagnosa Kerusakan Komputer dengan Metode *Forward Chaining* dan *Certainty Factor* di Universitas Kanjuruhan Malang. Tersedia di: <stmik-ikmi-cirebon.net/e-journal> [Diakses 31 Januari 2016]



LAMPIRAN A DATA GEJALA KERUSAKAN PADA HARDWARE KOMPUTER

No.	Gejala	Diagnosa pakar
1	<i>Flashdisk</i> tidak terdeteksi	<i>Port</i> atau <i>USB</i>
2	Komputer sering mengalami <i>hang</i> <i>Booting</i> terlalu lama	<i>Processor</i>
3	Komputer tidak bisa dinyalakan Tercium seperti bau hangus pada komputer Komputer tiba-tiba mati	<i>Processor</i>
4	Komputer tiba-tiba mati saat bermain <i>game</i> Komputer berbunyi beep 3 kali kemudian bunyi menghilang <i>Monitor blue screen</i>	<i>VGA</i>
5	Penggunaan <i>cursor mouse</i> patah-patah Komputer berbunyi beep secara terus menerus Komputer tiba-tiba restart	<i>RAM</i>
6	Suara komputer terlalu berisik	<i>Hard disk</i>
7	<i>LCD</i> mengeluarkan flek bergaris Monitor buram/warna tidak terang	<i>Monitor</i>
8	<i>Keyboard</i> tidak responsif <i>Mouse</i> sering <i>on/off</i> dengan sendirinya ketika digunakan	<i>Keyboard</i> dan <i>mouse</i>
9	Suara <i>speaker</i> pecah	<i>Sound card</i>
10	<i>Sound off/on</i> secara tiba-tiba	<i>Port</i> atau <i>USB</i>
11	Komputer tidak responsif/lelet	<i>Processor</i>
12	<i>CD/DVD ROM</i> tidak dapat membaca <i>CD/DVD</i> Suhu komputer tidak normal/cepat panas	<i>Power Supply</i>