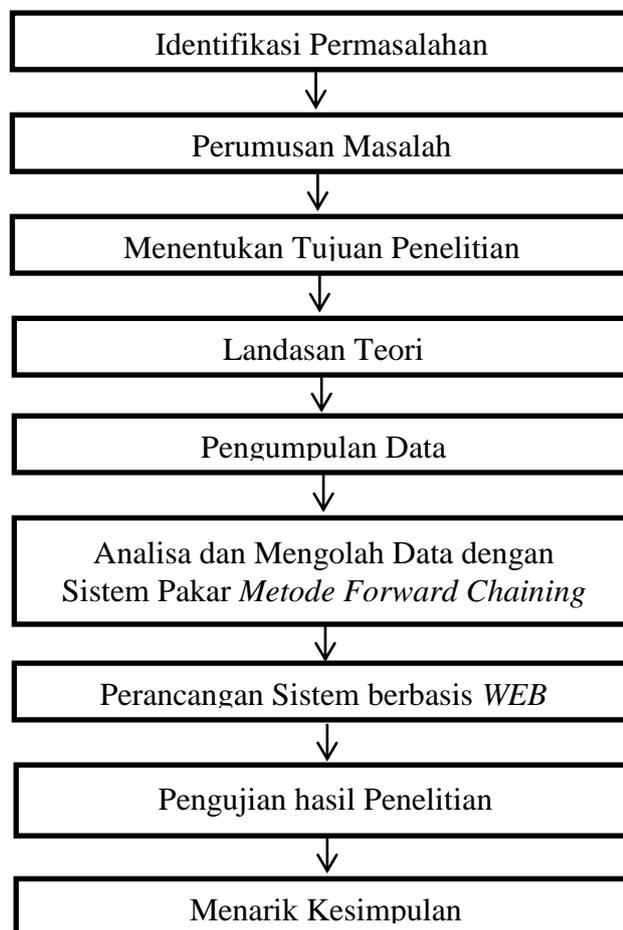


## BAB III

### METODE PENELITIAN

#### 3.1 Desain Penelitian

Penelitian ini dapat selesai dan sesuai dengan yang diinginkan. Oleh karena itu, membuat desain penelitian keadaan yang akan terjadi pada saat dilapangan. dibawah ini merupakan desain penelitian yang dibuat oleh peneliti.



**Gambar 3. 1** Desain Penelitian

(Sumber: Data Penelitian, 2018)

Sesuai dengan gambar 3.1 maka langkah-langkah penelitian yang akan dilakukan adalah sebagai berikut:

1. Mengidentifikasi masalah

Langkah awal yang dilakukan dalam penelitian ini adalah menggunakan identifikasi masalah atau pokok permasalahan dalam memutuskan objek sebagai materi yang akan distudi dalam penelitian ini. Masalah dalam objek yang akan diriset dari permasalahan yang ada dalam latar belakang, selanjutnya membagi jalan keluar atas permasalahan yang ada. Masalah dalam penelitian ini adalah diagnosis penyakit asam urat.

2. Perumusan Masalah

Peneliti merumuskan masalah yang telah didapatkan secara spesifik agar masalah tersebut dapat dijawab dengan benar melalui penelitian. Setelah itu kita melakukan tahapan berikutnya yaitu menentukan tujuan penelitian.

3. Menentukan Tujuan Penelitian

Tujuan penelitian agar mengetahui bagaimana sistem pakar diagnosa penyakit Asam Urat menggunakan metode *forward chaining* berbasis web. Selanjutnya peneliti melakukan landasan teori.

4. Landasan Teori

Untuk membantu jalannya penelitian, peneliti mencari dan mempelajari sumber-sumber pengetahuan berupa buku-buku teori, jurnal-jurnal penelitian, dan sumber-sumber pustaka otentik lainnya yang berkaitan dengan penelitian, diantaranya yaitu buku mengenai kecerdasan buatan, sistem pakar, penyakit asam urat, pemrograman *web*, dan *UML*. Selanjutnya dilakukan pengumpulan data.

## 5. Pengumpulan Data

Setelah data-data yang berkaitan didapatkan baik melalui studi maupun wawancara dengan Dokter sebagai pakar, peneliti menganalisa data-data yang dibutuhkan dalam sistem pakar kemudian data tersebut disederhanakan dan dikelompokkan agar lebih mudah dalam melakukan proses pengolahan data. Kemudian dilakukan analisa dan mengolah data dengan metode *forward chaining*.

## 6. Analisa dan Mengolah Data dengan metode *forward chaining*

Penelitian ini terlebih dahulu melakukan analisa variable dan indicator, kemudian mengolah metode yang dipakai. Sistem pakar ini menggunakan model representasi pengetahuan berbasis kaidah produksi. Sistem pakar dapat menghasilkan suatu kesimpulan berdasarkan atau kaidah yang ada. oleh sebab itu data yang sudah dianalisa kemudian diolah dengan metode *forward chaining* untuk membuat kaidahnya (*rule*) yang digunakan pada saat sistem pakar melakukan penelusuransebelum menyimpulkan hasil. Stelah itu dilakukan perancangan sistem berbasis *web*.

## 7. Perancangan Sistem Berbasis Web

Pada tahap ini, peneliti membuat perancangan mulai dari desain basis pengetahuan, desai *UML*, desain database, dan desain antar muka. Setelah itu dilakukan pengkodean menggunakan bahasa HTML, PHP, JavaScript, dan jQuery kemudian membuat database dengan menggunakan phpMyAdmin yang ada pada program XAMMP. Tahap selanjutnya mengimplementasikan sistem pakar yang dirancang dan pengujian hasil penelitian.

## 8. Pengujian hasil Penelitian

Setelah pembuatan aplikasi sistem pakar yang telah di rancang kemudian dilakukanlah pengujian hasil penelitian. proses ini bertujuan untuk mengurangi kesalahan dan memastikan keluran yang dihasilkan sesuai dengan yang diinginkan. Selanjutnya tahap terakhir yaitu menarik kesimpulan.

## 9. Menarik Kesimpulan

Penarikan kesimpulan dapat diketahui setelah data didapatkan, diidentifikasi, diolah dan menghasilkan *output* berdasarkan rancangan dan pengujian penelitian yang mana akan dibahas pada BAB V.

### 3.2 Pengumpulan Data

Teknik pengumpulan data adalah cara yang digunakan peneliti untuk mendapatkan data yang berkaitan dengan pokok pembahasan dalam penelitian yang sedang dilakukan. Teknik pengumpulan data yang digunakan dalam penelitian ini adalah :

#### 1. Wawancara

Wawancara adalah teknik pengumpulan data yang dilakukan dengan berhadapan secara langsung dengan orang atau pakar yang akan diwawancarai. Untuk mendapatkan data-data yang berkaitan dengan penelitian, peneliti melakukan wawancara Dr.Christiwaty, selaku pakar pada penelitian ini yang bekerja di Rumah Sakit Embung Fatimah . Pedoman wawancara yang digunakan berupa garis-garis besar permasalahan yang akan ditanyakan seperti penyakit, gejala, penyebab dan solusi penyakit asam urat.

## 2. Studi Literatur

Menurut (Noor j, 2011) studi literatur diperoleh melalui mengumpulkan, membaca dan memahami referensi teoritis yang berasal dari buku-buku teori, buku elektronik (*e-book*), jurnal-jurnal penelitian, dan sumber pustaka lainnya. Sifat utama dari studi literatur adalah tak terbatas pada ruang dan waktu sehingga memberi peluang kepada peneliti untuk mengetahui hal-hal yang pernah terjadi di waktu yang lalu. Studi literatur bertujuan untuk menemukan variabel yang akan diteliti, membedakan hal-hal yang sudah dilakukan dan menentukan hal yang perlu dilakukan, melakukan sintesa dan memperoleh perspektif baru, dan menentukan makna dan hubungan antar variabel.

### 3.3 Operasional Variabel

Variabel penelitian adalah suatu atribut atau sifat atau nilai dari orang, objek atau kegiatan yang mempunyai variasi tertentu yang ditetapkan oleh peneliti untuk dipelajari dan kemudian ditarik kesimpulannya. Berikut ini adalah table variabel dan indikator dalam penelitian ini yaitu:

**Tabel 3. 1** Variabel Dan Indikator

Variabel	Kode Indikator	Indikator / tahap Indikator
Penyakit Asam Urat	TP01	Tahap Asimtomik
	TP02	Tahap Akut
	TP03	Tahap Kronik

**Sumber** : Data Penelitian 2018

Dalam **Tabel 3.1** di atas menjelaskan hubungan antara variabel dan indikator serta pengkodean indikator. Variabelnya adalah penyakit asam urat, sedangkan indikatornya adalah 3 tahap dari penyakit asam urat yaitu: Tahap *Asintomatik*, Tahap *Akut*, dan Tahap *Kronik*.

### 3.4 Perancangan Sistem

Didalam pembangunan perangkat lunak merupakan upaya mengonstruksi sebuah sistem yang memberikan kepuasan terhadap kebutuhan fungsional, memenuhi target, memenuhi kebutuhan secara implisit atau eksplisit dari segi performansi, biaya, waktu dan perangkat (Rosa A.S, 2018).

#### 3.4.1 Desain Basis Pengetahuan

Desain basis pengetahuan dilakukan setelah peneliti melakukan proses pengumpulan data dan fakta melalui studi literatur tentang materi yang berkaitan dengan data-data dan wawancara dengan Pakar. Sumber fakta yang didapat berupa data-data yang berhubungan dengan penyakit Asam Urat.

Berikut ini adalah tabel yang menjelaskan mengenai solusi dari setiap tahap penyakit asam urat:

**Tabel 3. 2** Solusi Penyakit

<b>Kode Tahap Penyakit</b>	<b>Solusi</b>
TP01	1. Menerapkan Gaya Hidup sehat 2. Diet rendah purin 3. Pertahankan berat badan ideal dengan makan tidak

	<p>berlebihan dan olahraga teratur</p> <p>4. Jaga tekanan darah, kolesterol, dan gula darah yang normal</p> <p>5. Hindari merokok</p>
TP02	<p>1. Istirahatkan dahulu sendi yang terkena selama paling tidak 24 jam</p> <p>2. Angkat sendi yang sakit</p> <p>3. Gunakan cold pack atau kompres dengan es batu</p> <p>4. Jangan minum aspirin dan obat penurun asam urat saat kondisi nyeri akut</p> <p>5. segera berkonsultasi dengan dokter</p>
TP03	<p>Anda harus segera menemui dokter, karena anda harus melakukan pemeriksaan lebih lanjut, seperti cek darah dan rontgen benjolan pada sendi.</p>

**Sumber:** Data Penelitian (2018)

Berikut ini adalah tabel mengenai gejala-gejala penyakit asam yang diberikan kode untuk memudahkan pengolahan data di sistem pakar.

**Tabel 3. 3** Gejala Penyakit

<b>Kode Tahap penyakit</b>	<b>Kode Gejala</b>	<b>Gejala Penyakit</b>
TP01	G001	Kadar Asam Urat melebihi sedikit dari batas normal (Laki-laki 3,5-8 mg/dl, Wanita 2,8-6,8 mg/dl, Lansia 3,5-8,5 mg/dl)

TP02	G002	Persendiaan terasa panas dan merah
TP02	G003	Demam dan mengigil
TP02	G004	Nyeri mendadak pada sendi
TP03	G005	Rasa nyeri sendi terus menerus
TP03	G006	Sendi sulit digerakkan
TP03	G007	Sendi Bengkak dan ada benjolan
TP03	G008	Sendi terluka dan mengeluarkan cairan kental seperti kapur
TP01	G009	Tidak merasakan gejala apapun, hanya kadar asam urat lebih dari batas normal

**Sumber:** Data Penelitian (2018)

Data aturan merupakan data yang berisi relasi antara data jenis penyakit dan data gejala penyakit yang telah diberi kode sebelumnya. Relasi antar data tersebut disusun berdasarkan sumber pengetahuan dan fakta yang telah didapatkan. Data aturan ini disusun untuk memudahkan peneliti dalam menyusun kaidah yang akan digunakan sebagai basis pengetahuan dalam sistem pakar pada penelitian ini. Susunan data aturan yang digunakan dapat dilihat pada tabel dibawah ini:

**Tabel 3. 4** Data Aturan

<b>Kode Tahap Penyakit</b>	<b>Kode Gejala</b>
TP01	G001,G009

TP02	G001, G002, G003, G004
TP03	G001, G005, G006, G007, G008

**Sumber:** Data Penelitian (2018)

Berdasarkan data aturan yang telah disusun pada **Tabel 3.5**, dapat dijelaskan sebagai berikut:

1. Jika gejalanya yang dialami asam urat adalah Kadar Asam Urat melebihi sedikit dari batas normal (Laki-laki 3,5-8 mg/dl, Wanita 2,8-6,8 mg/dl, Lansia 3,5-8,5 mg/dl) (G001), tidak merasakan gejala apapun, hanya kadar asam urat yang melebihi batas normal (G009), maka jenis penyakitnya adalah tahap Asimtomtik (TP01).
2. Jika gejalanya yang dialami penyakit Asam urat adalah Kadar Asam Urat melebihi sedikit dari batas normal (Laki-laki 3,5-8 mg/dl, Wanita 2,8-6,8 mg/dl, Lansia 3,5-8,5 mg/dl) (G001), nyeri mendadak pada sendi (G002), persendiaan terasa panas dan merah (G003), demam dan menggigil (G004) merupakan tahap penyakitnya adalah Tahap Akut (TP02).
3. Jika gejalanya yang dialami kambing adalah Kadar Asam Urat melebihi sedikit dari batas normal (Laki-laki 3,5-8 mg/dl, Wanita 2,8-6,8 mg/dl, Lansia 3,5-8,5 mg/dl) (G001), rasa nyeri sendi yang terus-menerus (G005), sendi bengkak dan ada benjolan (G006), sendi sulit digerakkan (G007), dan sendi terluka dan mengeluarkan cairan kental seperti kapur (G008) merupakan tahap penyakitnya adalah Tahap Kronik (TP03).

Berdasarkan kaidah yang telah dibuat tersebut maka tabel keputusannya adalah berikut:

**Tabel 3. 5** Tabel Keputusan

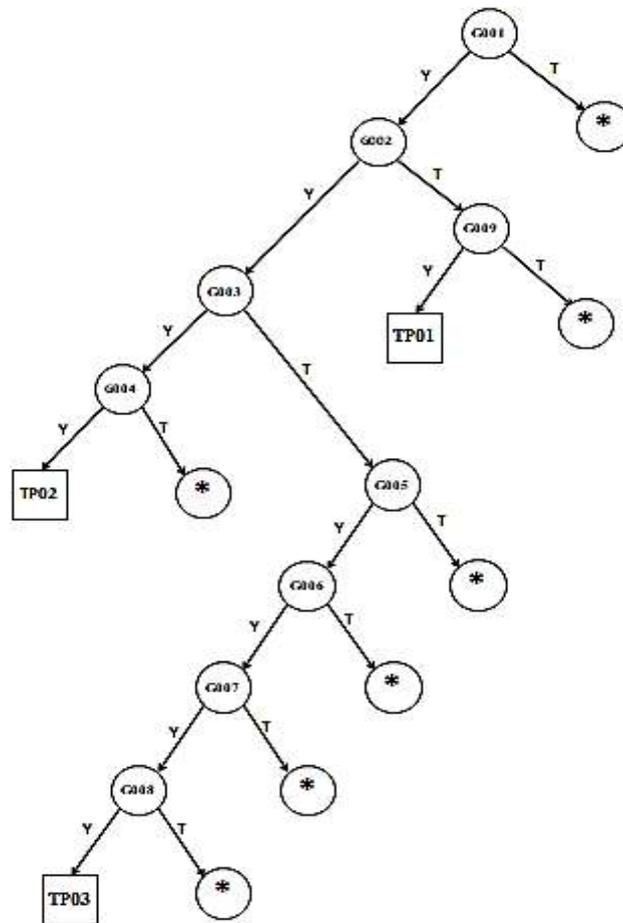
<b>Penyakit</b> <b>Gejala</b>	<b>TP01</b>	<b>TP02</b>	<b>TP03</b>
<b>G001</b>	√	√	√
<b>G002</b>		√	
<b>G003</b>		√	
<b>G004</b>		√	
<b>G005</b>			√
<b>G006</b>			√
<b>G007</b>			√
<b>G008</b>			√
<b>G009</b>	√		

**Sumber:** Data Penelitian (2019)

Pada **Tabel 3.6** diatas, baris gejala diberi tanda centang untuk kolom penyakit yang memenuhi aturan dari masing-masing gejala. Hal ini dibuat untuk memudahkan dalam menyusun aturan kaidah produksi sistem pakar yang akan dibuat.

---

Berdasarkan tabel keputusan (Tabel 3.6) diatas maka dapat dibuat pohon keputusan (Gambar 3.2) sebagai berikut:



**Gambar 3. 2** Pohon Keputusan

Keterangan:

Y: Ya

T: Tidak

\*: Tidak ada data /terdiagnosa

TP01 – TP03: Keterangan lengkap pada **Tabel 3.2**

TP01 - TP03 : Keterangan lengkap pada **Tabel 3.2**

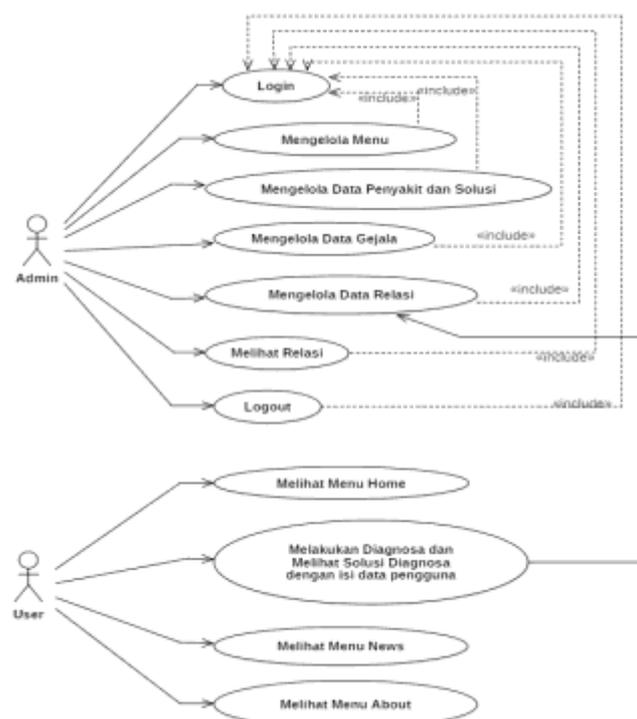
G001 – G008: Keterangan lengkap pada **Tabel 3.4**

### 3.4.2 Desain UML (*Unified Modeling Language*)

UML (*unified Modeling Language*) adalah salah satu tolak ukur bahasa yang digunakan didunia induatri untuk mengartikan requirement, melakukan analisis dan bentuk, serta melukiskan arsitektur dalam pemrograman berorientasi objek (Rosa A.S, 2018). Desain *UML* dibuat untuk memudahkan dalam pembuatan program. Pemodelan *UML* menggunakan alat bantu *software StarUML* versi 3.0.2. Berikut ini adalah diagram UML yang digunakan dalam perancangan program system pakar diagnose penyakit asam urat:

#### 1. *Use Case Diagram*

*Use case diagram* yang akan digunakan pada sistem diagnosa penyakit asam urat seperti pada Gambar 3.3:



**Gambar 3. 3** Use Case Diagram  
(Sumber: Data Penelitian, 2018)

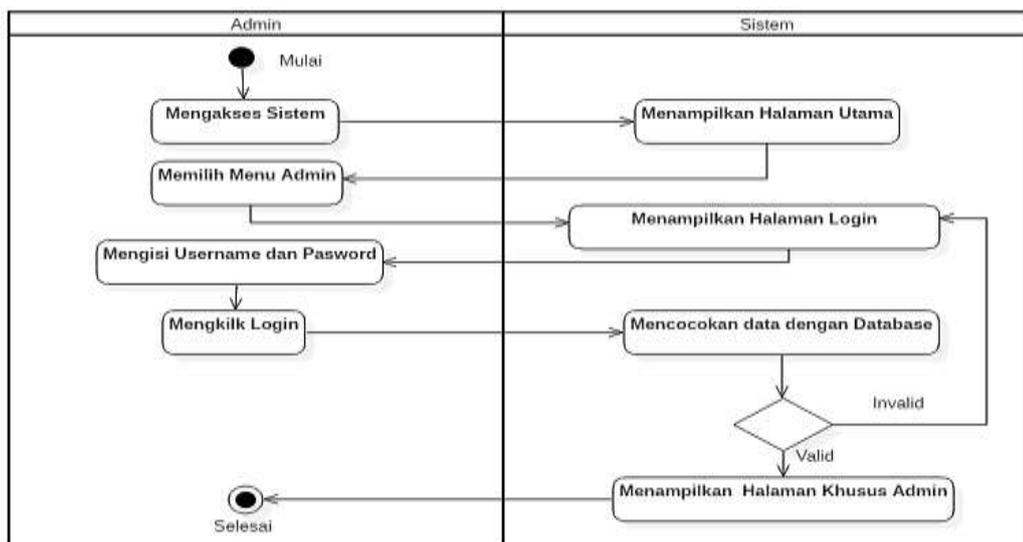
Terdapat 2 aktor yaitu *admin* dan *user*. *Admin* melakukan hubungan dengan sistem berupa mengelola menu, mengelola data penyakit dan solusi, mengelola data gejala, mengolah data relasi, melihat relasi dan *logout*. Semua hubungan dapat dilakukan *admin* setelah melakukan login pada halaman login *admin*. Sedangkan *user* berhubungan dengan sistem yaitu dapat melihat menu home, melihat menu news, melihat menu about dan melakukan diagnosa mengenai penyakit asam urat dengan mengisi data pengguna terlebih dahulu dan dapat melihat hasil dari diagnosa yang dilakukan yaitu berupa solusi.

## 2. Activity Diagram

Berikut ini adalah *activity diagram* yang dirancang dalam penelitian ini:

### a. Activity Diagram Login Admin

*Activity diagram login admin* merupakan *UML* yang menggambarkan kegiatan pengguna pada halaman khusus *admin*.

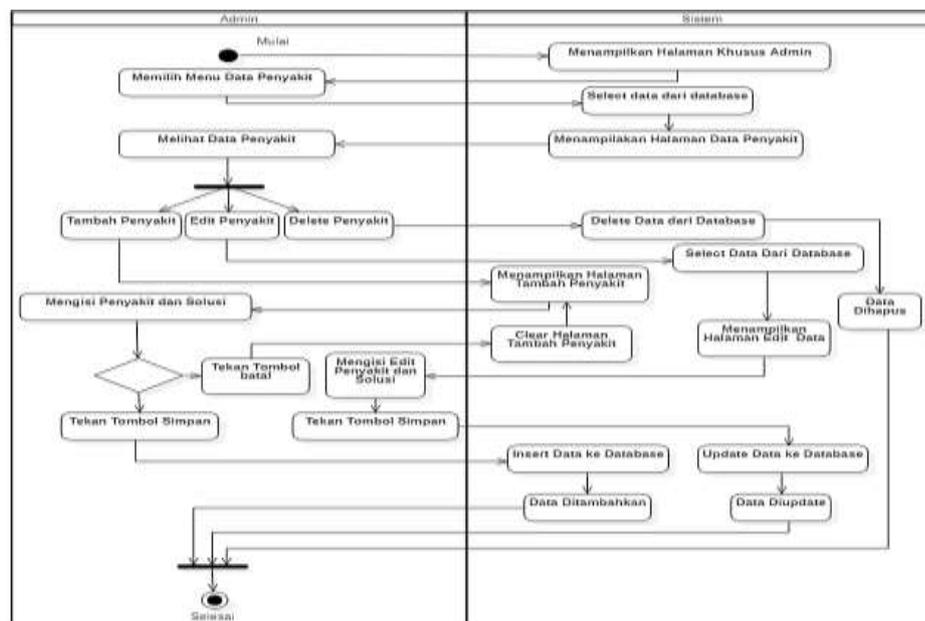


**Gambar 3. 4** Activity Diagram Login  
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.4** di atas, proses *login admin* adalah *admin* mengases system *admin*, kemudian sistem akan menampilkan halaman *login admin*. *Admin* akan mengisi *username* dan *password* pada halaman *login*, kemudian klik tombol *login*. Maka sistem akan mengecek *username* dan *password* kemudian dicocokkan dengan data yang ada di *database*. Jika *username* dan *password* tidak sesuai dengan yang ada di *database* maka sistem akan menampilkan pesan kesalahan dan sistem kembali menampilkan halaman *login*, apabila benar maka sistem akan menampilkan halaman khusus *admin*.

b. *Activity diagram* Mengelola Menu Penyakit

*Activity diagram* mengelola menu penyakit merupakan kegiatan *admin* dalam mengelola data jenis penyakit dan solusi yang digunakan pada penyakit asam urat. Berikut ini gambar *activity diagram* mengelola menu penyakit:

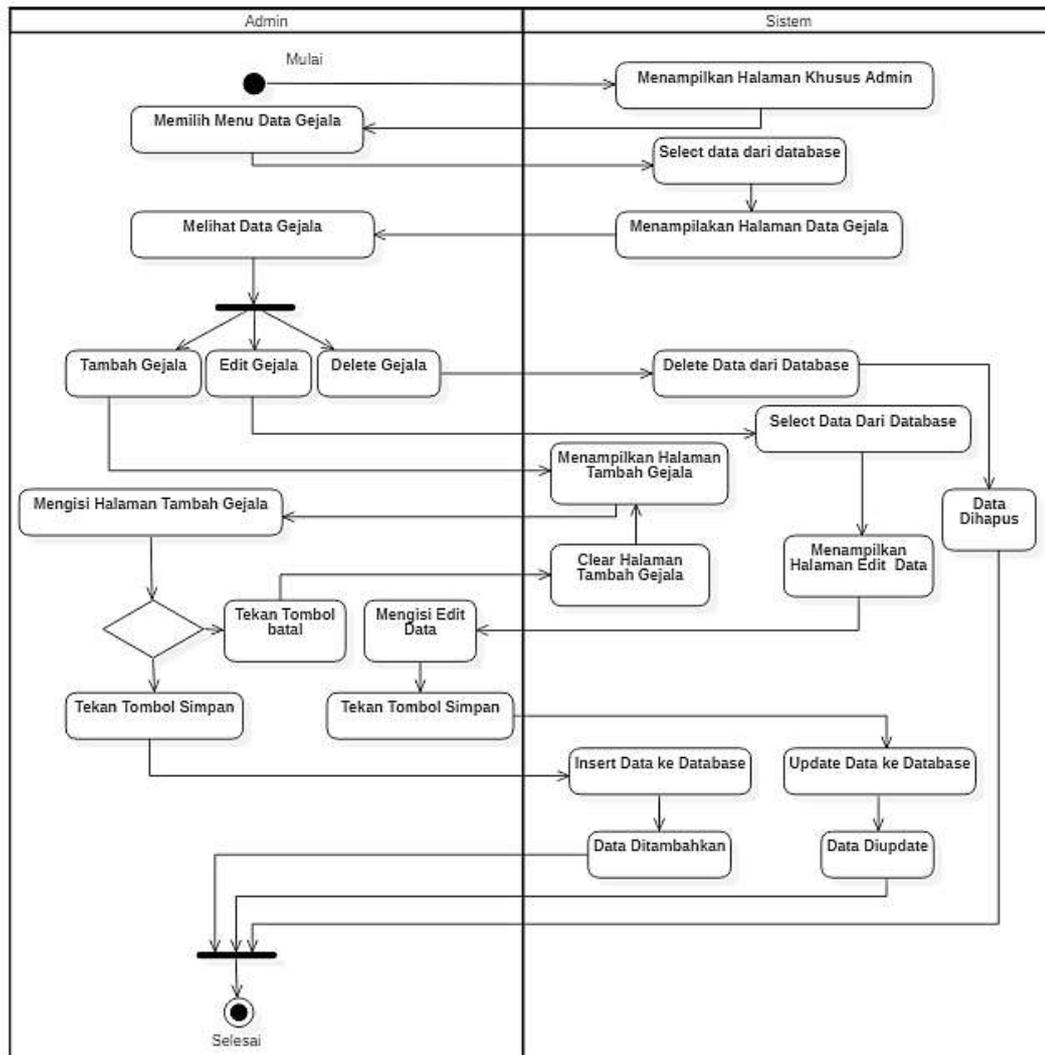


**Gambar 3.5** Activity Diagram Mengelola Menu Penyakit  
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.5** di atas, halaman khusus *admin* yang terbuka setelah *admin* melakukan *login*. Sistem akan menampilkan menu-menu pada halaman khusus *admin*, kemudian *admin* memilih menu penyakit. Sistem memanggil data dari database dan menampilkan halaman data penyakit. *Admin* melihat 3 pilihan yaitu tambah, *edit*, dan *delete*. Jika *admin* mengklik tombol tambah maka sistem akan menampilkan halaman tambah penyakit, kemudian *admin* mengisi data penyakit dan solusi. Kemudian memilih tombol simpan maka data akan dimasukkan ke *database* kemudian data ditambahkan di *database*, jika *admin* menekan tombol batal maka sistem akan membersihkan halaman tambah penyakit. Jika *admin* memilih tombol *edit*, maka sistem akan mengambil data dari *database*. *Admin* mengedit data kemudian klik tombol simpan. Sistem akan melakukan *update database*. Jika *admin* menekan tombol *delete* maka data yang ada di *database* akan terhapus maka proses selesai.

c. *Activity Diagram* Mengelola Gejala

*Activity diagram* mengelola menu gejala merupakan kegiatan *admin* dalam mengelola gejala penyakit asam urat yang digunakan. Berikut ini gambar *activity diagram* mengelola menu gejala:



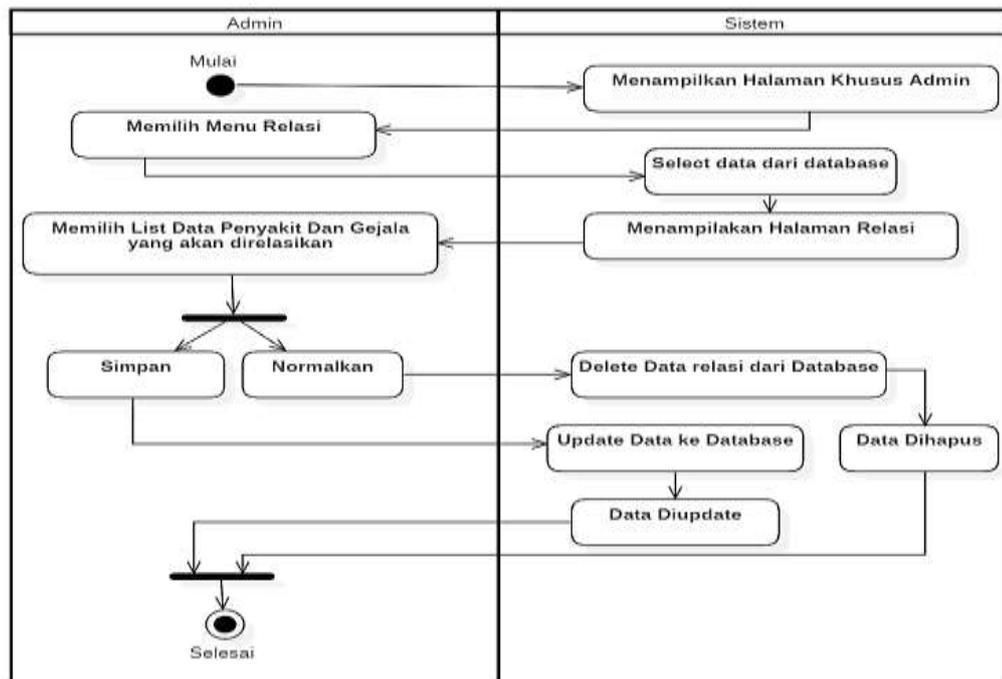
**Gambar 3. 6** Activity *Diagram* Mengelola Gejala  
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.6**, Halaman khusus *admin* yang terbuka setelah *admin* melakukan *login*. Sistem akan menampilkan menu-menu pada halaman khusus *admin*, kemudian *admin* memilih menu gejala. Sistem memanggil data dari database dan menampilkan halaman data gejala. *Admin* melihat 3 pilihan yaitu tambah, *edit*, dan *delete*. Jika *admin* mengklik tombol tambah maka sistem akan menampilkan halaman tambah gejala, kemudian admin mengisi data gejala. Kemudian memilih tombol simpan maka data akan dimasukkan ke

*database* kemudian data ditambahkan di *database*, jika *admin* menekan tombol batal maka sistem akan membersihkan halaman tambah gejala. Jika *admin* memilih tombol *edit*, maka sistem akan mengambil data dari *database*. *Admin* mengedit data kemudian klik tombol simpan. Sistem akan melakukan *update database*. Jika *admin* menekan tombol *delete* maka data yang ada di *database* akan terhapus maka proses selesai.

d. *Activity Diagram* Mengelola Menu Relasi

*Activity diagram* mengelola menu relasi merupakan diagram *UML* yang menggambarkan kegiatan *admin* dalam mengelola data relasi sesuai dengan data aturan penelitian ini. Berikut ini gambar *activity diagram* mengelola data relasi:



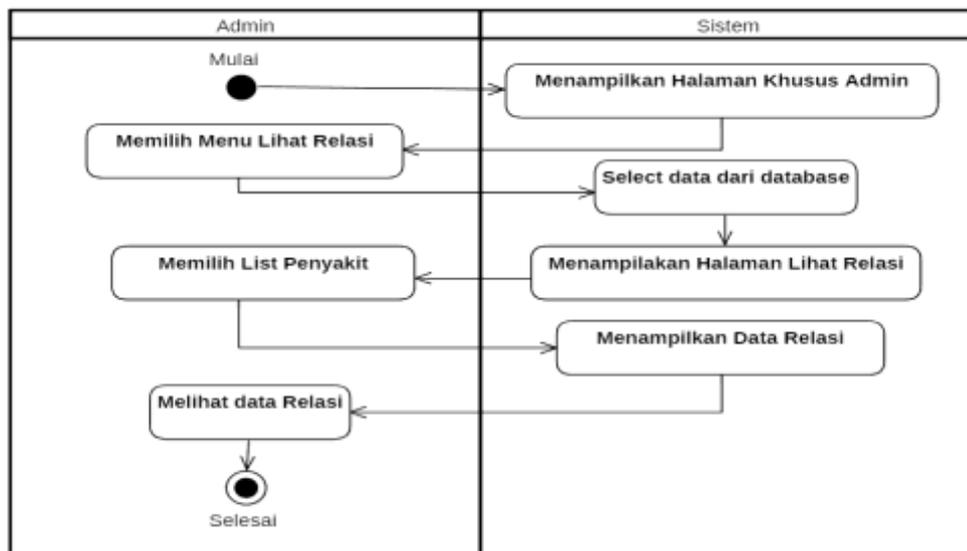
**Gambar 3.7** *Activity Diagram* Mengelola Menu Relasi  
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.7** diatas, *admin* mulai mengakses sistem dan sistem menampilkan halaman khusus admin. *Admin* memilih menu relasi kemudian

sistem menampilkan halaman menu relasi. *Admin* dapat memilih jenis penyakit dan gejala yang akan direlasikan. *Admin* dapat melakukan 2 pilihan yaitu simpan dan normalkan. Pertama, jika *admin* menekan tombol simpan maka sistem akan mengupdate data relasi ke *database*. Kedua, jika *admin* menekan tombol normalkan, maka sistem akan menghapus data relasi antara data penyakit dan gejala, data relasi akan dihapus dari *database* maka proses selesai.

e. *Activity Diagram* Mengelola Menu Lihat Relasi

*Activity diagram* mengelola menu lihat relasi merupakan diagram *UML* yang menggambarkan kegiatan *admin* dalam melihat data relasi atau aturan dari *rule forward chaining*. Berikut ini gambar *activity diagram* mengelola menu lihat relasi (**Gambar 3.8**):



**Gambar 3.8** *Activity Diagram* Mengelola Menu Lihat Relasi

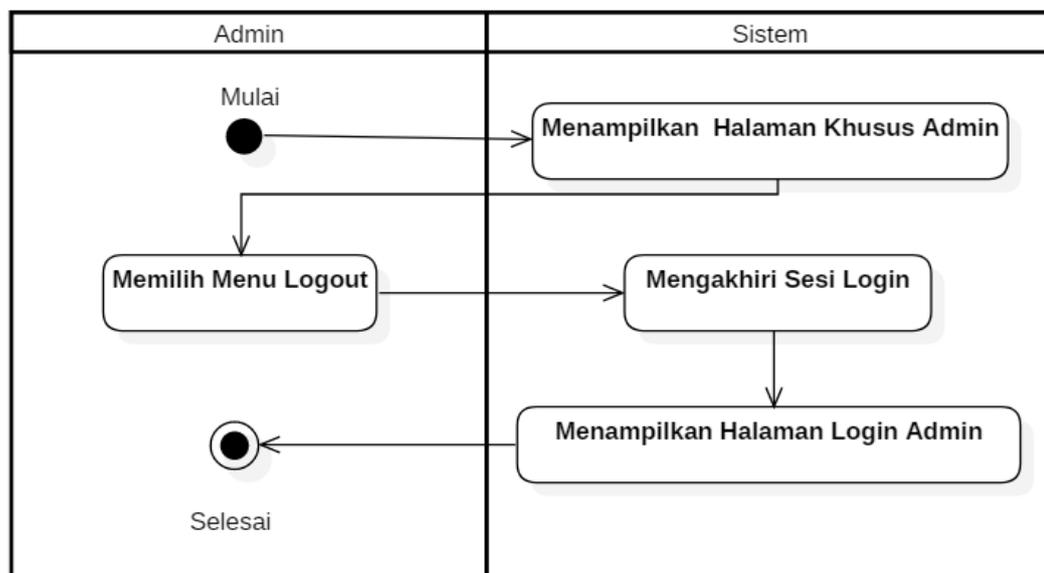
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.8** diatas, *admin* mulai dengan mengakses sistem dan sistem menampilkan halaman khusus admin. *Admin* memilih menu lihat relasi kemudian

sistem akan mengambil data dari *database* dan menampilkan halaman lihat relasi. *Admin* dapat melihat list penyakit yang akan dilihat relasinya, maka sistem akan menampilkan halaman data relasi yang telah dibuat. *Admin* akan melihat data relasi apakah telah sesuai dengan aturan penelitian ini, maka proses selesai.

f. *Activity Diagram Menu Logout*

*Activity diagram* menu *logout* merupakan diagram *UML* yang menggambarkan kegiatan *admin* dalam menggunakan menu *logout*. Berikut ini gambar *activity diagram* menu *logout* (**Gambar 3.9**):

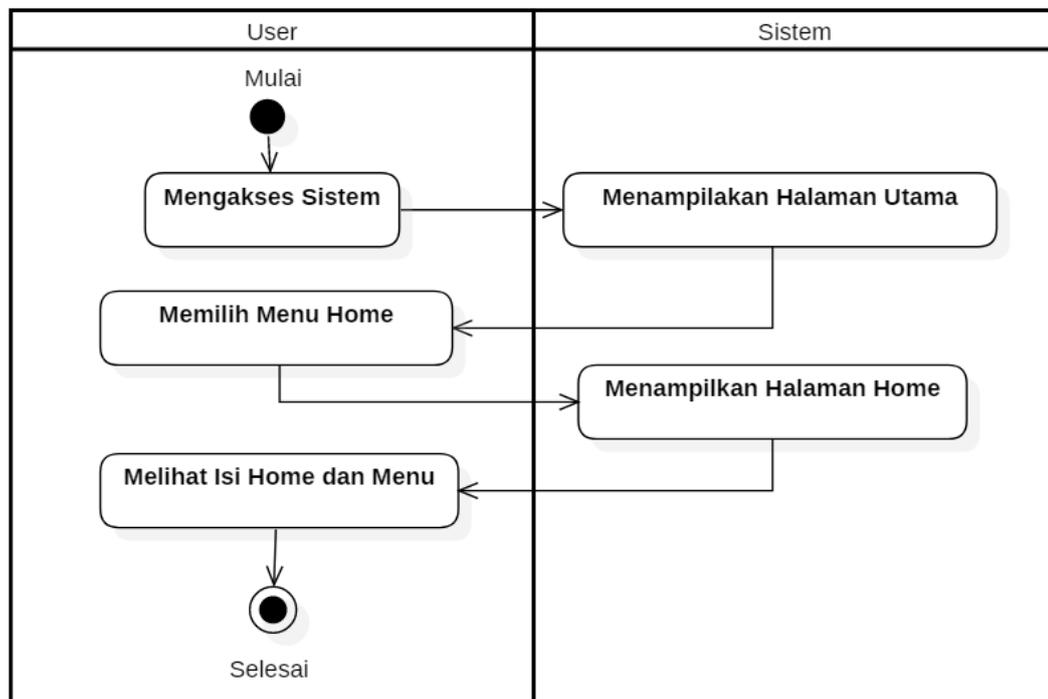


**Gambar 3.9** *Activity Diagram Logout*  
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.9** di atas, *admin* mulai dengan mengakses sistem dan sistem menampilkan halaman khusus *admin*. *Admin* memilih menu *logout* kemudian sistem akan mengakhiri sesi *login admin* dan sistem akan menampilkan halaman *login*.

g. *Activity Diagram Melihat Menu Home*

*Activity diagram* melihat menu home merupakan diagram *UML* yang menggambarkan kegiatan *user* melihat menu *home* dan menu lainnya saat pertama kali mengakses sistem. Berikut ini gambar *activity diagram* melihat menu *home*:

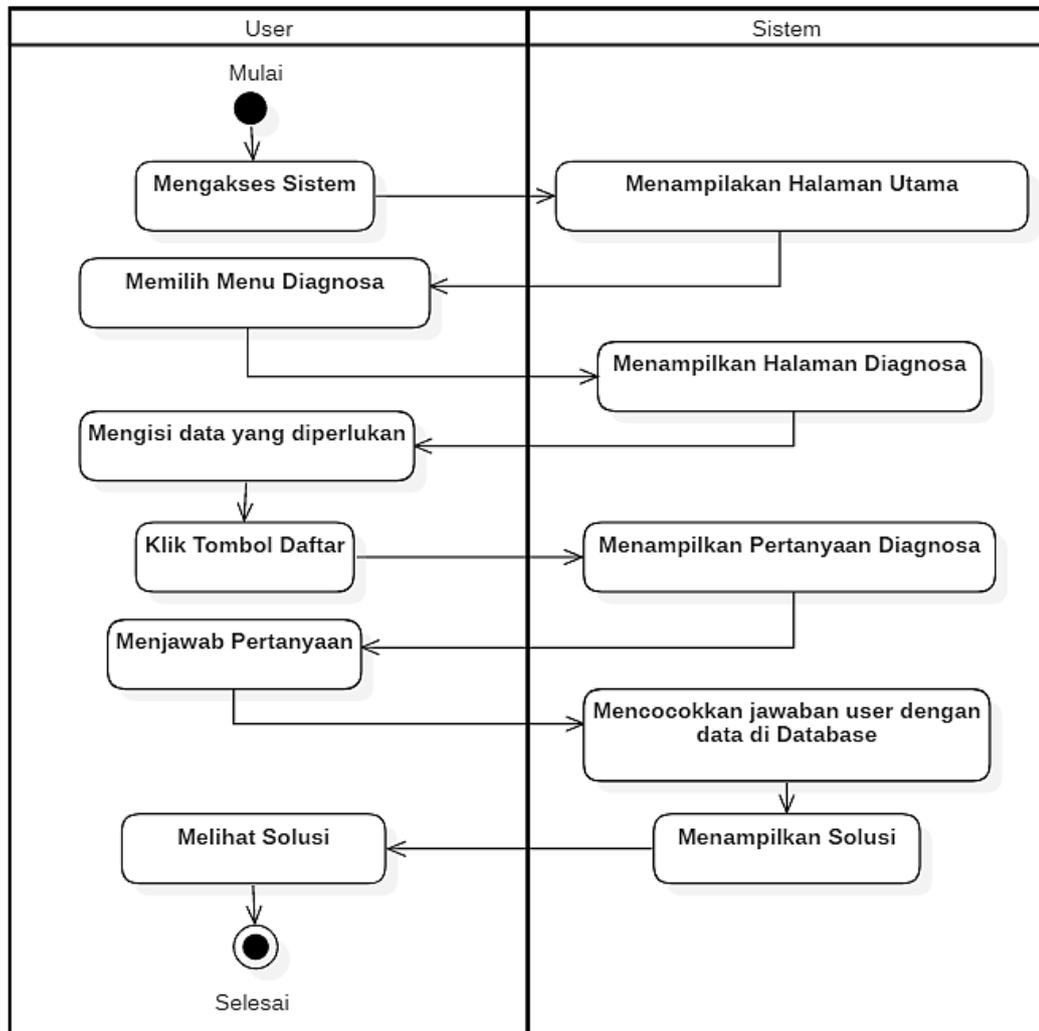


**Gambar 3. 10** *Activity Diagram Melihat Menu Home*  
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.10** di atas, *user* mulai dengan mengakses sistem dan sistem menampilkan halaman utama. *User* memilih menu home kemudian sistem menampilkan halaman home. *User* dapat melihat menu lainnya pada sistem maka proses melihat menu *home* selesai.

h. *Activity diagram Menu Diagnosa*

*Activity diagram* menu diagnosa merupakan diagram *UML* yang menggambarkan kegiatan *user* melakukan diagnosa penyakit asam urat. Berikut ini gambar *activity diagram* menu diagnos:



**Gambar 3. 11** Activity Diagram Menu Diagnosa

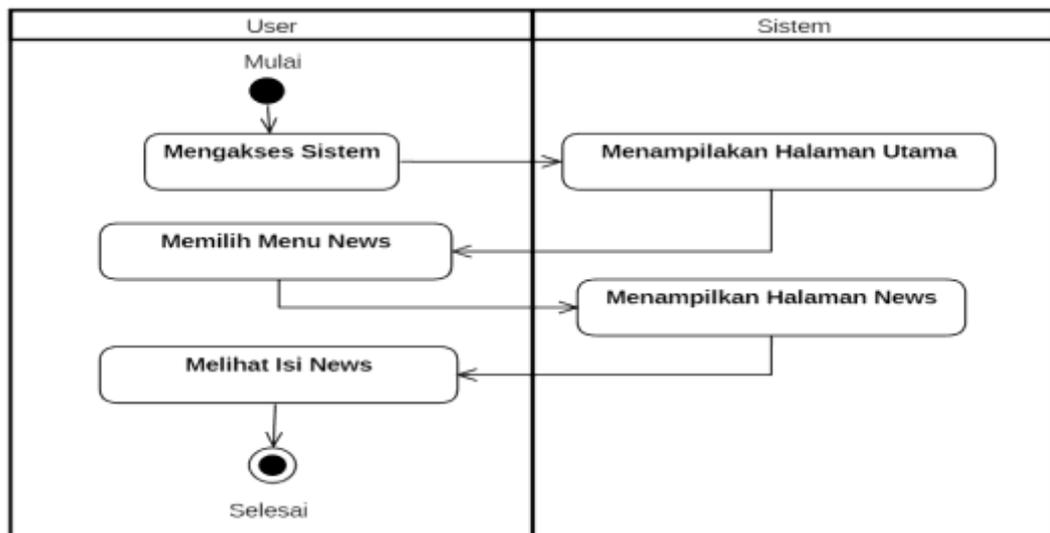
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.11** di atas, *user* mulai dengan mengakses sistem dan sistem menampilkan halaman utama. *User* memilih menu diagnosa kemudian sistem menampilkan halaman menu diagnosa. *User* akan mengisi data nama dan alamat terlebih dahulu dan menekan tombol daftar, Sistem akan menampilkan halaman pertanyaan diagnosa. *User* akan menjawab pertanyaan sesuai dengan apa yang

dialami *user*. Kemudian sistem akan mencocokkan jawaban dengan data yang ada didatabase lalu sistem akan menampilkan hasil atau solusinya. *User* dapat melihat solusi dari gejala yang dialaminya, maka proses melakukan diagnosa selesai.

i. *Activity Diagram* Melihat Menu *News*

*Activity diagram* melihat menu *news* merupakan diagram *UML* yang menggambarkan kegiatan *user* melihat menu *news* yang berisi mengenai informasi mengenai penyakit asam urat. Berikut ini gambar *activity diagram* melihat menu *news*:

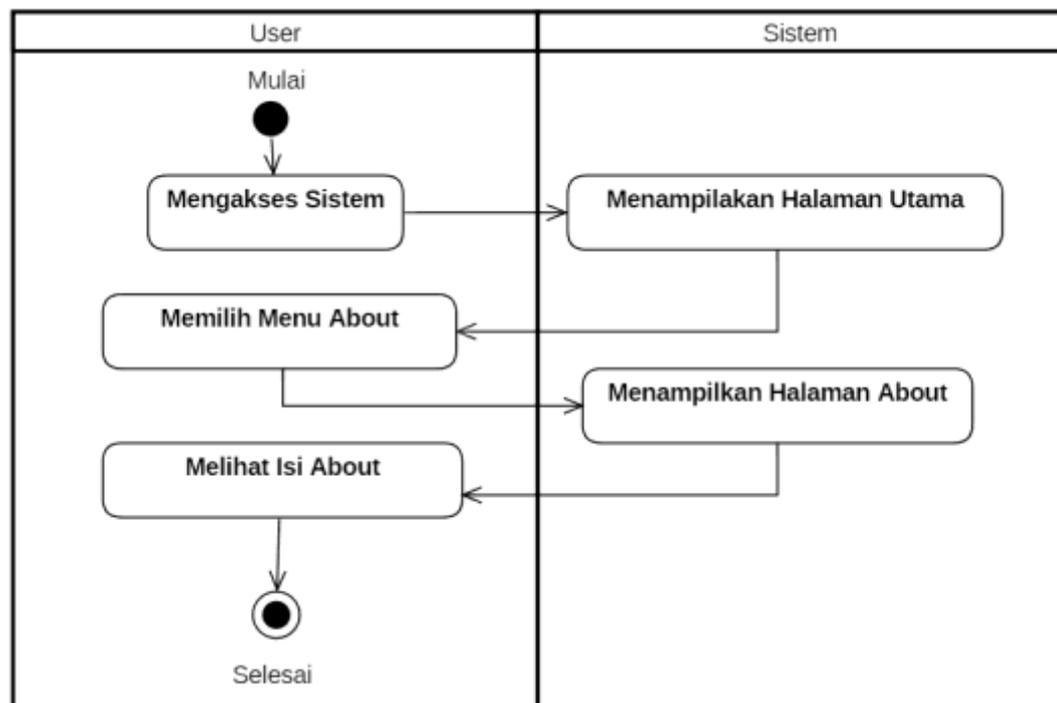


**Gambar 3. 12** *Activity Diagram* Melihat Menu *News*  
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.12** di atas, *user* mulai dengan mengakses sistem dan sistem menampilkan halaman utama. *User* memilih menu *news* kemudian sistem menampilkan halaman *news*. *User* dapat melihat informasi mengenai penyakit asam urat pada sistem maka proses melihat menu *news* selesai.

j. *Activity Diagram Melihat Menu About*

*Activity diagram* melihat menu *about* merupakan diagram *UML* yang menggambarkan kegiatan *user* melihat menu *about* yang berisi informasi pemilik sistem pakar diagnosa penyakit asam urat. Berikut ini gambar *activity diagram* melihat menu *about* :



**Gambar 3.13** *Activity Diagram* Melihat Menu About  
(Sumber: Data Penelitian, 2018)

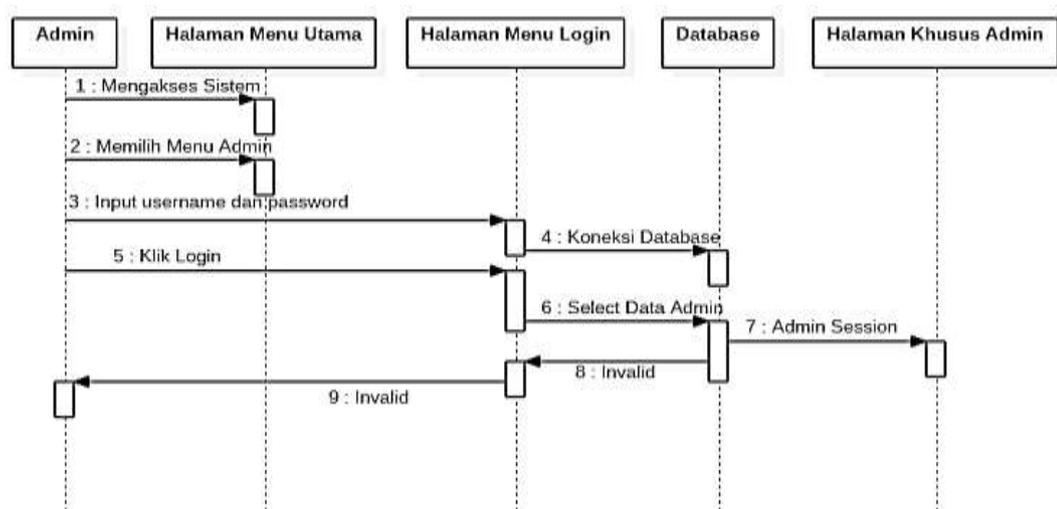
Pada **Gambar 3.13** diatas, *user* mulai dengan mengakses sistem dan sistem menampilkan halaman utama. *User* memilih menu *about* kemudian sistem menampilkan halaman *about*. *User* dapat melihat informasi mengenai pemilik sistem pakar diagnosa penyakit asam urat maka proses melihat menu *about* selesai.

### 3. Sequence Diagram

Sequence diagram melukiskan kelakuan objek pada *use case* dengan menggambarkan waktu hidup objek dan message yang dikirimkan dan diterima antar objek (Rosa A.S, 2018). Berikut ini adalah *sequence diagram* yang buat dalam penelitian ini:

#### a. Sequence Diagram Login Admin

*Sequence diagram login admin* merupakan urutan waktu kegiatan *admin* saat melakukan *login*. Berikut ini gambar *sequence diagram login admin* :



**Gambar 3. 14** *Sequence Diagram Login Admin*

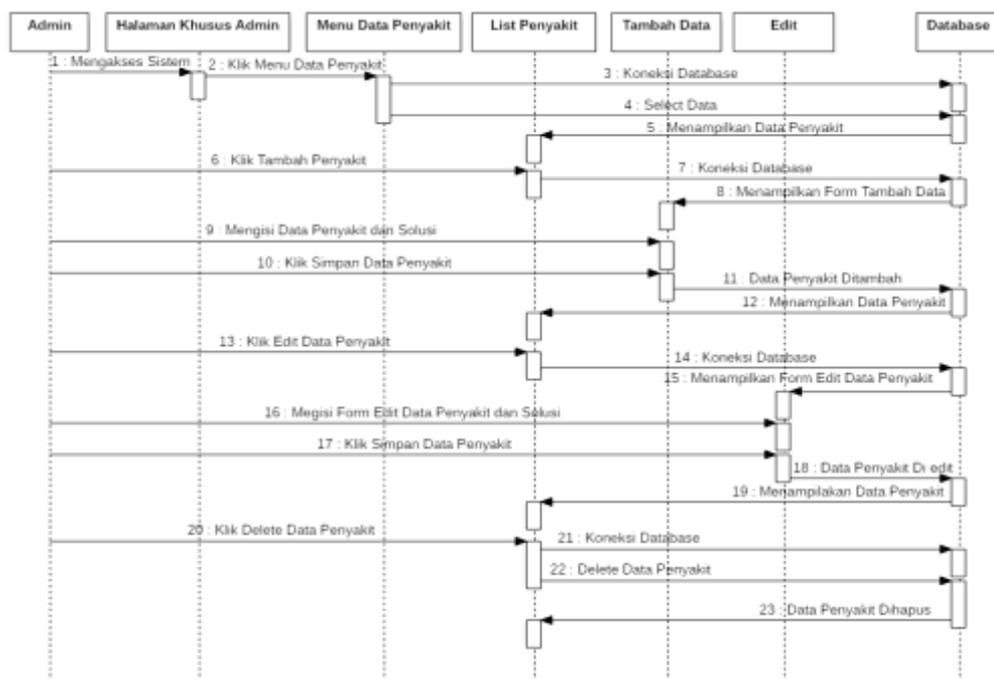
(Sumber: Data Penelitian, 2018)

*Admin* mengakses sistem lalu sistem menampilkan halaman menu *login*. *Admin* akan memasukkan *username* dan *password* ke menu *login* kemudian *admin* melakukan klik tombol *login*. Dari Halaman *Login Admin*, sistem akan mengecek *username* dan *password* yang sudah dimasukkan kemudian sistem akan terhubung dengan *database*, setelah itu *username* dan *password* akan dicocokkan oleh sistem. Jika *username* dan *password* *valid* maka halaman khusus *admin* akan

ditampilkan. Jika *username* dan *password invalid* (tidak sesuai dengan *database*) maka sistem akan menampilkan pesan gagal di halaman menu login.

#### b. Sequence Diagram Mengelola Menu Penyakit

*Sequence diagram* mengelola menu penyakit merupakan urutan waktu kegiatan *admin* saat mengelola data penyakit dan solusi. Berikut ini gambar *sequence diagram* mengelola menu penyakit:



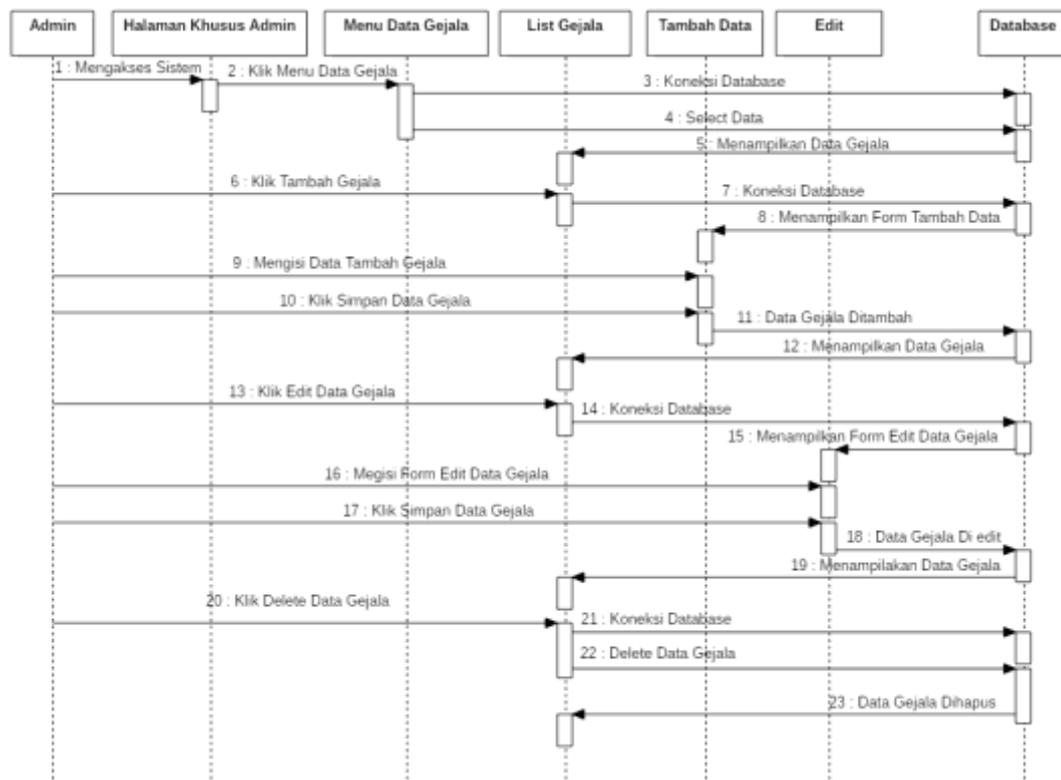
**Gambar 3. 15** Sequence Diagram Mengelola Menu Penyakit  
(Sumber: Data Penelitian, 2018)

*Admin* mengakses sistem, kemudian menampilkan halaman khusus admin. Lalu *admin* mengklik menu penyakit, *admin* menekan tombol tambah penyakit. Sistem akan menampilkan form tambah data. Kemudian *admin* akan mengisi form tambah penyakit dan solusi lalu *admin* mengklik tombol simpan. Data ditambahkan ke *database* dan sistem akan menampilkan kembali menu penyakit. Kemudian admin klik *edit* data penyakit dan sistem akan menampilkan form edit

data, kemudian *admin* akan mengisi form edit data penyakit dan solusi lalu mengklik tombol simpan. Data penyakit *diupdate* pada *database*, kemudian sistem akan menampilkan form menu data penyakit kembali. *Admin* mengklik tombol *delete*, sistem pun akan mendelete data penyakit pada *database*.

### c. Sequence Diagram Mengelola Gejala

*Sequence diagram* mengelola gejala merupakan urutan waktu kegiatan *admin* saat melakukan pengelolaan gejala penyakit. Berikut ini gambar *sequence diagram* mengelola data gejala (**Gambar 3.16**):



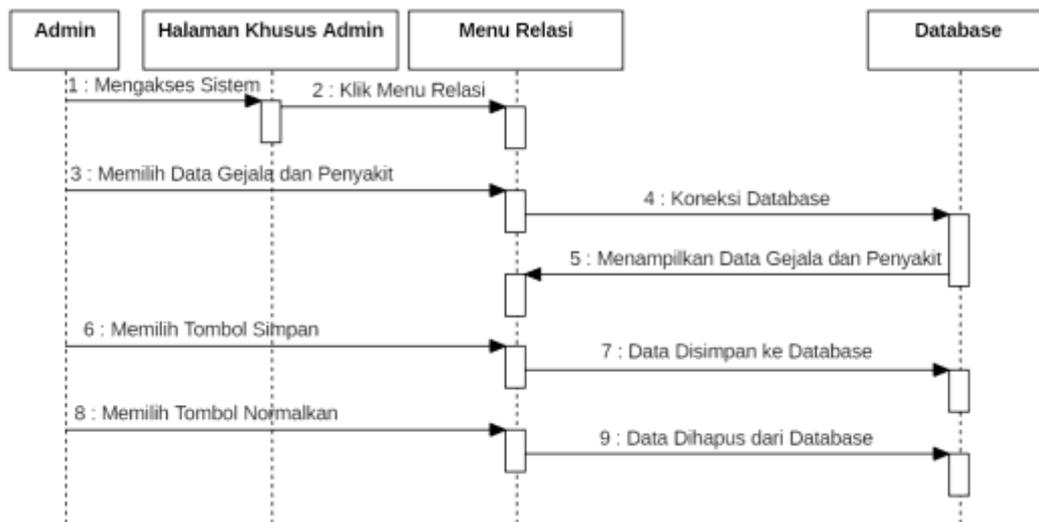
**Gambar 3. 16** Sequence Diagram Mengelola Gejala  
(Sumber: Data Penelitian, 2018)

*Admin* mengakses sistem, kemudian menampilkan halaman khusus admin. Lalu *admin* mengklik menu gejala, *admin* menekan tombol tambah gejala. Sistem akan menampilkan form tambah data. Kemudian admin akan mengisi form

tambah gejala lalu admin mengklik tombol simpan. Data ditambahkan ke *database* dan sistem akan menampilkan kembali menu gejala. Kemudian admin klik *edit* data gejala dan sistem akan menampilkan form *edit* data, kemudian admin akan mengisi *form edit* data gejala lalu mengklik tombol simpan. Data user diupdate pada *database*, kemudian sistem akan menampilkan *form* menu data gejala kembali. Admin mengklik tombol *delete*, sistem pun akan mendelete data gejala pada *database*.

#### d. *Sequence Diagram* Mengelola Menu Relasi

*Sequence diagram* mengelola menu relasi merupakan urutan waktu kegiatan *admin* saat mengelola data relasi sesuai aturan kaidah. Berikut ini gambar *sequence diagram* mengelola menu relasi (**Gambar 3.17**):



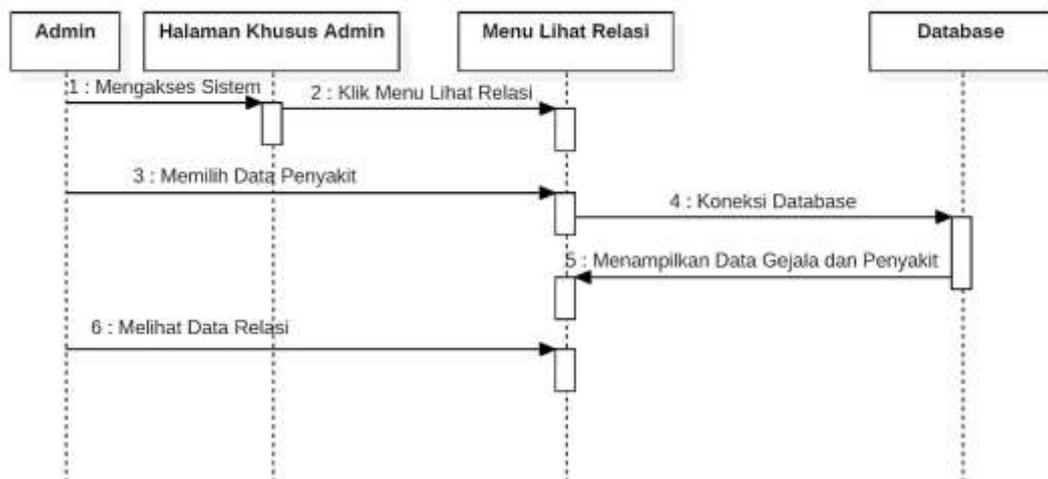
**Gambar 3. 17** Sequence Diagram Mengelola Menu Relasi  
(Sumber: Data Penelitian, 2018)

*Admin* mengakses sistem dan menampilkan halaman khusus *admin*. kemudian *admin* mengklik menu relasi, Sistem akan menampilkan halaman menu relasi dan *admin* akan memilih data penyakit yang akan direlasikan ke data gejala

kemudian sistem terkoneksi dengan *database* dan mengambil data, kemudian menampilkan data gejala dan penyakit. Jika *admin* menekan tombol simpan maka sistem akan menambahkan data ke *database*. Bila *admin* memilih tombol normalkan maka data gejala dan penyakit yang direlasikan akan dihapus dari *database*.

e. *Sequence Diagram* Mengelola Menu Lihat Relasi

*Sequence diagram* mengelola menu lihat relasi merupakan urutan waktu kegiatan *admin* saat melihat relasi yang telah sesuai dengan aturan rule pada sistem. Berikut ini gambar *sequence diagram* mengelola menu lihat relasi:

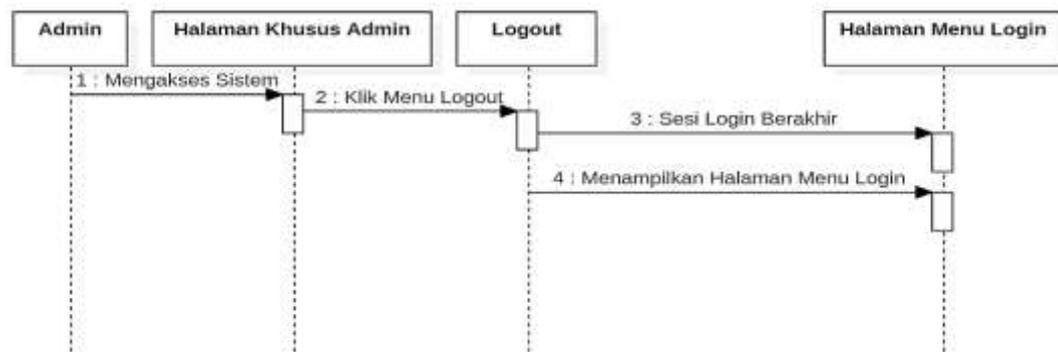


**Gambar 3. 18** *Sequence diagram* Mengelola Menu Lihat Relasi  
(Sumber: Data Penelitian, 2018)

*Admin* mengakses sistem dan menampilkan halaman khusus *admin*. kemudian *admin* mengklik menu lihat relasi, kemudian sistem menampilkan data relasi yang telah dibuat pada menu relasi. *Admin* memilih data penyakit yang akan dilihat relasinya. Sistem akan terkoneksi ke *database* dan sistem akan menampilkan data gejala dan penyakit yang telah direlasikan.

f. *Sequence Diagram Logout Admin*

*Sequence diagram logout admin* merupakan urutan waktu kegiatan admin saat keluar dari sistem. Berikut ini gambar *sequence diagram* logout admin:

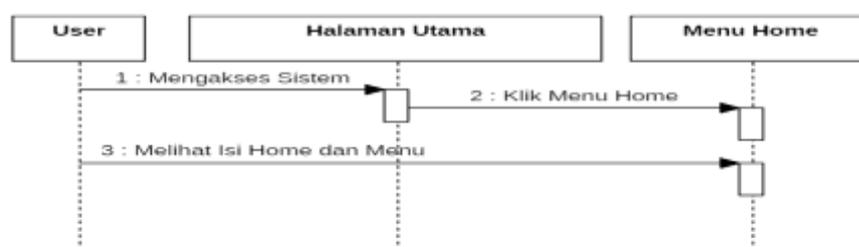


**Gambar 3. 19** *Sequence Diagram Logout Admin*  
(Sumber: Data Penelitian, 2019)

*Admin* mengakses halaman khusus *admin* lalu mengklik menu logout dan sesi login *admin* pada sistem pun berakhir kemudian akan menampilkan halaman *login*.

g. *Sequence Diagram Melihat Menu Home*

*Sequence diagram* melihat menu *home* merupakan urutan waktu kegiatan pengguna (*user*) saat mengakses sistem pertama kali dan melihat menu lainnya. Berikut ini gambar *sequence diagram* melihat menu *home*:

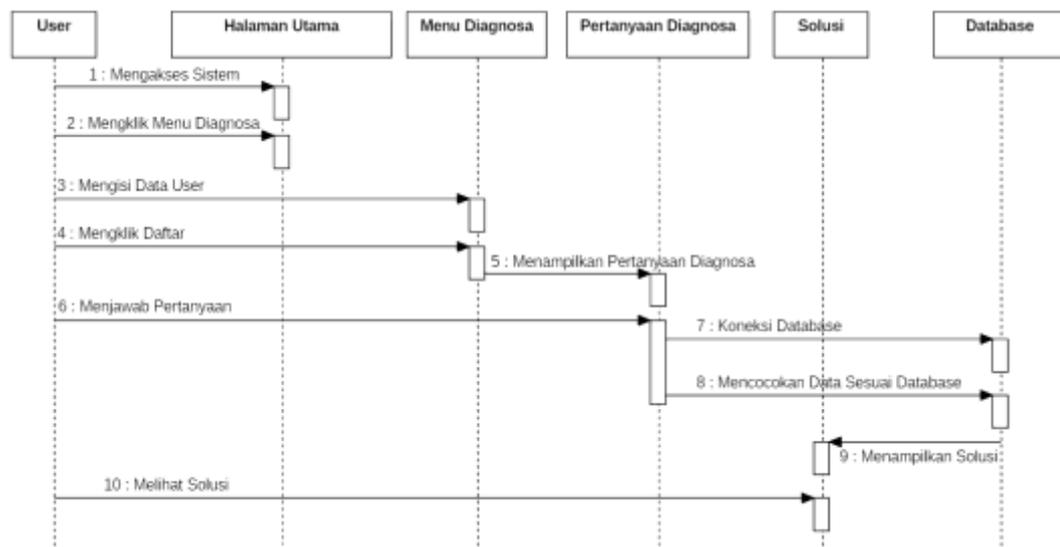


**Gambar 3. 20** *Sequence diagram Melihat Menu Home*  
(Sumber: Data Penelitian, 2019)

*User* mengakses sistem, sistem akan menampilkan halaman utama sistem. *User* memilih menu home dan dapat melihat menu lainnya pada sistem pakar penyakit asam urat.

#### h. *Sequence Diagram* Menu Diagnosa

*Sequence diagram* menu diagnosa merupakan urutan waktu kegiatan pengguna (*user*) saat melakukan diagnosa penyakit asam urat. Berikut ini gambar *sequence diagram* menu diagnosa:



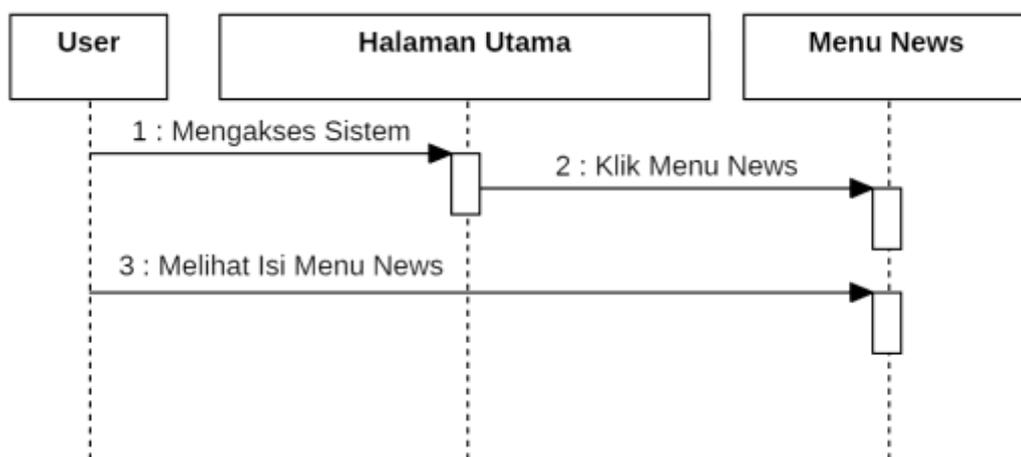
**Gambar 3. 21** *Sequence Diagram* Menu Diagnosa  
(Sumber: Data Penelitian, 2018)

*User* mengakses sistem dan mengklik menu diagnosa pada halaman utama sistem. Sistem akan menampilkan halaman menu diagnosa. *User* akan mengisi data yang diperlukan untuk buku tamu yang tersimpan pada database panel dengan menekan tombol daftar. Sistem akan menampilkan pertanyaan-pertanyaan diagnosa mengenai penyakit asam urat. Sistem akan melakukan koneksi ke database untuk mencocokkan jawaban dari setiap pertanyaan yang dijawab oleh

*user*, kemudian sistem akan menampilkan hasil diagnosa yaitu berupa solusi yang bisa dilihat oleh *user*.

i. *Sequence Diagram* Melihat Menu News

*Sequence diagram* melihat menu *news* merupakan urutan waktu kegiatan pengguna (*user*) melihat menu *news*. Berikut ini gambar *sequence diagram* melihat menu *news* (**Gambar 3.22**):

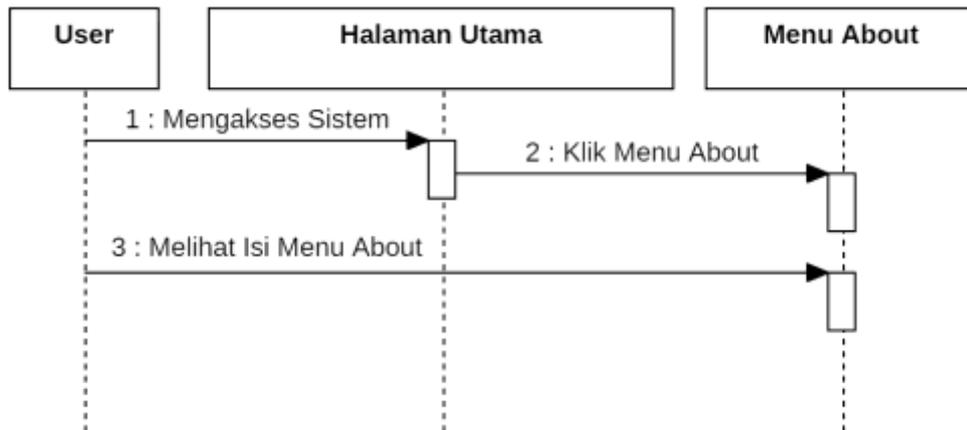


**Gambar 3. 22** *Sequence diagram* Melihat Menu News  
(Sumber: Data Penelitian, 2018)

*User* mengakses sistem, sistem akan menampilkan halaman utama sistem. *User* memilih menu *news* dan melihat informasi mengenai penyakit asam urat yang digunakan pada sistem pakar diagnosa penyakit asam urat.

j. *Sequence diagram* Melihat Menu About

*Sequence diagram* melihat menu *about* merupakan urutan waktu kegiatan pengguna (*user*) melihat menu *about*.. Berikut ini gambar *sequence diagram* melihat menu *about*:



**Gambar 3. 23** Sequence diagram Melihat Menu About  
(Sumber: Data Penelitian, 2018)

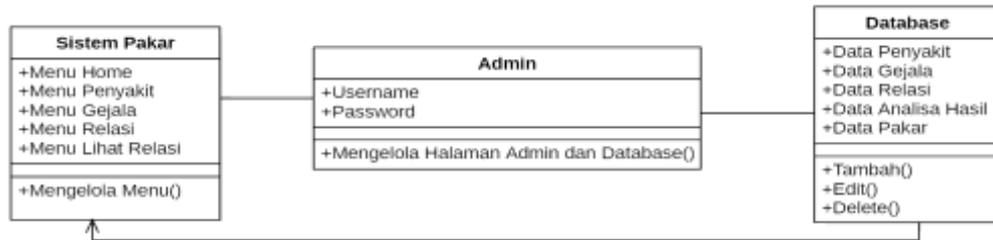
*User* mengakses sistem, sistem akan menampilkan halaman utama sistem. *User* memilih menu *about* dan melihat informasi mengenai pemilik sistem pakar diagnosa penyakit asam urat.

#### 4. *Class Diagram*

Dalam penelitian ini hanya dibuat 2 macam *class diagram* yaitu *class diagram user* dan *class diagram admin*. Berikut ini adalah gambar-gambar *class diagram* yang digunakan dalam sistem pakar pada penelitian ini:

##### a. *Class Diagram Admin*

*Class diagram admin* merupakan urutan kegiatan *admin* saat mengakses sistem pakar diagnosa penyakit asam urat. Berikut ini gambar *class diagram admin*:

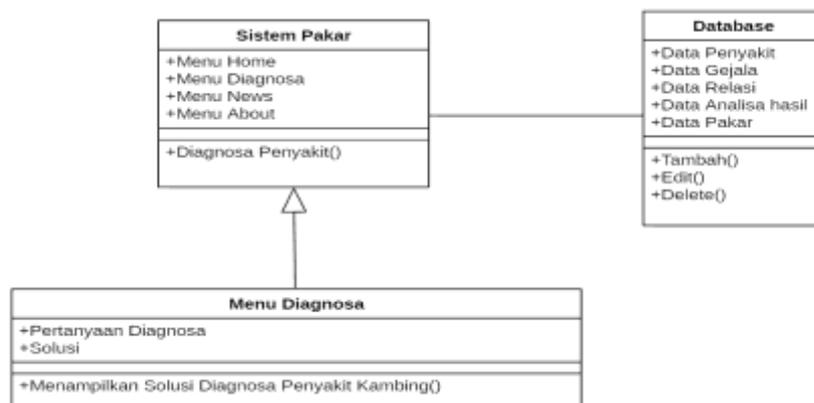


**Gambar 3. 24** Class Diagram Admin  
(Sumber: Data Penelitian, 2018)

Struktur *class diagram admin* pada **Gambar 3.24** Diatas menggambarkan bahwa pada sistem pakar memiliki atribut berupa menu *home*, menu penyakit, menu gejala, menu relasi dan menu lihat yang memiliki fungsi yaitu mengelola menu pada sistem pakar doagnosa penyakit asam urat. Sistem pakar terhubung dengan *database* yang memiliki atribut data penyakit, data gejala, data relasi, data analisa hasil dan data pakar serta memiliki fungsi tambah, *edit* dan *delete*, dan melakukan *login* terlebih dahulu pada sistem.

b. *Class Diagram User*

*Class diagram user* merupakan urutan kegiatan *user* saat mengakses sistem pakar diagnosa penyakit asam urat. Berikut ini gambar *class diagram user*:



**Gambar 3. 25** Class Diagram User  
(Sumber: Data Penelitian, 2018)

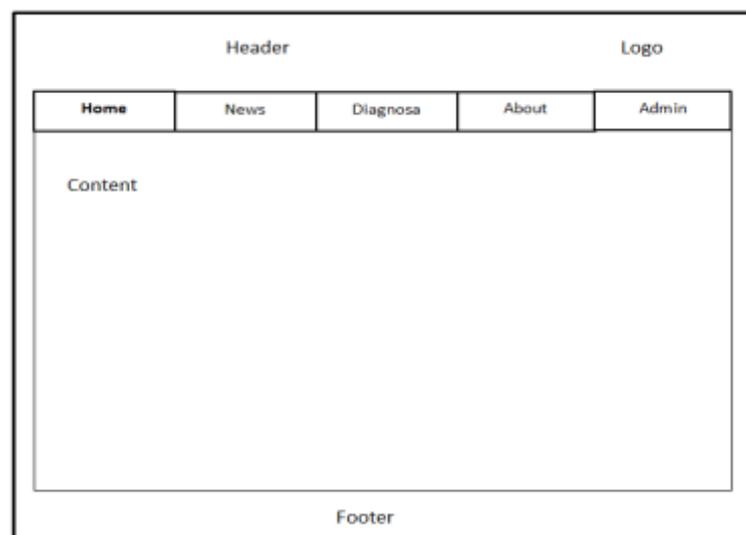
Struktur *class diagram user* pada **Gambar 3.25** diatas menggambarkan bahwa pada sistem pakar memiliki atribut berupa menu *home*, menu *diagnosa*, menu *news* dan menu *about* yang memiliki fungsi yaitu mendiagnosa penyakit asam urat. Sistem pakar terhubung dengan *database* yang memiliki atribut data penyakit, data gejala, data relasi, data analisa hasil dan data pakar serta memiliki fungsi tambah, *edit* dan *delete*. Pada menu *diagnosa* yang dapat diakses user memiliki pertanyaan-pertanyaan dan solusi untuk diagnosa penyakit asam urat.

### 3.4.3 Desain Antarmuka

Berikut ini adalah desain antarmuka yang akan dibuat pada aplikasi sistem pakar mendiagnosa penyakit asam urat:

#### 1. Halaman *Home*

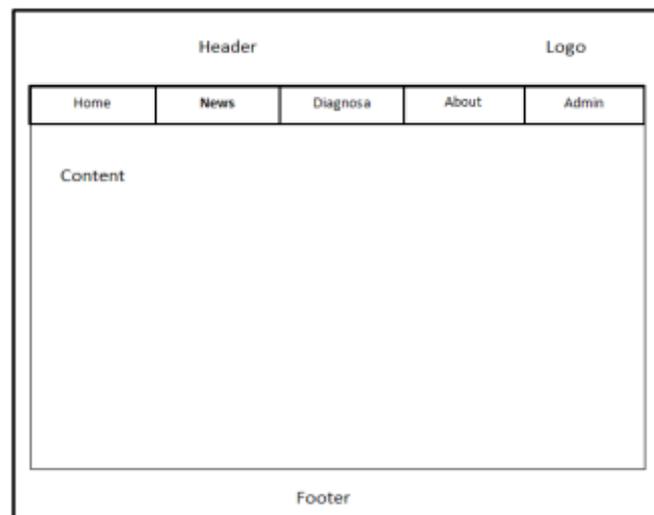
Halaman *home* menampilkan selamat datang pada sistem pakar mendiagnosa penyakit asam urat.



**Gambar 3. 26** Halaman Home  
(Sumber: Data Penelitian, 2018)

## 2. Halaman *News*

Halaman *news* menampilkan informasi mengenai penyakit asam urat yang digunakan pada sistem pakar.



**Gambar 3. 27** Halaman News  
(Sumber: Data Penelitian, 2018)

## 3. Halaman Diagnosa Daftar

Halaman Diagnosa Daftar difungsikan untuk user memasukkan data pribadi awal penggunaan sistem dan kemudian akan diarahkan ke pertanyaan diagnosa penyakit asam urat.

**Gambar 3. 28** Halaman Diagnosa Daftar  
(Sumber: Data Penelitian, 2018)

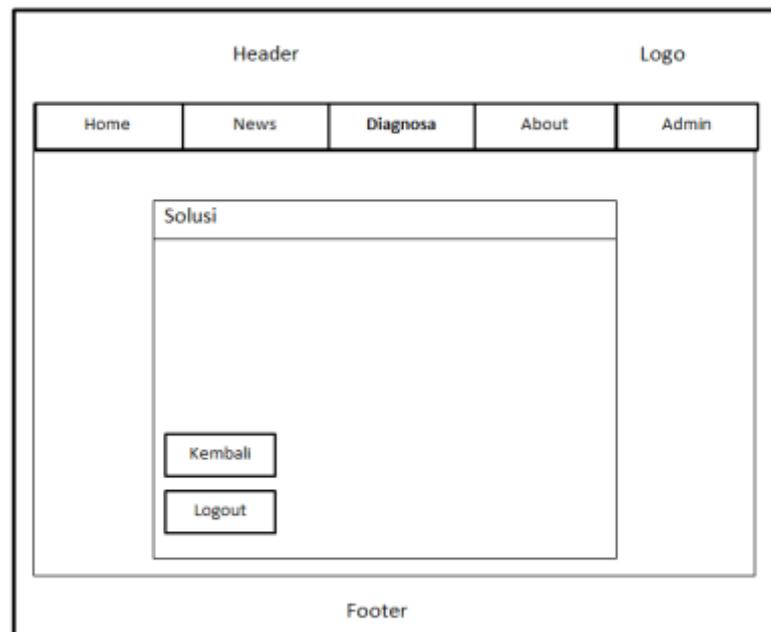
#### 4. Halaman Diagnosa Pertanyaan

Halaman Diagnosa Pertanyaan berfungsi untuk user melakukan diagnosa terhadap penyakit yang dialami *user*. *User* akan menjawab pertanyaan-pertanyaan yang ditampilkan oleh sistem

**Gambar 3. 29** Halaman Diagnosa Pertanyaan  
(Sumber: Data Penelitian, 2018)

## 5. Halaman Diagnosa Solusi

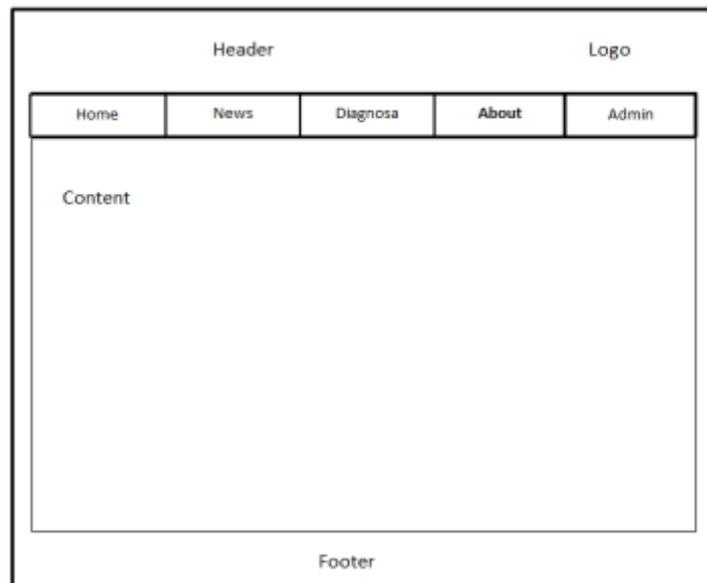
Halaman Diagnosa Solusi ini menampilkan solusi dari setiap pertanyaan yang telah *user* jawab. Solusi tersebut berasal dari database sistem.



**Gambar 3. 30** Halaman Diagnosa Solusi  
(Sumber: Data Penelitian, 2018)

## 6. Halaman *About*

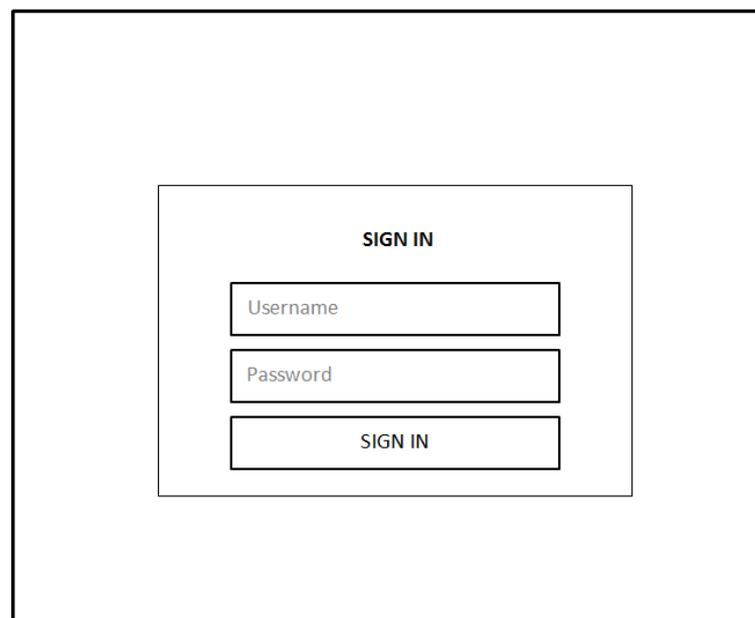
Halaman *about* menampilkan informasi mengenai pemilik sistem pakar diagnosa penyakit asam urat.



**Gambar 3. 31** Halaman About  
(Sumber: Data Penelitian, 2018)

#### 7. Halaman *Login*

Halaman *Login* berfungsi untuk *admin* akan melakukan aktivitas *login* ke sistem untuk dapat melakukan pengeditan data sistem pakar penyakit asam urat.

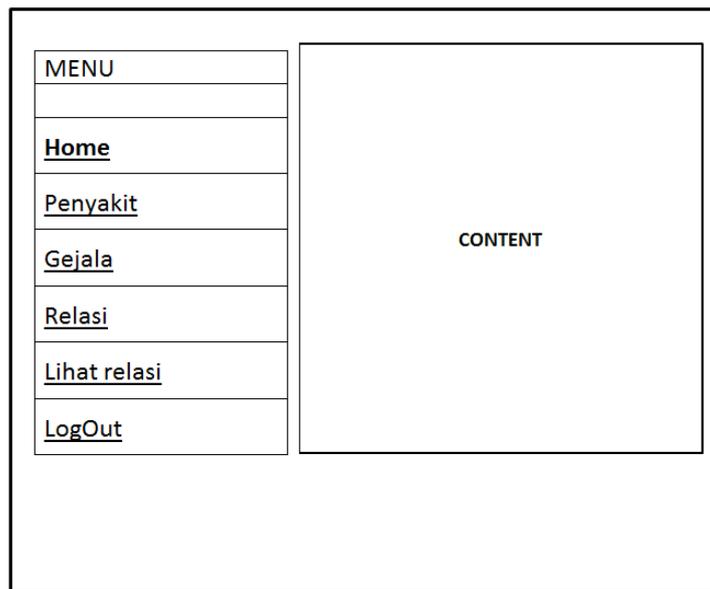


**Gambar 3. 32** Halaman Login  
(Sumber: Data Penelitian, 2018)

### 8. Halaman Khusus *Admin*

Halaman khusus *admin* berfungsi untuk menambahkan data gejala, data penyakit dan solusi, data relasi pada sistem pakar diagnosa penyakit asam urat.

*Admin* akan melakukan pengeditan data sesuai data yang didapat dari Pakar.



**Gambar 3. 33** Halaman Home Admin  
(Sumber: Data Penelitian, 2018)

### 9. Halaman Menu Penyakit

Halaman Menu Penyakit berfungsi untuk *admin* menambahkan, mengedit dan menghapus data penyakit dan solusi pada penyakit asam urat.

MENU		DAFTAR SEMUA TAHAP PENYAKIT			
<a href="#">Home</a>		<b>ID</b>	<b>Nama Penyakit</b>	<b>Pilihan</b>	
<b><a href="#">Penyakit</a></b>		TP01	Penyakit	<a href="#">Edit</a>	<a href="#">Delete</a>
<a href="#">Gejala</a>				<a href="#">Tambah</a>	
<a href="#">Relasi</a>					
<a href="#">Lihat relasi</a>					
<a href="#">LogOut</a>					

**Gambar 3. 34** Halaman Menu Penyakit  
(Sumber: Data Penelitian, 2018)

#### 10. Halaman Tambah Penyakit

Halaman tambah penyakit ini berfungsi untuk *admin* menambahkan mengedit dan menghapus data penyakit asam urat.

MENU		ENTRY DATA	
<a href="#">Home</a>		ID :	<input type="text" value="TP01"/>
<b><a href="#">Penyakit</a></b>		Penyakit :	<input type="text"/>
<a href="#">Gejala</a>		Solusi :	<input type="text"/>
<a href="#">Relasi</a>			
<a href="#">Lihat relasi</a>			
<a href="#">LogOut</a>			
		<input type="button" value="Simpan"/>	<input type="button" value="Batal"/>

**Gambar 3. 35** Halaman Tambah Penyakit  
(Sumber: Data Penelitian, 2018)

### 11. Halaman *Edit Penyakit*

Halaman *edit* penyakit ini berfungsi untuk *admin* mengedit data penyakit asam urat.

<b>MENU</b>	<b>EDIT DATA</b>
<a href="#">Home</a>	ID : <input type="text" value="TP01"/>
<b><a href="#">Penyakit</a></b>	Penyakit : <input type="text" value="a"/>
<a href="#">Gejala</a>	Solusi : <input type="text" value="1. 2. 3."/>
<a href="#">Relasi</a>	
<a href="#">Lihat relasi</a>	
<a href="#">LogOut</a>	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>

**Gambar 3. 36** Halaman *Edit Penyakit*  
(Sumber: Data Penelitian, 2018)

### 12. Halaman Menu Gejala

Halaman Menu Gejala ini berfungsi untuk admin menambahkan mengedit dan menghapus data gejala penyakit asam urat.

MENU	DAFTAR SEMUA GEJALA			
	<b>ID</b>	<b>Nama Gejala</b>	<b>Pilihan</b>	
<u>Home</u>	G001	Gejala	<u>Edit</u>	<u>Delete</u>
<u>Penyakit</u>			<u>Tambah</u>	
<u>Gejala</u>				
<u>Relasi</u>				
<u>Lihat relasi</u>				
<u>LogOut</u>				

**Gambar 3. 37** Halaman Menu Gejala  
(Sumber: Data Penelitian, 2018)

### 13. Halaman Tambah Gejala

Halaman tambah gejala ini berfungsi untuk admin menambahkan mengedit dan menghapus data gejala penyakit asam urat.

<b>MENU</b>	<b>MASUKKAN DATA GEJALA</b>
<a href="#">Home</a>	Kode : <input type="text" value="G001"/>
<a href="#">Penyakit</a>	Gejala : <input type="text" value="-"/>
<a href="#">Gejala</a>	<input type="text" value="-"/>
<a href="#">Relasi</a>	<input type="text" value="-"/>
<a href="#">Lihat relasi</a>	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
<a href="#">LogOut</a>	

**Gambar 3. 38** Halaman Tambah Gejala

(Sumber: Data Penelitian, 2018)

#### 14. Halaman *Edit* Gejala

Halaman *edit* gejala ini berfungsi untuk admin mengedit data gejala penyakit asam urat

<b>MENU</b>	<b>EDIT DATA GEJALA</b>
<a href="#">Home</a>	Kode : <input type="text" value="G001"/>
<a href="#">Penyakit</a>	Gejala : <input type="text" value="a"/>
<b><a href="#">Gejala</a></b>	<input type="button" value="Simpan"/> <input type="button" value="Batal"/>
<a href="#">Relasi</a>	
<a href="#">Lihat relasi</a>	
<a href="#">LogOut</a>	

**Gambar 3. 39** Halaman Edit Gejala  
(Sumber: Data Penelitian, 2018)

#### 15. Halaman Menu Relasi

Halaman relasi adalah halaman untuk admin membuat relasi antar data penyakit dengan data gejala sesuai dengan tabel keputusan yang telah dibuat.

<b>MENU</b>	<b>INPUT RELASI</b>
<a href="#">Home</a>	<input type="text" value="Data Penyakit"/>
<a href="#">Penyakit</a>	<input type="text" value="Data Gejala"/>
<b><a href="#">Gejala</a></b>	<input type="button" value="Simpan"/> <input type="button" value="Normalkan"/>
<a href="#">Relasi</a>	
<a href="#">Lihat relasi</a>	
<a href="#">LogOut</a>	

**Gambar 3. 40** Halaman Menu Relasi  
(Sumber: Data Penelitian, 2018)

## 16. Halaman Menu Lihat Relasi

Halaman menu lihat relasi ini adalah halaman yang menampilkan relasi yang telah dibuat antara data penyakit dan data gejala penyakit.

<b>MENU</b>		
<u>Home</u>		
<u>Penyakit</u>		
<u>Gejala</u>		
<u>Relasi</u>		
<b><u>Lihat relasi</u></b>		
<u>LogOut</u>		

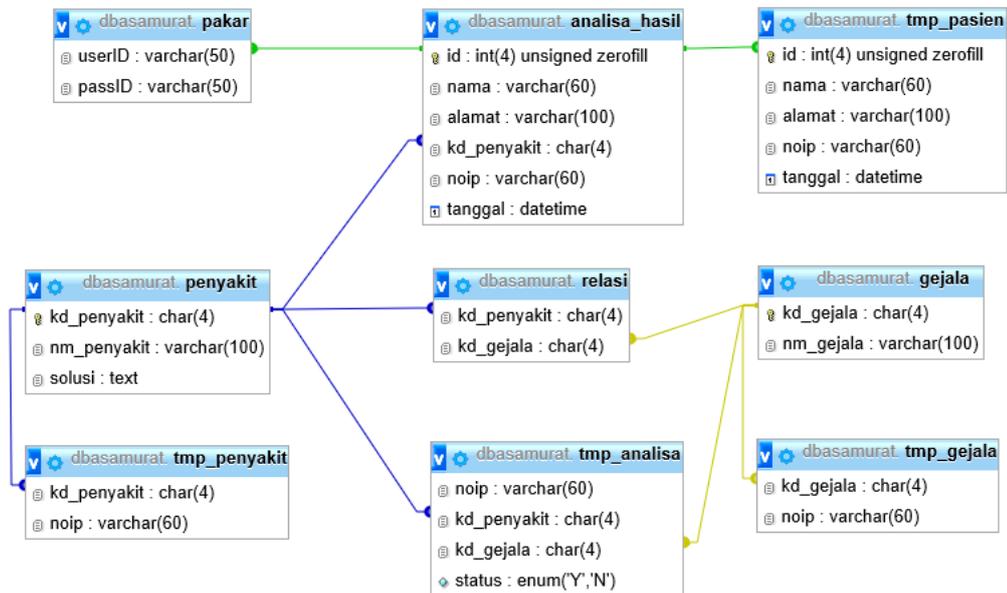
  

<b>TAMPILKAN GEJALA PER PENYAKIT</b>		
<b>Penyakit</b>	List Daftar Penyakit	
	<input type="button" value="Tampil"/>	
NAMA PENYAKIT :		
<b>DAFTAR GEJALA</b>		
<b>Penyakit</b>	<b>Gejala</b>	<b>Nama Gejala</b>
TP01	G001	

**Gambar 3. 41** Halaman Lihat Relasi  
(Sumber: Data Penelitian, 2018)

### 3.4.4 Desain Database

(Rosa A.S, 2018) PDM (*Physical Data Model*) adalah model yang dibuat dengan sejumlah table untuk membuat data serta hubungan antar data-data tersebut. PDM merupakan konsep yang menjelaskan detail bagaimana data disimpan di dalam *database*. Berikut ini adalah konsep database yang dibuat dalam penelitian:



**Gambar 3. 42** Desain Physical Data Model  
(Sumber: Data Penelitian, 2018)

Pada **Gambar 3.26** diatas, terdapat 5 tabel *database* yang terdiri dari: tabel analisa hasil, tabel penyakit, tabel gejala, tabel relasi dan tabel pakar, serta memiliki 4 tabel temporary yaitu tabel temp analisa, tabel temp penyakit, tabel temp gejala dan tabel temp analisa.

### 3.5 Lokasi dan Jadwal Penelitian

#### 3.5.1 Lokasi Penelitian

Penelitian ini dilakukan di Rumah Sakit Embung fatimah kota Batam Jl. R. Soeprpto. Blok D1-9. Batu Aji. Batam Batam. Alasan peneliti memilih rumah sakit ini sebagai lokasi penelitian adalah

1. Lokasi yang mudah ditempuh
2. Efisiensi biaya dan waktu



