

Penerapan Sistem Pakar Untuk Mendeteksi *Troubleshooting* Pada Hardware Laptop Lenovo Menggunakan Metode *Certainty Factor*

Leo Josten Sagala ^{*}, Darjat Saripurna, S.Kom., M.Kom ^{**}, Guntur Syahputra, S.Kom., M.Kom ^{*}.

^{#1}Program Studi Sistem Informasi, STMIK Triguna Dharma

^{#2,3}Program Studi Sistem Informasi, STMIK Triguna Dharma

Article Info

Article history:

Keyword:

Certainty Factor, hardware laptop Lenovo, sistem pakar

ABSTRACT

Pada masa saat ini yaitu masa dimana teknologi adalah salah satu penunjang manusia dalam hal melaksanakan kegiatan ataupun rutinitas sehari-hari nya dalam hal mengakses informasi dan yang paling mendasar terutama dalam membantu setiap pekerjaan dalam bidang komputerisasi.

Salah satu permasalahan yang sering terjadi di ruang lingkup pengguna laptop lenovo adalah masalah kerusakan pada laptop tersebut dikarenakan banyak pengguna yang hanya tau menggunakan tapi tidak memiliki pengetahuan di bagian komponen maupun kerusakan yang terjadi di laptop Lenovo tersebut. Bagian atau komponen hardware laptop lenovo dalam jangka waktu tertentu akan mengalami perubahan fisik maupun kerusakan yang menyebabkan laptop lenovo tersebut harus diperbaiki oleh karena itu, sangat dianjurkan bagi pengguna/user untuk mengetahui cara merawat dan memberikan pertolongan pertama ketika laptopnya bermasalah, sebelum memutuskan untuk menyerahkannya ke tempat servis. Informasi yang diharapkan dapat mengatasi kerusakan yang terjadi, bahkan buku manual yang disertakan pun tidak dapat mengakomodasi terhadap semua kemungkinan kerusakan, oleh karena itu dirasakan perlu dibuat sebuah sistem yang dapat membantu memecahkan permasalahan kerusakan pada Hardware laptop lenovo khususnya pada laptop lenovo sistem tersebut adalah sistem pakar.

Copyright © 2020 STMIK Triguna Dharma.

All rights reserved.

Corresponding Author :

Nama : Leo Josten Sagala

Program Studi : Sistem Informasi STMIK Triguna Dharma

E-Mail : leosagala620@gmail.com

1. PENDAHULUAN

Sistem Pakar merupakan sebuah sistem yang mampu mengidentifikasi sebuah permasalahan dengan menggunakan keahlian seorang pakar yang telah ditanamkan kedalam sebuah sistem dengan menggunakan algoritma tertentu. Sistem pakar adalah sebuah sistem yang dibangun dengan berbasis komputer yang menggunakan beberapa pengetahuan, fakta dan teknik penalaran dalam memecahkan suatu permasalahan yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut. Implementasi sistem pakar ini sangat banyak digunakan untuk kepentingan komersial karena sistem pakar dapat dipandang cara penyimpanan pengetahuan pakar dalam bidang tertentu kedalam program komputer dan dirancang sedemikian rupa sehingga dapat memberikan keputusan dan melakukan penalaran secara cerdas [3].

Metode *Certainty Factor* dapat digunakan untuk menggambarkan tingkat keyakinan seorang pakar terhadap masalah yang sedang dihadapi. Saat ini ada dua model yang sering digunakan untuk mendapatkan tingkat keyakinan [4]. Metode *Certainty Factor* merupakan metode di dalam sistem pakar yang tepat dalam pemakaian

aplikasi ini sebagai pengukur kepastian dari troubleshooting pada hardware laptop Lenovo, dalam hal ini ada beberapa yang menjadi tujuan penelitiannya antara lain ialah:

1. Untuk mendeteksi masalah kerusakan pada *hardware* Laptop Lenovo dengan menggunakan Sistem Pakar.
2. Untuk menerapkan metode *Certainty Factor* untuk mendeteksi *Troubleshooting* pada *hardware* laptop lenovo.
3. Untuk merancang dan membangun aplikasi system pakar untuk mendeteksi kerusakan pada Laptop Lenovo dengan menggunakan metode *Certainty Factor*.
4. Untuk menerapkan aplikasi sistem pakar, kerusakan pada Laptop Lenovo dengan menggunakan metode *Certainty Factor*.

Berdasarkan deskripsi di atas maka penelitian ini diberikan sebuah judul **Penerapan Sistem Pakar Untuk Mendeteksi Kerusakan Pada Hardware Laptop Lenovo Menggunakan Metode Certainty Factor”.**

2. KAJIAN PUSTAKA

2.1 Sistem Pakar

Aplikasi berbasis komputer yang banyak dipergunakan dalam penyelesaian permasalahan yang berkaitan dengan pemikiran ataupun keahlian seorang pakar disebut dengan Sistem pakar, yang mencoba dalam memecahkan masalah yang tidak dapat diselesaikan awam dan hanya bisa diselesaikan oleh seorang pakar dibidangnya, sistem pakar dikatakan berhasil jika mampu menghasilkan sebuah keputusan yang sama seperti yang dilakukan oleh pakar aslinya baik pada saat proses pengambilan keputusannya dan juga dari hasil keputusannya.

Mesin *Inferensi* adalah sebuah otak dari aplikasi sistem pakar. Dimana dalam mesin *inferensi* inilah kemampuan pakar ini disisipkan. Apa yang dikerjakan oleh mesin *inferensi*, didasarkan pada pengetahuan-pengetahuan yang ada dalam basis pengetahuan yang telah diambil dari seorang pakar [5].

Sistem pakar hadir menjadi pembantu atau asisten yang akan menuntun seseorang menyelesaikan permasalahan dengan dukungan data kepakaran yang disimpan dalam komputer. Dengan bantuan kepakaran, informasi dirangkum dalam *database* sebagai sumber penanganan diagnosa kerusakan sampai solusi yang akan dilakukan sebagai langkah penyelesaian permasalahan [6].

Istilah yang ada pada sistem pakar bersumber dari istilah *knowledge-based expert system*. Penyebab istilah ini muncul adalah untuk memecahkan sebuah masalah yang jarang dapat diselesaikan oleh awam [7].

Pengetahuan adalah informasi atau maklumat yang diketahui atau disadari oleh seseorang. Pengetahuan termasuk, tetapi tidak dibatasi pada deskripsi, hipotesis, konsep, teori, prinsip [8].

Pengetahuan merupakan suatu saringan atau inti sari dari informasi. Pengetahuan diklasifikasikan menjadi:

1. Pengetahuan prosedural (*procedural knowledge*), memberikan bagaimana cara dalam melakukan sesuatu.
2. Pengetahuan deklaratif (*declarative knowledge*), menjawab pertanyaan dengan jawaban yang bernilai salah atau benar.
3. Pengetahuan tacit (*tacit knowledge*), pengetahuan yang tidak bisa dijelaskan dengan bahasa.

Representasi pengetahuan dimaksudkan untuk menangkap seluruh sifat-sifat penting dalam permasalahan yang akan diselesaikan dan membuat informasi itu dapat digunakan oleh prosedur pemecahan masalah. Berikut ini adalah karakteristik dari representasi pengetahuan dalam sistem pakar:

1. Sistem harus dapat diprogram menggunakan bahasa pemrograman dan hasil dari proses tersebut disimpan ke dalam memori.
2. Dirancang dengan sedemikian rupa sehingga isinya bisa dipergunakan pada proses penalaran.

Model representasi pengetahuan merupakan suatu struktur data komputer yang bisa dimanipulasi oleh sebuah mesin *inferensi* dan melakukan pencarian dalam aktivitas pencocokan pola-pola kasus yang ditangani sistem.

2.2 Certainty Factor

Certainty Factor (CF) dikemukakan oleh Shortliffe dan Buchanan pada 1975 untuk mengakomodasikan ketidakpastian pemikiran (*Inexact reasoning*) seorang pakar. Seorang pakar, (misalnya dokter sering kali menganalisis informasi yang ada dengan ungkapan seperti “mungkin”, “kemungkinan besar”, “hampir pasti”.

Untuk mengakomodasi hal ini kita menggunakan *Certainty Factor* (CF) guna menggambarkan tingkat keyakinan pakar terhadap masalah yang dihadapi [11].

1. Metode ‘*Net Belief*’ yang diusulkan oleh E.H Shortliffe dan B.G Buchanan

$$CF(\text{rule}) = MB(H,E) - MD(H,E)$$

$$MB(H,E) = \frac{\frac{1}{\max[P(H|E), P(H)]} - P(H)}{\max[1,0] - P(H)}$$

$$MD(H,E) = \frac{\frac{1}{\min[P(H|E), P(H)]} - P(H)}{\min[1,0] - P(H)}$$

Dimana:

CF (Rule) = faktor kepastian

MB (H,E) = *measure of belief* (ukuran kepercayaan) terhadap hipotesis H, jika diberikan *evidence* E (antara 0 dan 1)

MD(H,E) = *measure of disbelief*, (ukuran ketidakpercayaan) terhadap *evidence* H, jika diberikan *evidence* E (antara 0 dan 1)

P(H) = probabilitas kebenaran hipotesis H

P(H|E) = probabilitas bahwa H benar karena fakta E

2. Dengan mewawancarai seorang pakar

Nilai CF (Rule) didapat dari interpretasi “*term*” dari pakar, yang diubah menjadi nilai CF tertentu sesuai table berikut:

dengan seorang Pakar

No	<i>Uncertain Term</i>	CF
1	<i>Definitely not</i> (pasti tidak)	-1.0
2	<i>Almost certainty not</i> (Hampir tidak pasti)	-0.8
3	<i>Probably not</i> (Kemungkinan besar tidak)	-0.6
4	<i>Maybe Not</i> (Mungkin Tidak)	-0.4
5	<i>Unknown</i> (Tidak tahu)	0.2
6	<i>Maybe</i> (Mungkin)	0.4
7	<i>Probably</i> (kemungkinan Besar)	0.6
8	<i>Almost certainty</i> (Hampir Pasti)	0.8
9	<i>Definitely</i> (Pasti)	1.0

2.2.1 Perhitungan *Certainty Factor*

Secara Umum, *rule* atau aturan direpresentasikan dalam bentuk sebagai berikut:

IF E1 AND E2.....AND En THEN H (CF Rule) Atau
IF E1 OR E2.....OREn THEN H (CF rule)

Dimana:

$E_1 \dots E_N$: Fakta-fakta (*evidence*) yang ada

H : Hipotesis atau konklusi yang dihasilkan

CF rule : Tingkat keyakinan terjadi hipotesis H akibat adanya fakta-fakta $E_1 \dots E_n$

1. *Rule* dengan *evidence* E tunggal dan Hipotesis H tunggal

IF E **THEN** H (*CF Rule*)

$CF(H,E) = CF(E) \times CF(\text{rule})$

2. *Rule* dengan *evidence* E ganda dan Hipotesis H tunggal

IF E_1 **AND** $E_2 \dots \dots$ **AND** E_N **THEN** H (*CF rule*)

$CF(H,E) = \min [CF(E_1), CF(E_2), \dots, CF(E_N)] \times CF(\text{rule})$

IF E_1 **OR** $E_2 \dots \dots$ **OR** E_N **THEN** H (*CF rule*)

$CF(H,E) = \max [CF(E_1), CF(E_2), \dots, CF(E_N)] \times CF(\text{rule})$

3. Kombinasi dua buah *rule* dengan *evidence* berbeda (E_1 dan E_2), Tetapi hipotesisnya sama.

IF E_1 **THEN** H *rule 1* $CF(H,E_1) = CF_1 = C(E_1) \times CF(\text{rule1})$

IF E_2 **THEN** H *rule 2* $CF(H,E_2) = CF_2 = C(E_2) \times CF(\text{rule2})$

$$\frac{CF_1 + CF_2(1 - CF_1)}{CF_1 + CF_2} = \begin{cases} \text{jika } CF_1 > 0 \text{ dan } CF_2 > 0 \\ 1 - \min[|CF_1|, |CF_2|] & \text{jika } CF_1 < 0 \text{ atau } CF_2 < 0 \\ CF_1 + CF_2 \times (1 + CF_1) & \text{jika } CF_1 < 0 \text{ dan } CF_2 < 0 \end{cases}$$

2.2.2 Kelebihan dan kekurangan Metode *Certainty Factor*

1. Kelebihan pada metode *Certainty Factor* adalah:

- Metode ini cocok dipakai dalam sistem pakar yang mengandung ketidakpastian.
- Dalam sekali proses perhitungan hanya dapat mengelola 2 data saja sehingga keakuratan data dapat dijaga.

2. Sedangkan kekurangan metode *Certainty Factor* adalah :

- Pemodelan ketidakpastian yang menggunakan perhitungan metode *Certainty Factor* biasanya masih diperdebatkan.

3. METODOLOGI PENELITIAN

Metode Penelitian merupakan sebuah proses atau cara ilmiah dalam mendapatkan data yang akan digunakan untuk menyelesaikan masalah dengan dengan mengadakan studi langsung kelapangan untuk mengumpulkan data. Kerangka kerja merupakan langkah-langkah yang dalam menyelesaikan permasalahan yang dibahas. Adapun kerangka kerja di dalam penelitian dapat dilihat pada table 1, berikut ini:

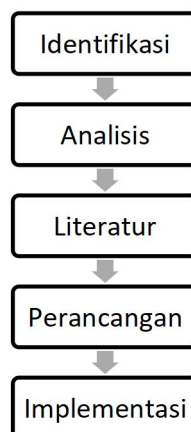


Table 1 kerangka kerja

Berikut ini adalah data berupa gejala ataupun data primer hasil wawancara dengan seorang pakar:

No	Kerusakan	Tanda-tanda (Gejala)
1	Kerusakan Ram/Slot Ram	Saat laptop Lenovo dinyalakan, tidak ada tampilan dilayar LCD
		Sesaat Laptop Lenovo dihidupkan, lampu LED menyala tapi layar tidak tampil
		Suhu Laptop Lenovo cepat sekali panas
		Laptop <i>restart</i> sendiri
		Laptop dihidupkan terdapat bunyi <i>beep</i> panjang
		Gagal <i>Booting</i> saat Laptop dihidupkan
		Sistem Operasi tidak berjalan dengan lancar
		<i>Blue Screen</i>
		Aplikasi atau <i>software</i> seringkali tidak bisa dijalankan
		Layar lapotop tidak menampilkan gambar apa apa (gelap)
2	Kerusakan Pada <i>MotherBoard</i>	Tidak ada tampilan di <i>monitor</i>
		Tidak ada lampu indikator (led) yang menyala
		Kipas <i>Fan</i> tidak berputar
		Tidak ada tampilan di <i>monitor</i>
		Lampu indikator (led) di panel depan menyala Perangkat mengalami panas berlebihan pada awal dinyalakan
		Tidak mau masuk ke dalam menu BIOS
		<i>Stuck</i> dalam menu BIOS, dan tidak masuk ke dalam Sistem Operasi
		Perangkat menyala, tetapi tidak muncul menu BIOS

3	Kerusakan Pada <i>Harddisk</i>	Perangkat mengalami panas berlebihan pada awal dinyalakan
		<i>Stuck</i> dalam menu BIOS, dan tidak masuk ke dalam Sistem Operasi
		Pada saat Laptop Lenovo dinyalakan kemudian melakukan proses <i>Post</i> setelah itu muncul pesan " <i>Operating system not found</i> "
		<i>Harddisk</i> Mengeluarkan Suara Aneh
		<i>Blue Screen</i>
		Sistem Operasi menjadi terasa lambat

Tabel 2 data hasil wawancara

Berikut adalah bagaimana menentukan Bobot nilai pakar merupakan data yang diberikan langsung oleh pakar terhadap gejala-gejala yang mendasari suatu hipotesis dari pengidentifikasian kerusakan *Hardware* Laptop Lenovo. Berikut ini pengetahuan dasar atau informasi tentang gejala kerusakan *Hardware* Laptop Lenovo dari beserta nilai MB dan MD untuk setiap gejalanya. Bobot nilai gejala diperoleh dari rumus:

$$MB(H, E) = \frac{\frac{1}{\text{Max}[P(H|E), P(H)]} - P(H)}{\text{Max}[1, 0] - P(H)}$$

$$MD(H, E) = \frac{\frac{1}{\text{Min}[P(H|E), P(H)]} - P(H)}{\text{Min}[1, 0] - P(H)}$$

MB (H,E) = Ukuran kepercayaan terhadap hipotesis h (antara 0 dan 1)
 MD (H,E) = Ukuran ketidakpercayaan terhadap hipotesis h (antara 0 dan 1)
 CF = *Factor* kepastian
 CF [H,E] = MB [H,E]-MD[H,E]

Dimana diasumsikan jumlah disini diasumsikan ada 70 responden terkait Jenis Kerusakan *Hardware* Laptop Lenovo. Dengan jumlah masing-masing tiap kerusakan adalah sebagai berikut:

K01 Kerusakan *Ram/Slot Ram* = 14

K02 Kerusakan Pada *MotherBoard* = 19

K03 Kerusakan Pada *Harddisk* = 37

Kemudian dihitung nilai premis masing-masing jenis kerusakan tersebut.

$$P(H1) = \frac{14}{70} = 0.2$$

$$P(H2) = \frac{19}{70} = 0.271$$

$$P(H3) = \frac{37}{70} = 0.529$$

Maka dihitung nilai premis H1 terhadap *Evidence* 1 (Gejala 1)

$$P(H1|E1) = \frac{10}{14} = 0.714$$

Selanjutnya dicari nilai MB dan MD nya dengan menggunakan rumus sebagai berikut:

$$MB(H1, E1) = \frac{\frac{1}{\text{Max}[P(H1|E1), P(H1)]} - P(H1)}{\text{Max}[1, 0] - P(H1)}$$

$$MB(H1, E1) = \frac{\frac{1}{\text{Max}[0.714, 0.2]} - 0.2}{\text{Max}[1, 0] - 0.2}$$

$$MB(H1, E1) = \frac{0.714 - 0.2}{1 - 0.2}$$

$$MB(H1, E1) = 0.643$$

$$MD(H, E) = \frac{\overset{1}{\text{Min}[P(H|E), P(H)]} - P(H)}{\text{Min}[1,0] - P(H)}$$

$$MD(H1, E1) = \frac{\text{Max}[0.714; 0.2] - 0.2}{\text{Max}[1,0] - 0.2}$$

$$MD(H1, E1) = \frac{0.714 - 0.2}{0 - 0.2}$$

$$MD(H1, E1) = 0$$

Jenis Kerusakan *Hardware* Laptop Lenovo Beserta Nilai MB dan MD

Kode	Tingkatan	Kode Gejala	MB	MD	CF
K01	Kerusakan Ram/Slot Ram	G01	0.643	0	0.643
		G02	0.821	0	0.821
		G03	0.464	0	0.464
		G04	0.732	0	0.732
		G05	0.464	0	0.464
		G06	0.643	0	0.643
		G07	0.464	0	0.464
		G08	0.821	0	0.821
		G09	0.464	0	0.464
		G10	0.821	0	0.821
		G11	0.464	0	0.464
K02	Kerusakan Pada <i>MotherBoard</i>	G12	0.494	0	0.494
		G13	0.567	0	0.567
		G14	0.422	0	0.422
		G15	0.567	0	0.567
		G16	0.494	0	0.494
		G17	0.205	0	0.205
		G18	0.278	0	0.278
		G19	0.133	0	0.133

K03		G20	0.000	0	0.000
		G21	0.205	0	0.205
		G22	0.278	0	0.278
	Kerusakan Pada <i>Harddisk</i>	G19	0.369	0	0.369
		G21	0.427	0	0.427
		G17	0.484	0	0.484
		G23	0.197	0	0.197
		G09	0.599	0	0.599
		G07	0.713	0	0.713

Table 3 perhitungan MB dan MD

Rule Kerusakan hardware Laptop Lenovo

Kode	Tingkatan	Kode Gejala	CF	CF User	CF user*CF Pakar
K01	Kerusakan Ram/Slot Ram	G01	0.643	0	0
		G02	0.821	0	0
		G03	0.464	0	0
		G04	0.732	0.6	0.4392
		G05	0.464	0	0
		G06	0.643	0	0
		G07	0.464	0.8	0.3712
		G08	0.821	0	0
		G09	0.464	0.8	0.3712
		G10	0.821	0.8	0.6568
		G11	0.464	0	0
K02	Kerusakan Pada <i>MotherBoard</i>	G12	0.494	0	0
		G13	0.567	0	0
		G14	0.422	0	0
		G15	0.567	0	0

		G16	0.494	0	0
		G17	0.205	0	0
		G18	0.278	0	0
K03	Kerusakan Pada <i>Harddisk</i>	G19	0.133	0	0
		G20	0.000	0	0
		G21	0.205	0	0
		G19	0.369	0	0
		G21	0.427	0	0
		G17	0.484	0	0
		G23	0.197	0	0
		G09	0.599	0.8	0.4792
		G07	0.713	0.8	0.5704

Tabel 4 rule kerusakan hardware laptop lenovo

Hasil menentukan CF *combine* untuk Kerusakan Ram/*Slot* Ram adalah sebagai berikut:

$$CF(h,e)_{g4,g7} = Cfgejala4 + Cfgejala7 * (1 - CFgejala1) 0.4392 + 0.3712 * (1 - 0.4392) = 0.647$$

$$CF(h,e)_{old1,g9} = CFold1 + Cfgejala9 * (1 - CFold1) 0.64736896 + 0.544 * (1 - 0.64736896) = 0.778$$

$$CF(h,e)_{old2,g10} = CFold2 + Cfgejala10 * (1 - CFold2) 0.77826 + 0.352 * (1 - 0.77826) = 0.929$$

Untuk hasil Kerusakan Pada *MotherBoard* tidak dihitung karena tidak ada gejala terkait dengan kasus. Maka akan dilanjutkan untuk penentuan CF *combine* untuk Kerusakan Pada *Harddisk* yaitu sebagai berikut:

$$\begin{aligned}
 CF(h,e)_{g9,g7} &= Cfgejala9 + Cfgejala7 * (1 \\
 &\quad Cfgejala1) \\
 &= 0.4792 + 0.5704 * (1 - 0.479 \\
 &\quad = 0.776
 \end{aligned}$$

4. PENGUJIAN DAN IMPLEMENTASI

4.1 Kebutuhan Sistem

Sistem program ini mencakup spesifikasi kebutuhan perangkat keras (*hardware*) dan perangkat lunak (*software*). Dalam implementasi dan pengujian program didalam sistem pendukung keputusan dengan metode *Weighted Agregated Sum Product Assesment* membutuhkan 2 buah perangkat yaitu, perangkat lunak (*Software*) dan perangkat keras (*Hardware*).

4.2 Implementasi Sistem

Implementasi merupakan langkah yang digunakan untuk mengoperasikan sistem yang akan dibangun. Pada bab ini akan dijelaskan bagaimana menjalankan sistem yang telah dibangun tersebut. Di bawah ini merupakan tampilan dari implementasi sistem Pakar dengan metode *Certainty factor* dalam menentukan Mendeteksi *Troubleshooting hardware* laptop lenovo. Implementasi sistem adalah suatu prosedur yang dilakukan untuk menyelesaikan sistem yang ada dalam dokumen rancangan yang telah disesuaikan.

Sistem informasi diharapkan mampu menyediakan informasi yang berguna dan berkualitas. Informasi yang berguna dapat dinilai dari ketepatan waktunya dan relevansi dari informasinya. Kebenaran dari hasil pengolahan data yang dikerjakan secara manual pada Bab III tersebut digunakanlah Aplikasi berbasis web. Berikut di bawah tahap dari pengaplikasian sistem yang telah berhasil dibangun:

1. Rancangan Halaman Menu Utama

Menu utama adalah tampilan awal ketika memasuki sistem. Halaman ini berisi tampilan luar tentang sistem pakar untuk mendeteksi *troubleshooting Hardware* laptop lenovo. Dapat dilihat pada gambar 1 di bawah ini:



Gambar 1 Form menu utama

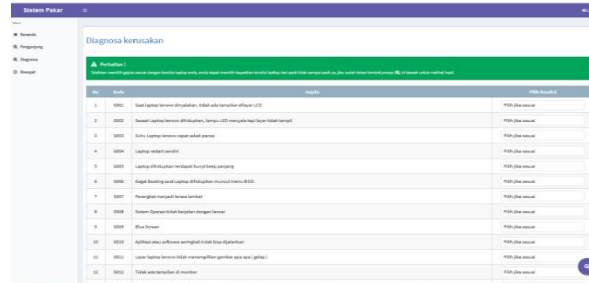
2. Form User

Form User merupakan *form* yang digunakan *user* untuk *Input* siapa saja yang telah menggunakan sistem yang dirancang ini, tetapi sebelum *User* melakukan diagnosa, mereka wajib mengisi *form* ini berikut tampilan menu *user* :

Gambar 1 Tampilan Form User

3. Form Diagnosa User

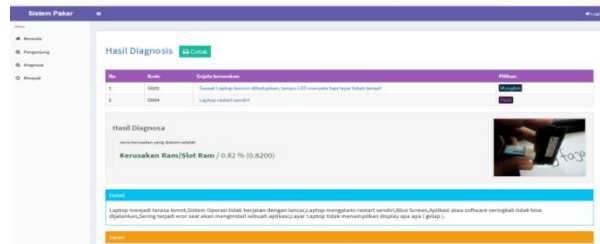
Form diagnosa digunakan oleh user. Pada form diagnosa ini user diharuskan untuk memilih gejala-gejala sesuai dengan Kerusakannya, dapat di lihat pada gambar 3 di bawah ini:



Gambar 2 Tampilan Form Diagnosa User

4. Form Hasil Deteksi

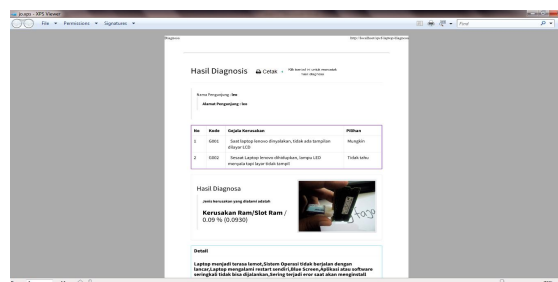
Form Hasil Deteksi ini merupakan form untuk menampilkan hasil deteksi berdasarkan gejala yang dipilih sebelumnya. dapat dilihat pada gambar 4 di bawah ini:



Gambar 3 Tampilan Form hasil deteksi

5. Form laporan user

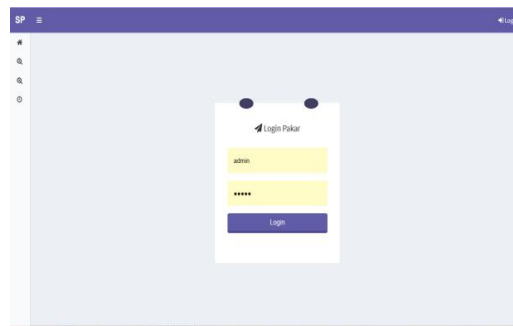
Form laporan user ini merupakan form untuk menampilkan laporan deteksi berdasarkan gejala yang sudah dipilih dan siap untuk dicetak dapat di lihat pada gambar 5 di bawah ini:



Gambar 4 Tampilan Form laporan user

6. Form Login Admin

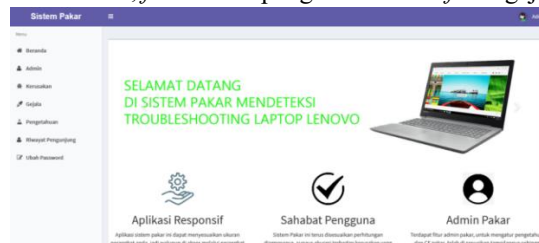
Pada bagian perancangan ini dilengkapi dengan rancangan halaman login. Halaman Login digunakan khusus untuk admin web yang dapat mengakses form Kerusakan, form gejala, form basis pengetahuan.



Gambar 6 Tampilan Form login admin

7. Halaman Admin

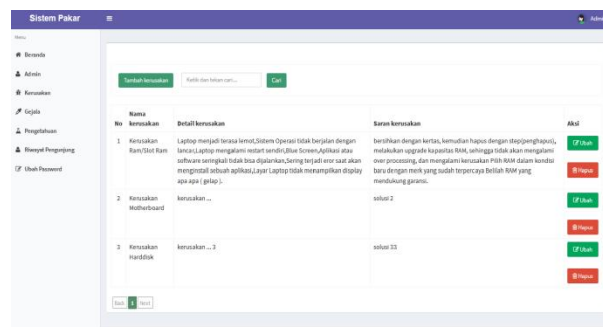
Halaman admin merupakan halaman untuk merancang bentuk desain halaman Web yang akan digunakan oleh admin untuk menuju ke *form* Kerusakan, *form* basis pengetahuan dan *form* gejala.



Gambar 7 Tampilan halaman admin

8. Form Kelola Jenis Kerusakan

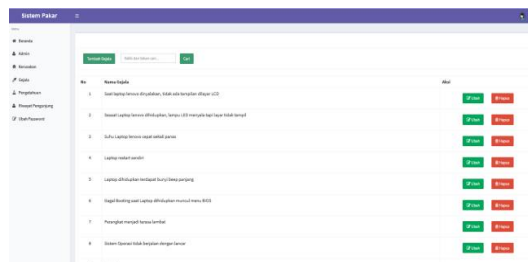
Form Kerusakan digunakan untuk melihat Kerusakan yang ada di *database*, menghapus Kerusakan, menambah Kerusakan dan mengubah Kerusakan. Berikut adalah tampilan untuk halaman *form* Kerusakan:



Gambar 8 Tampilan *form* kerusakan

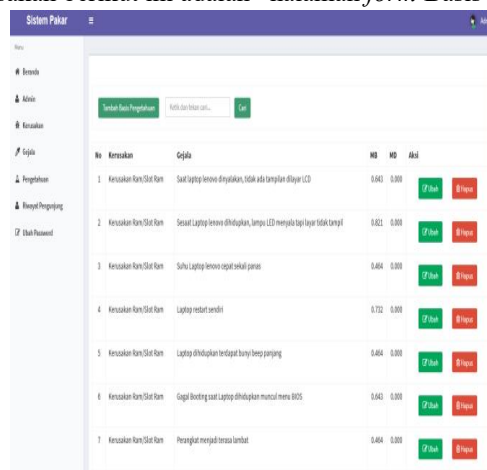
9. Form Kelola Jenis Gejala

Form gejala digunakan untuk melihat data gejala yang ada di *database*, menghapus data gejala, menambah data gejala dan mengubah data gejala. Berikut adalah tampilan untuk halaman *form* gejala:

Gambar 9 Tampilan *form* gejala

10. *Form* Kelola Basis pengetahuan

Form Basis Aturan digunakan untuk melihat data Basis Aturan yang ada di *database*, menghapus data Basis Aturan, menambah Basis Aturan dan mengubah Basis Aturan. *Form* Basis Aturan digunakan juga untuk membuat relasi antara gejala dan Kerusakan berikut ini adalah halaman *form* Basis Aturan:

Gambar 10 *form* kelola basis pengetahuan.

5. KESIMPULAN DAN SARAN

5.1 Kesimpulan

1. Dalam menentukan *Troubleshooting* pada *hardware* laptop Lenovo didapat data dari seorang pakar laptop Lenovo yang bekerja di sinar jaya komputer sehingga didapat lah kerukasakan antara lain: kerusakan pada *motherboard*, kerusakan pada *hardisk*,kerusakan pada RAM.
2. Metode *certainty factor* dapat mempermudah user laptop lenovodalam hal mendiagnosa kerusakan pada hardware laptop mereka dan juga mengefisiensikan waktu mereka tanpa harus langsung pergi ketem pat service laptop, hanya dengan menginputkan atau pun memilih jenis gejala saja mereka sudah dapat mengetahui apa yang menjadi kerusakan pada laptop mereka sehingga tidak menutup kemungkinan aplikasi sistem pakar yang dibangun ini sangat menguntungkan bagi si pengguna laptop Lenovo.
3. Aplikasi yang berbasis *web* dihasilkan dan dibangun dengan perancangan yang ditulis dalam UML dan *Interface* yang sederhana dan mudah digunakan, serta dapat membantu pengguna laptop Lenovo dalam mendeteksi kerusakan *hardware* mereka tanpa harus langsung pergi ketempat servis laptop.

5.2 Saran

1. Untuk pengembangan dimasa datang, sebaiknya menggunakan metode yang lain dalam proses menentukan kerusakan *hardware* laptop Lenovo sehingga dapat dibandingkan dan mendapatkan nilai yang lebih akurat dalam sistem pakar tersebut.
2. Pada sistem ini tampilan aplikasi masih sederhana. Untuk pengembangan selanjutnya bisa dibuat lebih menarik agar para *user* bisa lebih tertarik menggunakan aplikasi ini, yaitu buatlah aplikasi berbasis android atau lain sebagainya.
3. Untuk pengembangan dimana yang akan datang, sebaiknya aplikasi dapat digunakan bersifat *online*.

REFERENCES

- [1] ELIZABETH GINTING, "PENGARUH FAKTOR BUDAYA, SOSIAL, PRIBADI DAN PSIKOLOGIS T ERHADAP KEPUTUSAN PEMBELIAN LAPTOP MEREK "ASUS" (STUDI KASUS PADA MAHASISWA UNIVERSITAS BUDI LUHUR) PERIODE SEPTEMBER – DESEMBER 2014," *Jurnal Ekonomi dan Manajemen*, vol. 4, no. 2, 2015.
- [2] Suciati, "Penggunaan Laptop dalam Perkuliahan di Kelas Manfaat atau Mudharatkah?," *292 Jurnal Pendidikan dan Kebudayaan*, vol. 17, no. 3, 2017.
- [3] A. Irmayana, "Sistem Pakar Diagnosis Persalinan Ibu Hamil Menggunakan Metode Certainty Factor Expert System for Diagnosing Labor of Pregnant Women Using Certainty Factor Method," *Prosiding Seminar Nasional Komunikasi dan Informatika #3 Tahun*, vol. 1, no. 1, pp. 111-118, 2019.
- [4] Ayu Cantika, "sejarah-perkembangan-smartphonenovo," 2016. [Online].
- [5] H. T. Sihotang, "SISTEM PAKAR MENDETEKSI PENYAKIT KOLESTEROL PADA REMAJA DENGAN METODE CERTAINTY FACTOR (CF) BERBASIS WEB," *Vol 15 no.1 2014*.
- [6] Y. Yuliyana and A. S. R. M. Sinaga, "Sistem Pakar Diagnosa Penyakit Gigi Menggunakan Metode Naive Bayes," *Fountain of Informatics Journal*, vol. 4, no. 1, p. 19, 10 5 2019.
- [7] E. Ongko, "Perancangan Sistem Pakar Diagnosa Penyakit pada Mata, Vol.II No. 2: 10-17, 2013.
- [8] H. T. Sihotang, "SISTEM PAKAR UNTUK MENDETEKSI PENYAKIT PADA TANAMAN JAGUNG DENGAN METODE BAYES," *Vol 3 no. 1 2018*.
- [9] Y. Ramadhan Nasution, "SISTEM PAKAR DETEKSI AWAL PENYAKIT TUBERKULOSIS DENGAN METODE BAYES," vol. 1, no. 1, pp. 17-23, 2017.
- [10] Level Perdana, "SISTEM PAKAR UNTUK DIAGNOSIS PENYAKIT GINJAL DENGAN METODE FORWARD CHAINING," *Jurnal TIKomSiN*, no. ISSN : 2338-4018, 2018.
- [11] K. E. Setyaputri, A. Fadlil and D. Sunardi, "Analisis Metode Certainty Factor pada Sistem Pakar Diagnosa Penyakit THT".

- [12] Suendri, "Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle (Studi Kasus: UIN Sumatera Utara Medan)," *ALGORITMA: Jurnal Ilmu Komputer dan Informatika*, p. 1, 2018.
- [13] W. Aprianti, U. Maliha, J. Teknik Informatika, P. Negeri, T. Laut, J. A. Y. Km, P. T. Laut and K. Selatan, "SISTEM INFORMASI KEPADATAN PENDUDUK KELURAHAN ATAU DESA STUDI KASUS PADA KECAMATAN BATI-BATI KABUPATEN TANAH LAUT," 2016.
- [14] Sutejo, "Pemodelan UML Sistem Informasi Geografis Pasar Tradisional Kota Pekanbaru," *Jurnal Teknologi Informasi & Komunikasi Digital Zone*, Universitas Lancang Kuning, vol. 7, 2016.
- [15] S. Rosa, "Rekayasa perangkat lunak," Bandung, Informatika, 2017.
- [16] R. Nurmalina, J. A. Yani Km, T. Laut and K. Selatan, "Perencanaan dan Pengembangan Aplikasi Absensi Mahasiswa Menggunakan Smart Card Guna Pengembangan Kampus Cerdas (Studi Kasus Politeknik Negeri Tanah Laut)," 2017.
- [17] d. E. F. R. Rasim1), Wawan Setiawan2), "Metodologi Pembelajaran Berbasis Komputer Dalam Upaya Menciptakan Kultur Pembelajaran Berbasis Teknologi Informasi dan Komunikasi," 2, vol. 1, no. SSN:1979-9264, 2008.
- [18] N. David, M. Veronika and Y. Darnita, "RANCANG BANGUN APLIKASI TES TOEFL MENGGUNAKAN ALGORITMA QUICK SORT BERBASIS KOMPUTER," *Jurnal Pseudocode*, vol. 2, 2015.
- [19] N. E. Putri and S. Azpar, "Sistem Informasi Pengolahan Data Pendidikan Anak Usia Dini (PAUD) Terpadu Amalia Syukra Padang," *Jurnal Edik Informatika*, vol. 2, 2019.
- [20] T. Nata Lega and B. Eka Purnama, "PEMBANGUNAN SISTEM INFORMASI PERPUSTAKAAN PADA SEKOLAH MENENGAH ATAS NEGERI PUNUNG," *IJCSS - Indonesian Journal on Computer Science*, p. 1, 2019.

BIOGRAFI PENULIS

	<p>Data Diri</p> <p>Nama : Leo Josten Sagala Tempat/Tanggal Lahir : Batu bara,09 juni 1995 Jenis Kelamin : Laki-Laki Agama : Kristen Protestan Status : Belum Menikah Pendidikan Terakhir : Sekolah Menengah Atas Kewarganegaraan : Indonesia E-mail : leoagala620@gmail.com</p> <p>Pendidikan Formal</p> <ol style="list-style-type: none"> 1. Tahun 2001 - 2007 : SD Swasta Advent Tj.Sari Riau 2. Tahun 2007-2010 : SMP Swasta Advent Tj.Sari Riau 3. Tahun 2010-2013 : SMA Methodist 1 Medan
	<p>Darjat Saripurna, S.Kom., M.Kom Beliau merupakan dosen tetap STMIK Triguna Dharma, serta aktif sebagai dosen pengajar khusus pada bidang ilmu Sistem Informasi</p>
	<p>Guntur Syahputra, S.Kom., M.Kom Beliau merupakan dosen tetap di STMIK Triguna Dharma serta aktif sebagai dosen pengajar khusus bidang ilmu Sistem Informasi</p>