

BAB III

ANALISIS DAN DESAIN SISTEM

III.1. Analisa Sistem yang sedang berjalan

Analisa sistem yang sedang berjalan merupakan proses awal yang harus dilaksanakan untuk menentukan permasalahan yang sedang dihadapi. Tahap ini adalah sangat penting, karena proses analisa sistem yang kurang akurat akan menyebabkan hasil dari suatu sistem tidak sesuai dengan yang diharapkan. Jadi proses ini harus benar-benar sesuai dengan perencanaan agar menghasilkan suatu sistem yang baik.

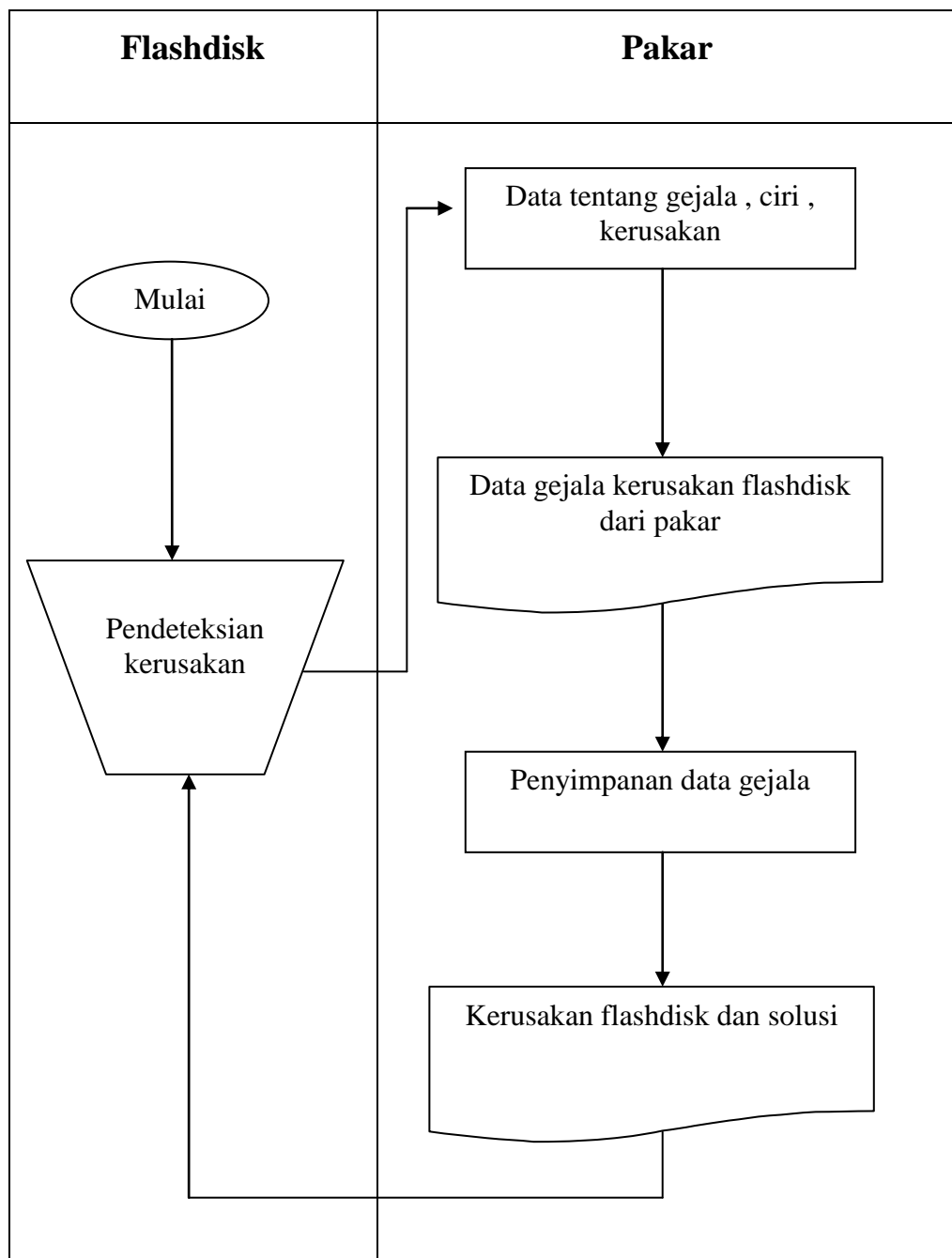
Judul yang diangkat oleh penulis tentang analisa kerusakan flashdisk , ternyata selama ini belum ada suatu sistem atau program yang di buat sebelumnya yang berkaitan dengan judul skripsi penulis dan selama ini apabila seorang user mengalami gangguan kerusakan flashdisk maka user tersebut hanya berkonsultasi langsung pada sumber / pakar tersebut.

III.1.1 Analisa Input

Melihat dari analisa sistem yang sedang berjalan diatas yang di mana selama ini belum adanya program / sistem yang di rancang untuk mendeteksi kerusakan flashdisk , maka selama ini seorang user hanya datang dan konsultasi langsung pada seorang pakar yang mengerti tentang kerusakan dan perbaikan flashdisk , dan sistem input ini tentu saja belum menggunakan sistem komputerisasi

III.1.2. Analisa Proses

Proses yang terjadi pada sistem dapat di jelaskan pada langkah berikut ini



Gambar. III.1 *Flow Of Document* dari sistem yang akan di rancang

III.1.3 Analisa Output

Melihat dari sistem yang lama , maka output yang dihasilkan berupa jenis kerusakan flashdisk dan cara perbaikannya , akan tetapi output tersebut hanya dalam bentuk lisan yang disampaikan langsung oleh pakar jadi pada sistem ini belum ada proses laporan (*Report*) yang diciptakan.

III.2. Evaluasi Sistem yang berjalan

Setelah dilakukan analisa terhadap ketiga poin diatas penulis menemukan kelemahan dari sistem yang berjalan. Dimana seorang user harus berkonsultasi langsung dengan pakarnya , jika dipandang dari segi efisiensi tentu saja sistem ini sangat merepotkan dan dapat membuang waktu.

Untuk itu penulis merasa perlu membuat sebuah sistem / program yang dapat mengatasi semua permasalahan dan kekurangan dari sistem yang berjalan guna menghasilkan sebuah informasi tentang kerusakan flashdik dan cara perbaikannya yang tepat , jelas & efisien.

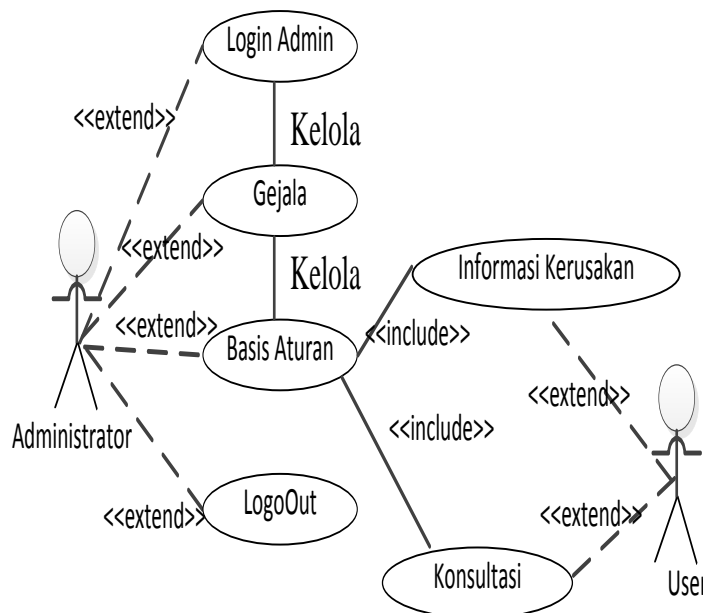
III.3. Disain Sistem

1II.3.1 Disain Sistem Secara Global

Untuk membangun sebuah sistem perlu adanya desain dari sebuah sistem yang dirancang. Berikut adalah desain sistem yang akan dirancang oleh penulis

III.3.1.1 Use Case Diagram

Use Case Diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem yang menekankan apa yang dibuat sistem dan merepresentasikan sebuah interaksi antara actor dengan sistem. Berikut adalah *use case diagram* dari analisa kerusakan flashdisk menggunakan metode Dempster Shafer.



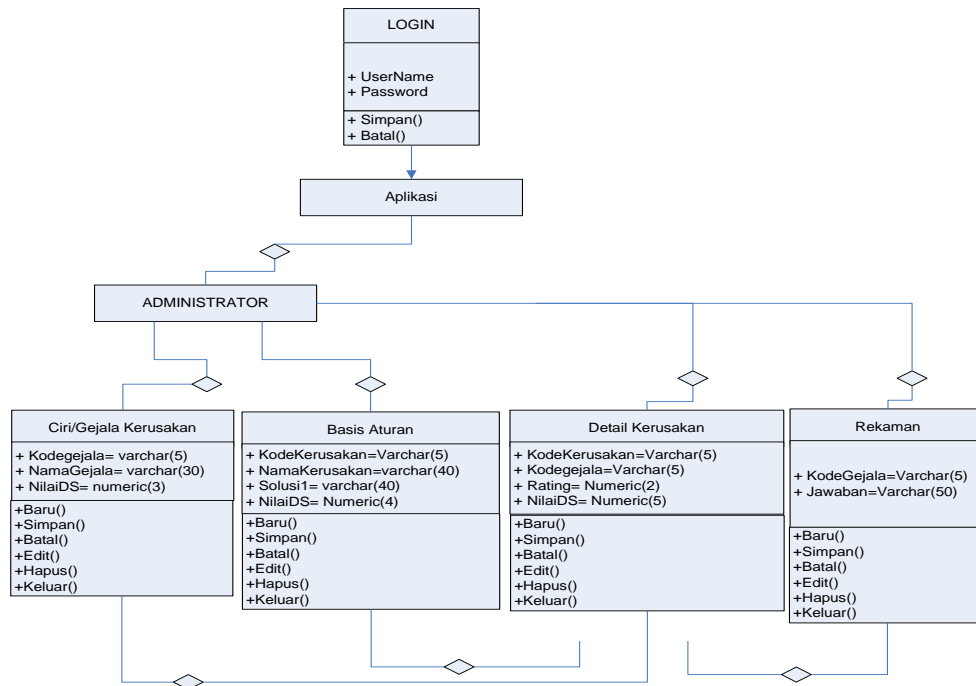
Gambar. III.2 Use Case Diagram Analisis Kerusakan Flashdisk Menggunakan Metode Dempster Shafer

III.3.1.2 Class Diagram

Diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem / perangkat lunak yang sedang kita kembangkan. • Diagram kelas (Class Diagram) memberi kita gambaran (diagram statis) tentang

sistem / perangkat lunak dan relasi-relasi yang ada di dalamnya. Bentuk Class

Diagram dari system yang dibangun dapat dilihat pada gambar dibawah ini.

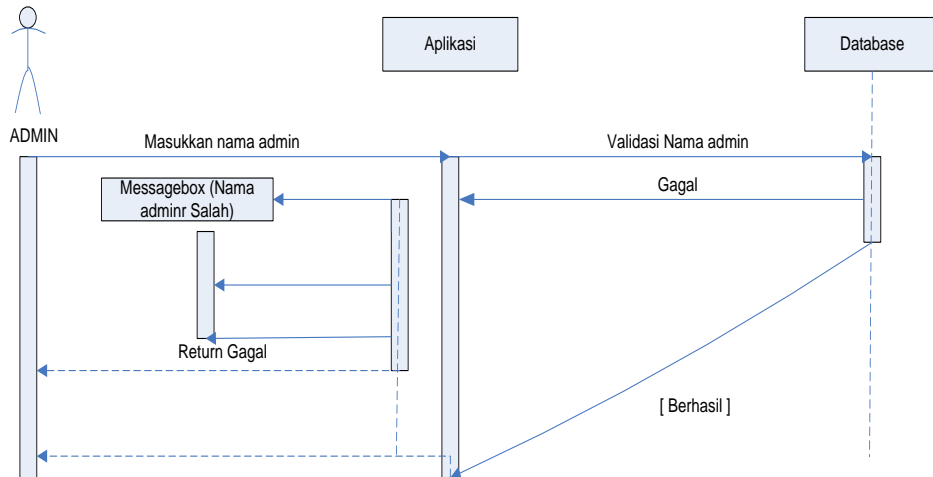


Gambar III.3 Class Diagram

III.3.1.3 Sequence Diagram

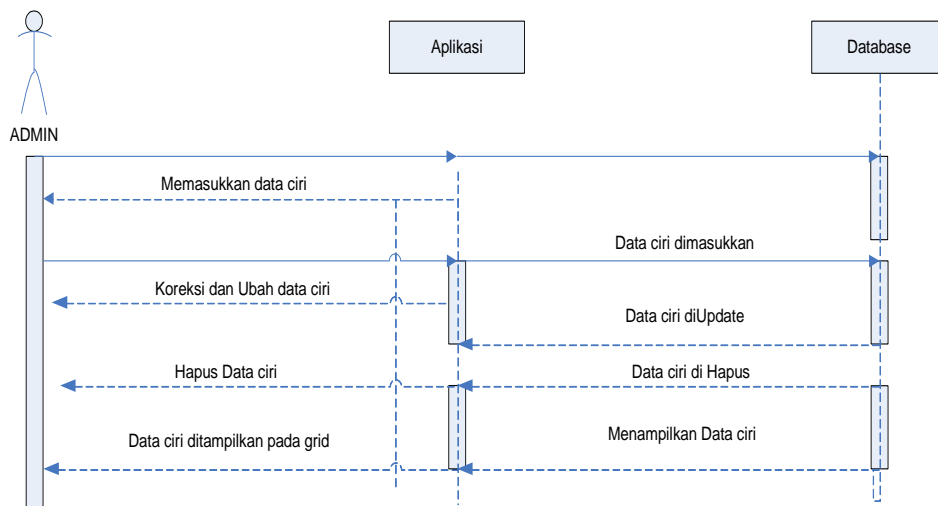
Sequence diagram adalah diagram yang merepresentasikan interaksi antar-objek. Bentuk *Sequence diagram* dari sistem yang dibangun adalah sebagai berikut :

a. Sequence diagram Login Ke Sistem



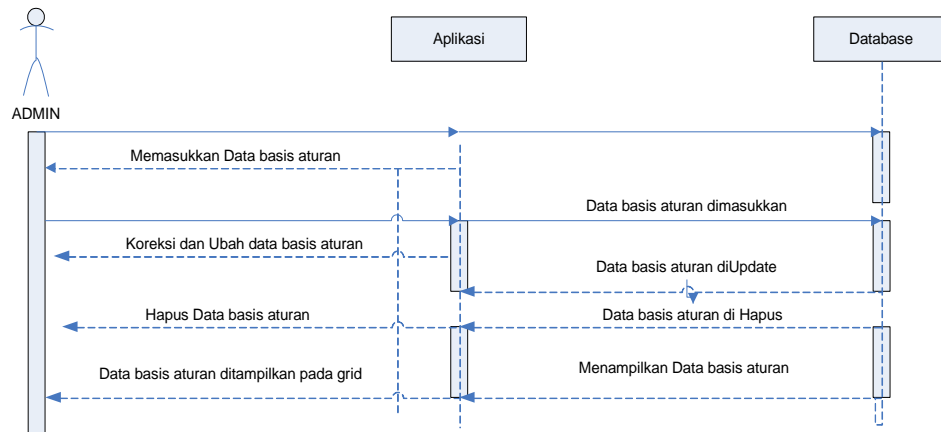
Gambar III.4 Sequence Diagram Login Ke Sistem

b. Sequence diagram Ciri/gejala kerusakan



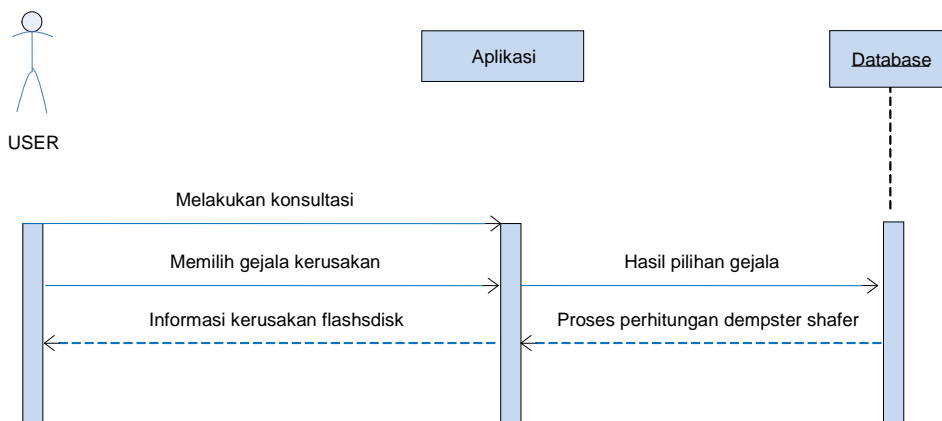
Gambar III.5 Sequence Diagram Ciri / Gejala Kerusakan

c. Sequence diagram basis aturan



Gambar III.6 Sequence Diagram Basis Aturan

d. Sequence diagram hasil konsultasi / laporan



Gambar III.7 Sequence diagram informasi kerusakan flashdisk

III.4.2 Disain Input

Pada penjelasan ini penulis akan memaparkan desain input dari sistem yang sedang dirancang penulis, pada tampilan desain input ini seorang user harus memilih atau menginputkan ke dalam sistem mengenai kode gejala, jenis gejala dan menentukan dengan pilihan antara YA atau TIDAK, setelah memilih kode gejala dan nama gejala maka user tersebut harus menekan tombol LANJUTKAN untuk memproses data yang telah diinputkan. Berikut adalah gambar desain sistem dari program yang akan dibuat.

Sadd

Kode Gejala :

Nama Gejala :

Nilai DS :

Kode Gejala	Nama Gejala	Nilai DS

Baru	Simpan	Batal	Ubah	Hapus	Keluar
------	--------	-------	------	-------	--------

Gambar III.9 Tampilan Input Dari Sistem Yang Akan Dirancang

III.4.3 Disain Database

Desain database berguna untuk menyimpan data – data yang saling berhubungan satu dengan yang lain nya. Dalam perancangan database di bentuk satu file yang berguna untuk menyimpan tabel – tabel yang diperlukan sebagai basis penyimpanan suatu data.

III.4.4 Kamus Data

Dibawah ini adalah kamus data atau refrensi data yang akan ada pada data sistem yang akan dibangun

1. Tabel Gejala = **kodegejala**+namagejala+nilaiDS
 Nama Database = db_Flashdisk
 Nama Tabel = dbo.TGejala
 nilaiDS : Nilai Dempster Shafer
2. Tabel kerusakan flashdisk = **kodekerusakan** + namakerusakan + solusi + nilaiDS
 Nama Database = db_Flashdisk
 Nama Tabel = dbo.TKerusakan
3. Tabel detail kerusakan flashdisk = **kodekerusakand** + kodegejala + rating + nilaiDS
 Nama Database = db_Flashdisk
 Nama Tabel = dbo.TDetail
4. Tabel rekaman = **kodegejala** + jawaban
 Nama Database = db_Flashdisk
 Nama Tabel = dbo.TRekaman
 Jawaban : berisi jawaban pertanyaan
5. Tabel login = username + password + status
 Nama Database = db_Flashdisk
 Nama Tabel = dbo.TLogin

III.4.5. Normalisasi

Normalisasi digunakan untuk menghindari ketergantungan antara 1 field dengan field lainya dan digunakan untuk menghindari redudansi data (kesamaan data).

1. Unnormal

Kode	Nama Kerusakan	Gejala	Solusi
0001	Flashdisk Bad Sector	Booting awal lambat	Perbaiki Flashdisk
-	-	Bunyi flashdisk keras	-
-	-	Proses lambat	-
-	-	Saat di scandisk tampil keterangan Bad Block di beberapa sector	-
-	-	Transfer data flashdisk terasa lambat	-
-	-	Gejala yang terjadi adalah bunyi flashdisk keras	-
0002	Flashdisk Failure	Proses komputer berhenti	Perbaiki Flashdisk
-	-	Setelah melakukan Power On Self Test, BIOS melaporkan pesan kesalahan “USB Failure”	-
-	-	Flashdisk gagal dibaca	-

2. Normal Pertama (1NF)

Kode	Nama Kerusakan	Gejala	Solusi
0001	Flashdisk Bad Sector	Booting awal lambat	Perbaiki Flashdisk
0001	Flashdisk Bad Sector	Bunyi flashdisk keras	Perbaiki Flashdisk
0001	Flashdisk Bad Sector	Proses lambat	Perbaiki Flashdisk
0001	Flashdisk Bad Sector	Saat di scandisk tampil keterangan Bad Block di beberapa sector	Perbaiki Flashdisk

0001	Flashdisk Bad Sector	Transfer data flashdisk terasa lambat	Perbaiki Flashdisk
0001	Flashdisk Bad Sector	Gejala yang terjadi adalah bunyi flashdisk keras	Perbaiki Flashdisk
0002	Flashdisk Failure	Proses komputer berhenti	Perbaiki Flashdisk
0002	Flashdisk Failure	Setelah melakukan Power On Self Test, BIOS melaporkan pesan kesalahan “USB Failure”	Perbaiki Flashdisk
0002	Flashdisk Failure	Flashdisk gagal dibaca	Perbaiki Flashdisk

3. Normal Kedua (2NF)

Tabel Gejala
KodeGejala
NamaGejala
NilaiDS

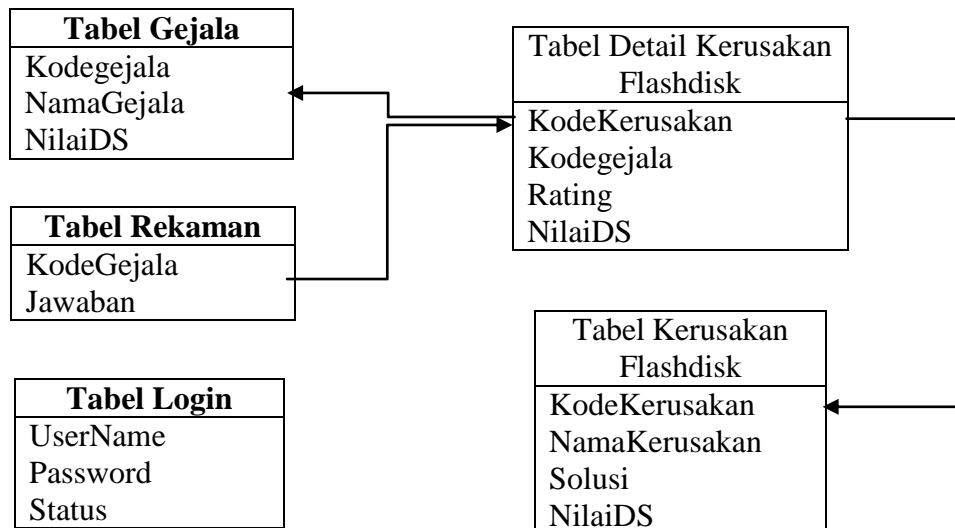
Tabel Rekaman
KodeGejala
Jawaban

Tabel Login
UserName
Password
Status

Tabel Detail Kerusakan Flashdisk
KodeKerusakan
Kodegejala
Rating
NilaiDS

Tabel Kerusakan Flashdisk
KodeKerusakan
NamaKerusakan
Solusi
NilaiDS

4. Normal Ketiga (3NF)



III.4.6. Disain Tabel/ File

Setiap database memiliki rancangan tabel yang digunakan untuk penyimpanan atau pengolahan data. Sehingga dalam database terdapat tabel – tabel dengan beberapa field yang mewakili sebuah klasifikasi data tertentu. Berikut ini desai dari tabel yang dirancang oleh penulis

1. Tabel Login

Ta

bel Login berguna untuk keamanan data. Jadi dalam hal ini hanya terdaftar didalam tabel pengembang yang berhak untuk melakukan perubahan terhadap sistem.

Database : db_Flashdisk

Tabel : dbo.Tlogin

Primary key : -

Tabel III.1 Tabel Login

Field name	Type	Size	Description	Keterangan
username	nvarchar	20	User Name	
password	nvarchar	20	Password	
Status	nvarchar	50	Status	

2. Tabel Kerusakan Flashdisk

Tabel Kerusakan Flashdisk ini berisi informasi tentang semua jenis kerusakan flashdisk

Database : db_Flashdisk

Tabel : dbo.TKerusakan

Primary key : Kodekerusakan

Foreign Key : -

Tabel III.2 Tabel Kerusakan Flashdisk

Field name	Type	Size	Description	Keterangan
KodeKerusakan	Char	10	Kode Kerusakan	<i>Primary Key</i>
NamaKerusakan	Varchar	50	Nama Kerusakan	
Solusi	Text	50	Solusi	
NilaiBN	Varchar	50	Nilai Demster Shafer	

3. Tabel Detail Kerusakan Flashdisk

Tabel ini berisi informasi Detail Kerusakan Flashdisk merupakan tabel untuk menampung gejala-gejala setiap kerusakan flashdisk.

Database : db_Flashdisk
 Tabel : dbo.TDetail
 Primary key : -
 Foreign Key : Kodekerusakan

Tabel III.3 Tabel Detail Kerusakan Flashdisk

Field name	Type	Size	Description	Keterangan
KodeKerusakan	Char	10	Kode kerusakan	<i>Foreign Key</i>
KodeGejala	Varchar	50	Kode gejala	
Rating	Numeric	10	Rating	

4. Tabel Rekaman

Tabel ini berisi informasi yang kerusakan flashdisk pada saat konsultasi. Jadi setiap hasil konsultasi disimpan pada tabel ini.

Database : db_Flashdisk
 Tabel : dbo.TRekaman
 Primary key : -
 Foreign key : Kodegejala

Tabel III.4 Tabel Rekaman Kerusakan Flashdisk

Field name	Type	Size	Description	Keterangan
kodegejala	Varchar	4	Kode Gejala	<i>Foreign key</i>
jawaban	Char	10	Jawaban Pertanyaan	

5. Tabel Gejala Kerusakan Flashdisk

Tabel gejala kerusakan flashdisk dapat digunakan untuk menyimpan data yang dapat memberikan informasi mengenai gejala kerusakan flashdisk

Database : db_Flashdisk

Tabel : dbo.Tlogin

Primary key : -

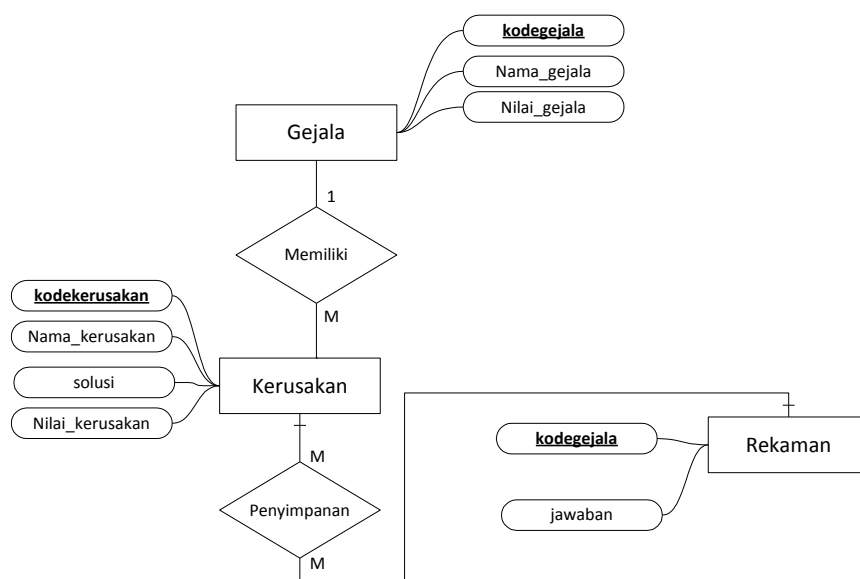
Foreign key : Kodegejala

Tabel III.5. Tabel Gejala

Field name	Type	Size	Description	Keterangan
Kodegejala	Nchar	4	Kode Gejala	<i>Primary key</i>
NamaGejala	Varchar	20	Nama Gejala	
NilaiBNgejala	Float	-	Nilai Dempster Shafer	

4.7. ERD (Entity Relationship Diagram) / Relasi Antar Tabel

Relasi menggambarkan hubungan antara tabel yang dapat dilihat pada Gambar III.9 Tabel login tidak dimasukkan dalam relasi karena bukan merupakan bagian dari data kepakaran (hanya menyimpan account pengguna pakar saja).



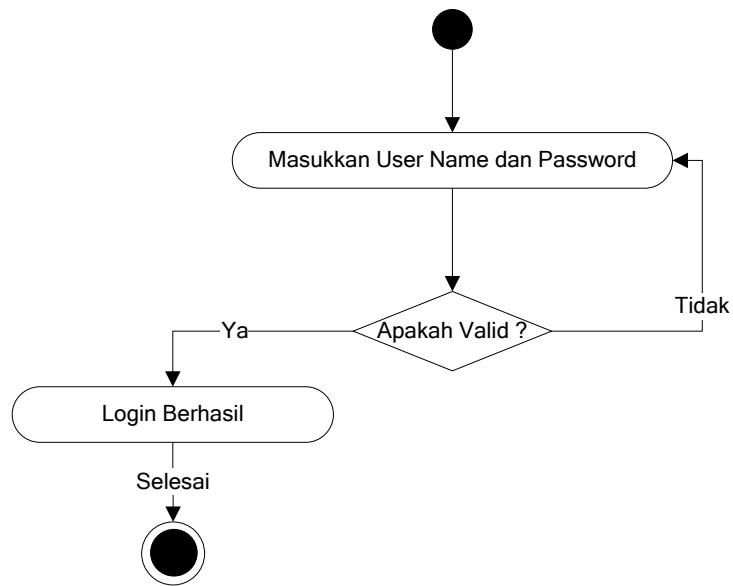
Gambar III.10 Relasi Antar Tabel

III.4.8. Logika Program

III.3.2.4.1 Activity Diagram

a. Activity Diagram Data Login

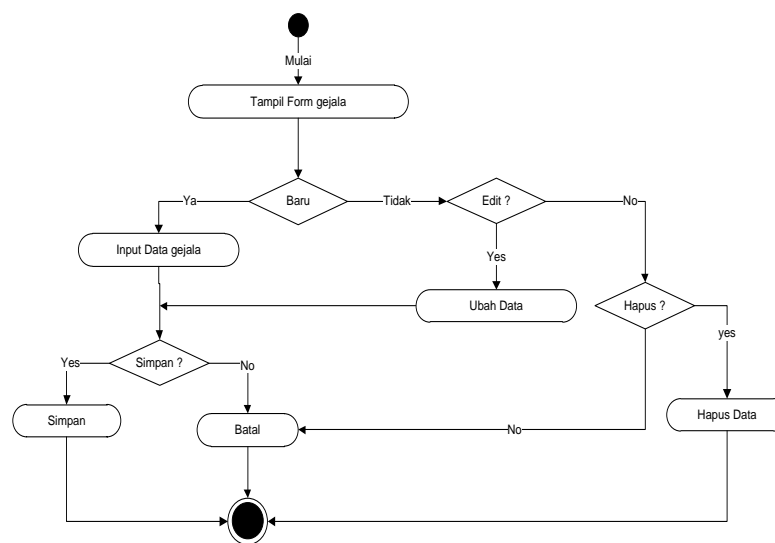
Adapun *Activity Diagram* form data login dapat dilihat h gambar dibawah ini.



Gambar III.11 Diagram Activity Login

b. *Activity Diagram* Data Gejala

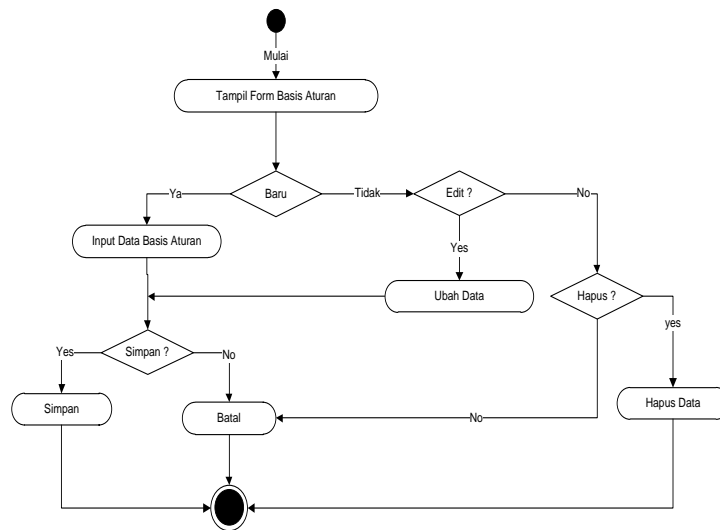
Adapun *Activity Diagram* form data gejala dapat dilihat pada gambar dibawah ini.



Gambar III.12 Diagram Activity Data Gejala

c. *Activity Diagram* Data Basis Aturan

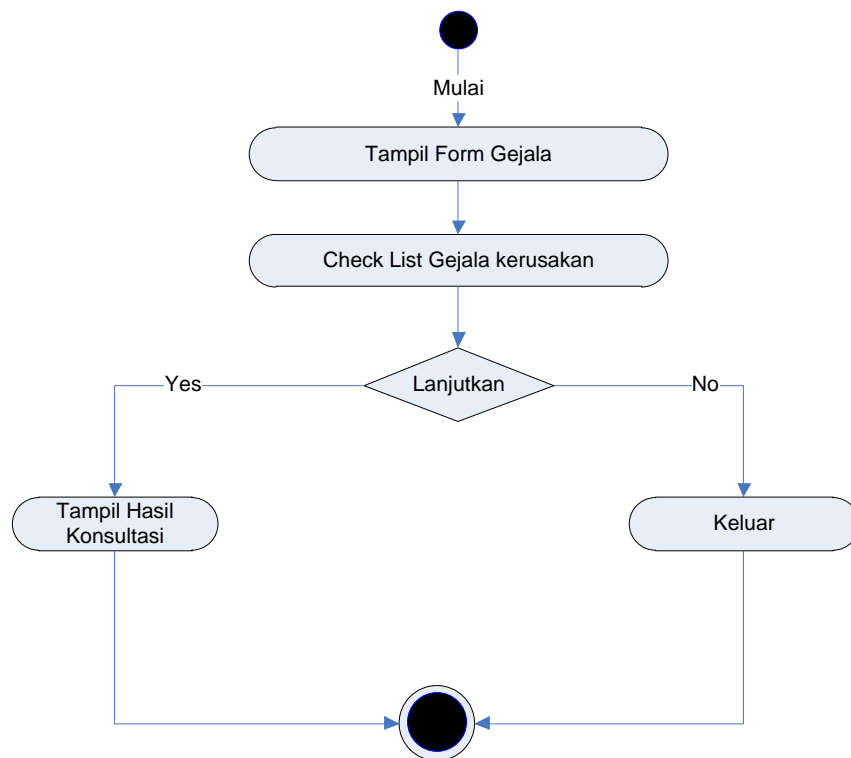
Adapun *Activity Diagram* form data basis aturan dapat dilihat pada gambar dibawah ini.



Gambar III.13 Diagram *Activity* Data Basis Aturan

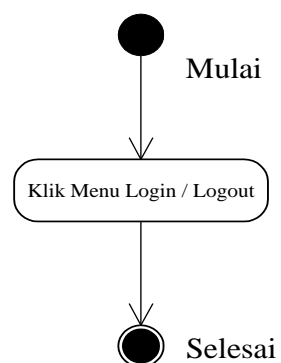
d. *Activity Diagram* Data Konsultasi

Adapun *Activity Diagram* form data Pembelian dapat dilihat pada gambar dibawah ini.



Gambar III.14 Diagram *Activity Data Konsultasi*

e. *Activity Diagram Data Logout*



Gambar III.15 Diagram *Activity Data Logout*

III.5 Basis Pengetahuan

Dalam perancangan basis pengetahuan ini digunakan kaidah produksi sebagai sarana untuk representasi pengetahuan. Kaidah produksi dituliskan dalam bentuk pernyataan **JIKA** [premis] **MAKA** [konklusi]. Pada perancangan basis pengetahuan sistem pakar ini premis adalah gejala-gejala yang terlihat pada flashdisk dan konklusi adalah jenis kerusakan flashdisk, sehingga bentuk pernyataannya adalah **JIKA** [gejala] **MAKA** [kerusakan]. Bagian premis dalam aturan produksi dapat memiliki lebih dari satu proposisi yaitu berarti pada sistem pakar ini dalam satu kaidah dapat memiliki lebih dari satu gejala. Gejala-gejala tersebut dihubungkan dengan menggunakan operator logika **DAN**. Bentuk pernyataannya adalah:

JIKA [gejala 1]

DAN [gejala 2]

DAN [gejala 3]

MAKA [kerusakan]

Adapun contoh kaidah sistem pakar mendeteksi kerusakan flashdisk adalah sebagai berikut :

a. Rule 1

JIKA Bunyi flashdisk keras **DAN** Proses lambat **DAN** Saat di scandisk
Tampil keterangan badblock di sector.

MAKA Flashdisk Failure

b. Rule 2

JIKA Driver usb masstorage bisa terdeteksi oleh PC (Komputer)
DAN pada windows explorer hanya muncul removable disk, tapi tidak ada driver flashdisk (Isi) **DAN** Kadang – kadang hilang driver USB **MAKA** Flashdisk Error X-tal card

c. Rule 3

JIKA Properties Flashdisk terbaca 0 byte. **DAN** Muncul pesan "Please insert a disk into drive x:" **DAN** Muncul pesan "device media is write-protected" **DAN** Muncul pesan "There is no media in the specified device" **MAKA** Flashdisk IC Controller USB

d. Rule 4

JIKA Tidak dapat membaca data. **DAN** Bisa di format tetapi tidak bisa dibaca **DAN** Tidak dapat melakukan aktivitas baca tulis file **DAN** Flashdisk terbaca tetapi tidak dapat di format **DAN** Pada saat di sambungkan ke komputer muncul permintaan format **DAN** Flashdisk tidak dapat di format **MAKA** Flashdisk Bad sector

e. Rule 5

JIKA Muncul pesan “ Device Not Recognized “ **DAN** Flashdisk sama sekali tidak dapat terbaca **DAN** Pada saat disambungkan kekomputer, komputer langsung mati **DAN** Muncul permintaan install driver USB **MAKA** Flashdisk Instalation driver

Pertanyaan-pertanyaan ini akan diproses dengan bentuk sebagai berikut :

1. If GK01 And GK02 and GK03 Then K01
2. If GK04 And GK05 And GK06 then K02
3. If GK07 And GK08 And GK09 And GK10 Then K03
4. If GK11 And GK12 And GK14 And GK15 And GK18 And GK19 Then K04
5. If GK13 And GK16 And GK17 And GK20 Then K05

a. Pengkonversian Tabel Keputusan Menjadi Kaidah Produksi

Representasi pengetahuan, kaidah produksi dibentuk dari perubahan tabel keputusan. Pembuatan suatu kaidah dilakukan dengan beberapa tahapan. Sebagai contoh perhatikan pembuatan kaidah Konklusi ini akan dapat tercapai bila kondisi-kondisi yang mendukung terpenuhi. Pembuatan kaidah 1 menggunakan goal dan kondisi yang telah diperoleh dari langkah 1 dan 2, seperti berikut :

Tabel III.6 Rule Gejala Kerusakan

RULE	IF	THEN
1	GK01,GK02,GK03	K01
2	GK04, GK05, GK06	K02
3	GK07, GK08, GK09, GK10	K03
4	GK11, GK12, GK14, GK15, GK18, GK19	K04
5	GK13, GK16, GK17, GK20	K05

Tabel III.7 Jenis Kerusakan Flashdisk

NO	KODE KERUSAKAN	JENIS KERUSAKAN
1	K01	Flashdisk Failure
2	K02	Error X-Tal Card
3	K03	IC Controller USB
4	K04	Bad Sector
5	K05	Instalation Driver

Demster shefer 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. Rumus dari Demster shefer

$$M_3(Z) = \frac{M_1(X)M_2(Y)}{N_1(X)N_2(Y)}$$

Pada Contoh dibawah ini, akan di cari persentase kemungkinan dari gangguan gangguan kerusakan flashdisk dengan menggunakan perhitungan pada tabel di bawah ini :

Tabel III.8 Tabel Gejala Pilihan kerusakan Flashdisk

Kode Gejala	Nama Gejala	Nilai
GK01	Bunyi flashdisk keras	0,7
GK02	Proses lambat	0,8
GK03	Saat di scandisk tampil keterangan Bad Block di sector	0,5
GK04	Driver usb masstorage bisa terdeteksi oleh PC (Komputer)	0,8
GK05	pada windows explorer hanya muncul removable disk	0,8
GK06	Kadang-kadang hilang driver usb	0,2
GK07	Properties Flashdisk terbaca 0 byte.	0,3
GK08	Muncul pesan "Please insert a disk into drive x:"	0,9
GK09	Muncul pesan "device media is write-protected"	0,7
GK10	Muncul pesan "There is no media in the specified device"	0,8
GK11	Tidak dapat membaca data	0,9
GK12	Bisa di format tetapi tidak bisa dibaca	0,9
GK13	Muncul pesan " Device not recognized "	0,2
GK14	Flashdisk yang masih kebaca tetapi tidak respon	0,5
GK15	Flashdisk yang masih kebaca tetapi tidak bisa diformat.	0,4
GK16	Flashdisk sama sekali tidak dapat terbaca	0,3
GK17	Pada saat disambungkan ke port komputer, komputer langsung mati	0,9
GK18	pada saat dicolokan muncul permintaan format flashdisk	0,2
GK19	Flashdisk tidak dapat diformat	0,4
GK20	muncul permintaan install driver USB	0,6

Dempster Shafer(DS) kerusakan flashdisk yang dipilih dengan menggunakan nilai *believe* yang telah ditentukan pada setiap gejala.

$$Pl(\Theta) = 1 - Bel$$

Dimana nilai bel (*believe*) merupakan nilai bobot yang diinput oleh pakar, maka untuk mencari nilai dari ketiga gejala diatas, terlebih dahulu dicari nilai dari, seperti yang dibawah ini.

Gejala 1 : Bunyi flashdisk keras “ (GK01)

Maka

$$GK01(bel) = 0.7$$

$$GK01(\Theta) = 1-0.7$$

$$= 0.3$$

Gejala 2 : proses lambat (GK02)

Maka

$$GK02(bel) = 0.8$$

$$GK02(\Theta) = 1-0.8$$

$$= 0.2$$

Gejala 3 : Saat di scandisk tampil keterangan Bad Block di beberapa sector

(GK03)

Maka

$$GK03(bel) = 0.5$$

$$GK03(\Theta) = 1-0.5$$

$$= 0.5$$

Maka untuk mencari nilai dari GPn, digunakan rumus :

$$M_3(Z) = \frac{M_1(X)M_2(Y)M_3(Z)}{1 - (N_1(X)N_2(Y)N_3(Z))}$$

$$GPn = \frac{0.7 * 0.8 * 0.5}{1 - (0.3 * 0.2 * 0.5)}$$

$$GPn = \frac{0.7 * 0.8 * 0.5}{1 - (0.3 * 0.2 * 0.5)}$$

$$GPn = \frac{0.28}{0.97} = 0.288$$

Maka nilai densitas dari ketiga gejala tersebut adalah 0,288. Dengan nilai densitas 0,288 (28 %) maka dapat dikatakan bahwa flashdisk memiliki eviden yang cukup kuat untuk mendekati kerusakan flashdisk failure

III.6 Mesin Inferensi (Inference Engine)

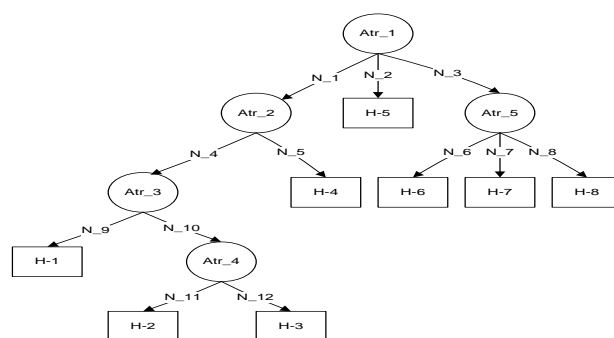
Mekanisme inferensi adalah bagian dari sistem pakar yang melakukan penelusuran dengan menggunakan isi daftar aturan berdasarkan urutan pola tertentu. Selama proses konsultasi antar sistem dan pemakai, mekanisme inferensi menguji aturan satu demi satu sampai kondisi aturan itu benar. Secara umum ada dua teknik utama yang digunakan dalam mekanisme inferensi untuk pengujian aturan, yaitu penelusuran maju (forward chaining) dan penelusuran mundur (backward chaining). Dalam penelusuran maju, aturan-aturan yang diuji satu demi satu dalam urutan tertentu. Urutan ini mungkin berupa urutan pemasukan aturan ke dalam basis aturan atau juga urutan lain yang ditentukan oleh pemakai. Saat tiap aturan diuji, sistem pakar akan mengevaluasi apakah kondisinya benar atau salah. Jika kondisi benar maka aturan itu disimpan kemudian aturan itu diuji.

Namun jika kondisinya salah, aturan itu tidak disimpan dan aturan berikutnya akan diuji. Proses ini akan berulang sampai seluruh basis aturan teruji dengan berbagai kondisi. Dalam mencari kerusakan flashdisk dan mencari penyebab kerusakan flashdisk akan dimulai dengan memberikan pertanyaan mengenai kerusakan flashdisk yang dialami atau dengan memberikan daftar macam kerusakan sehingga diperoleh suatu diagnosa kerusakan dan hasil akhir kesimpulan kerusakan mesin tersebut

III.7 Pohon Keputusan (*Decision Tree*)

Pohon atau *tree* adalah salah satu metode yang dapat digunakan untuk membuat suatu pemodelan. Struktur ini memiliki sifat-sifat atau ciri-ciri khusus, dan biasanya digunakan untuk menggambarkan hubungan yang bersifat hirarkis antara elemenelemen yang ada. Contoh paling sederhana yang bisa dilihat dalam kehidupan sehari-hari adalah struktur organisasi dari suatu perusahaan serta pada silsilah keluarga.

Hasil pembentukan pohon keputusan bisa seperti pohon keputusan yang tampak pada gambar dibawah ini



Gambar III.16 Decection Tree (Pohon Keputusan)

Node-node akar akan menjadi Premis dari aturan sedangkan node daun akan menjadi bagian konklusinya. Dari gambar pohon keputusan pada gambar 4, dapat dibentuk aturan sebagai berikut:

1. Jika $Atr_1 = N_1$
Dan $Atr_2 = N_4$
Dan $Atr_3 = N_9$
Maka H_1
2. Jika $Atr_1 = N_1$
Dan $Atr_2 = N_4$
Dan $Atr_3 = N_10$
Dan $Atr_4 = N_11$
Maka H_2
3. Jika $Atr_1 = N_1$
Dan $Atr_2 = N_4$
Dan $Atr_3 = N_10$
Dan $Atr_4 = N_12$
Maka H_2
4. $Atr_1 = N_1$
Dan $Atr_2 = N_5$
Maka H_4
5. Jika $Atr_1 = N_2$
Maka H_5

6. Jika $Atr_1 = N_3$
Dan $Atr_5 = N_6$
Maka H_6
7. Jika $Atr_1 = N_3$
Dan $Atr_5 = N_7$
Maka H_7
8. Jika $Atr_1 = N_3$
Dan $Atr_5 = N_8$
Maka H_8