

BAB III: METODOLOGI PENELITIAN

3.1 Instrumen Penelitian

Jenis penelitian yang digunakan adalah *explanatory research* yang bertujuan untuk mengkaji hubungan antar variabel yang di hipotesiskan sehingga dapat ditarik kesimpulan tentang masalah yang sedang dihadapi. Narasumber dalam penelitian ini adalah penyuluh petani kangkung bapak Irman Rudianto.

3.2 Analisis Kebutuhan

Untuk mendukung pembuatan sistem pakar ini perlu adanya sistem komputer yang mendukung, baik dari segi perangkat lunak (*software*) dan juga perangkat keras (*hardware*).

3.2.1 Analisis Software

Berikut adalah kebutuhan perangkat lunak (*software*) untuk membuat sistem pakar diagnosis hama dan penyakit tanaman kangkung :

- a. Sistem Operasi Windows 10 64 bit
- b. XAMPP
- c. *Browser* (Google Chrome, Mozilla Firefox, Microsoft Edge)
- d. *Microsoft Office* (*Word*, *Powerpoint*)
- e. *Visual Studio Code*

3.2.2 Analisis Hardware

Spesifikasi perangkat keras (*hardware*) yang digunakan untuk membuat sistem pakar diagnosis hama dan penyakit tanaman kangkung :

- a. Processor Intel Core i5
- b. RAM 8 GB
- c. Hardisk 1 TB
- d. VGA 2 GB
- e. Monitor LG
- f. *Keyboard*
- g. *Mouse*.

3.3 Teknik Pengumpulan Data

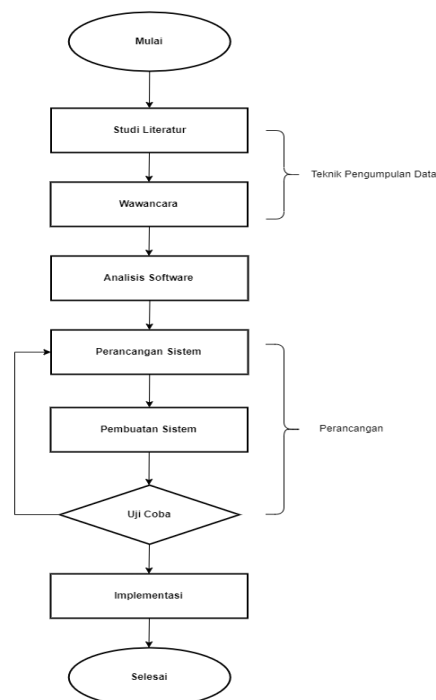
3.3.1 Teknik Wawancara

Teknik wawancara ini adalah mengumpulkan informasi atau data dengan cara tanya jawab langsung ke ahli dan narasumber petani kangkung yaitu pak Irman Rudianto. Metode wawancara ini bertujuan untuk mengetahui gejala-gejala, jenis penyakit dan hama dari tanaman kangkung.

3.3.2 Teknik Pustaka

Teknik pustaka ini adalah mengumpulkan referensi atau studi literatur dalam bentuk buku, karya ilmiah, referensi yang berasal dari internet ataupun jurnal.

Proses berpikir dalam pembuatan sistem pakar mulai dari pengumpulan informasi penelitian hingga pengimplementasian. Berikut adalah tahap-tahapannya :



Gambar 3.1 Kerangka Berpikir

Tahapan-tahapan kerangka berpikir di atas adalah :

- a. Tahapan teknik pengumpulan data meliputi metode studi literatur dan wawancara dengan narasumber, yaitu mempelajari pengetahuan

dari berbagai sumber baik dari buku, penelitian terdahulu, artikel, jurnal maupun internet. Metode wawancara adalah teknik pengumpulan data dengan melakukan tanya jawab kepada pihak atau seorang ahli yang mempunyai kredibilitas tentang pengetahuan yang dibutuhkan dalam penelitian ini.

- b. Tahap analisis *software* dilakukan sebelum tahapan perancangan. Pada tahap ini penulis melakukan analisis tentang kebutuhan *software* untuk mendukung proses pembuatan sistem pakar ini.
- c. Pada tahap perancangan meliputi perancangan sistem, pembuatan sistem dan uji coba. Perancangan merupakan serangkaian proses dari perancangan sistem hingga melakukan uji coba. Sistem pakar dirancang menggunakan bantuan UML (*Unified Modelling Language*) kerana sistem ini berbasis web. Perancangan sistem yang dimaksud seperti merancang desain antarmuka dan komponen penyusun sistem pakar.
- d. Pada tahap terakhir adalah tahap implementasi dari kerangka berpikir sebelum sistem ini dioperasikan ke pihak yang menjadi sasaran penelitian.

3.4 Analisis Sistem

Dalam mendiagnosis hama dan penyakit tanaman kangkung harus mengetahui dahulu gejala-gejala yang timbul. Berdasarkan gejala yang dapat diketahui dapat ditarik kesimpulan berupa suatu diagnosis hama dan penyakit. Untuk menentukan suatu hama dan penyakit tanaman kangkung perlu di diklasifikasikan menjadi beberapa jenis yaitu :

1. Hama Bekicot (*Achatina fulica*)
2. Hama Ulat Grayak (*Spodoptera*)
3. Hama Kutu Daun (*Aphidoidea*)
4. Hama Ulat Keket (*Agrius convolvuli*)
5. Penyakit Karat Putih
6. Penyakit Bercak daun.
7. Penyakit Karena Bakteri

8. Penyakit Karena Virus
9. Penyakit Karena Alga

3.5 Perancangan Sistem Pakar

Dalam perancangannya, sistem pakar diagnosis hama dan penyakit tanaman kangkung ini dirancang menggunakan UML (*Unified Modelling Language*) sebagai pemodelan. Berikut adalah tahapan perancangan UML menggunakan beberapa diagram antara lain :

3.5.1 *Software Requirement Analysis*

Untuk memahami dan mengevaluasi kerja sistem digunakanlah *use case diagram*. *Use case diagram* sistem pakar hama dan penyakit tanaman kangkung dapat dilihat pada gambar di bawah ini :



Gambar 3.2 Use Case Diagram Sistem Pakar

Dapat dilihat dari gambar 3.2 terdapat dua aktor, yaitu *user* dan *admin*. Pada *user* dapat melihat menu beranda dan ketika melakukan konsultasi *user* akan diminta untuk mengisi data konsultasi terlebih dahulu, lalu ketika konsultasi *user* akan diminta menjawab pertanyaan dari sistem, pertanyaan tersebut adalah gejala dari tanaman kangkung dengan metode *Forward*

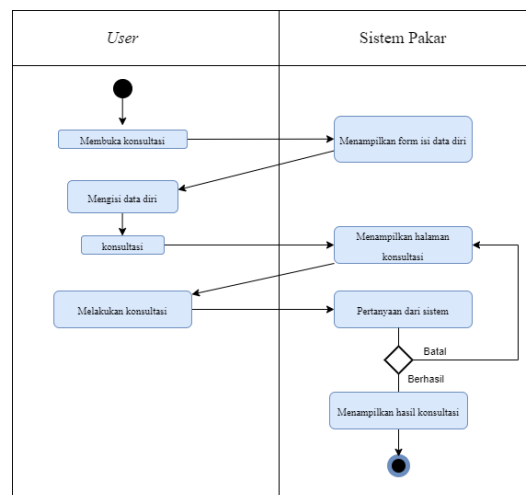
Chaining. Outputnya nanti sistem akan menampilkan riwayat pertanyaan pada saat konsultasi dan hasil konsultasi. Sedangkan *admin* dapat mengakses semua menu yang terdapat pada sistem serta *admin* dapat mengolah data yang ada pada sistem tersebut, guna memberikan pemeliharaan pada sistem.

3.5.2 Activity Diagram

Dari seluruh tahapan alur kerja merupakan representasi grafis *Activity Diagram*. Diagram sistem pakar diagnosis hama dan tanaman ini digunakan untuk menjelaskan alur kerja operasional langkah demi langkah dari komponen suatu sistem.

3.5.2.1 Activity Diagram Konsultasi

Ketika *user* melakukan konsultasi maka sistem akan menampilkan *form* isi data diri. *User* diminta untuk mengisi data diri, kemudian sistem akan menampilkan halaman konsultasi dengan pertanyaan-pertanyaan yang diajukan oleh sistem. *Activity Diagram* registrasi bisa ditinjau pada gambar 3.3 berikut ini.

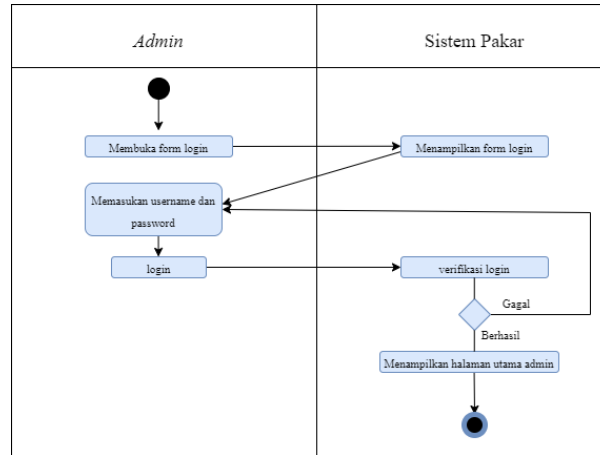


Gambar 3.3 Activity Diagram Konsultasi

3.5.2.2 Activity Diagram Log in Admin

Alur *activity diagram log in admin* dimulai dari *admin* melakukan *log in* menggunakan *username* serta *password* pada form yang sudah disediakan oleh sistem. Sistem selanjutnya akan memverifikasi *username* serta *password* sudah sinkron atau belum.

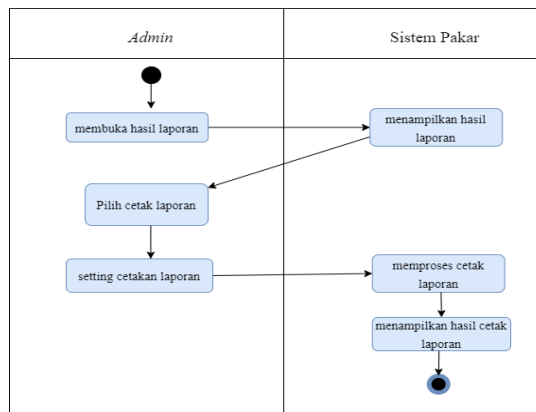
Activity Diagram Log in Admin data dilihat di gambar 3.4 berikut ini.



Gambar 3.4 Activity Diagram Log In Admin

3.5.2.3 Activity Diagram Lihat Hasil Laporan

Hasil diagnosis *user* dapat dilihat sesudah melakukan proses konsultasi hingga sistem mendiagnosis suatu penyakit. *Admin* dapat mencetak hasil diagnosis atau hanya menyimpan saja. *Activity diagram* lihat hasil diagnosis bisa dilihat di gambar 3.5 di bawah ini.

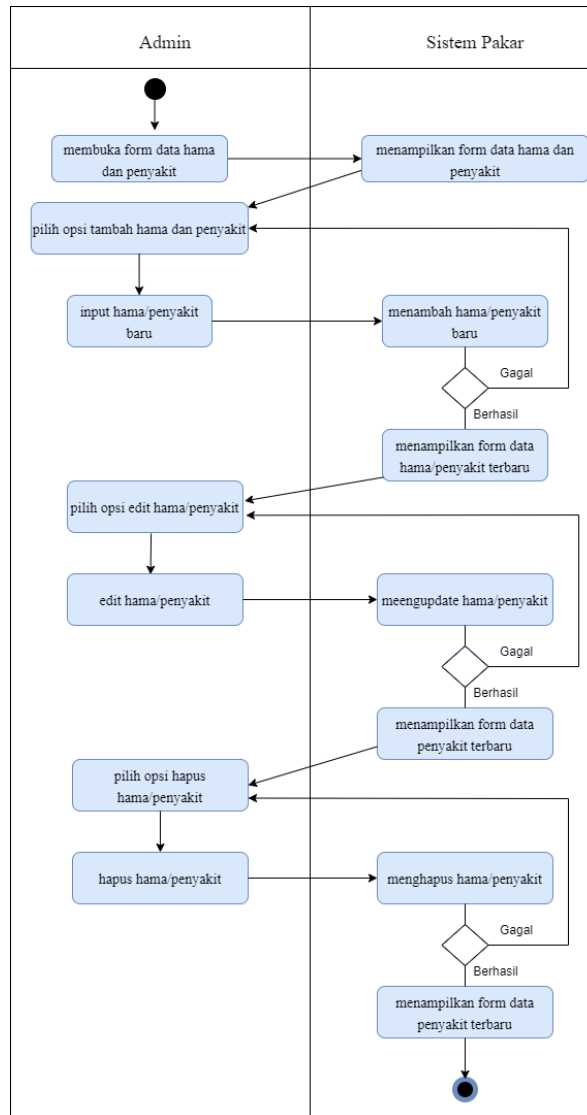


Gambar 3.5 Activity Diagram Lihat Hasil Diagnosis

3.5.2.4 Activity Diagram Data Penyakit

Data penyakit mempunyai fitur tambah, edit, dan hapus penyakit. *Admin* bisa menginput penyakit baru selanjutnya sistem akan memprosesnya serta menyimpan. Hal sama juga untuk edit

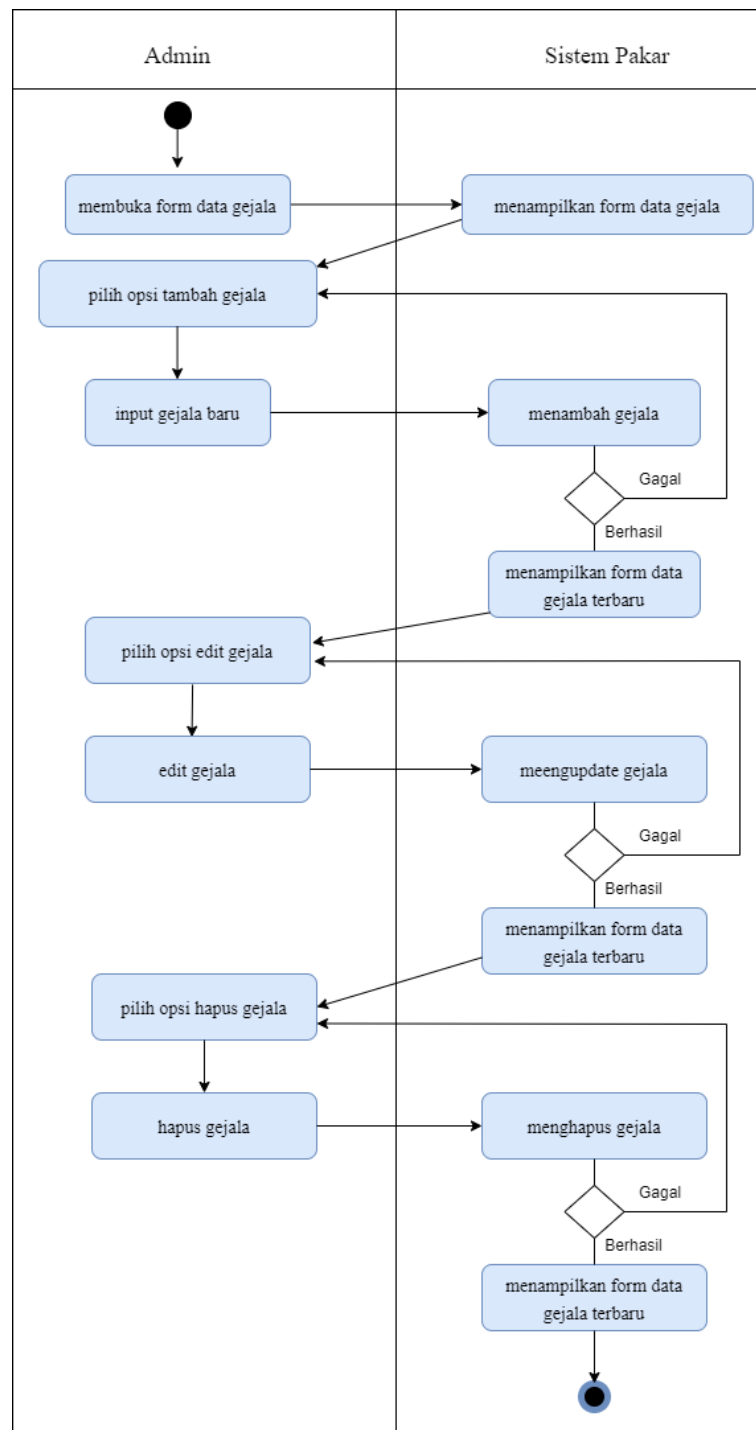
dan hapus penyakit. dapat dilihat *activity diagram* data penyakit di gambar 3.6 di bawah ini



Gambar 3.6 Activity Diagram Data Penyakit

3.5.2.5 Activity Diagram Data Gejala

Pada data gejala admin dapat menambah gejala, kemudian sistem akan memproses data baru itu berhasil maka menyimpannya, tetapi apabila gagal akan mengulangi proses sebelumnya. *Activity diagram* data gejala dapat dilihat pada gambar 3.7 di bawah ini.

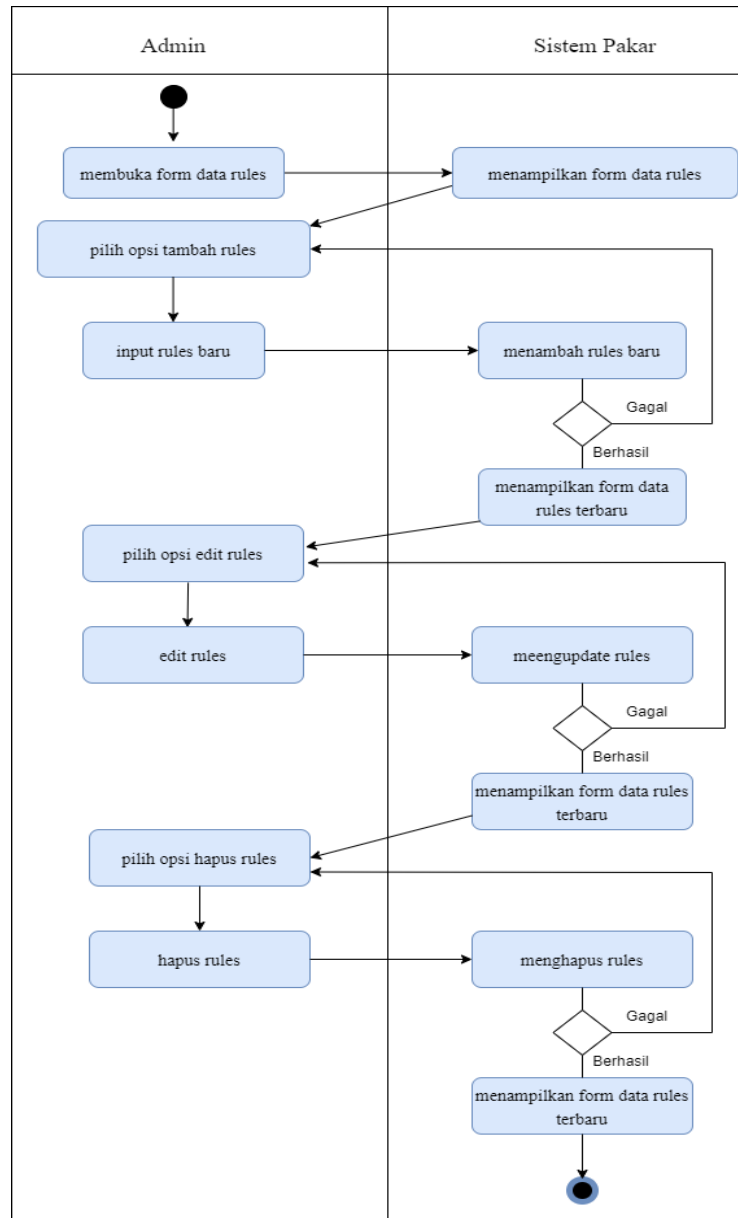


Gambar 3.7 Activity Diagram Data Gejala

3.5.2.6 Activity Diagram Data Rules

pada data *rules admin* dapat menambah *rules*, lalu sistem akan memproses data baru itu berhasil maka menyimpannya,

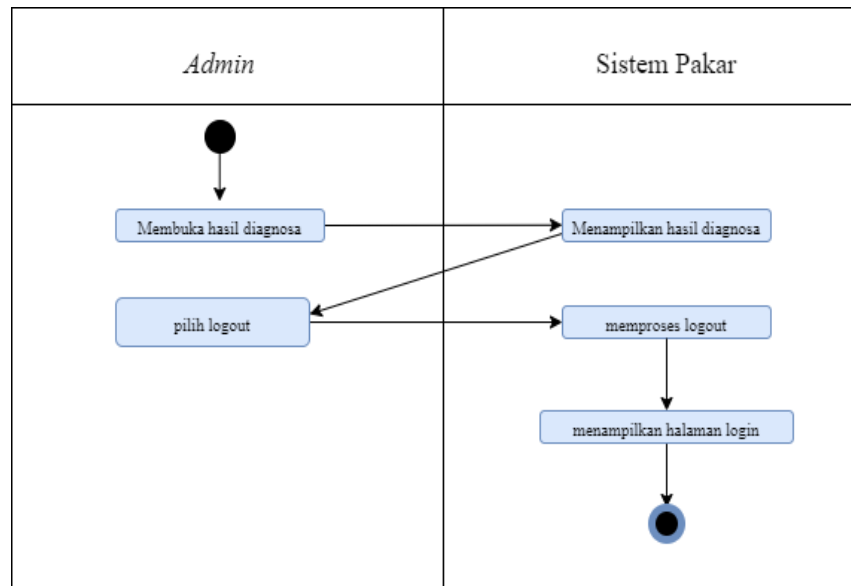
tetapi apabila gagal akan mengulangi proses sebelumnya. *Activity diagram data rules* dilihat pada gambar 3.8 di bawah ini.



Gambar 3.8 Activity Diagram Data Rules

3.5.2.7 Activity Diagram Log out Admin

Proses *log out* untuk keluar dari *menu admin* dimulai dari pilih *log out* lalu sistem memprosesnya untuk keluar dari halaman *admin*. Dapat kita lihat *activity diagram log out* pada gambar 3.9 di bawah ini.



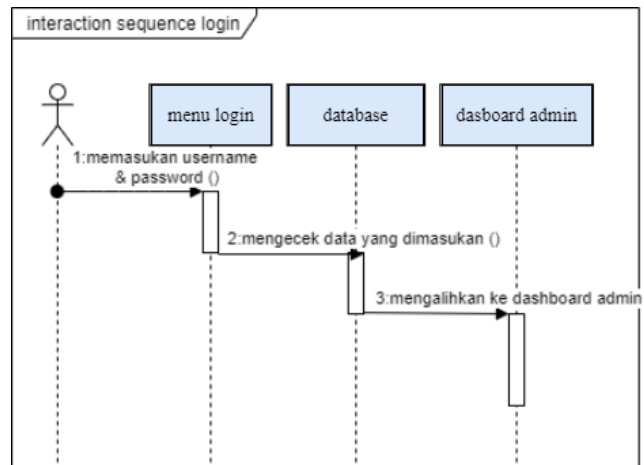
Gambar 3.9 Activity Diagram Log Out Admin

3.5.3 Sequence Diagram

Urutan proses yang dilakukan *user* dari awal konsultasi sampai akhir hasil diagnosis hama dan penyakit kangkung berdasarkan gejala dan ciri-ciri yang dijawab oleh *user* adalah inti dari *Sequence Diagram*. Selain itu, terdapat juga urutan proses yang dilakukan pakar untuk mengelola data. Berikut adalah beberapa *Sequence Diagram* sistem pakar hama dan penyakit kangkung.

3.5.3.1 Sequence Diagram Log in

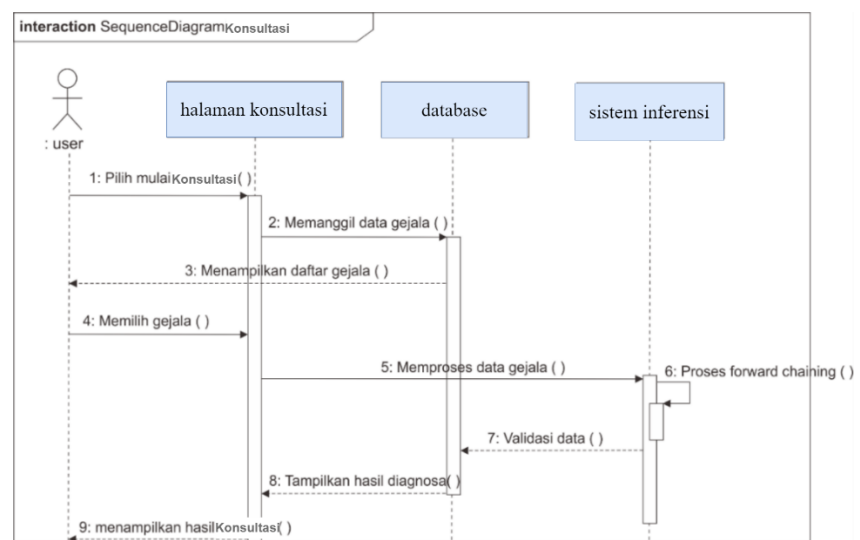
Admin melakukan *log in* dimulai dari menentukan *menu log in* kemudian memasukan *username* dan *password*. Selanjutnya *database* memproses *input user*. Jika berhasil dialihkan ke *dashboard admin* Jika gagal diulang ke step sebelumnya. pada gambar 3.10 dapat kita lihat *sequence diagram log in*.



Gambar 3.10 Sequence Diagram Log In

3.5.3.2 Sequence Diagram Konsultasi

Proses konsultasi diawali dari *user* memilih gejala yang diberikan oleh sistem dengan memanggil data gejala dari *database*. Kemudian data gejala yang terpilih akan diproses oleh sistem inferensi dan hasil diagnosis ditampilkan ke *user*. *Sequence diagram* konsultasi dapat dilihat pada gambar 3.11 di bawah ini.

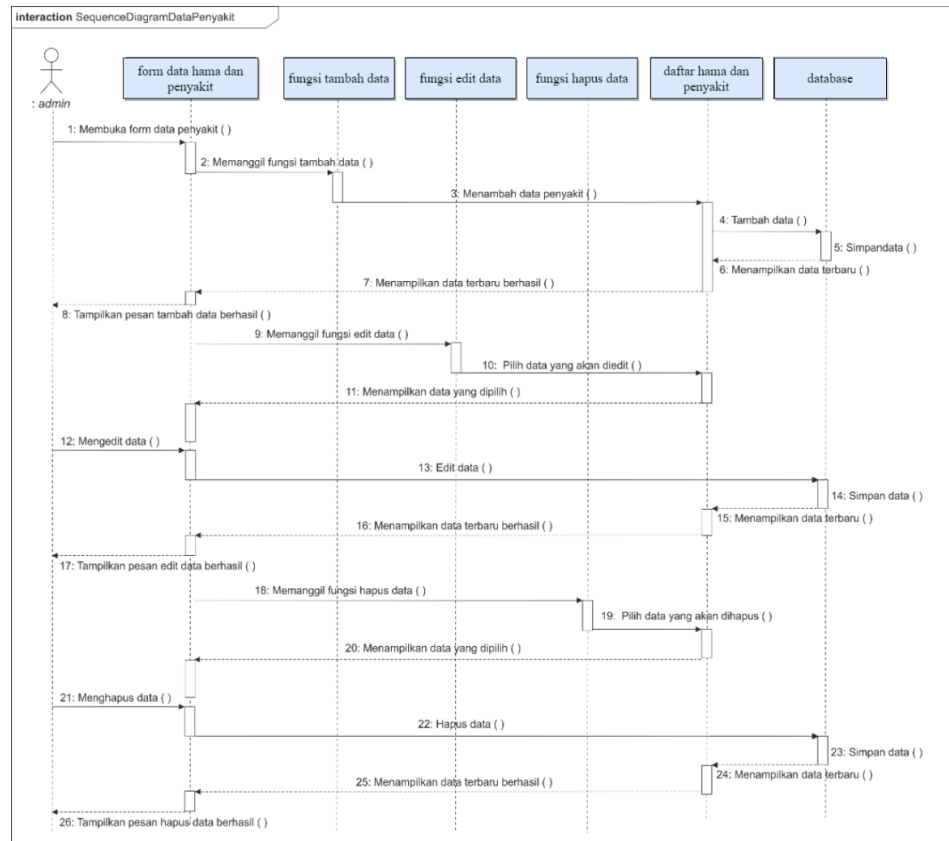


Gambar 3.11 Sequence Diagram Diagnosis Hama dan Penyakit

3.5.3.3 Sequence Diagram Data Hama dan Penyakit

Data penyakit memiliki fungsi untuk menambah, mengedit dan menghapus penyakit. Pemrosesan data dimulai dari memilih fungsi yang dipilih kemudian data tersebut dikirim ke *database*

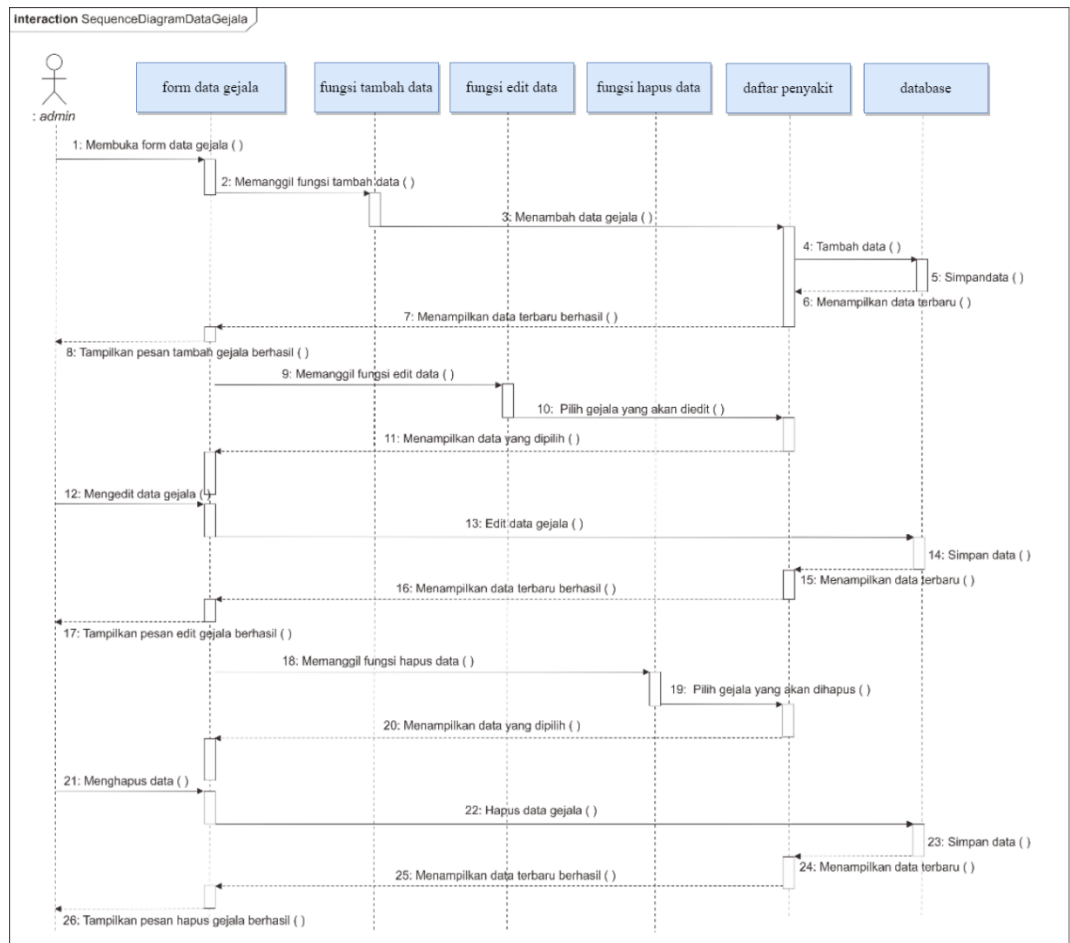
untuk diproses. *Sequence diagram* data penyakit dapat dilihat pada gambar 3.12 di bawah ini.



Gambar 3.12 Sequence Diagram Data Hama dan Penyakit

3.5.3.4 Sequence Diagram Data Gejala

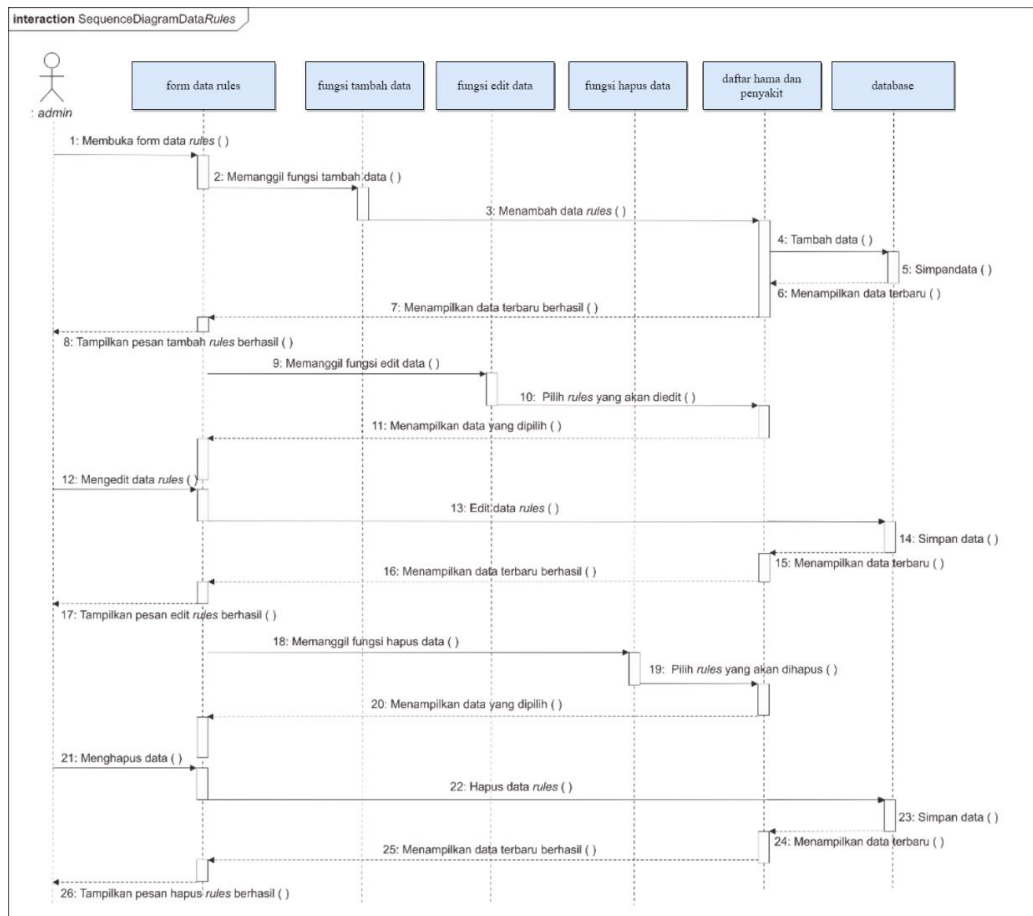
Data gejala memiliki fungsi untuk menambah, mengedit dan menghapus gejala. Pemrosesan data dimulai dari memilih fungsi yang dipilih kemudian data tersebut dikirim ke *database* untuk diproses. *Sequence diagram* data gejala dapat dilihat pada gambar 3.13 di bawah ini.



Gambar 3.13 *Sequence Diagram* Data Gejala

3.5.3.5 *Sequence Diagram* Data Rules

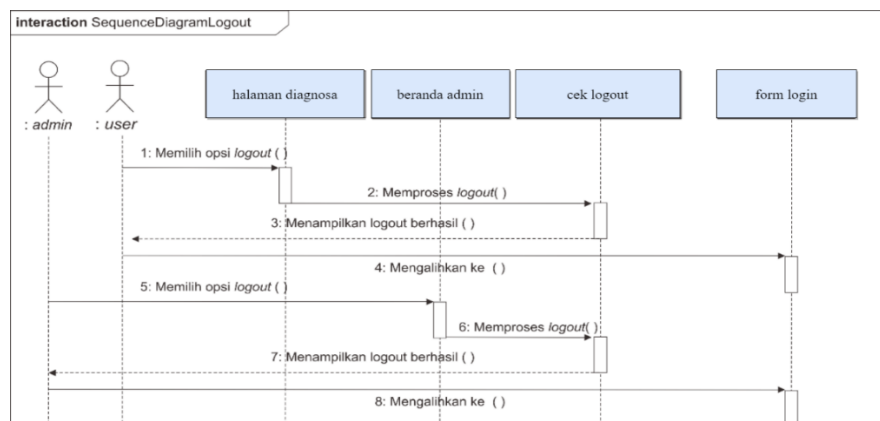
Data *rules* memiliki fungsi untuk menambah, mengedit dan menghapus *rules*. Pemrosesan data dimulai dari memilih fungsi yang dipilih kemudian data tersebut dikirim ke *database* untuk diproses. *Sequence diagram* data *rules* dapat dilihat pada gambar 3.14 di bawah ini.



Gambar 3.14 Sequence Diagram Data Rules

3.5.3.6 Sequence Diagram Log out

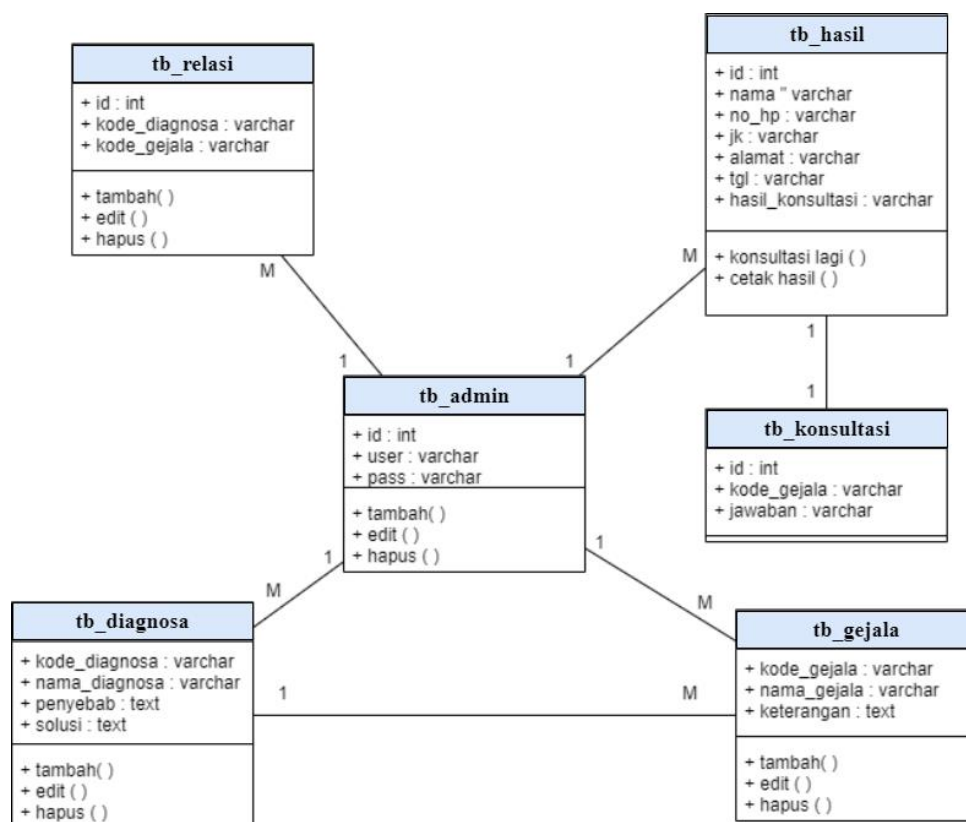
Proses *log out admin* diawali dari halaman admin pada opsi *log out*, setelah berhasil akan dialihkan ke halaman *log in*.



Gambar 3.15 Sequence Diagram Log Out

3.5.4 Class Diagram

Class diagram menunjukkan hubungan antar *class* dalam sistem yang sedang dibangun dan bagaimana *class* tersebut saling berkolaborasi untuk mencapai suatu tujuan. Di dalam *database* terdapat 6 tabel yaitu *tb_relasi*, *tb_konsultasi*, *tb_hasil*, *tb_diagnosis*, *tb_gejala*, dan *tb_admin*. Berikut adalah *class diagram* sistem pakar diagnosis penyakit hama dan penyakit kangkung pada gambar 3.16 berikut.



Gambar 3.16 Class Diagram

3.5.5 Perancangan Database

Perancangan *database* sistem pakar diagnosis hama dan penyakit tanaman kangkung ini menggunakan *database* MySQL. Berikut adalah tabel strukturnya :

1. Tabel Admin

Tabel ini berfungsi masuk ke dalam sistem sebagai *administrator*. Struktur tabel dapat dilihat pada tabel 3.1 berikut.

Tabel 3.1 Tabel Admin

<i>Field</i>	<i>Type</i>	Panjang	Keterangan
<i>id</i>	int	10	<i>Primary Key</i>
<i>user</i>	Varchar	16	
pass	Varchar	16	

2. Tabel Diagnosis

Tabel penyakit ini berfungsi untuk menyimpan data deskripsi hama dan penyakit, penyebab serta solusi. Struktur tabel dapat dilihat pada tabel 3.2 berikut.

Tabel 3.2 Tabel Diagnosis

<i>Field</i>	<i>Type</i>	Panjang	Keerangan
kode_diagnosis	varchar	16	<i>Primary Key</i>
nama_diagnosis	varchar	25	
Penyebab	text		
Solusi	text		

3. Tabel Gejala

Tabel gejala dalah tabel yang menyimpan data gejala hama dan penyakit tanaman kangkung. Struktur tabel dapat dilihat pada tabel 3.3 berikut.

Tabel 3.3 Tabel Gejala

<i>Field</i>	<i>Type</i>	Panjang	Keterangan
kode_gejala	varchar	16	<i>Primary Key</i>
nama_gejala	varchar	25	

<i>Field</i>	<i>Type</i>	Panjang	Keterangan
Keterangan	text		

4. Tabel Konsultasi

Tabel konsultasi berfungsi untuk menampung hasil jawaban dari petani. Struktur dapat dilihat pada tabel 3.4 berikut.

Tabel 3.4 Tabel konsultasi

<i>Field</i>	<i>Type</i>	Panjang	Keterangan
Id	int	11	<i>Primary Key</i>
kode_konsultasi	varchar	16	
Jawaban	varchar	6	

5. Tabel Relasi

Tabel relasi berfungsi untuk menyimpan hubungan antara data penyakit dan data gejala. Struktur tabel dapat dilihat pada tabel 3.5 berikut.

Tabel 3.5 Tabel Hasil Relasi

<i>Field</i>	<i>Type</i>	Panjang	Keterangan
id	int	11	<i>Primary Key</i>
kode_diagnosis	varchar	16	
kode_gejala	varchar	16	

6. Tabel hasil

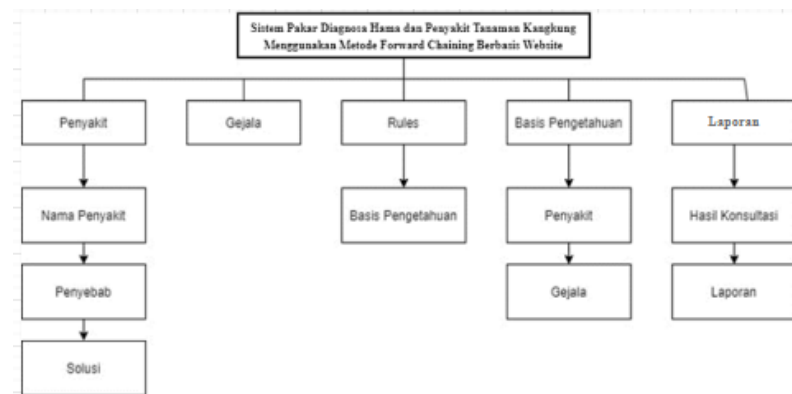
Tabel hasil berfungsi untuk menyimpan hasil diagnosis dari petani yang telah melakukan diagnosis. Struktur tabel dapat dilihat pada tabel 3.6 berikut.

Tabel 3.6 Tabel Hasil Diagnosis

<i>Field</i>	<i>Type</i>	Panjang	Keterangan
id	int	25	<i>Primary Key</i>
nama	varchar	50	
no_hp	varchar	50	
jk	varchar	50	
alamat	varchar	50	
tgl	varchar	50	
hasil_konsultasi	varchar	50	

3.5.6 Perancangan *Interface*

Proses perancangan *interface* ini merupakan acuan untuk membangun aplikasi yang akan penulis buat. Adapun struktur rancangan *interface* ini dapat dilihat pada gambar 3.17 berikut.



Gambar 3.17 Perancangan *Work Breakdown Structure* Desain *Interface*

Work Breakdown Structure (WBS) merupakan gambaran secara struktural tentang sistem yang akan dibuat. Pada gambar 4.1 di atas dapat dilihat bahwa terdiri dari beberapa bagian seperti ; penyakit, gejala, *rules*, basis pengetahuan, dan konsultasi. Berikut merupakan perancangan desain *interface*.

A. Rancangan Halaman Utama

Rancangan desain *interface* pada halaman utama ini merupakan menu awal yang pertama kali ditampilkan saat *user* membuka sistem pakar ini. Terdapat beberapa tombol seperti beranda, *log in* dan mulai konsultasi. Rancangan halaman utama dapat dilihat pada gambar 3.18 di bawah ini.

The image shows a web page layout. At the top, there are two buttons: 'Beranda' and 'Login'. Below them, the main heading reads 'Sistem Pakar Diagnosa Hama dan Penyakit Tanaman Kangkung'. Centered below the heading is a button labeled 'Mulai Konsultasi'.

Gambar 3.18 Rancangan Halaman Utama

Tombol konsultasi berfungsi untuk memulai konsultasi dan sistem akan mengalihkan secara otomatis ke halaman isi biodata konsultan sebelum memulai konsultasi. Rancangan form biodata konsultan dapat dilihat pada gambar 3.19 berikut.

The image shows a form titled 'Isi Data Konsultasi'. At the top, there are 'Beranda' and 'Login' buttons. The form contains the following fields:

- 'Nama *' with a text input field containing 'masukan nama...'
- 'No.Hp *' with a text input field containing 'masukan nama...'
- 'Jenis Kelamin *' with two radio button options: 'Laki-laki' and 'Perempuan'.
- 'Alamat *' with a text input field containing 'masukan nama...'
- 'Tanggal Konsultasi *' with a text input field containing 'hh/mm dd/mm/yyyy'.

 At the bottom of the form is a button labeled 'button Lanjutkan'.

Gambar 3.19 Rancangan *Form* Biodata Konsultan

Di dalam form biodata konsultan harus diisi semua kecuali tanggal konsultasi yang sudah diatur otomatis berdasarkan waktu pada saat konsultasi.

B. Rancangan Halaman Konsultasi

Rancangan konsultasi ini merupakan suatu rangkaian layanan dengan konsultan akan memilih serangkaian gejala yang diberikan oleh sistem berdasarkan gejala atau ciri-ciri yang muncul di tanaman kangkung. Kemudian konsultan akan diminta untuk memilih opsi ya atau tidak pada *button* yang disediakan. Pada saat pertanyaan kedua dan seterusnya konsultan dapat melakukan pembatalan dan melihat hasil konsultasi sesuai kebutuhan. Rancangan konsultasi atau halaman pilih gejala dapat dilihat pada gambar 3.20 di bawah ini.

Gambar 3.20 Rancangan Halaman Konsultasi

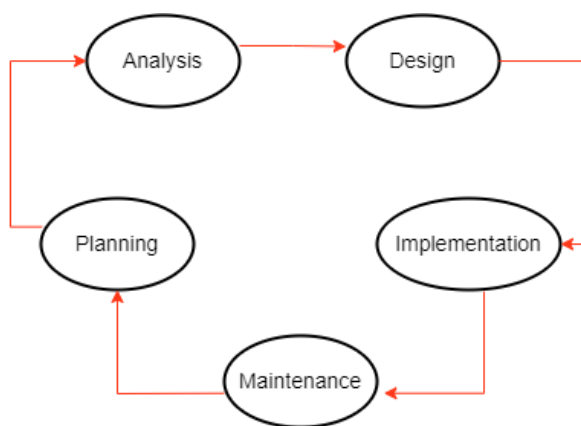
3.6 Metodologi Penelitian

Penelitian ini menggunakan metode deskriptif kualitatif. *Forward Chaining* merupakan fakta untuk mendapatkan kesimpulan (*conclusion*) dari fakta tersebut [9]. Penalaran ini berdasarkan fakta yang ada (*data driven*), metode ini adalah kebalikan metode *Backward Chaining*, dimana metode ini dijalankan dengan mengumpulkan fakta-fakta yang ada untuk menarik kesimpulan. Dengan kata lain, prosesnya dimulai dari *facts* (fakta-fakta yang ada) melalui proses *interface fact* (penalaran fakta-fakta) menuju suatu goal (suatu tujuan) [32]. Metode ini juga disebut menggunakan aturan *IF-THEN* dimana *premise (IF)* menuju *conclusion (THEN)* atau dapat juga dituliskan sebagai berikut: *THEN* (konklusi). Ada dua

pendapat mengenai pelaksanaan metode ini. Pertama dengan cara membawa seluruh data yang didapat ke sistem pakar. Kedua dengan membawa bagian-bagian penting saja dari data yang didapat ke sistem pakar. Cara pertama lebih baik digunakan jika sistem pakar terhubung dengan proses otomatis dan penerima seluruh data dari *database*. Kedua jenis strategi ini akan mengarah pada suatu kesimpulan. Namun, efisiensinya tergantung dari kondisi masalah yang dihadapi.

3.7 Metode Pengembangan Sistem

Proses pengembangan sebuah sistem pastinya melalui beberapa tahapan yang dilakukan untuk memudahkan merancang, mengidentifikasi masalah dan menyelesaikannya. Metode pengembangan sistem yang dipilih oleh penulis adalah metode *System Development Life Cycle* (SDLC). SDLC merupakan proses pengembangan atau mengubah suatu system perangkat lunak dengan menggunakan atau mengubah suatu system perangkat lunak dengan menggunakan model-model dan metodologi yang digunakan orang untuk mengembangkan system perangkat lunak. SDLC juga merupakan pola yang diambil untuk mengembangkan sistem perangkat lunak, yang terdiri dari tahap-tahap: rencana (*planning*), analisis (*analysis*), desain (*design*), implementasi (*implementation*), uji coba (*testing*) dan pengelolaan (*maintenance*) [33].



Gambar 3.21 Model Siklus Pengembangan Sistem [33]

1. Perencanaan (*Planning*)

Lebih menekankan pada aspek studi kelayakan pengembangan sistem (*feasibility study*). Aktivitas-aktivitas yang ada meliputi pembentukan dan konsolidasi tim pengembang, mendefinisikan tujuan dan ruang lingkup

pengembangan, mengidentifikasi apakah masalah-masalah yang ada bisa diselesaikan melalui pengembangan sistem, menentukan dan evaluasi strategi yang akan digunakan dalam pengembangan sistem, penentuan prioritas teknologi dan pemilihan aplikasi [34].

2. Analisis (*Analysis*)

Pada tahap ini, sistem akan dianalisis bagaimana akan dijalankan nantinya. Hasil analisis berupa kelebihan dan kekurangan sistem, fungsi sistem, hingga pembaharuan yang dapat diterapkan. Bagian ini termasuk dalam bagian perencanaan. Bagian lain yang termasuk dalam perencanaan ialah alokasi sumber daya, perencanaan kapasitas, penjadwalan proyek, estimasi biaya, dan penetapan. Dengan demikian, hasil dari tahap perencanaan ialah rencana proyek, jadwal, estimasi biaya, dan ketentuan. Idealnya manajer proyek dan pengembang dapat bekerja maksimal pada tahap ini [34].

3. Perancangan (*Design*)

Tahapan ini akan menghasilkan *prototype* dan beberapa *output* lain meliputi dokumen berisi desain, pola, dan komponen yang diperlukan untuk mewujudkan proyek tersebut. Setelah spesifikasi, kemudian dilakukan perancangan sistem sebagai tahapan kelanjutannya. Tahap ini disebut sebagai cetak biru, di mana sistem sudah siap untuk dikembangkan mulai dari implementasi, analisis sistem, hingga tenaga pendukung sistem yang akan dikembangkan [34].

4. Implementasi (*Implementation*)

Tahap berikutnya adalah implementasi yaitu mengimplementasikan rancangan dari tahap-tahap sebelumnya dan melakukan uji coba. Dalam implementasi, dilakukan aktivitas-aktivitas yaitu pembuatan *database* sesuai skema rancangan, pembuatan aplikasi berdasarkan desain sistem, pengujian dan perbaikan aplikasi (*debugging*) [34].

5. Pemeliharaan (*Maintenance*)

Dilakukan oleh *admin* yang ditunjuk untuk menjaga sistem tetap mampu beroperasi secara benar melalui kemampuan sistem dalam mengadaptasikan diri sesuai dengan kebutuhan [34].

3.8 Jadwal Penelitian

Penelitian dilaksanakan mulai bulan november 2020 sampai dengan mei 2022

Tabel 3.7 Jadwal Penelitian

Jenis Kegiatan	Bulan ke 1	Bulan ke 2	Bulan ke 4	Bulan ke 5
Observasi				
1. Persiapan				
2. Pelaksanaan				
Proposal Penelitian				
1. Persiapan				
2. Penyusunan Proposal				
3. Perijinan Penelitian				
Pelaksanaan Penelitian				
1. Persiapan				
2. Pelaksanaan				
3. Evaluasi				
Laporan Skripsi				
1. Persiapan				
2. Penyusunan Laporan				
3. Evaluasi				