

## **BAB III**

### **ANALISIS DAN DESAIN SISTEM**

#### **III.1. Analisis Masalah**

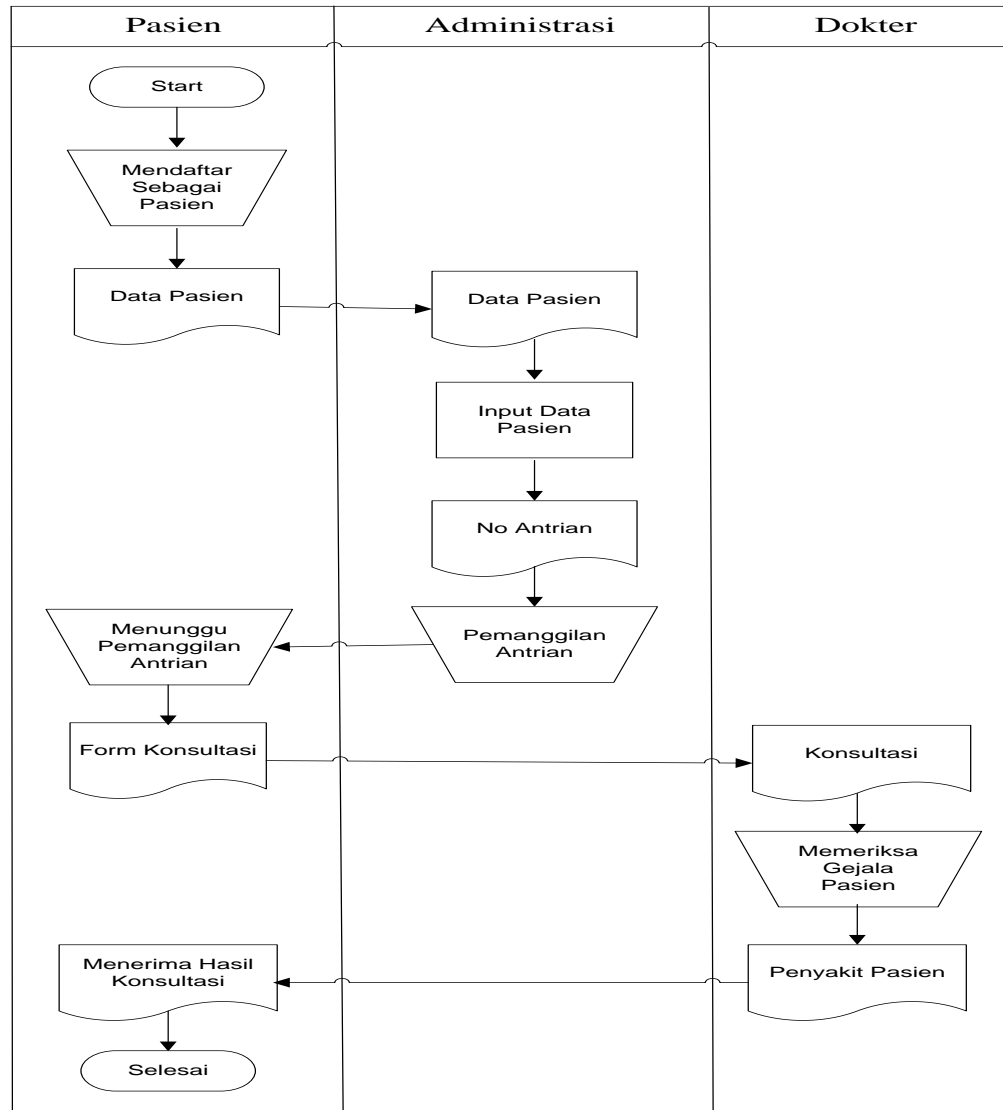
Analisa sistem yang di jelaskan pada bab ini adalah sebagai bahan perbandingan dengan sistem yang akan dirancang. Penulis akan memaparkan proses konsultasi yang sedang berjalan.

##### **III.1.1. Analisis Input**

Agar proses konsultasi dapat dilakukan dan menghasilkan *output* (keluaran) sesuai dengan yang diharapkan maka dokter spesialis mata perlu mengetahui data *input* (masukan) dari pasien yang merupakan gejala-gejala yang dialami oleh pasien. Pasien menyampaikan langsung keluhannya kepada dokter.

##### **III.1.2. Analisis proses**

Proses penentuan jenis penyakit pada pasien , data gejala penyakit dicatat. Kemudian dokter spesialis mata menyimpulkan penyakit yang diderita pasien dengan melihat kembali jenis gejala yang di tunjukkan oleh pasien, proses yang dilakukan pada sistem yang sedang berjalan dapat dijelaskan pada gambar III.1 dibawah ini:



**Gambar III.1. FOD Analisis Proses Mengidentifikasi Penyakit Lazy Eye Pada Anak**

**Sumber : Rumah Sakit Maya Sari**

Keterangan dari FOD ( Flow Of Dokumen) di atas adalah :

1. Pasien mendaftarkan sebagai pasien ke administrasi.
2. Administrasi menginput data pasien, kemudian memberikan no antrian ke pasien.

3. Pasien menunggu panggilan antrian. Setelah no antrian di panggil si pasien memberikan form konsultasi ke dokter kemudian dokter memeriksa gejala pasien.
4. Setelah dokter memeriksa gejala yang di alami pasien, dokter memberitahukan penyakit yang dialami oleh pasien.

### **III.1.3. Analisis Output**

Output merupakan dari hasil konsultasi yang di berikan dokter. Output atau hasil konsultasi dari identifikasi penyakit *lazy eye* adalah menentukan penyakit yang di derita si pasien beserta solusinya.

## **III.2. Penerapan Metode / Algoritma**

### **III.2.1. Metode *Dempster-Shafer***

Ada berbagai macam penalaran dengan model yang lengkap dan sangat konsisten, tetapi pada kenyataannya banyak permasalahan yang tidak dapat terselesaikan secara lengkap dan konsisten. Ketidak konsistenan yang tersebut adalah akibat adanya penambahan fakta baru. Penalaran yang seperti itu disebut dengan penalaran *non monotonis*. Untuk mengatasi ketidak konsistenan tersebut maka dapat menggunakan penalaran dengan teori *Dempster Shafer*

Sebelum mendesain sistem terlebih dahulu penulis akan menguraikan tabel yang akan digunakan, antara lain :

**Tabel III.1. Tabel Penyakit**

Id_Penyakit	Jenis Penyakit	Solusi
P001	<i>Lazy Eye</i> Ringan	Memberikan Obat tetes mata, Melakukan Penutupan bagian mata yang sehat 3 atau 6 jam setiap hari sehingga bagian mata yang malas akan belajar untuk melihat dengan baik. sebaiknya terapi ini dilakukan dalam pengawasan paramedis
P002	<i>Lazy Eye</i> Berat	Pemakaian kaca mata korektif kepada pasien dan jika pemakain kacamata korektif tidak berhasil, segera hubungin dokter spesialis mata untuk melakukan pemeriksaan lanjut sebelum melakukan operasi kecil bedah mata. Operasi kecil ini harus dalam pengontrolan dokter spesialis

**Tabel III.2. Tabel Gejala**

Id Gejala	Gejala-Gejala Penyakit <i>Lazy Eye</i>	Nilai
G-01	Anak harus lebih maju saat menonton TV ataupun belajar di kelas	0.2
G-02	penurunan tajam penglihatan	0.6
G-03	Miopia (rabun jauh)	0.3
G-04	Sakit kepala yang sering dan berkepanjangan	0.2
G-05	Anisometropik (ukuran mata tidak setara)	0.7
G-06	Ptosis (Mata turun)	0.3
G-07	Perbedaan pada ukuran kacamata	0.4
G-08	Hiperopia (rabun dekat)	0.3
G-09	Riwayat Keluarga	0.2
G-10	Strabismus (mata juling)	0.5
G-11	deprivasi	0.8

G-12	Isometropia	0.4
G-13	Astigmatisma yang besar	0.3
G-14	Gangguan penglihatan karena cacat sejak lahir	0.2
G-15	luka pada mata	0.3
G-16	Adanya efek destinas filter	0.3

Berikut ini adalah gambar tabel keputusan pada Sistem Pakar Identifikasi

Penyakit *Lazy Eye* pada Anak Dengan Metode *Dempster Shafer* adalah :

**Tabel III.3. Tabel Keputusan Sistem Pakar Identifikasi Penyakit *Lazy Eye* pada Anak Dengan Metode *Dempster Shafer***

Kode Gejala	Kode penyakit	
	P001	P002
G-01	*	
G-02	*	
G-03	*	
G-04	*	
G-05	*	
G-06	*	
G-07	*	
G-08	*	
G-09		*
G-10		*
G-11		*
G-12		*
G-13		*
G-14		*
G-15		*
G-16		*

### III.2.2. Rule Base

1. **If G-01 And G-02 And G-03 And G-04 And G-05 And G-06 And G-07 And G-08 Then Penyakit *Lazy Eye* Ringan.**

2. **If G-09 And G-10 And G-11 And G-12 And G-13 And G-14 And G-15 And G-16 Then Penyakit *Lazy Eye* Berat.**

Untuk mengetahui hasil konsultasi penyebab penyakit ini, dilakukan pengujian proses konsultasi. Proses pengujian sistem berupa masukan data gejala yang dialami pasien. Pada pengujian pertama diberikan beberapa gejala yang di alami pasien antara lain :

Rumus metode Dempster Shafer :

$$m_3(Z) = \frac{\sum_{X \cap Y = Z} m_1(X) \cdot m_2(Y)}{1 - \sum_{X \cap Y = \emptyset} m_1(X) \cdot m_2(Y)}$$

**Contoh Perhitungan Dempster Shafer :**

- a. Penyakit *Lazy Eye* Ringan :
1. penurunan tajam penglihatan
 
$$m_1(G-02) = 0.6$$

$$m_1(\emptyset) = 1 - 0.6 = 0.4$$
  2. Anisometropik (ukuran mata tidak setara)
 
$$m_2(G-05) = 0.7$$

$$m_2(\emptyset) = 1 - 0.7 = 0.3$$
  3. Ptosis (Mata turun)
 
$$m_3(G-06) = 0.3$$

$$m_3(\emptyset) = 1 - 0.3 = 0.7$$

#### 4. Perbedaan pada ukuran kacamata

$$m_4(G-07) = 0.4$$

$$m_4(\emptyset) = 1 - 0.4 = 0.6$$

#### 5. Hiperopia (rabun dekat)

$$m_5(G-08) = 0.3$$

$$m_5(\emptyset) = 1 - 0.3 = 0.7$$

Kalkulasi Penyakit *Lazy Eye Ringan* :

P001 {G-02,G-05,G-06,G-07,G-08}:

$$G-02 \cap G-05 = 0.6 * 0.7 / [ 1 - (0.4 * 0.3)] = 0.477$$

$$G-02 \cap G-05 \cap G-06 = 0.477 * 0.3 / [ 1 - (0.523 * 0.7)] = 0.226$$

$$G-02 \cap G-05 \cap G-06 \cap G-07 = 0.226 * 0.4 / [ 1 - (0.774 * 0.6)] = 0.169$$

$$G-02 \cap G-05 \cap G-06 \cap G-07 \cap G-08 = 0.169 * 0.3 / [ 1 - (0.831 * 0.7)] = 0.121$$

$$\sum G-01 \cap G-02 \cap G-03 \cap G-04 \cap G-05 = 0.477 + 0.226 + 0.169 + 0.121$$

$$= 0.993$$

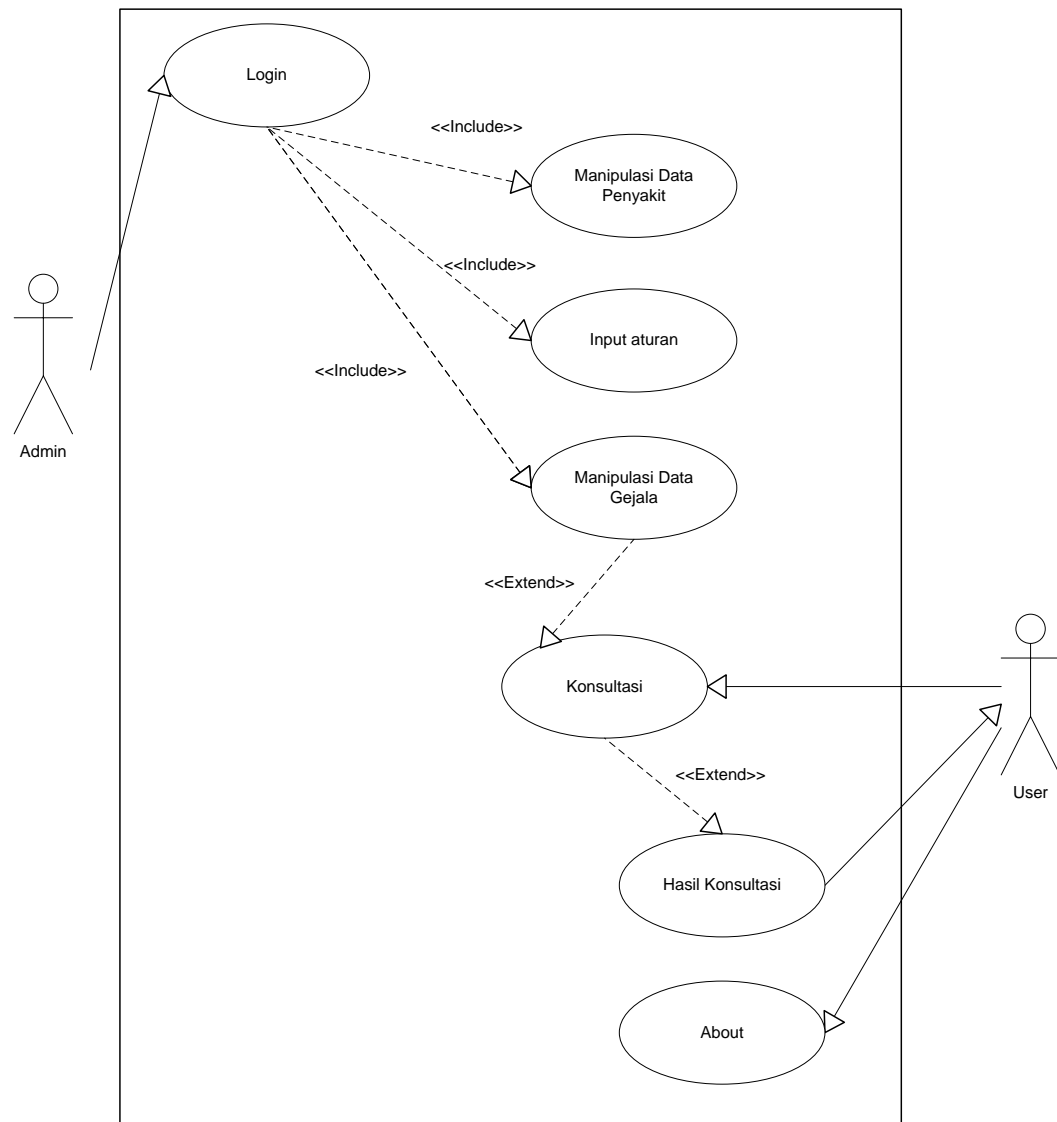
Dari nilai keyakinan 5 gejala diatas, maka Nilai kekuatan terhadap evidence penyakit *Lazy Eye Ringan* sebesar **99,3 %**

### III.3. Perancangan Sistem

Perancangan desain sistem yang akan dibangun menggunakan pemodelan UML. (*Unified Modeling Language*). Diagram-diagram yang digunakan *Use Case Diagram, Class Diagram, Activity Diagram, dan Sequence Diagram*.

### III.3.1. Use Case Diagram

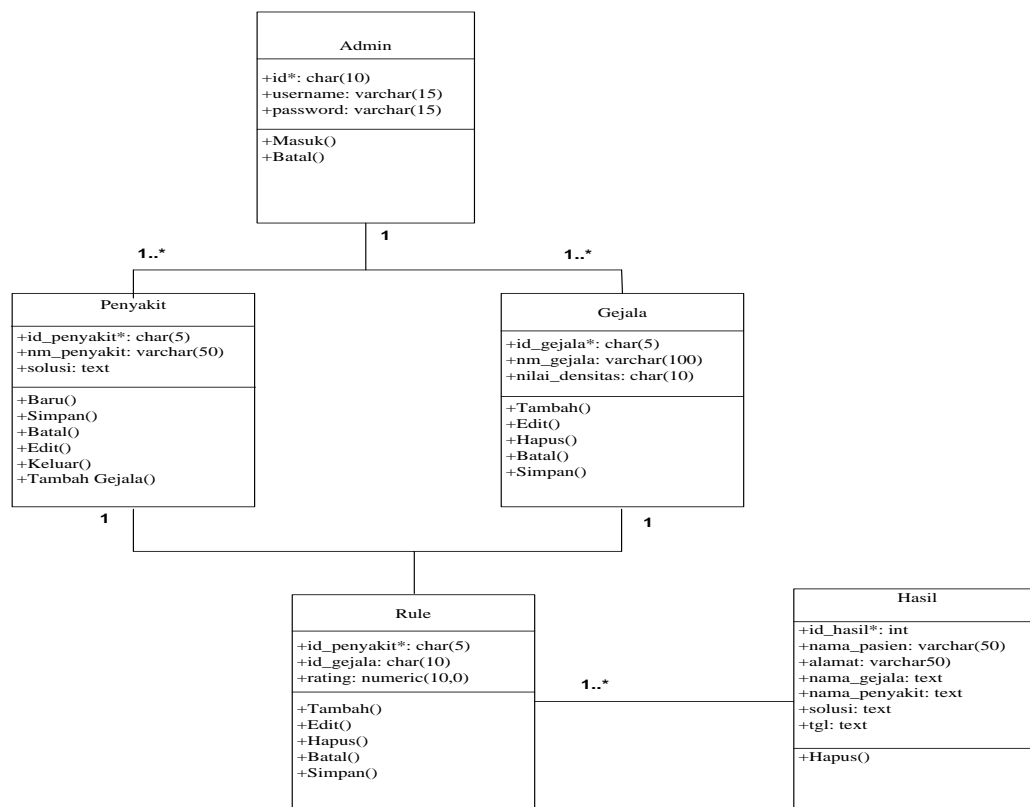
Diagram ini menggambarkan interaksi beberapa aktor dengan sistem di gambarkan pada gambar III.2 berikut ini :



**Gambar III.2. Use Case Diagram**

### II.3.2. Class Diagram

*Class diagram* pada aplikasi yang akan di bangun untuk penggunanya seorang pakar yaitu dimulai dari login seorang dokter untuk proses selanjutnya yaitu tampilan *home*, dan di akhiri dengan tampilan hasil konsultasi, berikut ini adalah gambar *class*. Berikut ini adalah *class diagram* dari aplikasi sistem pakar identifikasi penyakit *Lazy eye*.



**Gambar III.3. Class diagram**

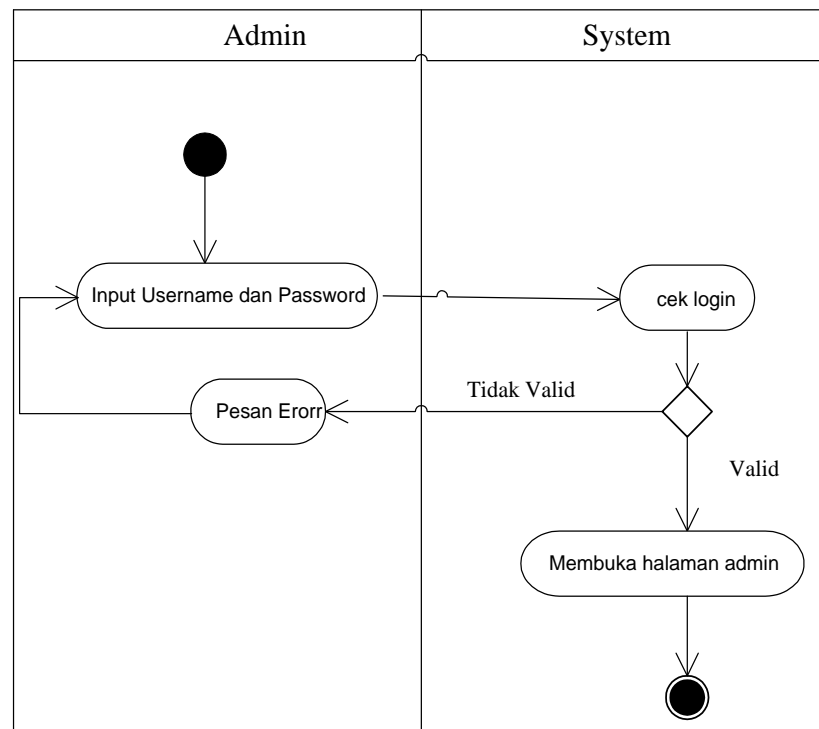
### III.3.3. Activity Diagram

*Activity diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing – masing alir berawal, *Decision* yang

mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

## 1. Login Admin

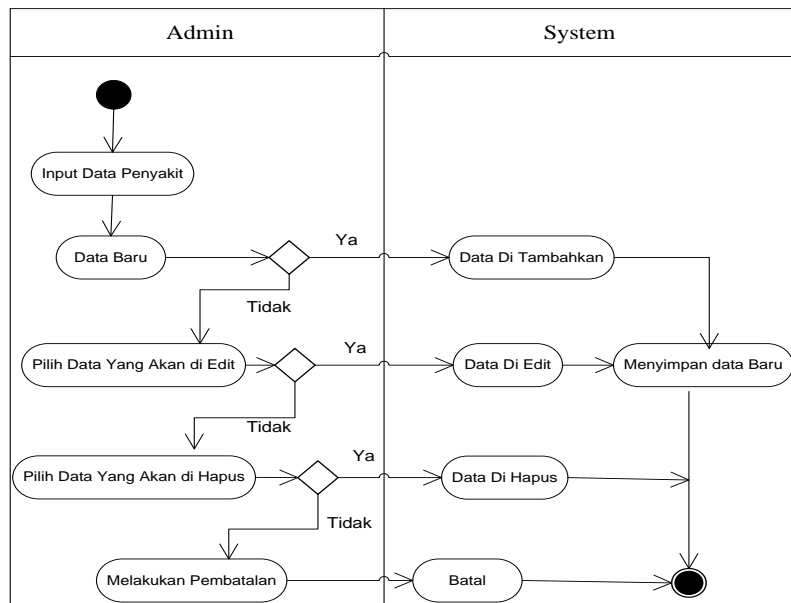
Berikut ini merupakan gambar *activity diagram* login admin.



**Gambar III.4. Logika Program Login**

## 2. Manipulasi Data Penyakit

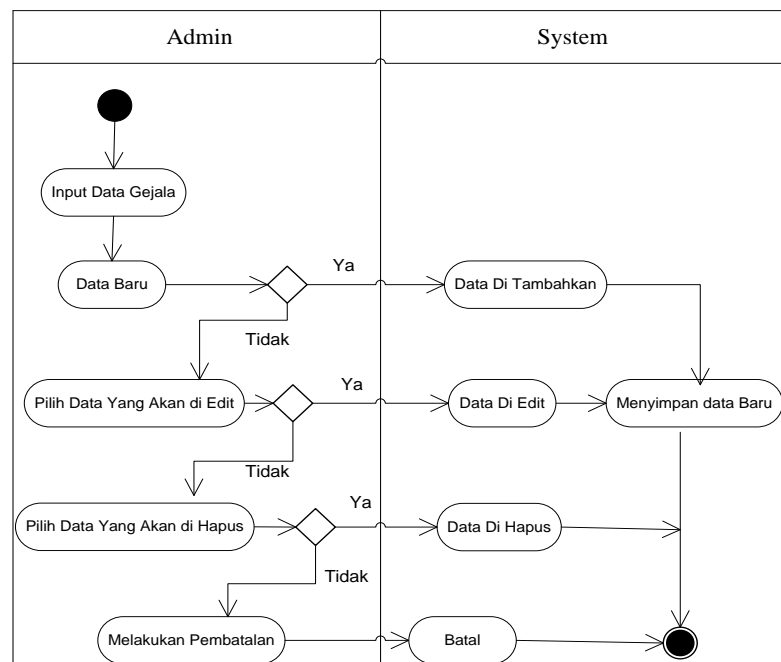
Berikut ini merupakan gambar *activity diagram* manipulasi data penyakit.



**Gambar III.5. Logika Program Manipulasi Data Penyakit**

### 3. Manipulasi Data gejala

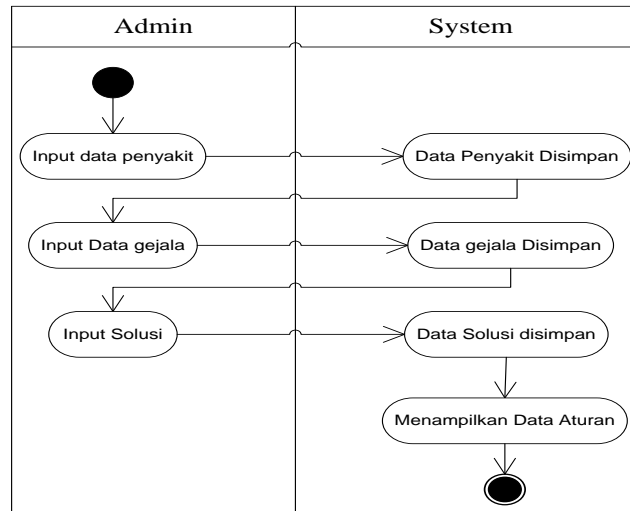
Berikut ini merupakan gambar *activity diagram* input gejala.



**Gambar III.6. Logika Program Input Data Gejala**

#### 4. Input Aturan

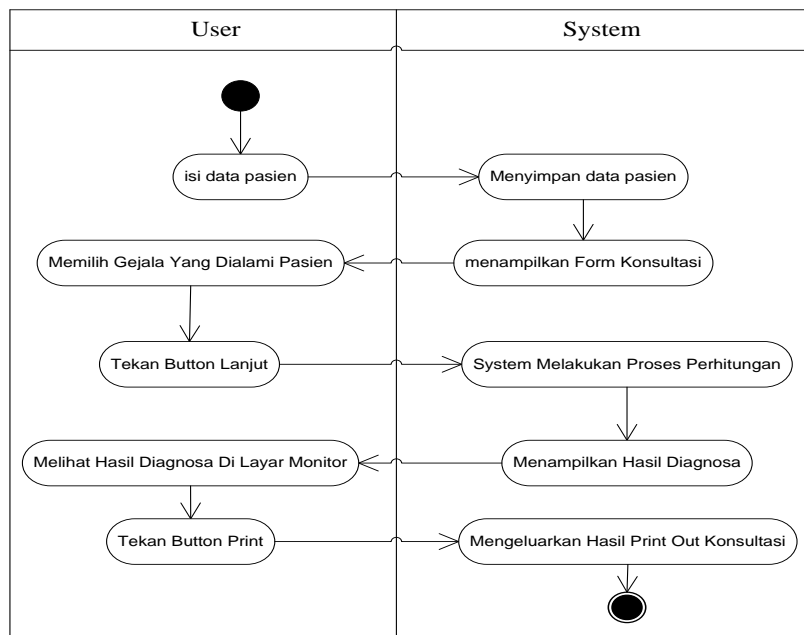
Berikut ini merupakan gambar *activity diagram* Input Aturan.



**Gambar III.7. Logika Program Manipulasi Aturan**

#### 5. Konsultasi.

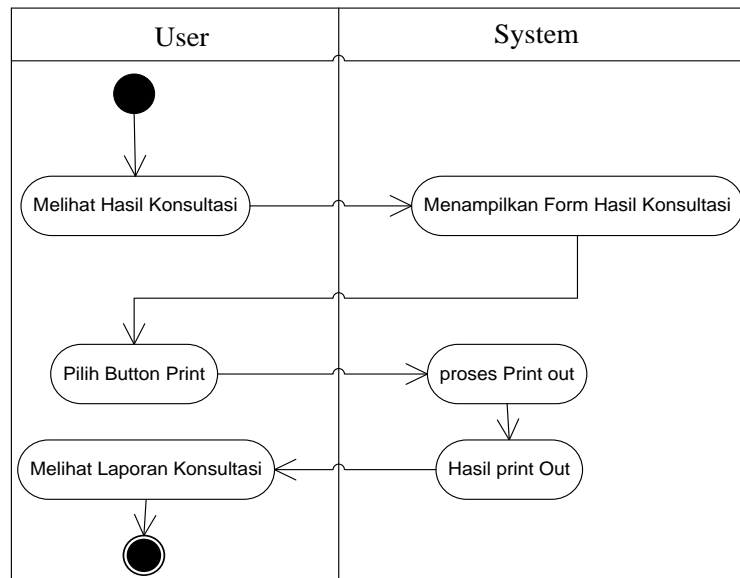
Berikut ini adalah *activity diagram* konsultasi.



**Gambar III.8. Logika Program Konsultasi**

## 6. Hasil Konsultasi

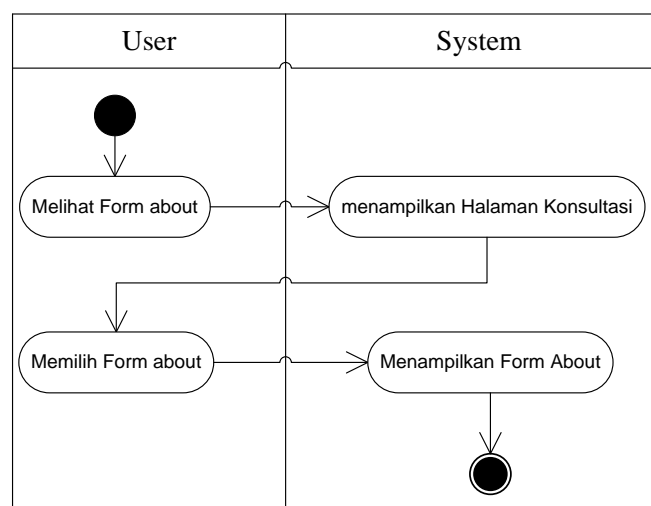
Berikut ini merupakan gambar *activity diagram* Hasil Konsultasi.



**Gambar III.9. Logika Program Hasil Konsultasi**

## 7. About

Berikut ini merupakan gambar *activity diagram* About.



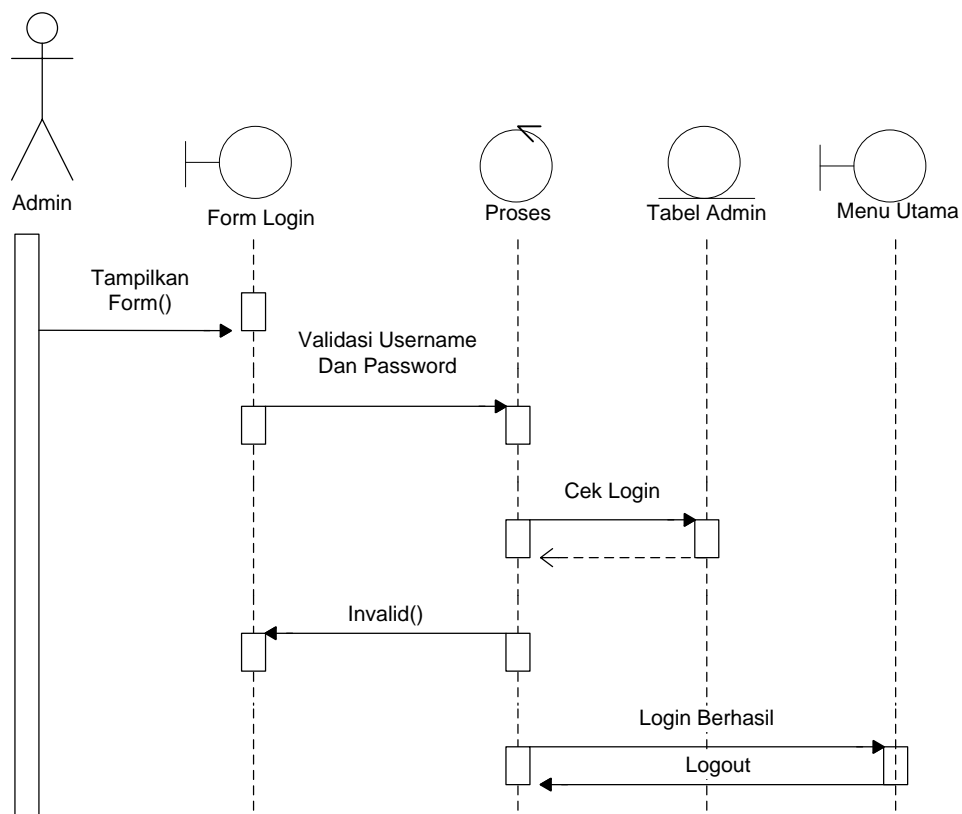
**Gambar III.10. Logika Program About**

### III.3.4. Sequence Diagram

*Sequence diagram* menggambarkan perilaku pada sebuah skenario, diagram ini menunjukkan sejumlah contoh objek dan pedan yang diletakkan diantara objek-objek ini didalam *use case*, berikut ini adalah gambar *sequence diagram*.

#### III.3.4.1. Sequence Diagram Login Admin

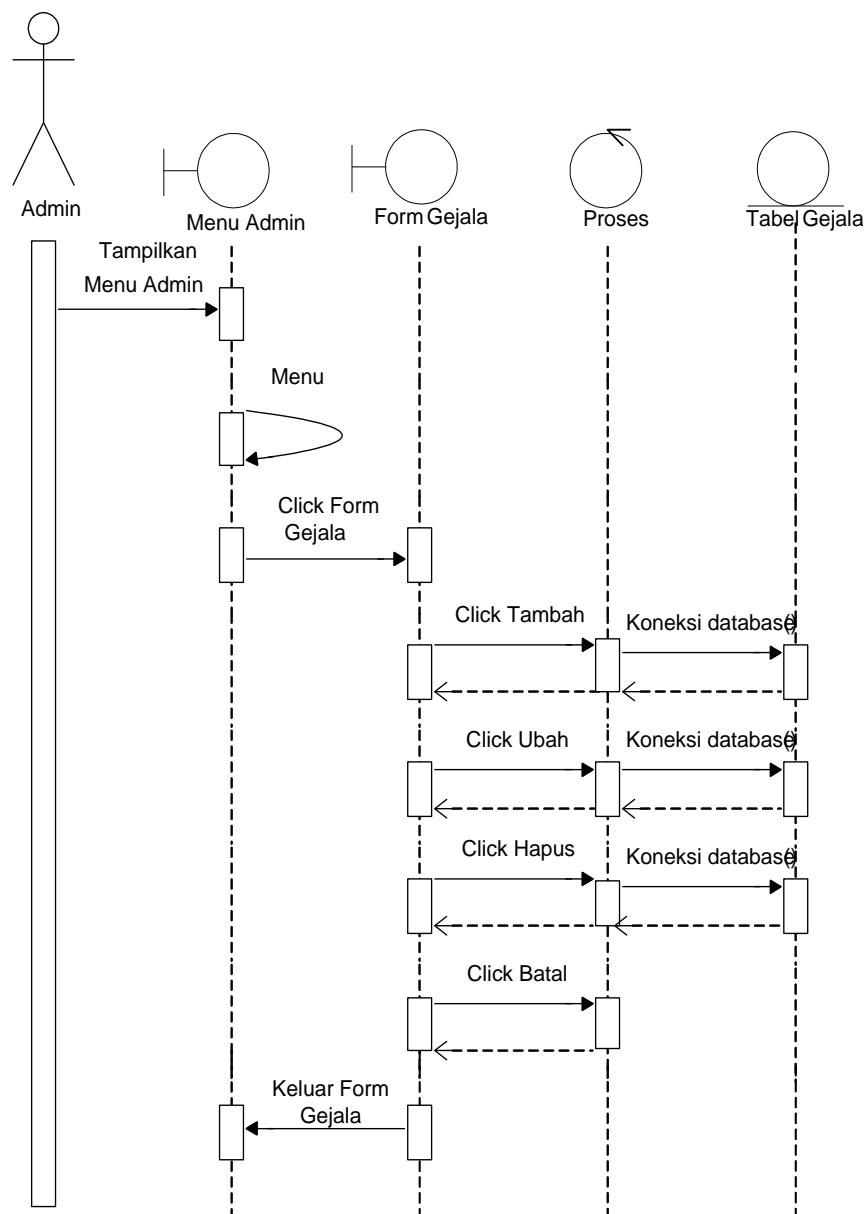
*Sequence Diagram login* menggambarkan interaksi yang terjadi antara objek yang menghasilkan *new user* dan tampilan menu utama. *Sequence diagram login* ditunjukkan pada gambar III.11. berikut ini :



**Gambar III.11. Sequence Diagram Login Admin**

### III.3.4.2. Sequence Diagram Manipulasi Data Gejala

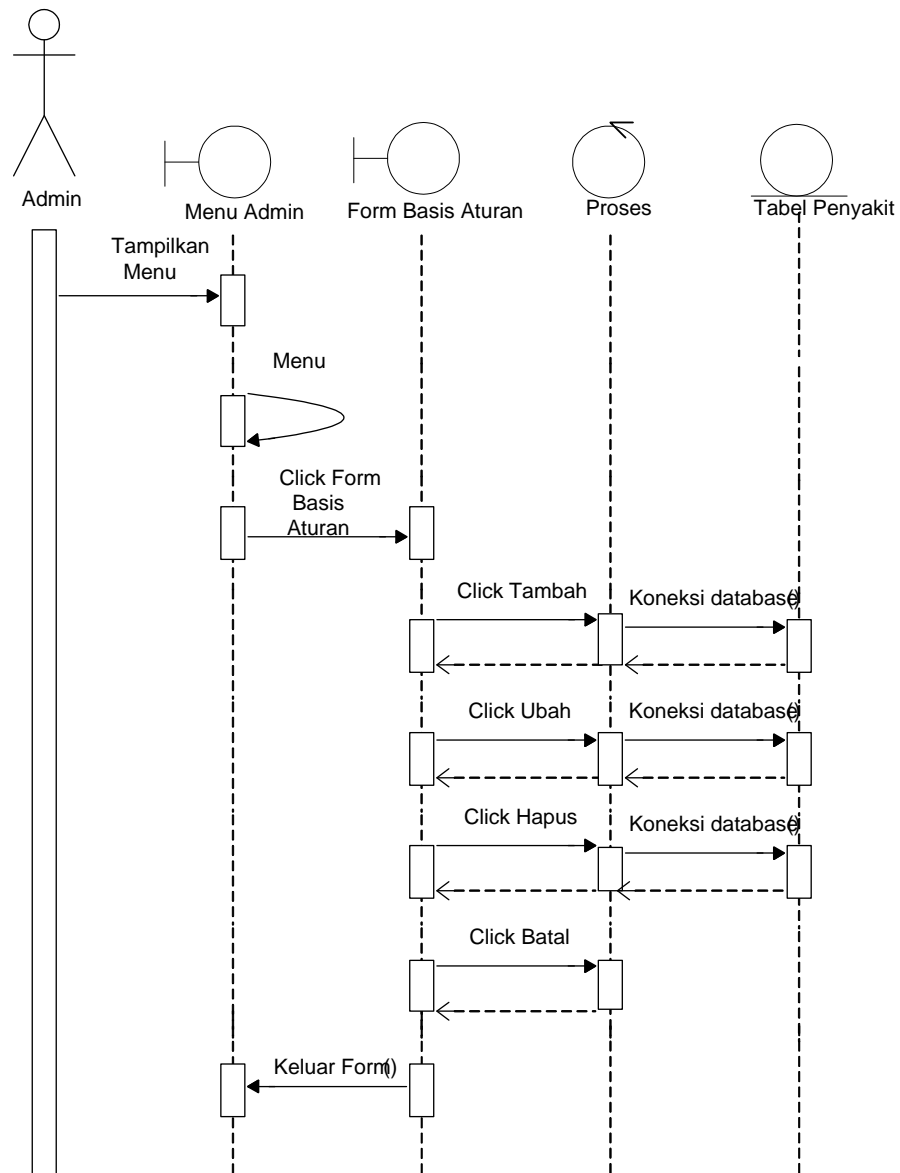
Sequence diagram data gejala menggambarkan interaksi yang terjadi antara objek yang menghasilkan data gejala. Sequence diagram data gejala ditunjukkan pada gambar III.12. berikut ini :



Gambar III.12. Sequence Diagram Manipulasi Data Gejala

### III.3.4.3. SequenceDiagram Manipulasi Data Penyakit

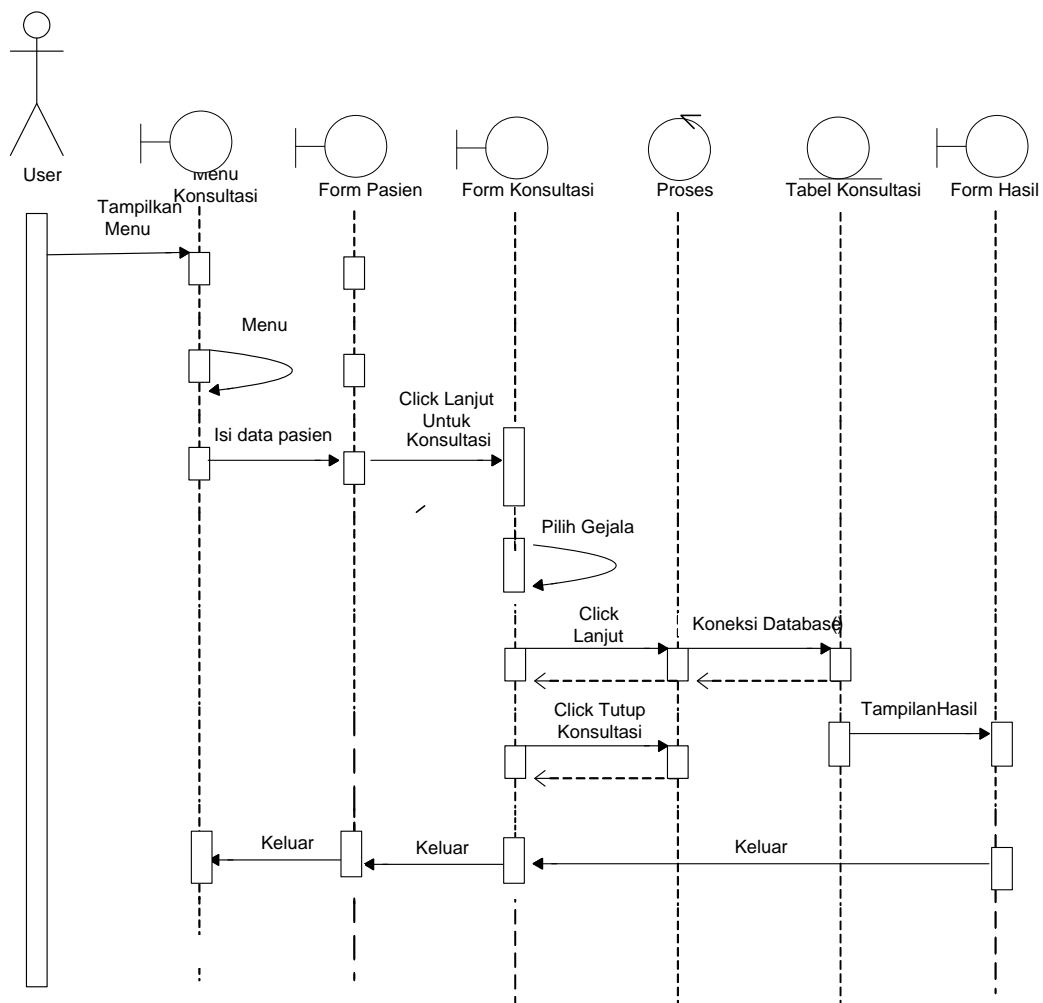
*Sequence diagram* data penyakit menggambarkan interaksi antara objek pada proses manipulasi penyakit. *Sequence diagram* penyakit dapat dilihat pada gambar III.13 berikut ini :



**Gambar III.13. Sequence Diagram Manipulasi Data Penyakit**

### III.3.4.4. Sequence Diagram Konsultasi

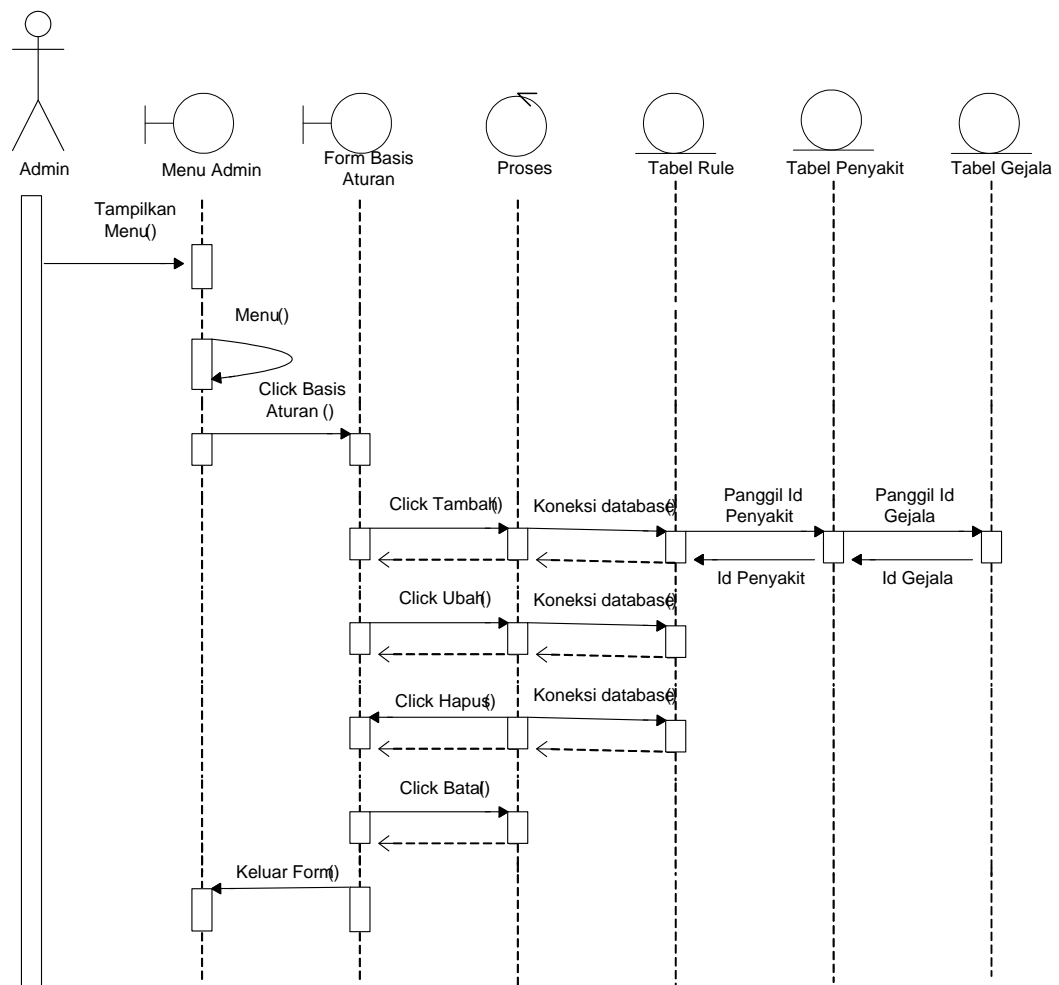
*Sequence diagram* Konsultasi menggambarkan interaksi yang terjadi antara objek yang menghasilkan konsultasi. *Sequence diagram* konsultasi ditunjukkan pada gambar III.14. berikut ini :



**Gambar III.14. Sequence Diagram Konsultasi**

### III.3.4.5. Sequence Diagram Manipulasi Aturan

*Sequence diagram* manipulasi aturan menggambarkan interaksi antara objek pada proses manipulasi aturan. *Sequence diagram* manipulasi aturan dapat dilihat pada gambar III.15 berikut ini :



**Gambar III.15. Sequence Diagram Manipulasi Aturan**

### III.3.4.7. Sequence Diagram About

*Sequence diagram* about menggambarkan interaksi antara objek pada proses about. *Sequence diagram* about dapat dilihat pada gambar III.17. berikut ini :



## 2. Bentuk Normal kedua (2NF).

## a. Tabel Admin

Id	Nama	Password

## b. Tabel Penyakit

Id_penyakit	Nm_Penyakit	Solusi

## c. Tabel Gejala

Id_Gejala	Nm_Gejala	Nilai Densitis

## 3. Bentuk Normal ketiga (3NF)

## a. Tabel Rule

Id_Rule	Id_Penyakit	Id_Gejala

## b. Tabel Gejala

Id_Gejala	Jawaban	Nilai Densitas

### III.4.2. Desain Tabel

Perancangan struktur databas adalah untuk menentukan *file database* yang digunakan seperti *field*, tipe data, ukuran data. Sistem ini dirancang dengan menggunakan database *SQL Server*.

Berikut adalah desain database dan tabel dari sistem yang dirancang :

#### 1. Tabel Admin

Nama Database : db\_pakar

Nama Tabel : tadmin

Primary Key : id\_admin

Foreign Key : -

**Tabel III.4. Struktur Tabel Admin**

No	Nama Field	Type Data	Nilai	Keterangan
1	Id	Nchar	10	Menyimpan nomer urut admin
2	Nama	Varchar	15	Menyimpan nama admin
3	Password	Varchar	15	Menyimpan password admin

#### 2. Tabel Rule

Nama Database : db\_pakar

Nama Tabel : trule

Primary Key : id\_penyakit

Foreign Key : id\_gejala

**Tabel III.5. Struktur Tabel Rule**

No	Nama Field	Type Data	Nilai	Keterangan
1	Id_penyakit	Nchar	5	Menyimpan nomer urut penyakit
2	Id_gejala	Nchar	10	Menyimpan nomer urut gejala
3	Rating	Number	(10,0)	Menyimpan nilai reteng

### 3. Tabel Gejala

Nama Database : db\_pakar

Nama Tabel : tgejala

Primary Key : -

Foreign Key : -

**Tabel III.6. Struktur Tabel Gejala**

No	Nama Field	Type Data	Nilai	Fungsi
1	Id_gejala	Nchar	10	menyimpan nomer urut gejala
2	Nm_gejala	Varchar	100	Menyimpan nama gejala
3	Nilai_densitis	Nchar	10	Menyimpan nilai densitis dari tiap gejala

### 4. Tabel Penyakit

Nama Database : db\_pakar

Nama Tabel : tpenyakit

Primary Key : id\_penyakit

Foreign Key : -

**Tabel III.7. Struktur Tabel Penyakit**

No	Nama Field	Type Data	Nilai	Fungsi
1	Id_penyakit	Nchar	5	menyimpan nomer urut penyakit
2	Nm_penyakit	Varchar	50	Menyimpan nama penyakit
3	Solusi	Text	-	Menyimpan solusi penyakit

## 5. Tabel Hasil

Nama Database : db\_pakar

Nama Tabel : tpenyakit

Primary Key : -

Foreign Key : -

**Tabel III.8. Struktur Tabel Hasil**

No	Nama Field	Type Data	Nilai	Fungsi
1	Id_hasil	int	-	menyimpan nomer urut penyakit
2	nm_pasien	Varchar	50	menyimpan nama pasien
3	alamat	Varchar	50	menyimpan alamat pasien
4	nm_penyakit	text	-	Menyimpan nama penyakit pasien
5	solusi	text	-	Menyimpan solusi pasien
6	nilai	text	-	Menyimpan nilai pasien
7	tgl	text	-	Menyimpan tanggal pasien

### III.5. Desain User Interface

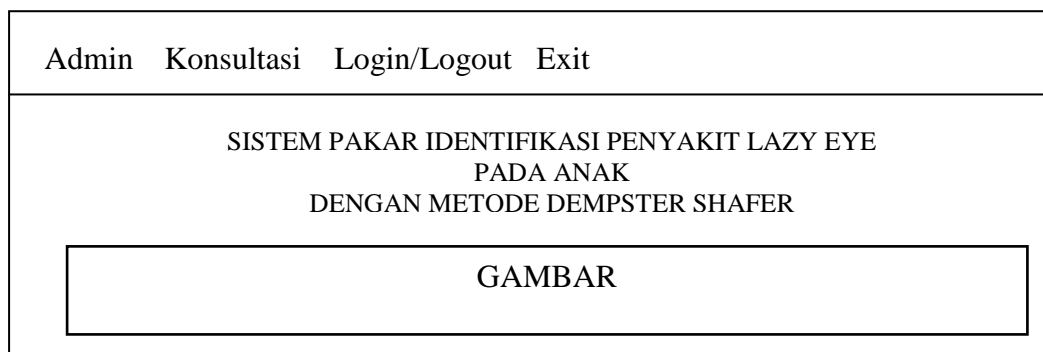
Desain User Interface dari sistem pakar identifikasi penyakit *Lazy eye* pada anak adalah sebagai berikut :

#### III.5.1. Desain Output

Desain ini berisikan pemilihan menu dan hasil pencarian yang telah dilakukan. Adapun bentuk rancangan output dari sisten pakar identifikasi penyakit *Lazy eye* adalah sebagai berikut :

##### 1. Menu Utama

Tampilan menu utama merupakan tampilan awal pada saat aplikasi dijalankan. Dibawah ini merupakan tampilan dari menu utama.



**Gambar III.18. Menu Utama**

##### 2. Profil penulis

Profil penulis merupakan halaman dari biodata si penulis.. Dibawah ini adalah tampilan dari halaman Profil Penulis

Sistem pakar identifikasi penyakit lazy eye pada anak dengan metode demster shafer

Profil penulis

Gambar

Nama :

Tempat, Tgl Lahir :

Agama :

Alamat :

Email :

Universitas :

Jurusan :

Fakultas :

Alamat Universitas :

OK

**Gambar III.19. Form Profil Penulis**

### 3. Form Hasil konsultasi

Form Hasil konsultasi berfungsi untuk melihat hasil konsultasi yang kita lakukan di form konsultasi. Dibawah ini merupakan gambar III.22. yaitu rancangan form hasil konsultasi.

gejala

Nama penyakit

Nilai dempster

solusi

Simpan

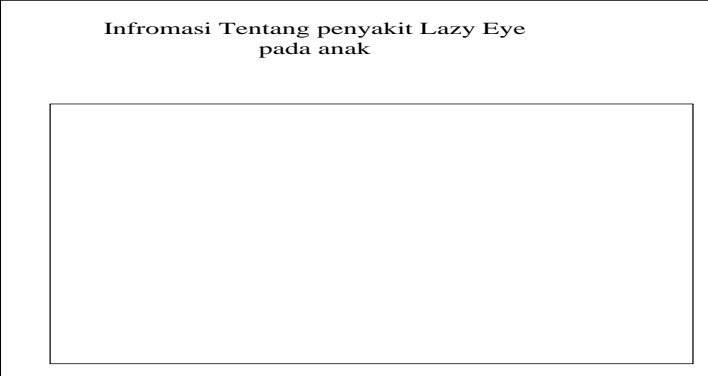
Print

Tutup Konsultasi

**Gambar III.20. Form Hasil Konsultasi**

#### 4. Form Informasi

Form ini merupakan rancangan untuk informasi tentang penyakit *lazy eye*, berikut ini adalah tampilan dari form Informasi.



The image shows a rectangular form with a double-line border. At the top center, the text reads "Infomasi Tentang penyakit Lazy Eye pada anak". Below this text is a large, empty rectangular box, likely intended for displaying information or an image.

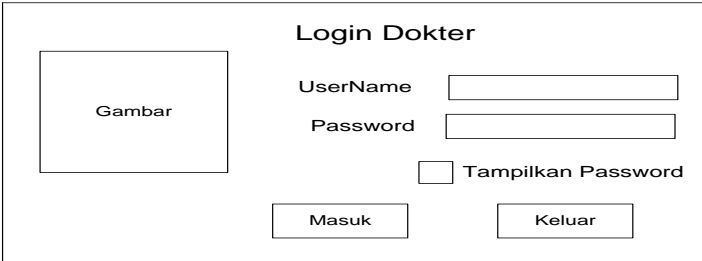
**Gambar III.21. From Informasi**

#### III.5.2. Desain Input

Berikut ini adalah rancangan *form* masukan (input) yang penulis gunakan dalam pembuatan Identifikasi penyakit *Lazy eye*.

##### 1. Form Login

Pada desain login yang menjadi inputan adalah username dan password. Tampilannya ada pada gambar III.23 sebagai berikut



The image shows a login form titled "Login Dokter". On the left side, there is a rectangular box labeled "Gambar". To the right of this box, the form contains the following elements: a "UserName" label followed by a text input field; a "Password" label followed by a text input field; a checkbox labeled "Tampilkan Password"; and two buttons at the bottom, "Masuk" and "Keluar".

**Gambar III.22. Form Login**

## 2. Form Gejala

Form ini merupakan rancangan untuk memanipulasi data dari gejala penyakit *Lazy eye*. berikut ini adalah tampilan dari form gejala.

**FORM GEJALA**

Id Gejala

Nama Gejala

Nilai Densitis

Kode	Gejala	Nilai Densitis

**Gambar III.23. From Gejala**

## 3. Form Input Aturan

Form ini berfungsi untuk menginputkan aturan atau rule baru oleh admin atau dokter. Di bawah ini merupakan gambar III.25. yaitu gambar rancangan form input aturan.

Kode Aturan :

Nama Penyakit :

KODE	GEJALA

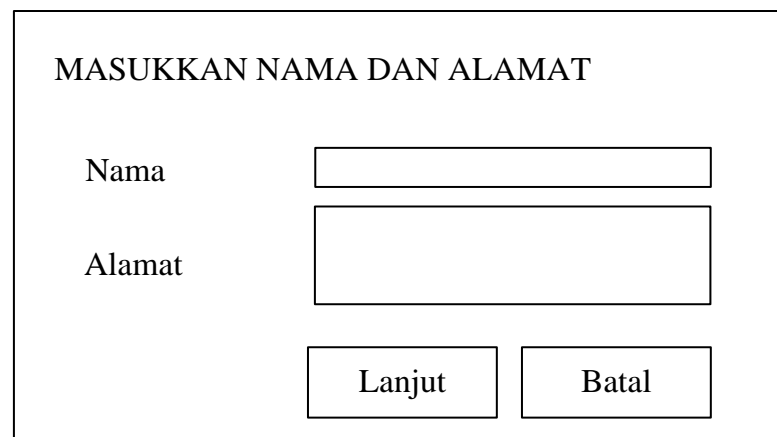
Gejala Dari Penyakit

solusi

**Gambar III.24. Form Input Aturan**

#### 4. Form Data Pasien

Form data pasien merupakan tampilan untuk menginputkan nama dan alamat pasien sebelum melakukan konsultasi. tampilan dari form data pasien dapat di lihat pada gambar III.27. berikut ini :



MASUKKAN NAMA DAN ALAMAT

Nama

Alamat

**Gambar III.25. Form Data Pasien**