

## **BAB III**

### **ANALISA DAN DESAIN SISTEM**

#### **III.1. Analisa Sistem Yang Berjalan**

Analisa sistem yang dijelaskan pada bab ini adalah sebagai bahan perbandingan dengan sistem yang akan dirancang. Penulis akan memaparkan proses konsultasi untuk mengetahui apakah seseorang menderita penyakit *Typus* dengan cara mendiagnosa berdasarkan gejala.

##### **III.1.1. Analisa *Input***

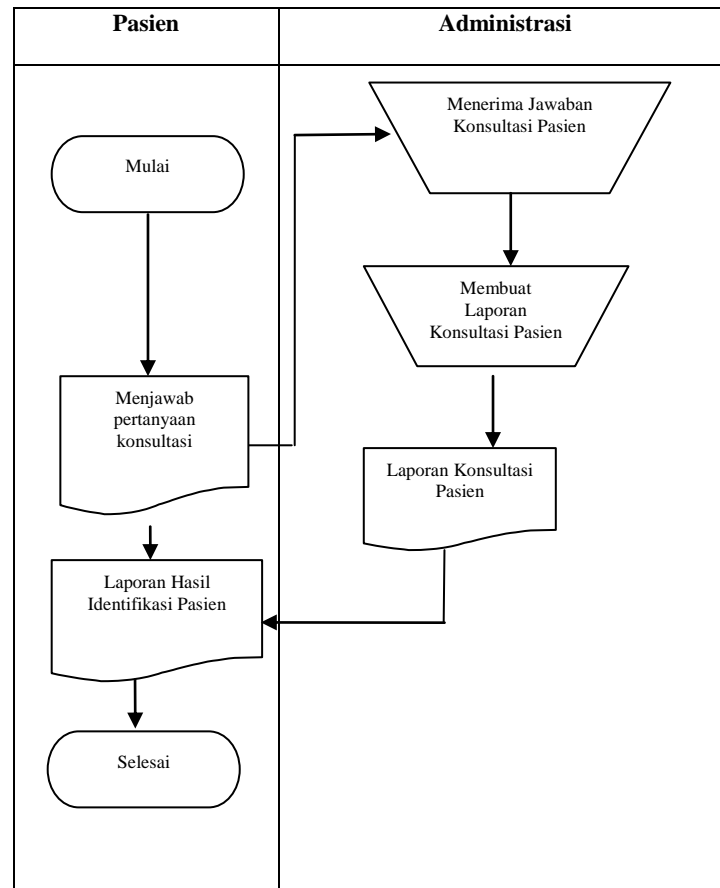
Agar proses mendiagnosa dapat dilakukan dan menghasilkan keluaran sesuai dengan yang diharapkan maka pakar perlu mengetahui data gejala yang dirasakan pasien, maka pakar akan mengajukan beberapa pertanyaan untuk kemudian mendiagnosa penyakit berdasarkan gejala-gejala yang dirasakan pasien.

##### **III.1.2. Analisa Proses**

Berdasarkan sistem yang sedang berjalan, tahapan-tahapan kerja atau proses mendiagnosa penyakit *Typus* adalah sebagai berikut :

1. Pasien di minta untuk menjawab pertanyaan yang di ajukan pakar.
2. Setelah pasien menjawab semua data yang diinputkan, pakar mendiagnosa penyakit *Typus*, apakah pasien menderita penyakit *Typus* atau tidak serta memberikan solusi dari penyakit *Typus*.

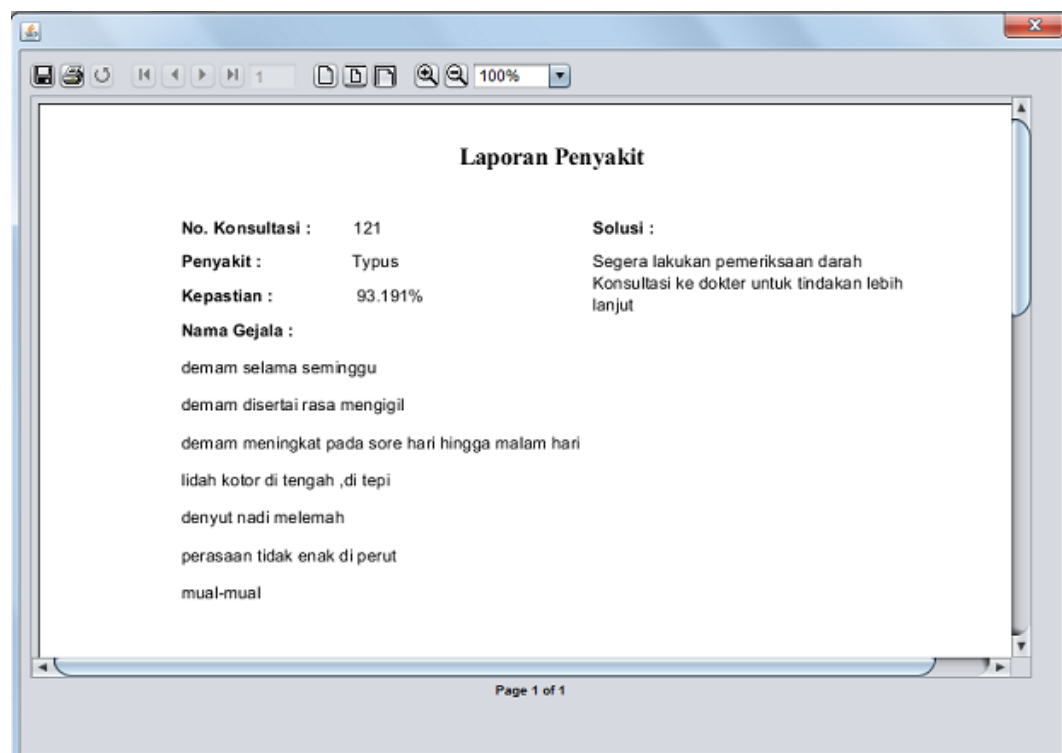
Adapun proses mendiagnosa penyakit Typus yang sedang berjalan dapat digambarkan dalam bentuk FOD ( *Flow Of Document* ) berikut ini :



**Gambar III.1. FOD ( *Flow Of Document* ) Sistem Pakar Mendiagnosa Penyakit Typus**

### III.1.3. Analisa *Output*

*Output* merupakan hasil dari pengolahan data yang telah diinputkan. *Output* atau hasil keluaran dari sistem pakar ini adalah apakah seseorang terkena penyakit *Typus* atau tidak, serta solusi dari hasil analisisnya. Hasil output dapat dilihat pada gambar III.2 sebagai berikut :



**Gambar III.2 Hasil Output dari sistem pakar diagnosa penyakit *Typus***

### III.2. Evaluasi sistem yang berjalan

Berdasarkan Analisa terhadap *input*, proses dan *output* pada sistem pakar mendiagnosa penyakit *Typus* yang sedang berjalan penulis menemukan beberapa kelemahan antara lain sebagai berikut :

1. Waktu yang dibutuhkan untuk melakukan konsultasi relatif tidak efektif karena pada umumnya pasien yang akan melakukan konsultasi harus membuat janji dan mengantri untuk bertemu dengan pakar.

2. Biaya yang dikeluarkan relatif mahal.
3. Waktu yang lama untuk menunggu hasil pemeriksaan darah.

Untuk menangani kelemahan-kelemahan sistem yang ada salah satu solusi yang ditawarkan adalah dengan merancang sistem pakar untuk mendiagnosa penyakit *Typus*. Sistem ini diharapkan mampu memberikan kontribusi positif bagi penanganan penyakit *Typus*.

Tabel keputusan untuk gejala-gejala yang terjadi adalah seperti ditunjukkan oleh tabel III.1 dibawah ini:

**Tabel III.1 Kumpulan *rule-rule* penyakit *Typus***

<b>Id_Gejala</b>	<b>Nama_Gejala</b>	<b>Penyakit</b>
G01	Apakah demam tinggi selama seminggu ?	P01
G02	Apakah lidah kotor ditengah,tepi dan ujung merah ?	P01
G03	Apakah nadi 80-100 kali permenit ?	P01
G04	Apakah muntah ?	P01
G05	Apakah sakit kepala ?	P01
G06	Apakah demam semakin meningkat hingga memasuki minggu ke dua?	P02
G07	Apakah lidah tampak kering,merah mengkilat ?	P02
G08	Apakah diare berwarna gelap ?	P02
G09	Apakah mengalami gangguan kesadaran ?	P02
G10	Apakah demam kembali normal?	P03
G11	Apakah otot-otot bergerak terus ?	P03
G12	Apakah terjadi tekanan diperut/nyeri perut hebat ?	P03
G13	Apakah keringat dingin ?	P03
G14	Apakah gelisah?	P03

**Sumber berdasarkan hasil wawancara dan Buku Ajar Ilmu Penyakit Dalam Jilid III Edisi V**

Adapun tabel diagnose penyakit dapat di lihat pada tabel III.2 sebagai berikut :

**Tabel III.2 Tabel Penyakit *Typus***

Id_Penyakit	Nama_Penyakit	SOLUSI
P01	Gejala Typus 1	Ini merupakan awal infeksi. segera lakukan pemeriksaan darah untuk memastikan hasil diagnosa. Konsultasi kedokter untuk mendapatkan penangan lebih lanjut.
P02	Gejala Typus 2	Ini merupakan gejala Typus lanjutan. Segera lakukan pemeriksaan darah untuk memastikan hasil diagnosa. Konsultasi kedokter untuk mendapatkan penangan lebih lanjut. Karena di tingkat ini gejala sudah menunjukan adanya perdarahan di perut.
P03	Gejala Typus 3	Ini merupakan tingkat dimana telah terjadi perdarahan dan perforasi usus. Segera lakukan tes darah. Melakukan konsultasi kedokter. Menjalani perawatan rawat inap untuk mendapatkan penanganan khusus.

**Sumber berdasarkan hasil wawancara dan Buku Ajar Ilmu Penyakit Dalam Jilid III Edisi V**

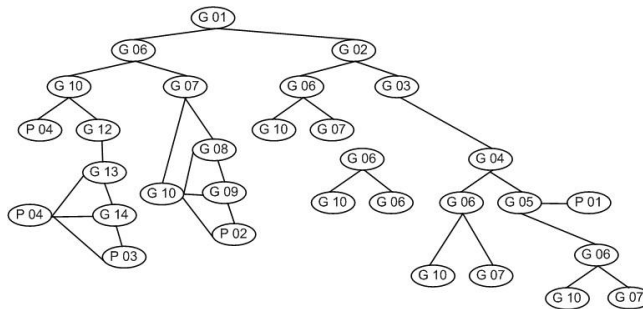
Tabel nilai kepastian dapat anda lihat pada gambar III.3 dibawah ini :

**Tabel III.3 Tabel Nilai Kepastian (Certainty Factor) untuk Gejala Penyakit *Typus***

Id_Gejala	Nama_Gejala	MB	MD	Penyakit
G01	Apakah demam tinggi selama seminggu ?	0.7	0.01	P01
G02	Apakah lidah kotor ditengah,tepi dan ujung merah ?	0.8	0.01	P01
G03	Apakah nadi 80-100 kali permenit ?	0.7	0.01	P01
G04	Apakah muntah ?	0.7	0.01	P01
G05	Apakah sakit kepala ?	0.7	0.01	P01
G06	Apakah demam semakin meningkat hingga memasuki minggu ke dua?	0.7	0.01	P02
G07	Apakah lidah tampak kering,merah mengkilat ?	0.7	0.01	P02

G08	Apakah diare berwarna gelap ?	0.7	0.01	P02
G09	Apakah mengalami gangguan kesadaran ?	0.7	0.01	P02
G10	Apakah demam kembali normal?	0.7	0.01	P03
G11	Apakah otot-otot bergerak terus ?	0.8	0.01	P03
G12	Apakah terjadi tekanan diperut/nyeri perut hebat ?	0.8	0.01	P03
G13	Apakah keringat dingin ?	0.7	0.01	P03
G14	Apakah gelisah?	0.7	0.01	P03

Berikut contoh pohon keputusan dari diagnose penyakit *Typus* dapat dilihat pada gambar III.3 di bawah ini :



**Gambar III.3 Contoh Pohon Keputusan Diagnosa Penyakit *Typus***

**Tabel III.4 Pembuatan Rule Diagnosa Penyakit *Typus***

Kode_Rule	ID_Gejala	Ya	Tidak
R01	G01	G02	G06
R02	G02	G03	G06
R03	G03	G04	G06
R04	G04	G05	G06
R05	G05	P01	G06
R06	G06	G07	G10
R07	G07	G08	G10
R08	G08	G09	G10
R09	G09	P02	G10
R10	G10	G11	P04
R11	G11	G12	P04

R12	G12	G13	P04
R13	G13	G14	P04
R14	G14	P03	P04

### III.3 Metode *Certainty Factor*

Metode *certainty factor* yang akan diterapkan dalam pembuatan sistem pakar ini adalah metode dengan rumus *certainty factor* sebagai berikut :

$$\text{CF}[h,e] = \text{MB}[h,e] - \text{MD}[h,e]$$

Keterangan :

CF[h,e] = Certainty Factor dalam hipotesis h yang dipengaruhi oleh fakta e.

MB[h,e] = *Measure of Believe*, merupakan nilai kenaikan dari kepercayaan hipotesis h dipengaruhi oleh fakta e.

MD[h,e] = *Measure of Disbelieve*, merupakan nilai kenaikan dari ketidakpercayaan hipotesis h dipengaruhi oleh fakta e.

h = Hipotesis.

e = Evidence.

Untuk mendapatkan nilai CF di masing-masing penyakit, terlebih dahulu harus dicari nilai MB dan MD penyakit tersebut. Proses perhitungan dilakukan berdasarkan rumusan dari CF.

### III.4. Desain Sistem

Untuk membantu membangun rancang bangun sistem pakar mendiagnosa penyakit Typus, penulis mengusulkan pembuatan sebuah sistem dengan menggunakan aplikasi program yang lebih akurat dan lebih mudah dalam

pengolahannya. Dengan menggunakan bahasa pemograman *Java* dengan editor *Netbeans* dan database *MySQL*. Desain perancangan menggunakan UML (*Unified Modeling Language*).

### **III.4.1 Desain Sistem**

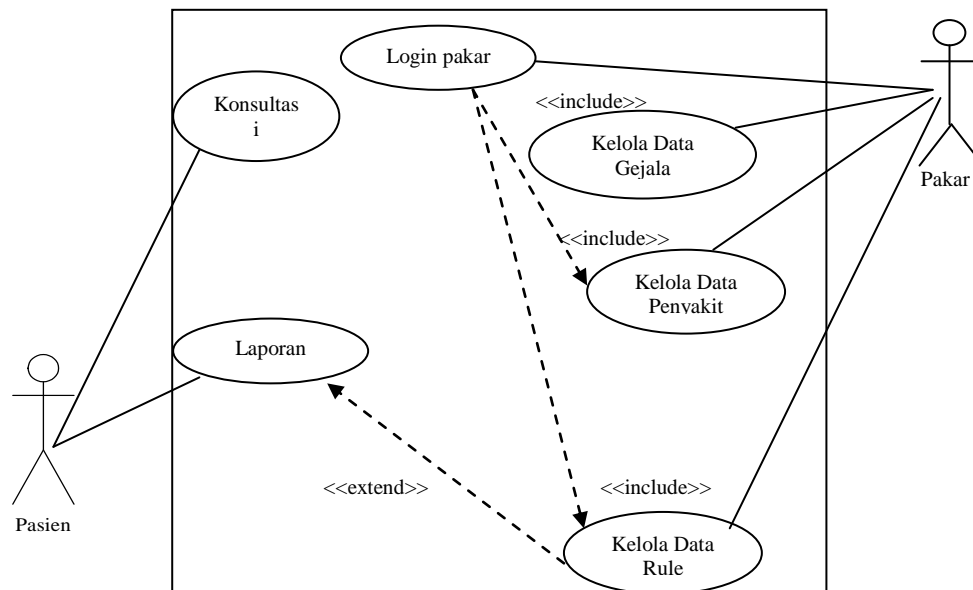
Pada perancangan sistem ini terdiri dari tahap perancangan yaitu :

1. Perancangan *Use Case Diagram*
2. Perancangan *Class Diagram*
3. Perancangan *Sequence Diagram*
4. Perancangan *Activity Diagram*
5. Perancangan *Rule*

#### **III.4.1.1. Use Case Diagram**

Diagram yang menggambarkan *actor*, *use case* dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk *actor*. Sebuah *use case* digambarkan sebagai elips horizontal dalam suatu diagram UML *use case*, dapat dilihat pada gambar III.4. Sebagai berikut :

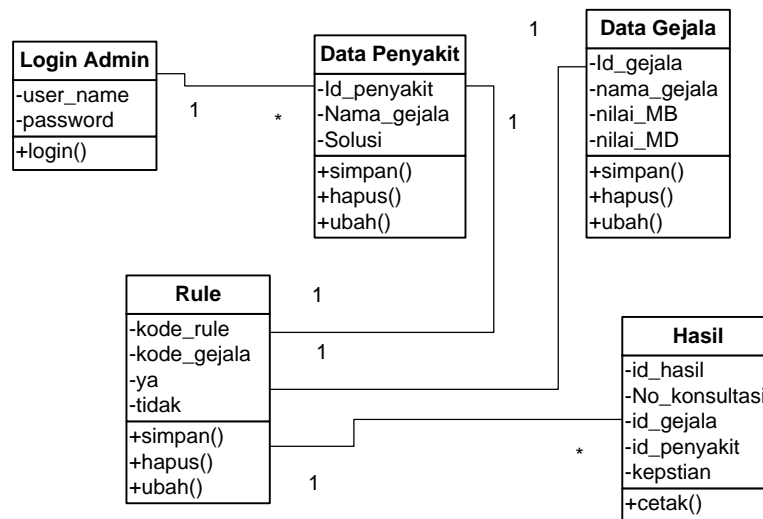




**Gambar III.4. Use Case Diagram Sistem Pakar Mendiagnosa Penyakit Typus dengan Menggunakan Metode Certainty Factor**

#### III.4.1.2. Class Diagram

*Class Diagram* adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi), berikut gambar *Class Diagram* dapat dilihat pada Gambar III.5 sebagai berikut :



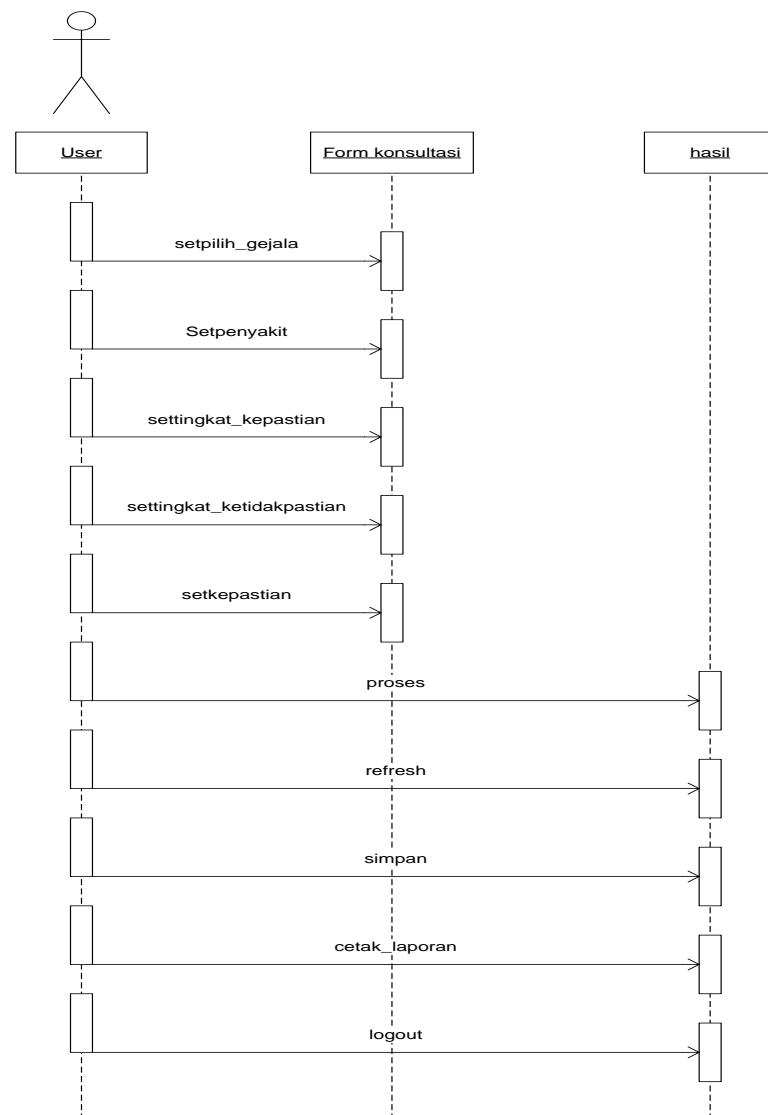
**Gambar III.5. Class Diagram Sistem Pakar Mendiagnosa Penyakit Typus  
dengan Menggunakan Metode Certainty Factor**

#### III.4.1.3. Sequence Diagram

*Sequence diagram* menjelaskan interaksi objek yang disusun berdasarkan urutan waktu. Secara mudahnya *sequence diagram* adalah gambaran tahap demi tahap, termasuk kronologi (urutan) perubahan secara logis yang seharusnya dilakukan untuk menghasilkan sesuatu sesuai dengan *use case diagram*, berikut beberapa gambar *sequence diagram* :

##### a. Sequence Diagram Konsultasi

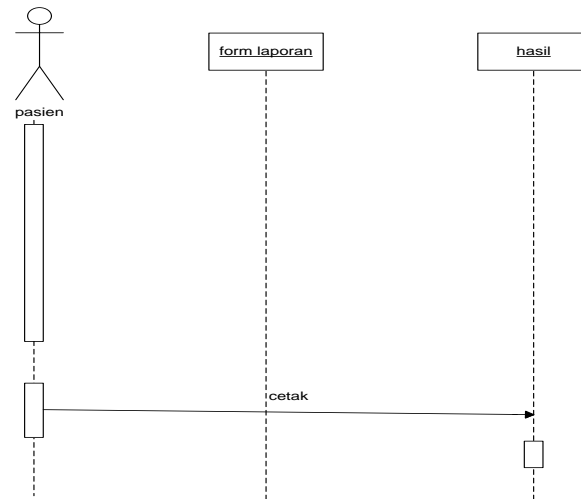
*Sequence Diagram* Konsultasi menggambarkan aktivitas *user* untuk melakukan konsultasi penyakit dan mengetahui bagaimana solusinya. *Sequence diagram* konsultasi ditunjukkan pada gambar III.6 berikut ini:



**Gambar III.6. Sequence Diagram Konsultasi**

**b. Sequence Diagram Hasil**

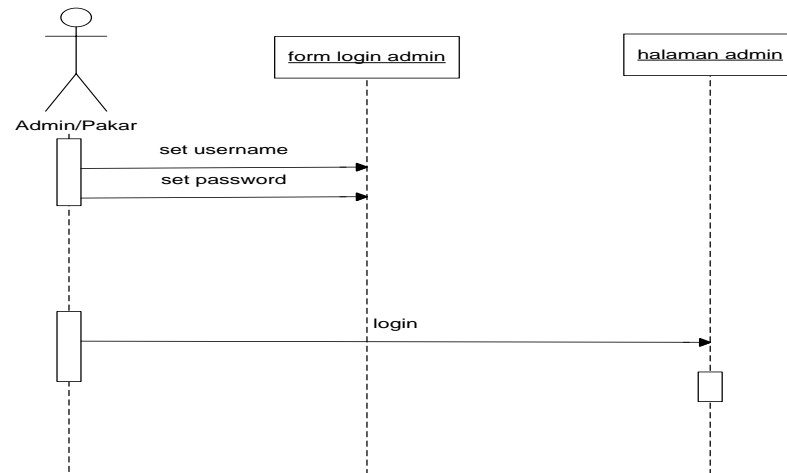
*Sequence diagram* hasil menggambarkan interaksi antar objek pada proses hasil. *Sequence diagram* hasil ditunjukkan pada gambar III.7 berikut ini:



**Gambar III.7. Sequence Diagram Hasil**

c. *Sequence Diagram Login Admin/Pakar*

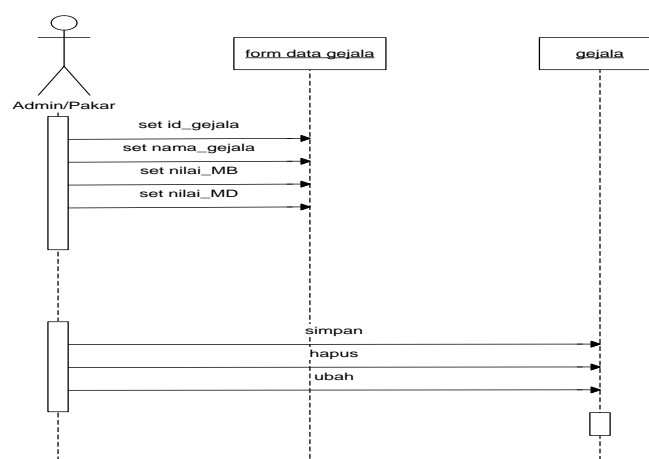
*Sequence Diagram Login Admin* menggambarkan aktivitas administrasi melakukan *login* sebelum masuk ke halaman admin untuk melakukan aktivitas yang lain administrasi harus menginputkan *username* dan *password* yang valid dan kemudian akan diproses dan akan diketahui *username* dan *password* tersebut valid atau tidak. *Sequence diagram* login admin ditunjukkan pada gambar III.8 berikut ini:



**Gambar III.8. Sequence Diagram Login Admin/Pakar**

d. *Sequence Diagram Data Gejala*

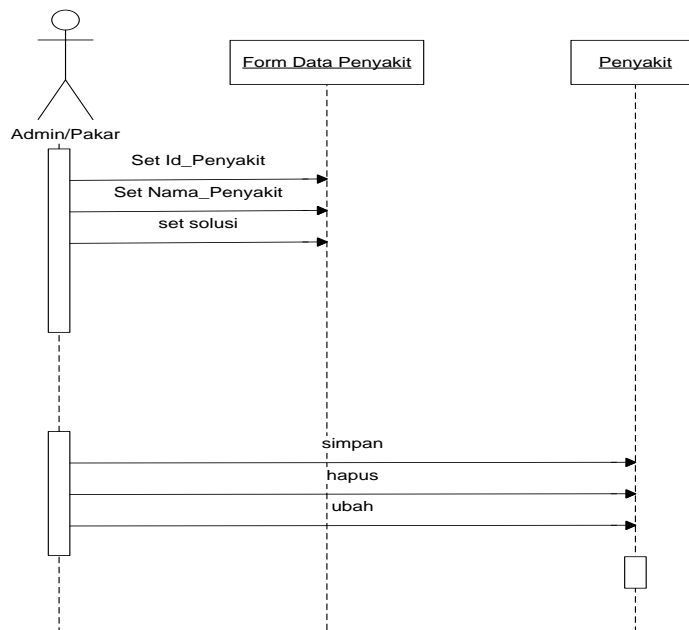
*Sequence Diagram Data Gejala* menggambarkan untuk melakukan entri data gejala ke *database* dimana administrasi dapat melakukan menyimpan dan menghapus data gejala tersebut. *Sequence diagram* ditunjukkan pada gambar III.9 berikut ini:



**Gambar III.9. Sequence Diagram Data Gejala**

e. *Sequence Diagram Data Penyakit*

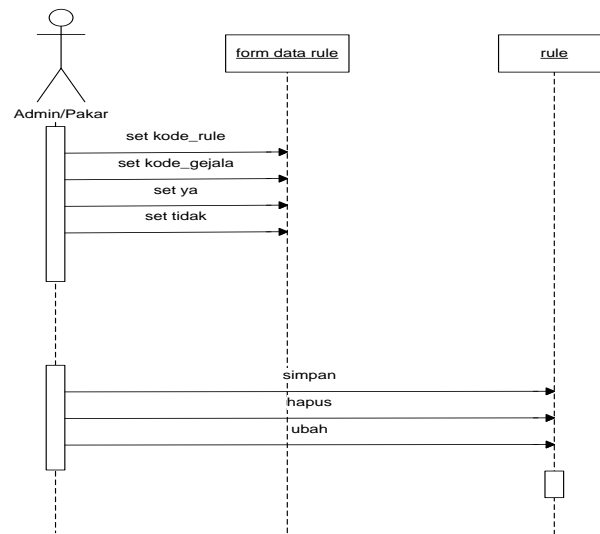
*Sequence Diagram Data penyakit* menggambarkan aktivitas untuk melakukan entri data kerusakan ke *database* dimana administrasi dapat melakukan menyimpan dan menghapus data penyakit tersebut. *Sequence diagram* ditunjukkan pada gambar III.10 berikut ini:



**Gambar III.10. *Sequence Diagram Data Penyakit***

f. *Sequence Diagram Data Rule*

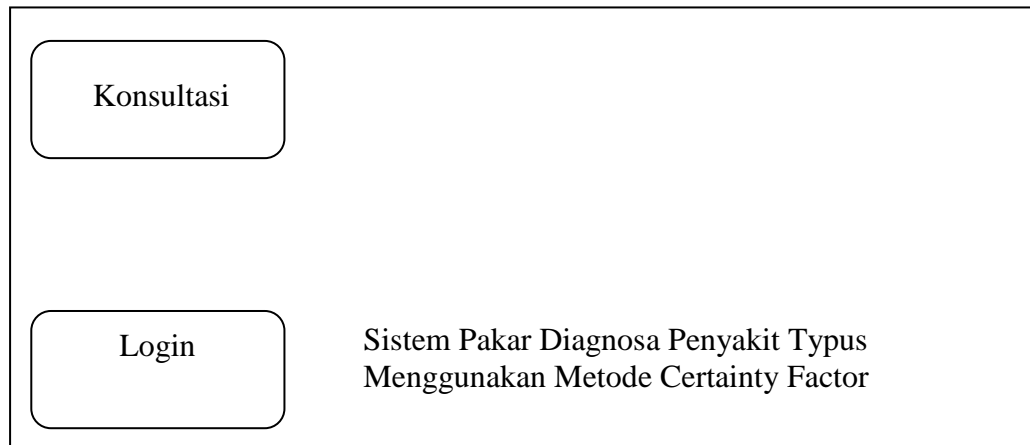
*Sequence Diagram Data Rule* menggambarkan aktivitas administrasi untuk melakukan kumpulan *rule - rule* penyakit dimana administrasi dapat melakukan menyimpan dan menghapus data kumpulan *rule - rule* penyakit tersebut. *Sequence diagram* ditunjukkan pada gambar III.11 berikut ini:



**Gambar III.11. Sequence Diagram Data Rule**

#### III.4.1.4. Desain Ouput

Terdapat beberapa antar muka yang menjadi output dari sistem yang akan di bangun yaitu tampilan awal, konsultasi dan laporan hasil, desain output tampilan dapat di lihat pada gambar III.12 sebagai berikut :



**Gambar III.12 Desain Output Tampilan Awal**

Tampilan desain output konsultasi dapat dilihat pada gambar III.13 dibawah ini :

Konsultasi :

Pertanyaan

☐ Ya
☐ Tida

Simpan

Penyakit yang di derita

Tingkat Kepastian (MB)

Tingkat Kepastian (MD)

Tingkat Kepastian

Solusi

Log in

Sistem Pakar Diagnosa Penyakit Typus Menggunakan Metode Certainty Factor

Cetak

Refresh

**Gambar III.13 Desain output konsultasi**

Desain output laporan hasil dapat di lihat pada gambar III.14 sebagai berikut :



**Laporan Penyakit**

<b>No. Konsultasi :</b>	121	<b>Solusi :</b>	
<b>Penyakit :</b>	Typus	<b>Solusi :</b>	Segera lakukan pemeriksaan darah
<b>Kepastian :</b>	93.191%	<b>Solusi :</b>	Konsultasi ke dokter untuk tindakan lebih lanjut
<b>Nama Gejala :</b>	demam selama seminggu demam disertai rasa mengigil demam meningkat pada sore hari hingga malam hari lidah kotor di tengah ,di tepi denyut nadi melemah perasaan tidak enak di perut mual-mual		

Page 1 of 1

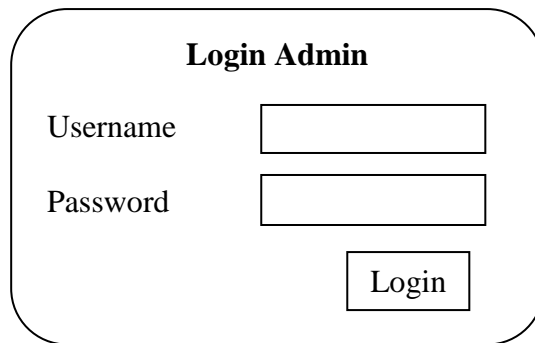
**Gambar III.14 Desain output laporan hasil**

#### III.4.1.5 Desain Input

Perancangan *input* merupakan masukan yang penulis desain guna lebih memudahkan dalam *entry* data. *Entry* data yang dirancang akan lebih mudah dan cepat untuk meminimalisir kesalahan penulisan dan memudahkan perubahan. Perancangan *input* tampilan yang dirancang adalah sebagai berikut :

##### 1. Desain *Input Login Admin/Pakar*

Perancangan *input login admin/pakar* berfungsi untuk verifikasi pengguna yang berhak menggunakan sistem. Adapun rancangan *login admin/pakar* dapat dilihat pada Gambar III.15 sebagai berikut :



**Login Admin**

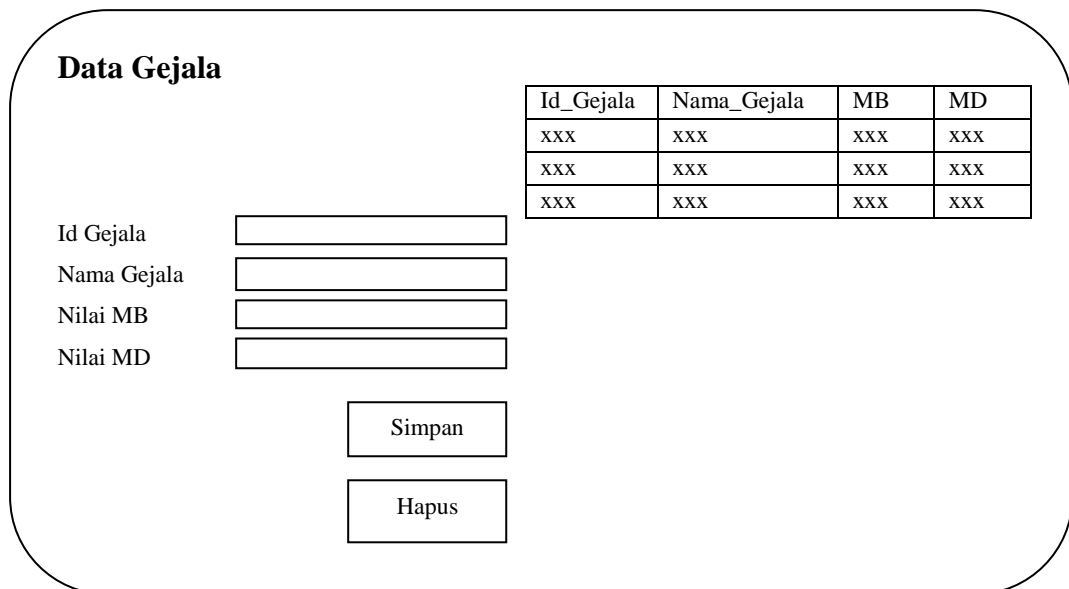
Username

Password

**Gambar III.15. Desain Input Login Admin/Pakar**

## 2. Desain *Input* Data Gejala

Perancangan *input* data gejala merupakan form untuk penyimpanan data - data gejala, nilai kepastian dan nilai ketidakpastian. Adapun bentuk *input* data gejala dapat dilihat pada Gambar III.16 Sebagai berikut :



**Data Gejala**

Id_Gejala	Nama_Gejala	MB	MD
xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx
xxx	xxx	xxx	xxx

Id Gejala

Nama Gejala

Nilai MB

Nilai MD

**Gambar III.16. Desain Input Data Gejala**

### 3. Desain *Input* Data Penyakit

Perancangan *input* data penyakit merupakan form untuk penyimpanan data - data penyakit dan solusi. Adapun bentuk *input* data penyakit dapat dilihat pada Gambar III.17 Sebagai berikut :

#### Data Penyakit

Id\_penyakit

Nama\_penyakit

Solusi

Id_penyakit	Nama_penyakit	Solusi
xxx	xxx	xxx
xxx	xxx	xxx
xxx	xxx	xxx

**Gambar III.17. Desain *Input* Data Penyakit**

#### III.4.1.6 Desain *Database*

##### 1. Kamus data (*Data Dictionaries*)

Kamus data merupakan suatu daftar terorganisasi tentang komposisi elemen data, aliran data dan data store yang digunakan. Pengisian data dictionary dilakukan setiap saat selama proses pengembangan berlangsung, ketika diketahui adanya data atau saat diperlukan penambahan data item ke dalam sistem. Berikut kamus data dari sistem pakar mendiagnosa penyakit *Typus* sebagai berikut :

1. Login = ({username + password})

2. Data\_gejala = ({id\_gejala + nama\_gejala + nilai\_MB + nilai\_MD})
3. Data\_Penyakit = ({id\_penyakit + nama\_penyakit + solusi})
4. Rule = ({id\_penyakit + id\_gejala + Ya + Tidak})
5. Hasil = ({id\_hasil + No\_konsultasi + Id\_gejala + Id\_penyakit + kepastian})

## 2. Normalisasi

Normalisasi adalah proses pengelompokkan data ke dalam bentuk tabel atau relasi atau file untuk menyatakan entitas dan hubungan mereka sehingga terwujud satu bentuk database yang mudah untuk dimodifikasi. Berikut ini adalah bentuk normalisasi yang dilakukan dalam sistem mendiagnosa penyakit *Typus* sebagai berikut :

### First Normal Form (1NF)

Bentuk normal yang pertama atau 1NF mensyaratkan beberapa kondisi dalam sebuah database, berikut adalah fungsi dari bentuk normal pertama ini.

- Menghilangkan duplikasi kolom dari tabel yang sama.
- Buat tabel terpisah untuk masing-masing kelompok data terkait dan mengidentifikasi setiap baris dengan kolom yang unik (primary key).

Contoh Normalisasi Database 1NF :

Id_gejala	Nama_gejala	MB	MD
G01	Apakah demam tinggi selama seminggu ?	0.7	0.01
G02	Apakah lidah kotor ditengah,tepi dan ujung merah ?	0.8	0.01
G04	Apakah nadi 80-100 kali permenit ?	0.7	0.01

Id_kerusakan	Nama_kerusakan	Solusi
P01	Gejala Typus 1	Ini merupakan awal infeksi.

		segera lakukan pemeriksaan darah untuk memastikan hasil diagnosa. Konsultasi kedokter untuk mendapatkan penanganan lebih lanjut.
P02	Gejala Typus 2	Ini merupakan gejala typus lanjutan. Segera lakukan pemeriksaan darah untuk memastikan hasil diagnosa. Konsultasi kedokter untuk mendapatkan penanganan lebih lanjut. Karena di tingkat ini gejala sudah menunjukkan adanya perdarahan di perut.

kode_Rule	Id_gejala	Ya	Tidak
R01	G01	G02	G06
R02	G02	G03	G06
R03	G03	G04	G06
R04	G04	G05	G06
R05	G05	P01	G06

Pada intinya bentuk normalisasi 1NF ini mengelompokkan beberapa tipe data atau kelompok data yang sejenis agar dapat dipisahkan sehingga data dapat di atasi. Contoh adalah ketika kita ingin menghapus, mengupdate, atau menambahkan data di tabel gejala, maka kita tidak bersinggungan dengan data tabel kerusakan atau data tabel rule. Sehingga data dapat mulai di jaga.

## Second Normal Form (2NF)

Syarat untuk menerapkan normalisasi bentuk kedua ini adalah data telah dibentuk dalam 1NF, berikut adalah beberapa fungsi normalisasi 2NF.

- a. Menghapus beberapa subset data yang ada pada tabel dan menempatkan mereka pada tabel terpisah.
- b. Menciptakan hubungan antara tabel baru dan tabel lama dengan menciptakan foreign key.

Contoh Normalisasi Database 2NF :

Nama_Gejala	Id_Gejala	Id_Penyakit	Id_Rule	Nama_Penyakit
Demam selama seminggu	G01	P01	R01	Gejala Typus 1
Apakah lidah kotor di tengah, tepi dan ujung merah	G02	P01	R02	Gejala Typus 1

Bentuk 2NF dari tabel diatas

Id_Gejala	Id_Kerusakan	Id_rule
G01	P01	R01
G02	P01	R02

pada intinya bentuk kedua ini adalah tidak boleh ada field yang berhubungan dengan field lainnya secara fungsional. Contoh nama gejala tergantung dengan id\_gejala sehingga dalam bentuk 2NF nama gejala dapat di hilangkan karena telah memiliki tabel master tersendiri.

#### III.4.1.7. Desain Tabel/File

Perancangan struktur database adalah untuk menentukan *file database* yang digunakan seperti *field*, tipe data, ukuran data. Sistem ini didesain dengan

menggunakan database *MYSQL*. Berikut adalah desain database dan tabel dari sistem yang dirancang.

a. Tabel Admin

Nama Database : sp\_tipas

Nama Tabel : admin

Primary Key : username

**Tabel III.5. Tabel Admin**

Nama Field	Tipe Data	Ukuran	Keterangan
*username	Varchar	20	Primary key
password	Varchar	20	Password

b. Tabel Gejala

Nama Database : sp\_tipas

Nama Tabel : gejala

Primary Key : id\_gejala

**Tabel III.6. Tabel Gejala**

Nama Field	Tipe Data	Ukuran	Keterangan
*Id_gejala	Varchar	10	Primary key
Nama_gejala	Varchar	50	Nama_gejala
MB	Double	-	MB
MD	Double	-	MD

c. Tabel Penyakit

Nama Database : sp\_tipas

Nama Tabel : penyakit

Primary Key : id\_penyakit

**Tabel III.7. Tabel Penyakit**

Nama Field	Tipe Data	Ukuran	Keterangan
*Id_Penyakit	Varchar	10	Primary key
Nama_Penyakit	Varvhar	30	Nama_penyakit
Solusi	Text	-	Solusi

## d. Tabel Rule

Nama Database : sp\_tipus

Nama Tabel : rule

Primary Key : kode\_rule

**Tabel III.8. Tabel Rule**

Nama Field	Tipe Data	Ukuran	Keterangan
*kode_rule	Int	20	Primary key
Id_gejala	Varchar	20	Id_gejala
Ya	Varchar	20	Ya
Tidak	Varchar	20	Tidak

## e. Tabel Hasil

Nama Database : sp\_tipus

Nama Tabel : Hasil

Primary key : Id\_hasil

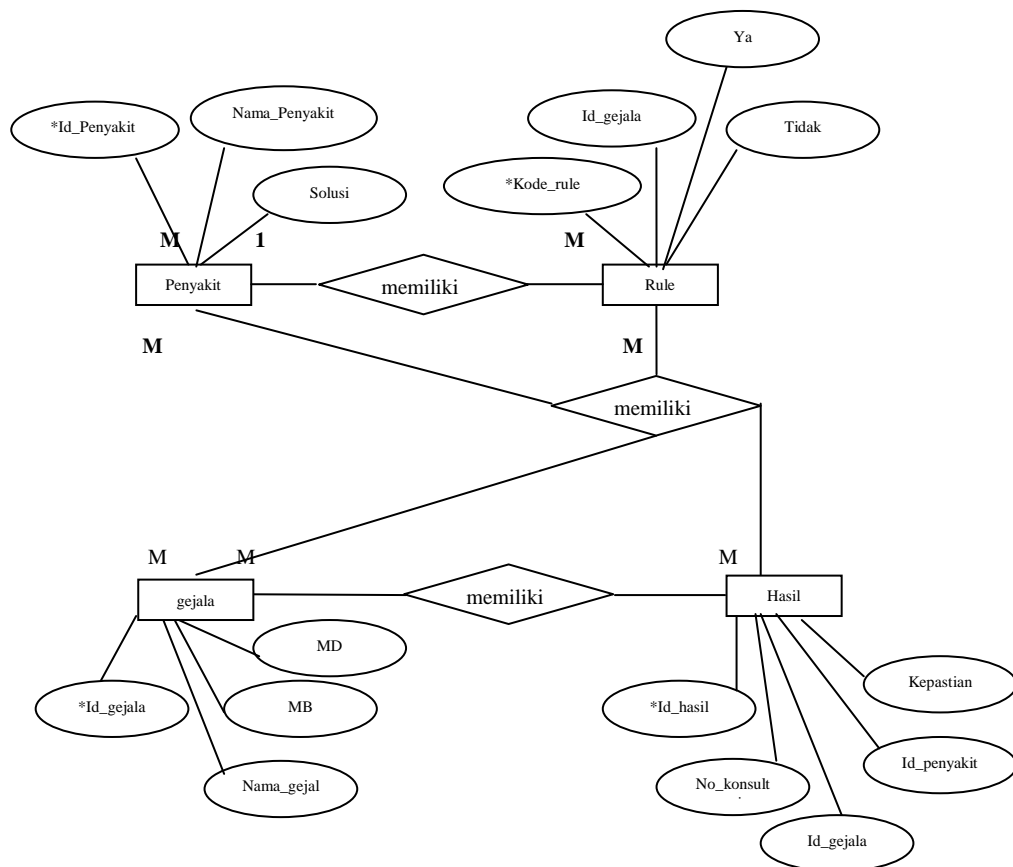
**Tabel III.9 . Tabel Hasil**

Nama Field	Tipe Data	Ukuran	Keterangan
Id_hasil	Int	11	Primary key
No_konsultasi	Int	11	No_konsultasi
Id_gejala	Varchar	10	Id_gejala
Id_penyakit	Varchar	20	Id_penyakit
kepastian	Varchar	20	Kepastian



### III.4.1.5. ERD (*Entity Relationship Diagram*)

Adapun ERD (*Entity Relationship Diagram*) dari aplikasi yang akan dibangun ditunjukkan pada gambar III.18 berikut ini:



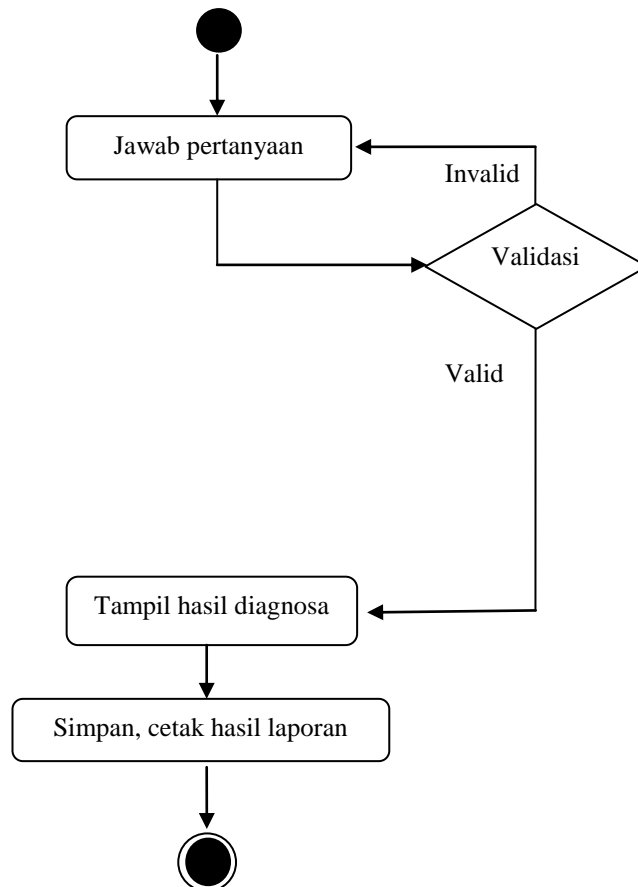
**Gambar III.18 ERD (*Entity Relationship Diagram*)**

### III.4.1.8. Activity Diagram

Menggambarkan aktifitas - aktifitas, objek, *state*, transisi *state* dan *event*. Dengan kata lain kegiatan diagram alur kerja menggambarkan perilaku sistem untuk aktivitas, berikut beberapa gambar *Activity Diagram* :

a. *Activity Diagram* Konsultasi

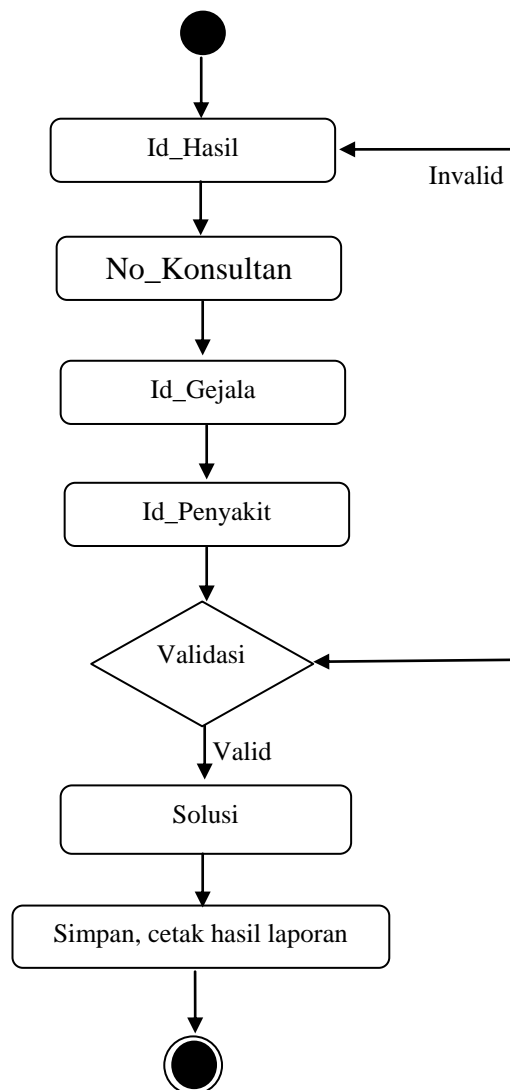
*Activity diagram* konsultasi merupakan *activity diagram* untuk proses konsultasi. *Activity diagram* konsultasi ditunjukkan padagambar III.19. berikut ini:



**Gambar III.19. *Activity Diagram* Konsultasi**

b. *Activity Diagram* Hasil

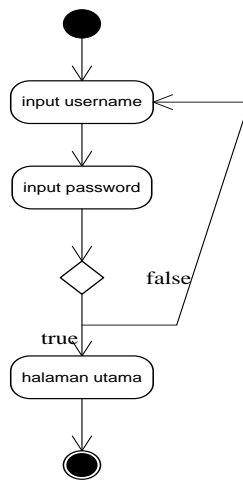
*Activity diagram* hasil merupakan *activity diagram* untuk melihat hasil. *Activity diagram* hasil ditunjukkan padagambar III.20. berikut ini:



**Gambar III.20. Activity Diagram Hasil**

c. *Activity Diagram Login Admin/Pakar*

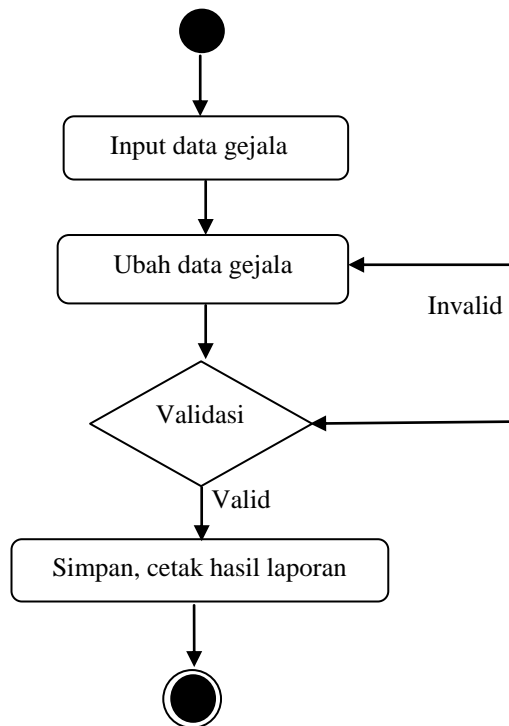
*Activity diagram* login admin/pakar merupakan *activity diagram* untuk proses login admin/pakar. *Activity diagram* login pakar ditunjukkan padagambar III.21. berikut ini:



**Gambar III.21. Activity Diagram Login Admin/Pakar**

d. *Activity Diagram Data Gejala*

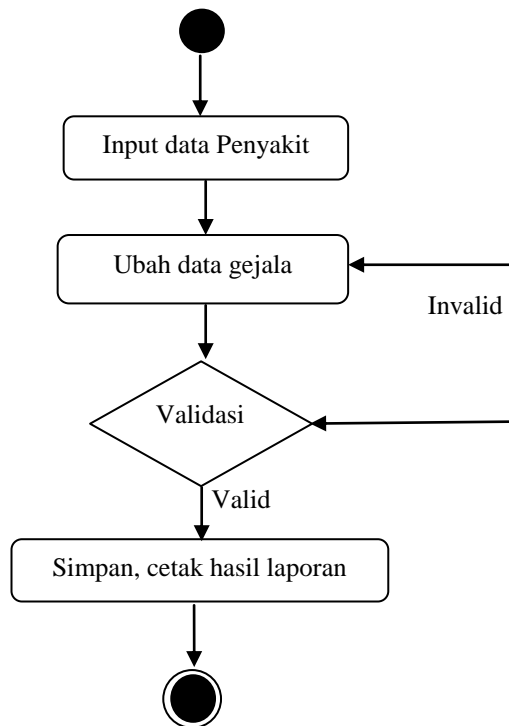
*Activity diagram* data gejala merupakan *activity diagram* untuk proses menginput data gejala. *Activity diagram* data gejala ditunjukkan pada gambar III.22. berikut ini:



**Gambar III.22. Activity Diagram Data Gejala**

e. *Activity Diagram Data Penyakit*

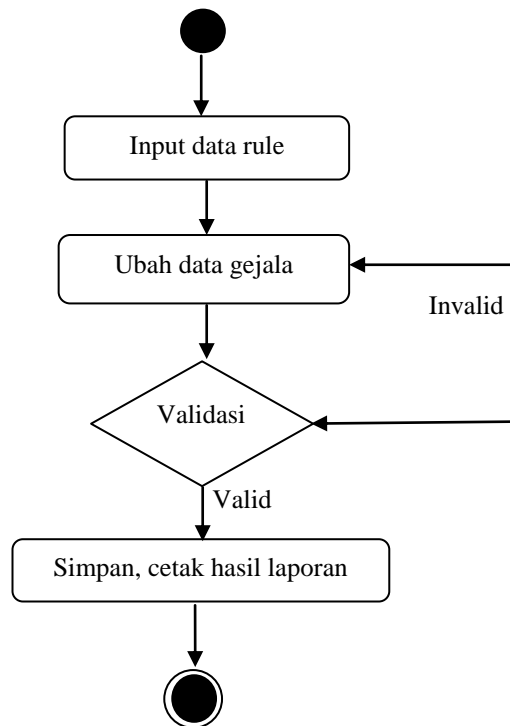
*Activity diagram data penyakit* merupakan *activity diagram* untuk proses input data penyakit. *Activity diagram data penyakit* ditunjukkan pada gambar III.23. berikut ini:



**Gambar III.23. Activity Diagram Data Penyakit**

f. *Activity Diagram Data Rule*

*Activity diagram data rule* merupakan *activity diagram* untuk proses input data rule. *Activity diagram login pakar* ditunjukkan pada gambar III.24. berikut ini:



**Gambar III.24. Activity Diagram Data Rule**





