

SISTEM PAKAR MENDETEKSI KERUSAKAN EMMC PADA HANDPHONE VIVO DENGAN MENGGUNAKAN METODE CERTAINTY FACTOR

**Muhammad Dwi Galih Rizki¹, Riah Ukur Ginting², Dini M.Hutagalung³, Rianto
Sitanggang⁴**

Program Studi Sistem Informasi Universitas Sari Mutiara Indonesia Medan
E-Mail : massgalih23@gmail.com

Abstrak

Tujuan penulisan ini adalah untuk menyelesaikan permasalahan bagaimana merancang suatu sistem pakar yang berbasis komputer untuk mendeteksi kerusakan EMMC pada Handphone Vivo secara cepat dan akurat. Terbatasnya jumlah pakar yang dapat memahami dalam memperbaiki EMMC akan memakan waktu yang cukup lama untuk mengidentifikasi kerusakan EMMC pada Handphone Vivo. Dengan metode Certainty Factor menentukan kerusakan yang menyediakan solusi-solusi yang spesifik. Yang merupakan salah satu cabang dari Artificial Intellident yang membuat pengguna secara luas pengetahuan yang khusus untuk penyelesaian masalah tingkat manusia yang pakar dalam bidang tertentu dari metode Certainty Factor yang menentukan hasil penelitian di lapangan. Maka dihasilkanlah sebuah sistem yang dapat mengidentifikasi kerusakan EMMC pada Handphone Vivo dengan menggunakan metode Certainty Factor.

Kata Kunci : Sistem Pakar, deteksi kerusakan EMMC

I. PENDAHULUAN

1.1 Latar Belakang

Pada masa sekarang ini perkembangan teknologi dan komunikasi dari waktu ke waktu dirasakan semakin meningkat dengan pesat, terlebih lagi perkembangan di bidang teknologi komputer yang mendorong penggunaan dan pemanfaatan perkembangan teknologi tersebut secara luas diberbagai bidang dan aspek kehidupan, terutama dibidang kecerdasan buatan. Bidang yang menggunakan kecerdasan buatan antara lain adalah system pakar.

Handphone merupakan alat telekomunikasi elektronik dua arah yang bisa dibawa kemana-mana dan memiliki kemampuan untuk mengirimkan pesan berupa suara. Dalam keseharian kini manusia hampir tidak bisa lepas dari handphone. Apalagi dengan semakin berkembangnya handphone sehingga handphone memiliki berbagai macam fungsi sekaligus, bukan hanya sebagai alat komunikasi saja namun telah berkembang

menjadi alat dengan fungsi lainnya seperti sebagai media hiburan, media bisnis, dan sebagainya yang bisa digunakan untuk melakukan banyak hal. Akan tetapi hanya sedikit pengguna handphone yang mengerti akan gejala-gejala kerusakan handphone seperti kerusakan EMMC pada handphone, sehingga akhirnya handphone pun menjadi lambat dan keluar dari aplikasi secara tiba-tiba saat dibuka, dan apabila dibiarkan terus menerus seperti ini handphone akhirnya akan mati sendiri dan tidak bisa digunakan kembali. Ketidaktahuan pengguna handphone terhadap kerusakan pada EMMC dapat diatasi dengan Sistem Pakar.

1.2 Rumusan Masalah

Berdasarkan uraian diatas pada latar belakang maka yang menjadi rumusan masalah adalah:

1. Bagaimana sistem menjalankan metode *Certainty Factor* untuk mendeteksi kerusakan pada EMMC Handphone Vivo ?

2. Bagaimana sistem dapat mendeteksi gangguan yang dapat membantu pengguna maupun teknisi dalam mendeteksi gangguan kerusakan pada EMMC Handphone Vivo ?

1.3 Manfaat Penelitian

Adapun manfaat penelitian yang dapat diperoleh dari penulisan skripsi ini antara lain :

1. Sistem ini dapat digunakan oleh siapa saja selaku *user* untuk mendeteksi kerusakan EMMC Handphone Vivo berdasarkan gejala-gejala yang ada.
2. Dapat memberikan informasi sebagai deteksi awal kerusakan pada EMMC Handphone Vivo sesuai dengan pengetahuan sistem pakar yang dibuat.
3. Dapat memberikan informasi bagi *user* untuk melakukan tindakan yang harus diambil jika mengetahui ada beberapa besar kemungkinan terjadi kerusakan pada EMMC Handphone Vivo.

II. TINJAUAN PUSTKA

2.1 Sistem Pakar

Sistem pakar merupakan cabang dari *Artificial Intelligence* (AI) yang membuat penggunaan secara luas *knowledge* yang khusus untuk penyelesaian masalah tingkat manusia yang pakar. Seorang pakar adalah orang yang mempunyai keahlian dalam bidang tertentu, yaitu pakar yang mempunyai *knowledge* atau kemampuan khusus yang orang lain tidak mengetahui atau mampu dalam bidang yang dimilikinya. Ketika sistem pakar dikembangkan pertama kali sekitar tahun 70-an sistem pakar hanya berisi *knowledge* yang eksklusif. Namun demikian sekarang ini istilah sistem pakar sudah digunakan untuk berbagai macam sistem yang menggunakan teknologi sistem pakar. Teknologi sistem pakar ini meliputi bahasa sistem pakar, pemrograman dan perangkat keras yang dirancang untuk membantu

pengembangan dan pembuatan sistem pakar.

Pengetahuan dari suatu sistem pakar mungkin dapat direpresentasikan dalam sejumlah cara. Salah satu metode yang paling umum untuk pengetahuan adalah dalam bentuk tipe aturan (*rule*) **IF...THEN** (Jika...Maka)..

2.2 EMMC (*Embedded Multi Media Controller*)

EMMC merupakan salah satu tempat penyimpanan data pada handphone smartphone android di mana dalam EMMC tersebut di letakan semua data seperti *Operating system*, IMEI, GPRS, Kamera dan data data lainnya. Kalau di dalam suatu handphone smartphone android tidak memiliki EMMC atau EMMC itu rusak maka handphone tersebut tidak bisa digunakan.

2.3 *Certainty Factor* (Faktor Kepastian)

Faktor kepastian (CF) pertama kali diperkenalkan oleh *Shortliffe* dan Buchanan pada tahun 1975 untuk mengakomodasi ketidakpastian pemikiran (*inexact reasoning*) seorang pakar. Seorang pakar (misalnya dokter) seringkali menganalisis informasi yang ada dengan ungkapan seperti “mungkin”, “kemungkinan besar”, “hampir pasti”. Untuk mengakomodasi hal ini maka digunakan *certainty factor* (CF) guna menggambarkan tingkat keyakinan pakar terhadap masalah yang sedang dihadapi. Defenisi menurut David McAlister, *Certainty factor* adalah suatu metode untuk membuktikan apakah suatu fakta itu pasti ataukah tidak pasti yang berbentuk matrik yang biasanya digunakan dalam mendeteksi sesuatu yang belum pasti. Didalam metode ini terdapat dua cara yang bisa dilakukan dalam mendapatkan tingkat keyakinan (CF) dari sebuah *rule*, yaitu :

1. Metode *Net belief* yang diusulkan oleh E. H. Shortfille dan B. G. Bachnan

$$CF (Rule) = MB (H,E) \left\{ \begin{array}{l} 1 \\ \max [P(H|E), P(H)] - P(H) \end{array} \right. \quad P(H)=1$$

$$MD(H,E) = \begin{cases} 1 & P(H)=0 \\ \frac{\max [P(H|E), P(H)] - P(H)}{\max [1.0] - P(H) \text{ lainnya}} & \end{cases}$$

Dimana :

- CF (Rule) : Factor kepastian
- MB (H,E) : *Measure of Beliefe* (ukuran kepercayaan) terhadap hipotesis H, jika diberikan *evidence* E (antara 0 dan 1)
- MD (H,E) : *Measure of Disbeliefe* (ukuran ketidakpercayaan) terhadap *evidence* E (antara 0 dan 1)
- P(H) : Probabilitas kebenaran hipotesis H
- P(H|E) : Probabilitas bahwa H benar karena fakta E

2. Dengan cara mewawancarai seorang pakar Nilai CF (Rule) didapat dari interpretasi "term" dari pakar, yang diubah menjadi nilai CF.

2.3.1 Perhitungan Certainty Factor

Secara umum, *rule* direpresentasikan dalam bentuk sebagai berikut :

IF E1 AND E2.....AND En THEN H (CF rule)

Atau

IF E1 OR E2.....OR En THEN H (CF rule)

Dimana :

- E1....En: Fakta-fakta (*evidence*) yang ada
- H : Hipotesis atau konklusi yang dihasilkan
- CF Rule : Tingkat keyakinan terjadinya hipotesis H akibat adanya fakta-fakta E1....En

1. *Rule* dengan *evidence* E tunggal dan hipotesis H tunggal
IF E₁AND E₂ AND E_nTHEN H (CF Rule)
 $CF(H,E) = CF(E) \times CF(C)$
2. *Rule* dengan *evidence* E ganda dan hipotesis H tunggal
IF E₁AND E₂ AND E_nTHEN H (CF Rule)
 $CF(H,E) = \min [CF(E_1), CF(E_2), \dots, CF(E_n)] \times CF(rule)$
IF E₁OR E₂ OR E_nTHEN H
 $CF(H,E) = \max [CF(E_1), CF(E_2), \dots, CF(E_n)] \times CF(rule)$
3. Kombinasi dua buah *rule* dengan *evidence* berbeda (E₁ dan E₂), tetapi hipotesisnya sama.

IF E₁THEN H Rule 1 $CF(H,E_1) = CF_1 = C(E_1) \times CF(rule 1)$

IF E₂THEN H Rule 2 $CF(H,E_1) = CF_2 = C(E_2) \times CF(rule 2)$

$CF_1 + CF_2 (1 - CF_1)$ Jika $CF_1 > 0$ dan $CF_2 > 0$

$CF(CF_1, CF_2) \quad CF_1 + CF_2$ Jika $CF_1 > 0$ dan $CF_2 < 0$

$1 - \min[CF_1, CF_2]$

$CF_1 + CF_2(1 + CF_1)$ Jika $CF_1 < 0$ dan $CF_2 < 0$

2.3.2 Kelebihan Dan Kekurangan Metode Certainty Factor

Kelebihan metode *Certainty Factor* adalah :

- a. Metode ini cocok dipakai dalam sistem pakar yang mengandung ketidakpastian.
- b. Dalam sekali proses perhitungan hanya dapat mengelola dua data saja sehingga keakuratan data dapat terjaga.

2.4 UML (Unified Modeling Language)

Unified Modeling Language merupakan salah satu alat bantu yang dapat digunakan dalam bahasa pemrograman yang berorientasi objek, saat ini UML akan menjadi *standart* masa depan bagi industri pengembangan sistem/perangkat lunak yang beorientasi objek sebab pada dasarnya UML, digunakan oleh banyak perusahaan raksasa seperti IMB, *Microsoft*, dan sebagainya.

UML adalah metode pengembangan perangkat lunak (sistem informasi) dengan metode grafis serta merupakan bahasa untuk visualisasi, spesifikasi, konstruksi serta dokumentasi dari komponen-komponen perangkat lunak dan digunakan untuk pemodelan bisnis. Pemodelan UML berarti menggambarkan yang ada dalam dunia nyata kedalam bentuk yang dapat dipahami dengan menggunakan UML. Berikut proses dalam membuat gambaran UML (*Unified Modeling Language*).

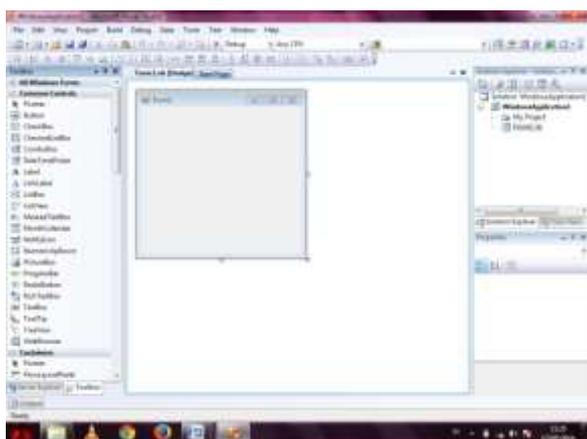
2.5 Microsoft Visual Basic

Cikal bakal bahasa pemrograman *Visual Basic* adalah *BASIC (Beginer's AllPurpose Symbolic Instruction Code)* yang diciptakan sekitar tahun 1964 oleh Profesor Jhon Kemeny dan Thomas Kurtz.

Bahasa pemrograman *BASIC* sangat cepat perkembangannya, mudah dipelajari dan digunakan bagi programmer pemula terdapat pada Gambar 1.

1. Pada tahun 1975, *Microsoft* mengeluarkan satu produk pertamanya yaitu *Microsoft Basic* dan produk suksesnya adalah *Quick Basic (Qbasic)* yang
2. banyak digunakan pada *computer* pribadi atau PC (*Personal Computer*). Produk yang sangat *revolusioner* adalah saat *Microsoft* mengeluarkan versi terbaru dari bahasa *Basic* yaitu *Visual Basic 1.0*. bahasa pemrograman ini sangat populer karena mudah dipelajari dan bekerja di lingkungan grafis atau lebih dikenal dengan istilah GUI (*Graphical User Interface*).
3. Pada tahun 1993, *Microsoft Visual Basic 3.0* dirilis yang disertai dengan diikutsertakannya kontrol data standar. Kontrol ini menyediakan akses *database* dalam aplikasi-aplikasi pengkodean minimal. Kehadiran bahasa pemrograman *Visual Basic 3.0* ini juga disertai dengan mesin *database Jet* versi 1.1 (*Jet Engine*). Mesin *Jet* ini dipakai untuk konektivitas *database* melalui DAO (*Data Access Object*) atau kontrol data. Keberadaan mesin *Jet* ini diteruskan hingga sekarang dengan hadirnya mesin *Jet* versi 4.0
4. *Visual Basic 4.0* diluncurkan pada tahun 1996 dan menjadi sebuah revisi yang menentukan. Perangkat lunak ini mengakomodasi kecanggihan pengembangan perangkat lunak dengan memasukkan teknologi OLE (*Object Linking and Embedding*) dan kemampuan untuk membuat obyek. Versi ini juga mendukung kontrol *costum 32-bit* yang disebut OCX. Kehadiran *Visual Basic 4.0* juga ditandai dengan adanya fasilitas pemrograman *database RDO (Remote Data Objects)*. RDO adalah sebuah metode akses data yang dirancang dari bawah ke atas untuk menggantikan DAO. Pustaka *ActiveX 32-bit* ini lebih cepat dan lebih kecil dari DAO, dan dirancang bersama sebuah hierarki obyek yang sama dengan ODBC API.
5. Pada tahun 1997, *Visual Basic 5.0* dirilis dengan mendukung standar *COM Microsoft* dan memungkinkan pembuatan kontrol-kontrol *ActiveX*. Versi ini bisa dibilang menjadi sebuah terobosan nyata karena sekarang pengembang program bisa membuat kontrol dan DLL mereka sendiri dengan menggunakan bahasa *Visual Basic*. Bahasa pemrograman *Visual Basic 5.0* mempunyai kemampuan membuat bahasa interpreter atau *compile*, dengan *ferforma* yang lebih baik.
6. Pada tahun 1998, *Visual Basic 6.0* derilis dengan memuat metode baru yang telah ditingkatkan kemampuannya untuk berinteraksi dengan *SQL Server*. Salah satu fasilitas terbaru yang dibawa *Visual Basic 6.0* adalah ADO (*ActiveX Data Object*) versi 2.0. ADO dirancang bagi kinerja dalam perusahaan besar dan *aplikasi internet* serta merupakan salah satu komponen utama UDA (*Universal Data Acces*). Versi terbaru dari ADO adalah 2.5, yang disertakan bersama sistem operasi *Windows 2000*.
7. Perkembangan *Microsoft.NET* pertama kali dipaparkan oleh *Microsoft* pada bulan Juli 2000 dala PDC (*Profesional Develover Conference*) di orlando, Amerika Serikat. Rentang waktu dua tahun dirasa cukup bagi *Microsoft* untuk mengembangkan *software* ini sampai pada tahapan dengan diluncurkannya versi betanya. Sekitar bulan Februari 2002 secara resmi *Microsoft* merilis *Visual Studio.NET 2002* terdiri dari bahasa pemrograman *Visual Basic.NET*, *Visual C++.NET*, *Visual C#.NET* dan *Visual J#.NET*.
8. Tidak berselang lama *Microsoft* merilis *Visual Studio. NET 2003* yang

memperbaiki performa dan aspek keamanan dari *Visual Studio .NET* 2002 pada versi *Visual Studio .NET*, *Microsoft* juga menyertakan teknologi *.NET Framework* 1.1. Kabarnya, setiap tahun *Microsoft* akan merilis *Visual Studio .NET* versi baru sebagai jawaban untuk menutup kekurangan dari versi-versi sebelumnya (Primanda Arif Aditya, 2013, 1)



Gambar 1 Tampilan Ruang Kerja *Visual Basic 2008.NET*

III. METODE PENELITIAN

Beberapa metode penelitian dalam mengumpulkan data-data untuk penyelesaian aplikasi perangkat lunak mengenai kerusakan EMMC pada Handphone Vivo yang menggunakan metode *Certainty Factor* sebagai berikut :

1. Wawancara
Teknik wawancara ini dilakukan untuk mendapatkan informasi tambahan dari pihak-pihak yang memiliki wewenang dan data. Dalam hal ini saya melakukan wawancara langsung dengan pakar dibidang Handphone.
2. Metode Perpustakaan
Penulis memperoleh data dengan cara mengutip dari beberapa bahan bacaan yang berkaitan dengan skripsi ini. Hal yang dikutip dapat berupa data, informasi, teori ataupun beberapa pendapat dari buku yang diperoleh dari perpustakaan atau bahan kuliah.
3. Analisa

Analisa merupakan sebuah proses dalam sebuah penelitian, dengan analisa awal menginformasikan data yang kemudian dikumpulkan. Maka langkah berikutnya ialah menganalisis data yang telah diperoleh.

4. Perancangan
Pada tahap ini penulis mulai membuat rancangan-rancangan dari aplikasi yang akan dibuat, antara lain desain dari tampilan, perancangan menu-menu, serta penentuan dari fungsi setiap tombol yang ada.
5. Pengujian
Dalam tahapan ini akan dilihat bagaimana menentukan hasil dari apa yang telah dirancang.
6. Implementasi
Hasil dari tahapan-tahapan tersebut akan dipindahkan kedalam sistem informasi.

IV. ANALISA DAN PERANCANGAN

4.1 Analisa Perancangan

Terhambatnya penanganan penyebab kerusakan EMMC handphone Vivo masih kurang diketahui oleh masyarakat. Untuk mengatasi hal ini diperlukannya seorang pakar atau ahli dalam bidang tersebut serta suatu metode untuk menyebarkan kepakaran yang dimiliki oleh seorang pakar kedalam suatu program komputer yang dinamakan sistem pakar dengan mendeteksi kerusakan EMMC yang melibatkan sistem kerusakan umum, yang diharapkan membantu masyarakat mengenali gejala-gejala yang timbul. Oleh karena itu berdasarkan analisa masalah diatas, maka melalui sistem ini diharapkan menjadi pilihan alternatif konsultasi serta informasi mengenai penyebab kerusakan EMMC pada handphone vivo. Untuk menyelesaikan permasalahan yang terjadi digunakan metode *Certainty Factor* untuk dapat mengetahui penyebab kerusakan EMMC pada handphone vivo terdapat pada Tabel 1 dan Tabel 2.

Tabel 1 Gejala Penyebab Kerusakan EMMC Pada Handphone Vivo

(Sumber : Teknisi Handphone)

No	Kode	Gejala Kerusakan EMMC
1	G01	Aplikasi yang dibuka sering keluar sendiri secara tiba-tiba dengan notifikasi "Sayangnya aplikasi telah terhenti"
2	G02	Hp hanya tampil logo atau hanya getar ketika di hidupkan sehingga perlu pengecekan mesin handphone
3	G03	Pernah terjadi kegagalan saat flasing
4	G04	Handphone sering menjadi lambat dan terestart sendiri

Tabel 2 Basis Pengetahuan Tentang Kerusakan EMMC Handphone Vivo

Kode	Nama Gejala	MB	MD
G01	Aplikasi yang dibuka sering keluar sendiri secara tiba-tiba dengan notifikasi "Sayangnya aplikasi telah terhenti"	0.4	0.2
G02	Hp hanya tampil logo atau hanya getar ketika di hidupkan sehingga perlu pengecekan mesin handphone	0.6	0.4
G03	Pernah terjadi kegagalan saat flasing	0.8	0.6
G04	Handphone sering menjadi lambat dan terestart sendiri	0.6	0.4

(Sumber : Teknisi Handphone)

4.2 Algoritma Sistem

Menyelesaikan permasalahan yang terjadi tentang mendeteksi kerusakan EMMC pada handphone vivo berdasarkan gejala-gejala kerusakan yang terjadi pada EMMC handphone vivo maka dibutuhkan suatu sistem yang mampu mengambil proses dan cara berpikir seorang pakar yang nantinya dapat diaplikasikan dalam sebuah sistem komputer dengan menggunakan metode *certainty factor*. Penerapan metode *certainty factor*

digunakan untuk mengukur tingkat kepastian dalam mendeteksi gejala-gejala kerusakan yang terjadi pada EMMC handphone vivo.

Rumus :

$$CF[H,E] = MB[H,E] - MD[H,E]$$

$$MB(H,E1^E2) = MB[H,E1] + MB[H,E2] * (1 - MB[H,E1])$$

$$MD(H,E1^E2) = MD[H,E1] + MD[H,E2] * (1 - MD[H,E1])$$

Kaidah Kerusakan Pada EMMC Handphone vivo :

IF Aplikasi yang dibuka sering keluar sendiri secara tiba-tiba dengan notifikasi "Sayangnya aplikasi telah terhenti"

AND Hp hanya tampil logo atau hanya getar ketika di hidupkan sehingga perlu pengecekan mesin handphone

AND Pernah terjadi kegagalan saat flasing

AND Handphone sering menjadi lambat dan terestart sendiri

THEN EMMC handphone rusak

Langkah pertama, pakar menentukan nilai CF untuk masing-masing gejala sebagai berikut :

$$MB(G01) : 0.4$$

$$MD(G01) : 0.2$$

$$MB(G02) : 0.6$$

$$MD(G02) : 0.4$$

$$MB(G03) : 0.8$$

$$MD(G03) : 0.6$$

$$MB(G04) : 0.6$$

$$MD(G04) : 0.4$$

Kaidah awal yang memiliki 4 premis dipecah menjadi kaidah yang memiliki premis tunggal, sehingga menjadi :

Kaidah 1.1 :

IF Aplikasi yang dibuka sering keluar sendiri secara tiba-tiba dengan notifikasi "Sayangnya aplikasi telah terhenti"

THEN EMMC handphone rusak

Kaidah 1.2 :

IF Hp hanya tampil logo atau hanya getar ketika di hidupkan sehingga perlu pengecekan mesin handphone

THEN EMMC handphone rusak

Kaidah 1.3 :

IF Pernah terjadi kegagalan saat flasing

THEN EMMC handphone rusak

Kaidah 1.4 :

IF Handphone sering menjadi lambat dan terestart sendiri

THEN EMMC handphone rusak

$$MB(\text{EMMC handphone rusak}, G01^G02) = 0.4 + 0.6 * (1-0.4) = 0.4 + (0.6 * 0.6) = 0.4 + 0.36 = 0.76$$

$$MD \text{ (EMMC handphone rusak, } G01^{\wedge}G02) = 0.2 + 0.4 * (1-0.2) = 0.2 + (0.4 * 0.8) = 0.2 + 0.32 = 0.52$$

$$CF[H,E]1 = 0.76 - 0.52 = 0.24$$

$$MB \text{ (EMMC handphone rusak, } G01^{\wedge}G02^{\wedge}G03) = 0.76 + 0.8 * (1-0.76) = 0.76 + (0.8 * 0.24) = 0.76 + 0.192 = 0.952$$

$$MD \text{ (EMMC handphone rusak, } G01^{\wedge}G02^{\wedge}G03) = 0.52 + 0.6*(1-0.52) = 0.52 + (0.6 * 0.48) = 0.52 + 0.288 = 0.808$$

$$CF[H,E]1 = 0.952 - 0.808 = 0.144$$

$$MB \text{ (EMMC handphone rusak, } G01^{\wedge}G02^{\wedge}G03^{\wedge}G04) = 0.952 + 0.6 * (1-0.952) = 0.952 + (0.6 * 0.048) = 0.952 + 0.0288 = 0.9808$$

$$MD \text{ (EMMC handphone rusak, } G01^{\wedge}G02^{\wedge}G03^{\wedge}G04) = 0.808 + 0.4 * (1-0.808) = 0.808 + (0.4 * 0.192) = 0.808 + 0.0768 = 0.8848$$

$$CF[H,E]1 = 0.9808 - 0.8848 = 0.096$$

Hasil Perhitungan *Certainty Factor*. Berikut ini adalah hasil perhitungan dari metode *certainty factor*.

- a. Nilai CF MB adalah = 0.9808
- Nilai CF MD adalah = 0.8848
- Mencari Nilai Maximal
- Max = 0.9808 - 0.8848
- = 0.096

4.3 Implementasi

Implementasi dalam penulisan skripsi ini merupakan proses atau langkah dari pengguna sistem yang menerapkan metode *Certainty Factor* ke dalam bahasa pemrograman. Penjelasan tentang implementasi sistem dijabarkan pada langkah – langkah berikut :

1. Login

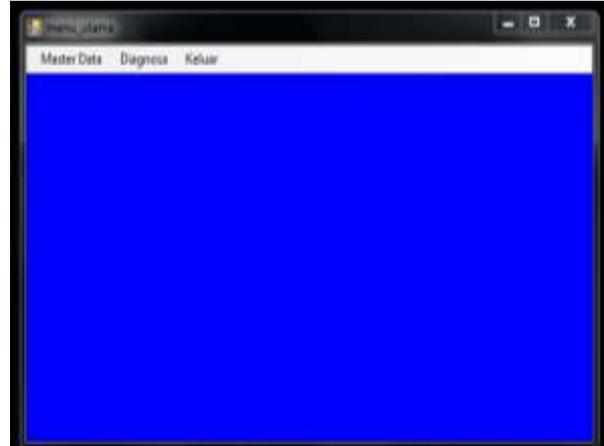
Seperti pada umumnya sebuah aplikasi harus memiliki sistem *login* untuk memberikan kemudahan pada pengguna untuk menggunakan sistem yang telah dirancang pada Gambar 2.



Gambar 2 Form Login

2. Menu Utama

Menu utama merupakan halaman yang pertama kali muncul setelah *admin login* ke dalam sistem yang terdapat pada Gambar 3.



Gambar 3 Menu Utama

3. Form Input Data Kerusakan

Pada *menu input* data kerusakan merupakan *Form* yang digunakan untuk menginput data.

4. Form Input Data Gejala

Form Input data gejala merupakan *Form* yang digunakan untuk menginput dan merubah database tanpa harus masuk ke dalam database langsung, berikut ini merupakan tampilan *Form* data gejala terdapat pada Gambar 4.



Gambar 4 Form Data Gejala

5. Form Diagnosa

Form Diagnosa merupakan *Form* yang digunakan untuk melihat hasil diagnosa yang telah dilakukan oleh pakar atau teknisi *HandPhone* terdapat pada Gambar 5.



Gambar 5 Form Diagnosa

6. Form Laporan Hasil Diagnosa

Form Laporan Hasil Diagnosa Kerusakan EMMC dengan *Cristal Report Form* yang digunakan untuk melihat hasil yang sudah selesai dari perhitungan *Certainty Factor* terdapat pada Gambar 6.



Gambar 6 Form Laporan Hasil Diagnosa

V. KESIMPULAN DAN SARAN

5.1 Kesimpulan

Adapun kesimpulan yang diperoleh adalah :

1. Aplikasi yang dibangun dapat memudahkan para pengguna Handphone khususnya bagi pengguna Handphone Vivo untuk mendeteksi kerusakan MC sebelum membawanya ke teknisi Handphone, sehingga pengguna dapat mengurangi biaya perbaikan.
2. Dapat memberikan informasi sebagai deteksi awal kerusakan pada EMMC Handphone vivo sesuai dengan pengetahuan sistem pakar yang telah dibuat.
3. Aplikasi yang dibangun sederhana akan tetapi sangat berguna dan bermanfaat bagi pengguna Handphone.

5.2 Saran

Adapun saran yang dapat diberikan setelah melakukan pembahasan mengenai perangkat lunak Sistem Pakar Mendeteksi Kerusakan EMMC Pada Handphone Vivo menggunakan Metode *Certainty Factor* yang berbasis desktop adalah :

1. Sistem pakar dengan menggunakan metode *Certainty Factor* ini sebaiknya dapat di fungsikan untuk banyak kalangan pengguna sehingga aplikasi ini dapat dikunjungi oleh banyak orang.
2. Sistem ini hendaknya dilakukan pengembangan secara terus-menerus untuk dapat memudahkan pengguna dalam menentukan kerusakan EMMC pada Handphone Vivo tanpa harus membawanya ke teknisi Handphone langsung untuk dilakukan perbaikan EMMC.
3. Sistem ini di harapkan terus dikembangkan sehingga dapat memenuhi standar pengembangan untuk mendeteksi kerusakan EMMC pada Handphone Vivo.berupa Studi Literatur, dapat memberikan ringkasan yang disarankan

DAFTAR PUSTAKA

- Muhammad Arhami, 2015, *Konsep Dasar Sistem Pakar*, Yogyakarta: (Penerbit Andi)
- Sutojo, S.Si., M.Kom., Edy Mulyanto, S.Si., M.Kom., Dr.Vincent Suhartono, 2011, *Kecerdasan Buatan*.Yogyakarta: (Penerbit Andi).
- Wahana Komputer, 2009, *Panduan Aplikasi Dan Solusi (PAS) Membangun Aplikasi Dengan Visual Basic* 2008, Yogyakarta: (Penerbit Andi).
- Rahmat Priyanto, 2009, *Langsung Bisa Visual Basic.Net* 2008. Bandung: ANDI.
- Harip Santoso. 2010. *Membangun Aplikasi Web Menggunakan Vb.Net/Asp.Net* Andi, Jakarta.

- Kusrini, 2008, Aplikasi Sistem Pakar, Edisi I, Andi, Yogyakarta.
- Marlina., Iqbal., 2016, Implementasi Metode Certainty Factor pada Unggas, Jurnal TIKTA No 2 Vol 1, hal 48-63.
- Gaung Rimba Putra Dirgantara., Suprpto., & Bayu rahayudi, 2018, *Implementasi Metode Certainty factor Pada Identifikasi Kerusakan Kendaraan Bermotor Roda Dua*, Jurnal Pengembangan Teknologi Informasi Dan Ilmu komputer, Vol.2, No,6 hlm,2046 – 2050.
- Jogiyanto. (2009). *Analisa & Desain Sistem Informasi*. Yogyakarta: Andi.
- Kurniawan Yahya, 2005. *Vb Script, Pt Elex Media Komputindo*. Jakarta
- Connolly, Thomas and Carolyn Begg. (2009). *Database System: A Practical Approach to Design, Implementation, and Management*, 4th Edition. Addison Wesley: Harlow, Englan