

## **BAB III**

### **ANALISA DAN PERANCANGAN**

#### **3.1. Analisa Sistem**

Analisis sistem adalah proses meninjau sistem secara keseluruhan, dimulai dengan analisis sistem, analisis masalah, perancangan logika, dan pengambilan keputusan berdasarkan hasilnya.

##### **3.1.1. Analisa Sistem Berjalan**

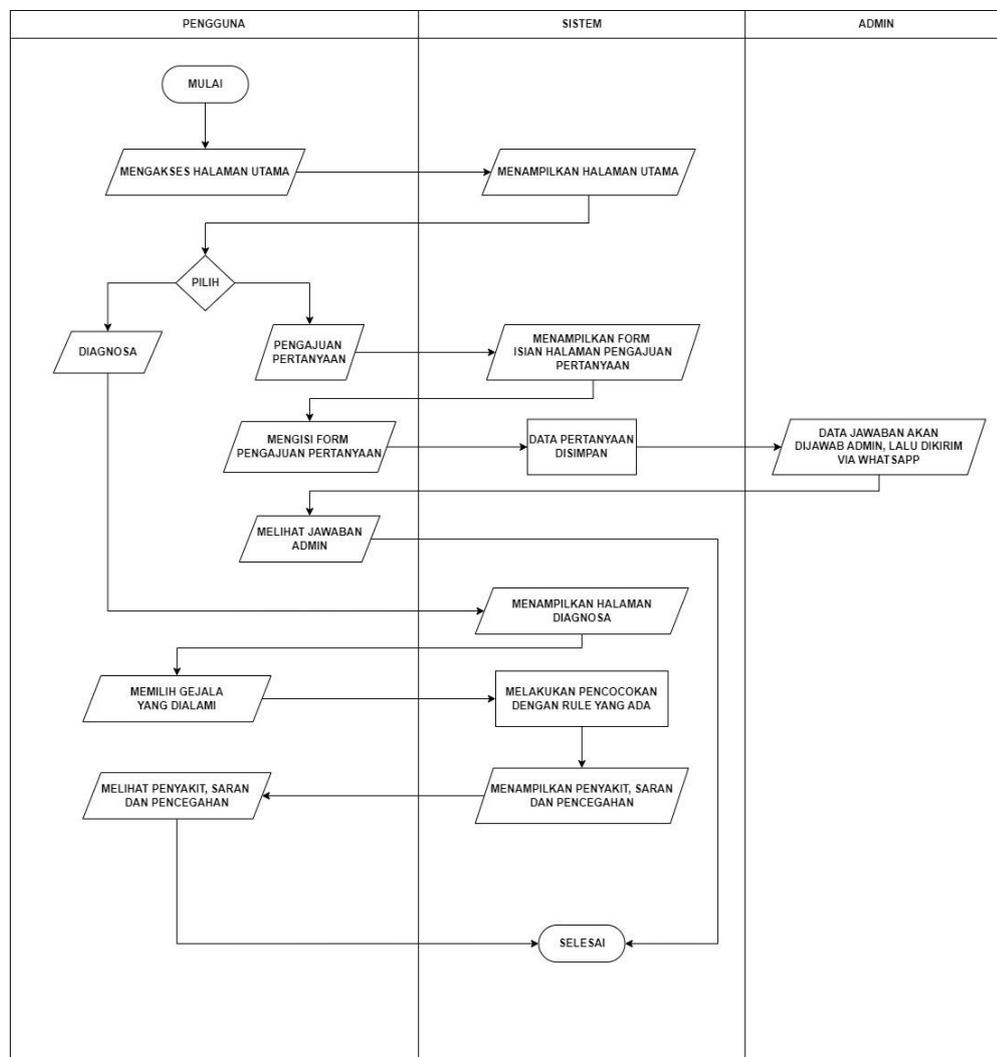
Untuk mendiagnosa penyakit pada tanaman tomat secara tradisional, biasanya dimulai dengan inspeksi visual yang rutin. Pemilik tanaman tomat akan melihat tanda-tanda yang tidak biasa, seperti bercak, perubahan warna, pembusukan, atau adanya serangga, di daun, batang, dan buah. Gejala ini dapat menunjukkan jenis penyakit atau hama yang menyerang. Selain itu, mereka sering memeriksa kondisi akar dan kelembaban tanah untuk memastikan bahwa tanaman tidak menghadapi stres air atau masalah nutrisi. Selain itu, pemilik tanaman menggunakan pengetahuan turun-temurun tentang pola serangan penyakit dan hama yang bergantung pada musim dan cuaca.

##### **3.1.2. Analisa Sistem Usulan**

Proses penting dalam pengembangan sistem informasi adalah analisis sistem usulan, yang melibatkan langkah-langkah mendalam mulai dari perancangan konsep hingga pengajuan proposal sistem. Pengguna akan pertama kali mengakses halaman utama sistem, seperti yang ditunjukkan dalam *flowchart*. Di halaman utama ini, dua opsi utama diberikan: diagnosa atau pengajuan pertanyaan. Saat pengguna memilih diagnosa, sistem akan menuntun mereka melalui serangkaian pertanyaan yang berkaitan dengan gejala tanaman tomat mereka. Pertanyaan-pertanyaan ini membantu menemukan masalah yang mungkin dihadapi tanaman mereka. Di sisi lain, pengguna dapat memilih opsi pengajuan pertanyaan jika mereka mengalami kendala atau masalah tertentu dengan tanaman mereka. Mereka akan diminta untuk mengisi formulir yang

menjelaskan masalahnya.

Sistem ini tidak hanya memberikan informasi dan bantuan yang diperlukan tetapi juga meningkatkan pengalaman pengguna dalam menangani masalah pertanian mereka. Setelah pertanyaan atau masalah dikirimkan, admin sistem akan memberikan respons melalui platform komunikasi seperti WhatsApp, yang memungkinkan interaksi yang langsung dan efisien antara pengguna dan tim admin, memungkinkan solusi yang tepat waktu dan responsif terhadap kebutuhan pengguna.

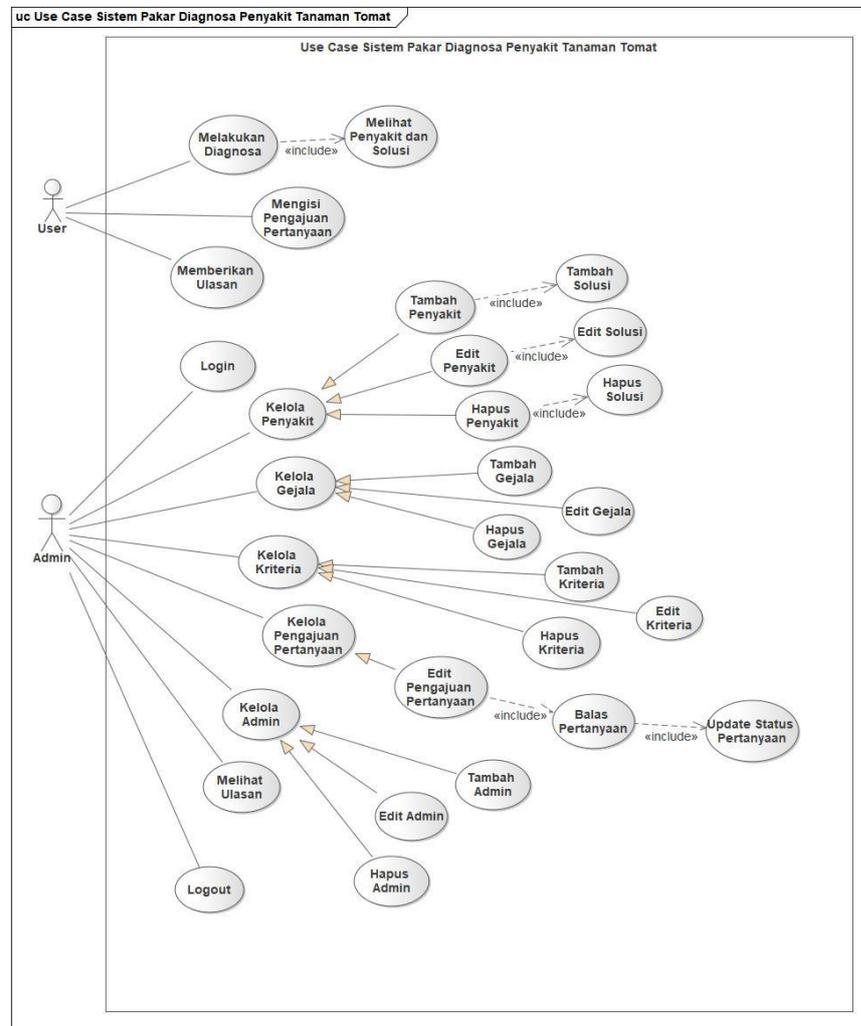


Gambar 3. 1 Analisa Sistem Usulan

### 3.2. Perancangan *Usecase Diagram*

*Use Case* adalah deskripsi mengenai fungsi yang diharapkan dari suatu

sistem, yang menggambarkan interaksi antara aktor dan sistem tersebut. Berikut adalah use case yang akan dikembangkan dalam penelitian ini:



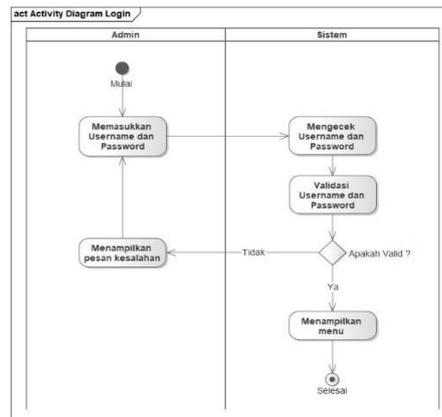
Gambar 3. 2 Usecase Diagram

### 3.3. Perancangan Activity Diagram

*Activity diagram* memungkinkan pemodelan berbagai proses yang terjadi dalam suatu sistem. Diagram ini menggambarkan alur proses secara vertikal, mirip dengan runtutan langkah-langkah yang terjadi selama sistem beroperasi.

#### 3.3.1. Activity Diagram Login

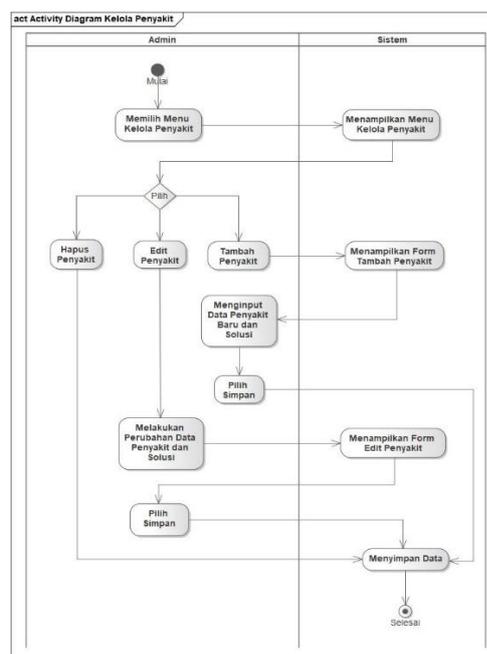
Perancangan *activity diagram login* pada sistem adalah sebagai berikut:



Gambar 3. 3 Activity Diagram Login

### 3.3.2. Activity Diagram Kelola Penyakit

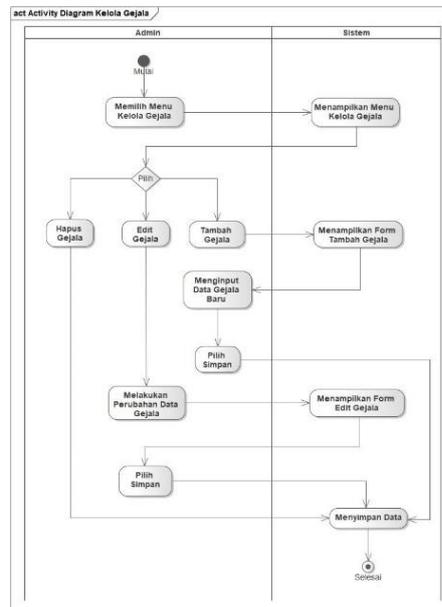
Perancangan *activity diagram* Kelola penyakit pada sistem adalah sebagai berikut:



Gambar 3. 4 Activity Diagram Kelola Penyakit

### 3.3.3. Activity Diagram Kelola Gejala

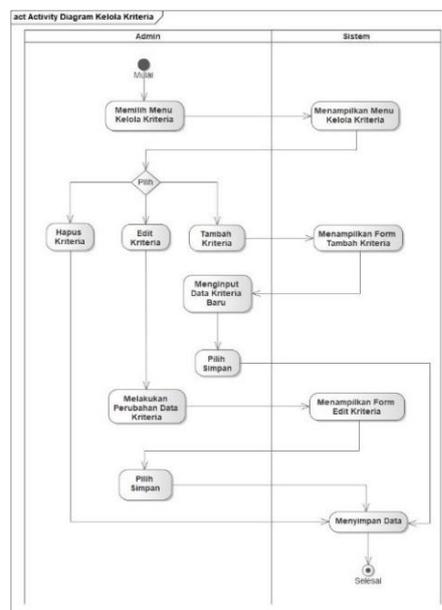
Perancangan *activity diagram* Kelola gejala pada sistem adalah sebagai berikut:



Gambar 3. 5 Activity Diagram Kelola Gejala

### 3.3.4. Activity Diagram Kelola Kriteria

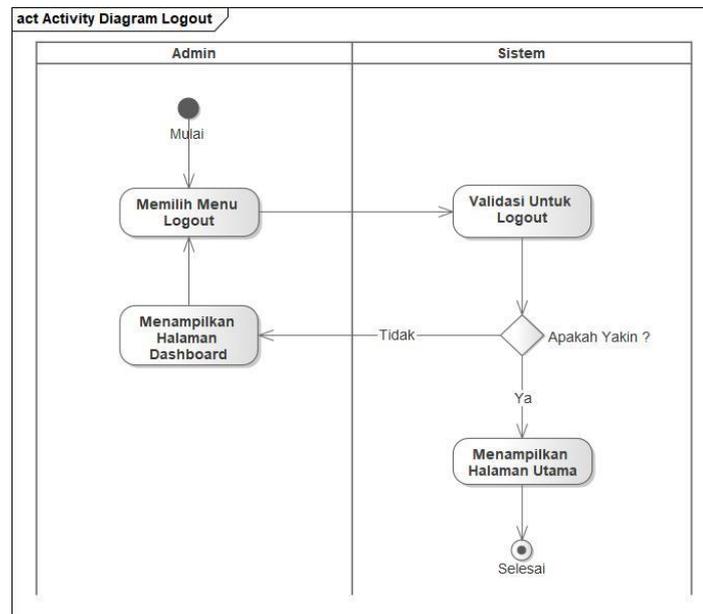
Perancangan *activity diagram* Kelola kriteria pada sistem adalah sebagai berikut:



Gambar 3. 6 Activity Diagram Kelola Kriteria

### 3.3.5. Activity Diagram Logout

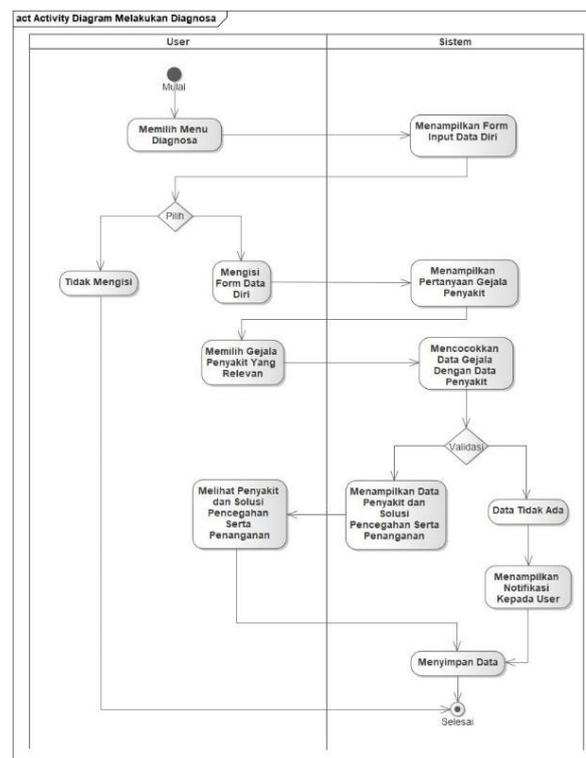
Perancangan *activity diagram logout* pada sistem adalah sebagai berikut:



Gambar 3. 7 Activity Diagram Logout

### 3.3.6. Activity Diagram Melakukan Diagnosa

Perancangan *activity diagram* melakukan diagnosa pada sistem adalah sebagai berikut:



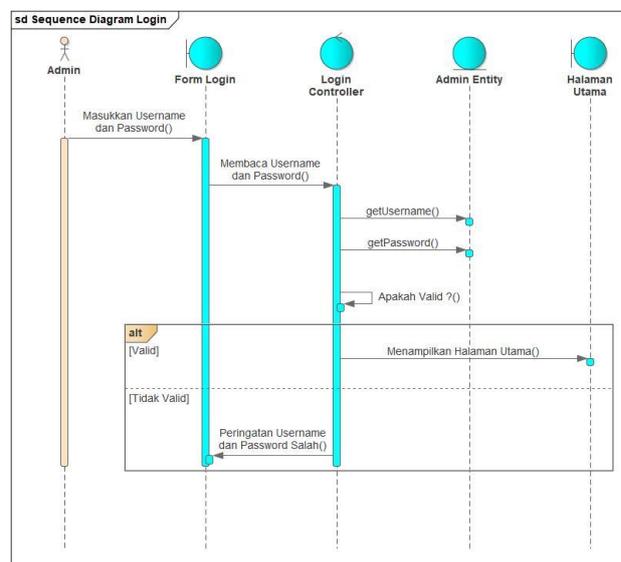
Gambar 3. 8 Activity Diagram Melakukan Diagnosa

### 3.4. Perancangan Sequence Diagram

*Sequence Diagram* menunjukkan tindakan yang diambil sebagai tanggapan terhadap sebuah peristiwa untuk menghasilkan *output* tertentu. Diagram ini juga menunjukkan perubahan internal yang terjadi dan *output* yang dihasilkan oleh peristiwa tersebut.

#### 3.4.1. Sequence Diagram Login

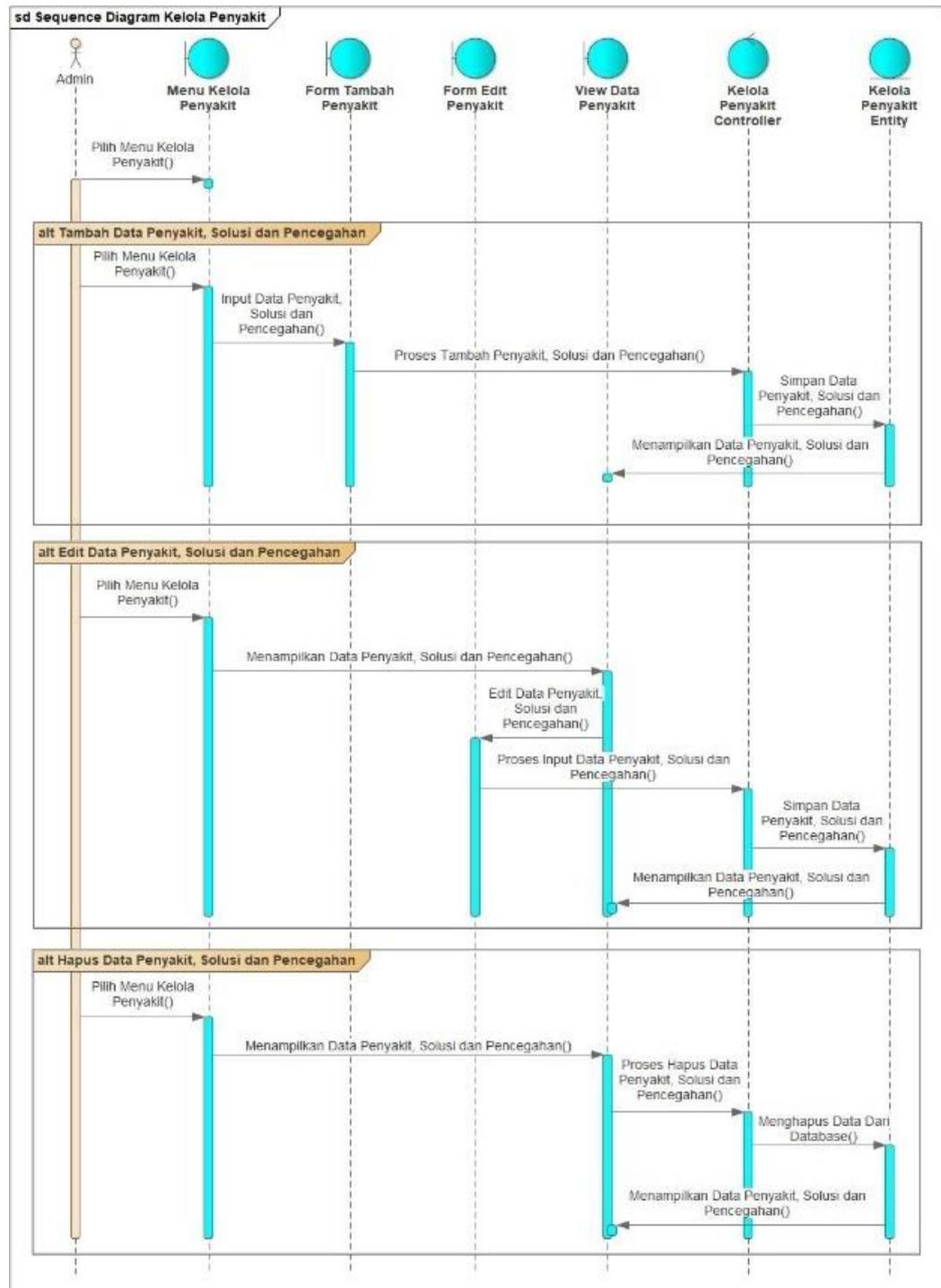
Perancangan *sequence diagram* login pada sistem adalah sebagai berikut :



Gambar 3.9 Sequence Diagram Login

#### 3.4.2. Sequence Diagram Kelola Penyakit

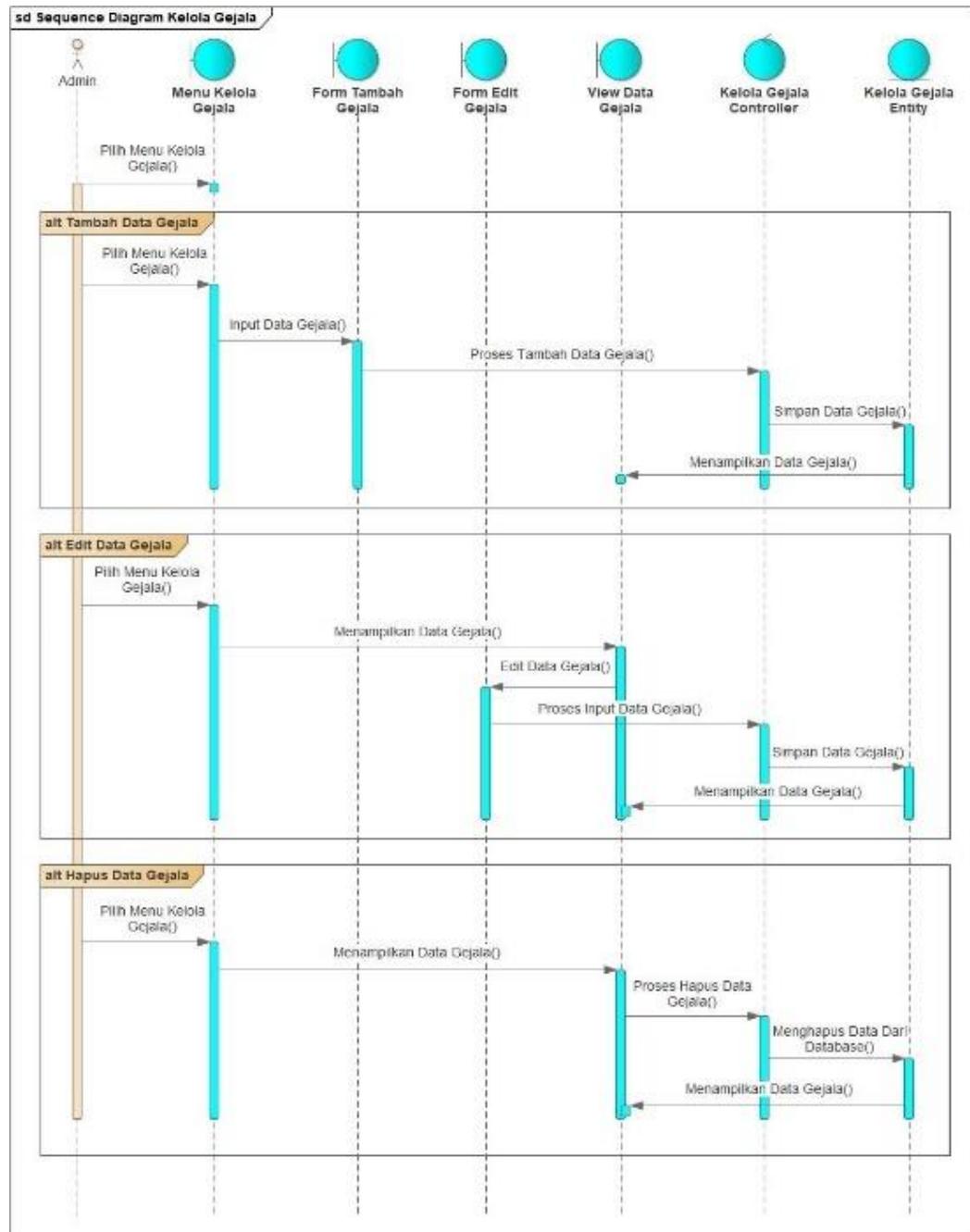
Perancangan *sequence diagram* Kelola penyakit pada sistem adalah sebagai berikut :



Gambar 3. 10 Sequence Diagram Kelola Penyakit

### 3.4.3. Sequence Diagram Kelola Gejala

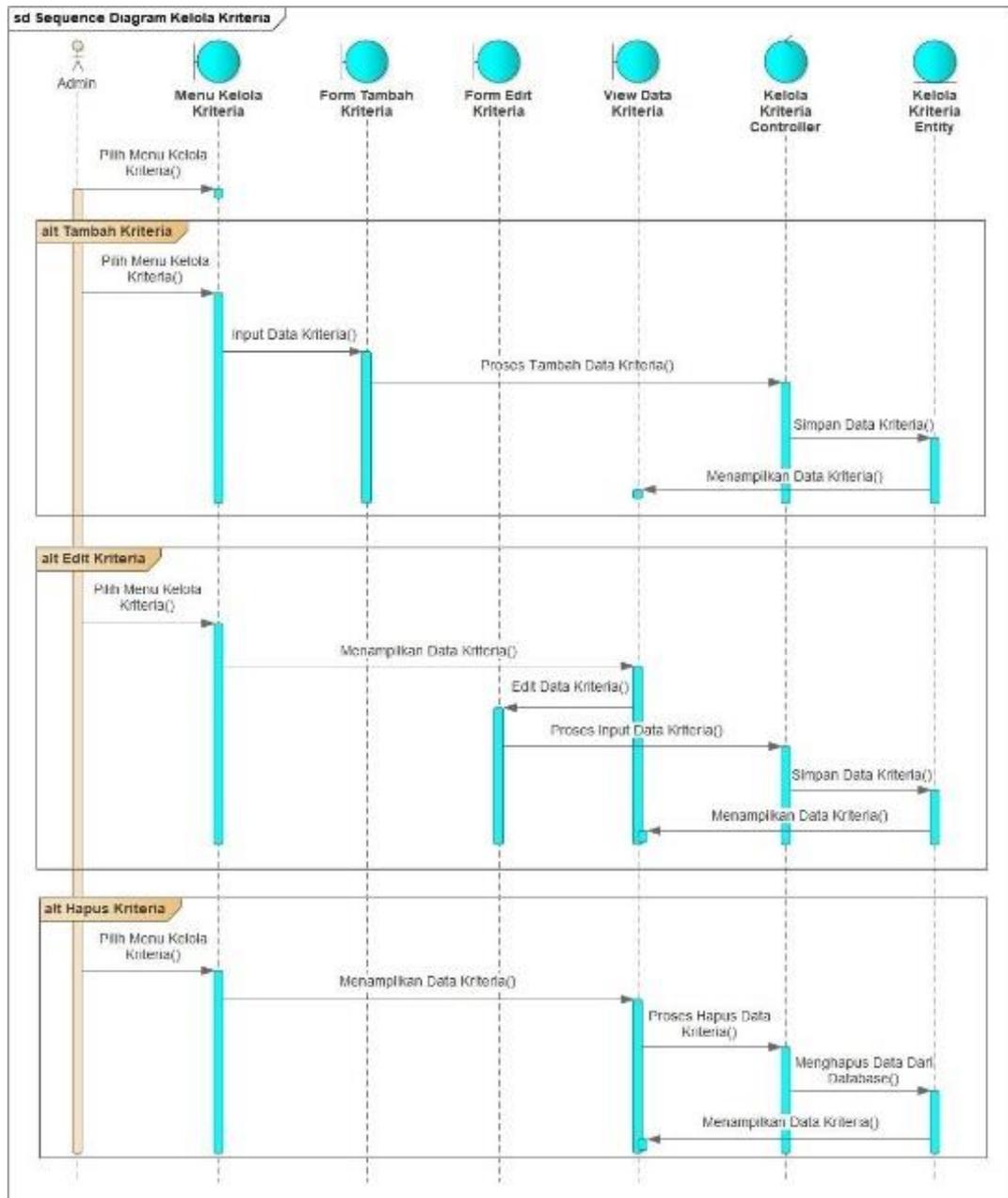
Perancangan *sequence diagram* Kelola gejala pada sistem adalah sebagai berikut :



Gambar 3. 11 Sequence Diagram Kelola Gejala

### 3.4.4. Sequence Diagram Kelola Kriteria

Perancangan *sequence diagram* Kelola kriteria pada sistem adalah sebagai berikut :

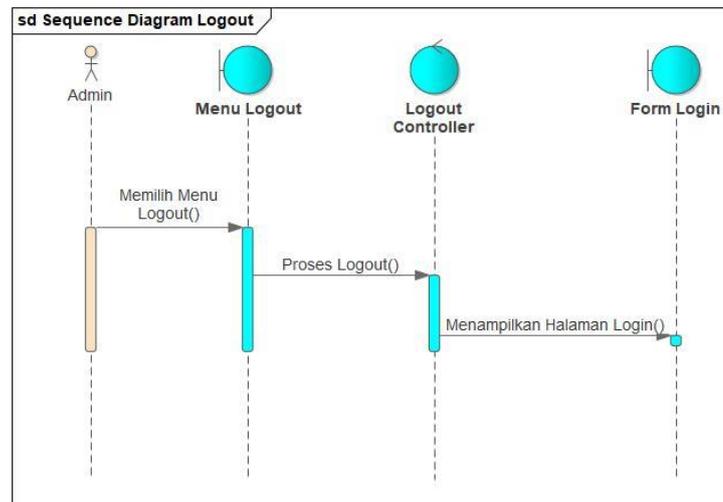


Gambar 3. 12 Sequence Diagram Kelola Kriteria

### 3.4.5. Sequence Diagram Logout

Perancangan *sequence diagram logout* pada sistem adalah sebagai berikut

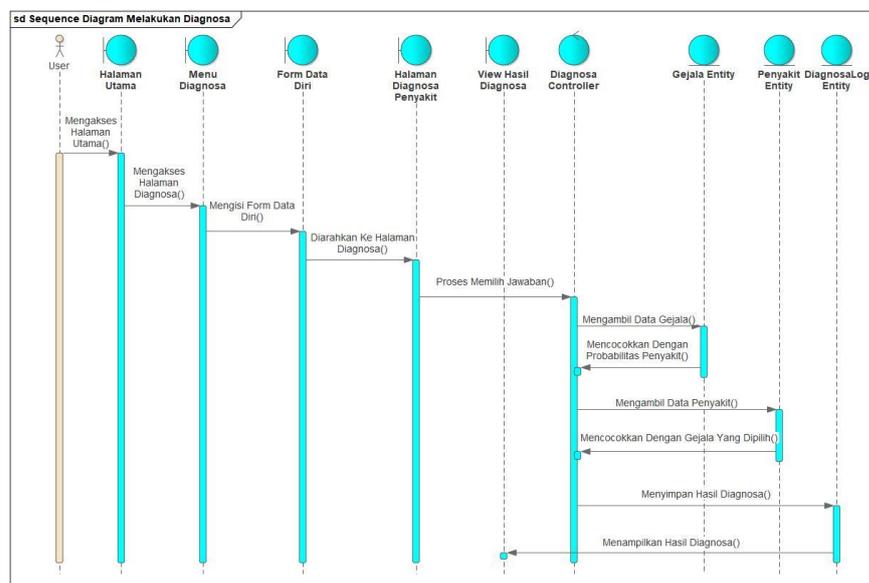
:



Gambar 3. 13 Sequence Diagram Logout

### 3.4.6. Sequence Diagram Melakukan Diagnosa

Perancangan *sequence diagram* melakukan diagnosa pada sistem adalah sebagai berikut :



Gambar 3. 14 Sequence Diagram Melakukan Diagnosa

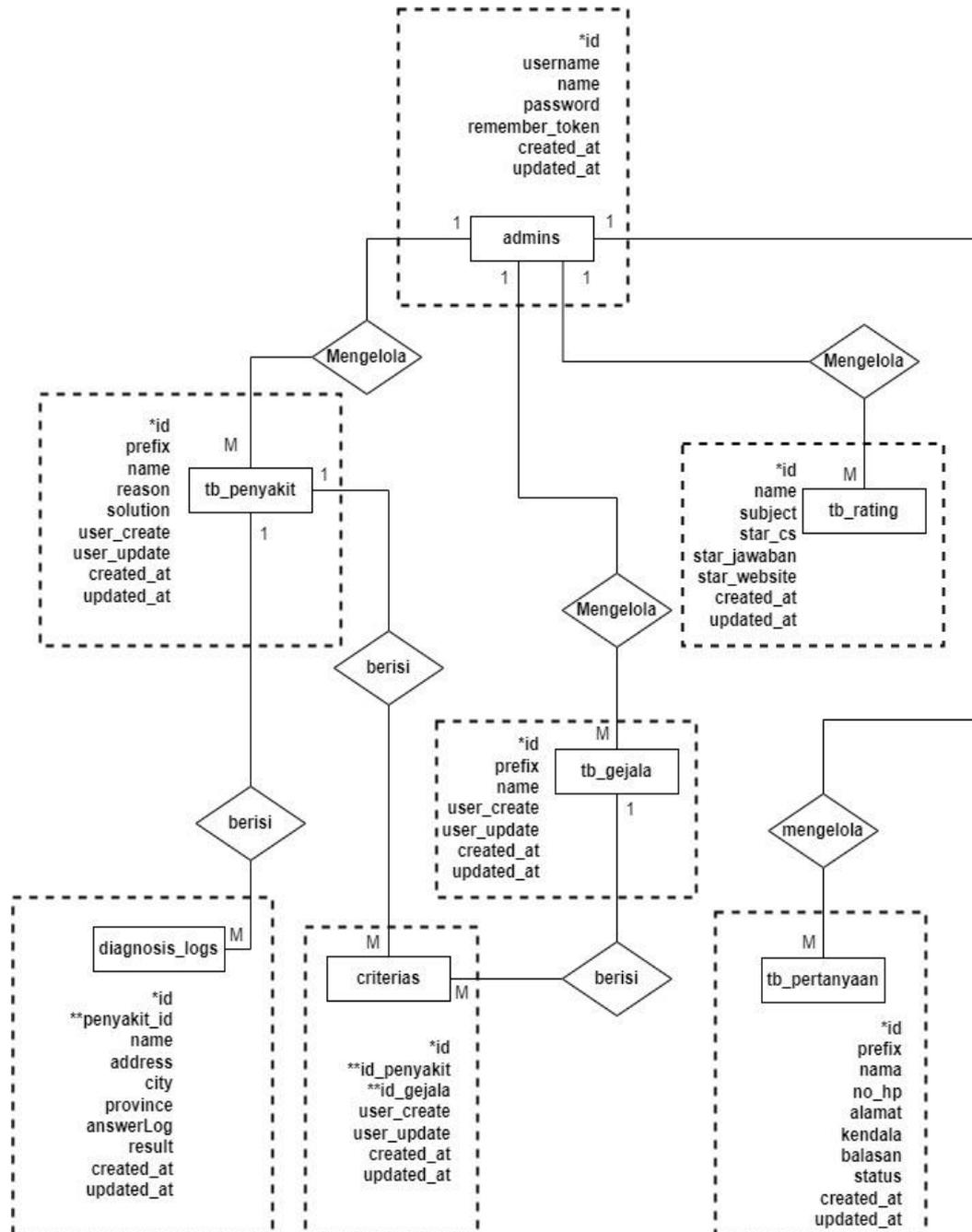
### 3.5. Perancangan Class Diagram

Perancangan *class diagram* merepresentasikan entitas-entitas yang ditangani oleh sistem, dengan mempertimbangkan karakteristik sistem aplikasi sebagai berikut:



### 3.6.2. Transformasi ERD ke LRS

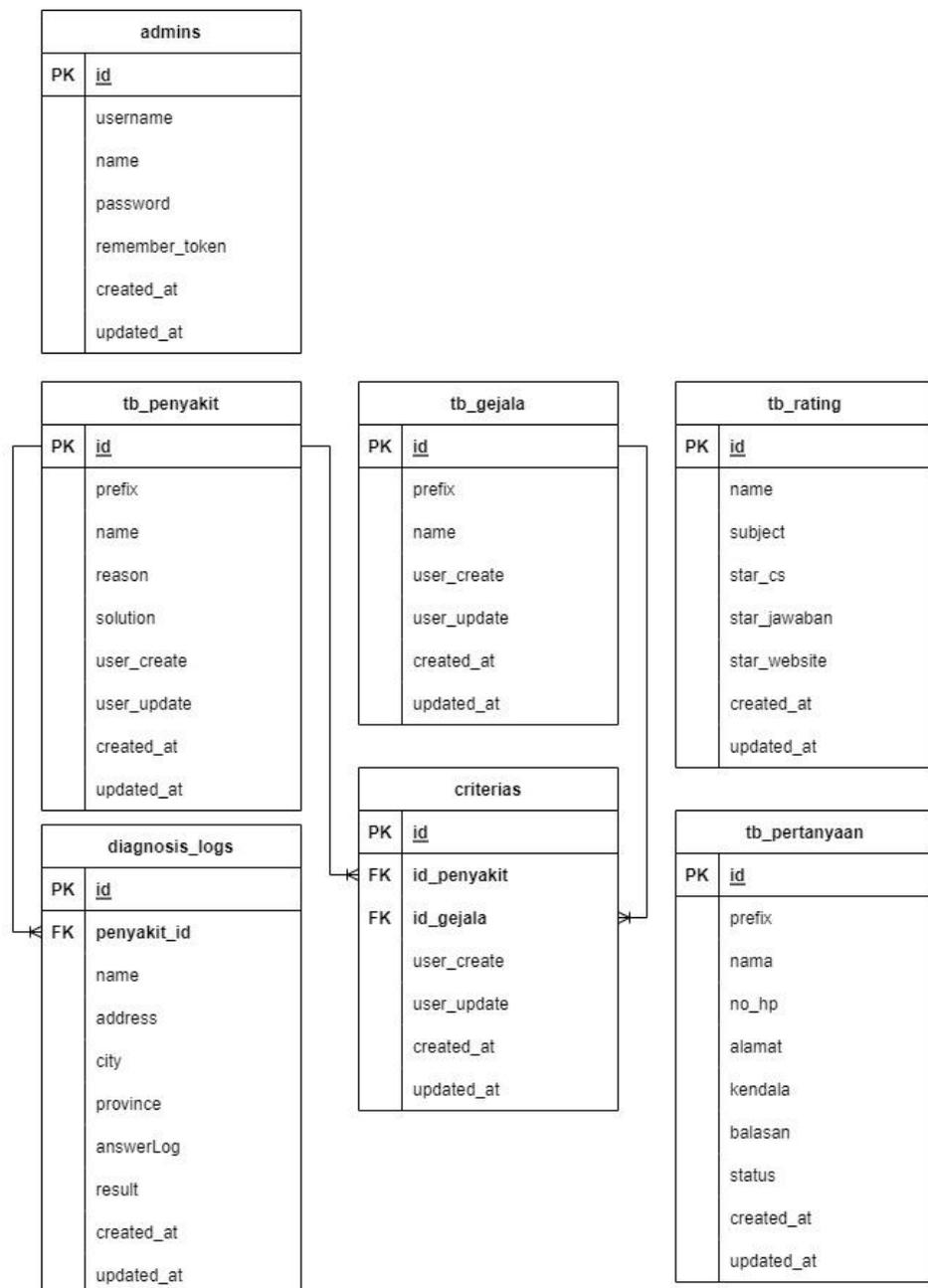
Berikut adalah transformasi dari *Entity Relationship Diagram* (ERD) ke bentuk *Logical Record Structure* (LRS):



Gambar 3. 17 Transformasi ERD Ke LRS

### 3.6.3. Logical Record Structure (LRS)

Setelah ditransformasikan ERD ke LRS, maka bentuk *Logical Record Structure* (LRS) yang sudah terbentuk adalah seperti pada gambar berikut:



**Gambar 3. 18** Logical Record Structure (LRS)

### 3.6.4. Spesifikasi Basis Data

Pada spesifikasi basis data ini tabel-tabel yang dibutuhkan untuk membuat sistem pakar diagnosa penyakit tomat antara lain:

a. Spesifikasi Tabel Admin

Nama Tabel : admins

*Primary Key* : id

**Tabel 3. 1** Spesifikasi Tabel Admins

Field	Type	Size	Null	Default	Extra	Keterangan
id	int	5	no	-	Auto increment	id admin
username	varchar	255	no	-		username admin
name	varchar	255	no	-		Nama admin
password	varchar	255	no	-		password admin
Remember_token	varchar	255	no	-		Token tersimpan
created_at	timesta mp	-	null	-	-	tanggal dan jam dibuat
updated_at	timesta mp	-	null	-	-	tanggal dan jam diubah

b. Spesifikasi Tabel Penyakit

Nama Tabel : tb\_penyakit

*Primary Key* : id

**Tabel 3. 2** Spesifikasi Tabel Penyakit

Field	Type	Size	Null	Default	Extra	Keterangan
id	smallint	5	no	-	auto increment	id
prefix	char	1	no	P	-	identitas penyakit
name	varchar	255	no	-	-	nama penyakit
reason	text	-	no	-	-	penyebab penyakit
solution	text	-	no	-	-	solusi dari penyakit
user_create	varchar	255	no	-	-	user yang membuat pertama kali

## c. Spesifikasi Tabel Gejala

Nama Tabel : tb\_gejala

Primary Key : id

**Tabel 3. 3** Spesifikasi Tabel Gejala

Field	Type	Size	Null	Default	Extra	Keterangan
id	smallint	5	no	-	auto increment	id
prefix	char	1	no	G	-	identitas penyakit
name	varchar	255	no	-	-	nama penyakit
user_create	varchar	255	no	-	-	user yang

						membuat
user_update	varchar	255	no	-	-	user yang merubah
created_at	timestamp	-	null	-	-	tanggal dan jam dibuat

d. Spesifikasi Tabel Kriteria

Nama Tabel : criterias

Primary Key : id

**Tabel 3. 4** Spesifikasi Tabel Kriteria

Field	Type	Size	Null	Default	Extra	Keterangan
id	smallint	5	no	-	auto increment	id
id_penyakit	smallint	5	no	-	-	id_penyakit
id_gejala	smallint	5	no	-	-	id_gejala
user_create	varchar	255	no	-	-	user yang membuat
user_update	varchar	255	no	-	-	user yang merubah
created_at	timestamp	-	null	-	-	tanggal dan jam dibuat
updated_at	timestamp	-	null	-	-	tanggal dan jam diubah

## e. Spesifikasi Tabel Histori Diagnosis

Nama Tabel : diagnosis\_logs

Primary Key : id

**Tabel 3. 5** Spesifikasi Tabel Histori Diagnosis

Field	Type	Size	Null	Default	Extra	Keterangan
id	smallint	5	no	-	auto increment	id
penyakit_id	smallint	5	no	-	-	berisi id penyakit
name	varchar	255	no	-	-	nama pengguna
address	varchar	-	no	-	-	alamat pengguna
city	varchar	-	no	-	-	kota pengguna
province	varchar	255	no	-	-	provinsi pengguna
answerlog	longtext	-	no	-	-	jawaban yang dipilih

## f. Spesifikasi Tabel Pertanyaan

Nama Tabel : tb\_pertanyaan

Primary Key : id

**Tabel 3. 6** Spesifikasi Tabel Pertanyaan

Field	Type	Size	Null	Default	Extra	Keterangan
id	smallint	5	no	-	auto increment	id

prefix	char	10	no	PERTANYAAN	-	identitas awal
nama	varchar	255	no	-	-	nama pengguna
no_hp	text	-	no	-	-	Nomor telepon pengguna
alamat	varchar	255	no	-	-	Alamat pengguna
kendala	varchar	255	no	-	-	Kendala yang dialami
balasan	varchar	255	null	-	-	Berisi balasan admin

g. Spesifikasi Tabel Rating

Nama Tabel : tb\_rating

Primary Key : id

**Tabel 3. 7** Spesifikasi Tabel Rating

Field	Type	Size	Null	Default	Extra	Keterangan
id	smallint	5	no	-	auto increment	id
nama	varchar	255	no	-	-	nama pemberi rating
subject	varchar	255	no	-	-	subject ulasan
star_cs	int	255	no	-	-	rating untuk customer service

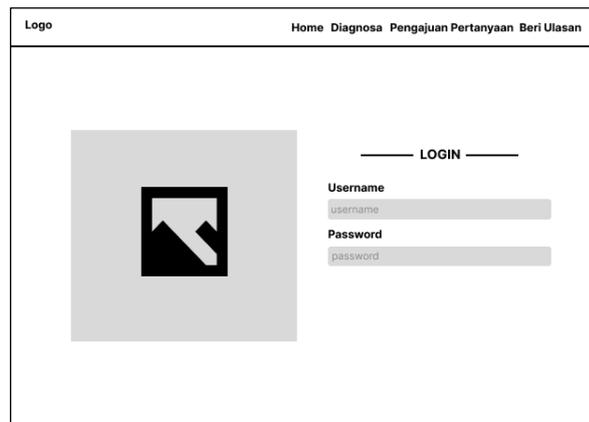
star_jawaban	int	255	no	-	-	rating untuk jawaban pertanyaan
star_website	int	255	null	-	-	rating untuk website
created_at	timestamp	-	null	-	-	tanggal dan jam dibuat

### 3.7. Perancangan *User Interface*

Tujuan UI adalah untuk memberikan gambaran tentang aplikasi yang akan dibangun sehingga implementasinya lebih mudah. Untuk pembangunan aplikasi ini, ini adalah rancangan antarmuka pengguna yang akan digunakan.

#### 3.7.1. Perancangan Halaman Login

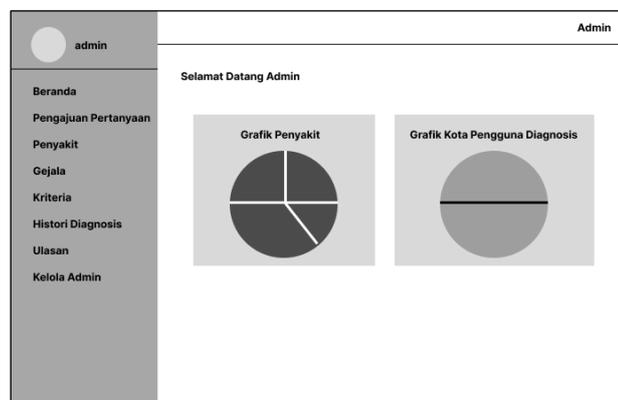
Sebuah formulir telah dibuat untuk mendesain halaman login sistem yang diberikan kepada Administrator. Tujuan dari langkah validasi ini adalah untuk memastikan bahwa Administrator adalah orang yang benar-benar memiliki identitas dengan membandingkan data yang dimasukkan dengan data yang tersimpan dalam tabel Administrator. Langkah validasi ini dimaksudkan untuk memastikan bahwa hanya Administrator yang memiliki akses ke sistem dengan data yang sesuai dan terotorisasi.



**Gambar 3. 19** Perancangan Halaman Login

### 3.7.2. Perancangan Halaman Dashboard Admin

Halaman dashboard admin ini dirancang untuk memberikan tampilan kondisi sistem yang jelas. Ada dua komponen utama di dalamnya: grafik penyakit dan grafik kota pengguna. Grafik penyakit menampilkan jenis penyakit yang sering dialami tanaman pengguna. Ini membantu manajer memahami masalah yang paling sering dihadapi pengguna sistem. Untuk saat ini, lokasi geografis pengguna yang menggunakan sistem pakar ini dapat dilihat melalui peta kota pengguna. Dengan kedua grafik ini, admin dapat membuat keputusan yang lebih terinformasi dan strategis tentang manajemen dan peningkatan sistem agar lebih responsif terhadap kebutuhan lokal pengguna. Ini penting karena dapat membantu dalam menyesuaikan atau menargetkan informasi atau layanan sesuai dengan kebutuhan lokal pengguna.



**Gambar 3. 20** Perancangan Halaman Dashboard Admin

### 3.7.3. Perancangan Halaman Penyakit

Tujuan dari desain halaman ini adalah untuk memberi admin kemampuan untuk mengakses informasi lengkap tentang penyakit yang sering menyerang tanaman tomat. Halaman ini akan mencakup daftar penyakit bersama dengan penyebabnya yang paling umum dialami oleh tanaman tomat, solusi atau metode yang disarankan untuk mencegahnya, dan kode identifikasi penyakit yang digunakan dalam sistem.

NO	KODE PENYAKIT	NAMA PENYAKIT	PENYEBAB	SOLUSI	TANGGAL	AKSI
1	P1	aaa	aaa	aaa	dd/mm/yyyy	AKSI
2	P2	bbb	bbb	bbb	dd/mm/yyyy	AKSI

Gambar 3. 21 Perancangan Halaman Penyakit

### 3.7.4. Perancangan Halaman Gejala

Admin dapat melihat informasi tentang gejala dan kode gejala yang ada di halaman ini. Halaman ini memiliki akses ke daftar gejala tanaman yang paling umum dan kode identifikasi yang digunakan dalam sistem. Dengan informasi ini, administrator dapat lebih memahami kondisi pengguna dan mengelola data gejala dengan lebih baik.

NO	KODE GEJALA	NAMA GEJALA	TANGGAL	AKSI
1	G1	aaa	dd/mm/yyyy	AKSI
2	G2	bbb	dd/mm/yyyy	AKSI
3	G3	ccc	dd/mm/yyyy	AKSI
4	G4	ddd	dd/mm/yyyy	AKSI

Gambar 3. 22 Perancangan Halaman Gejala

### 3.7.5. Perancangan Halaman Kriteria

Perancangan halaman ini bertujuan untuk memungkinkan admin memeriksa kriteria yang terkait dengan gejala dan penyakit tomat.

NO	PENYAKIT	GEJALA	TANGGAL	AKSI
1	aaa	aaa	dd/mm/yyyy	AKSI
2	bbb	bbb	dd/mm/yyyy	AKSI
3	ccc	ccc	dd/mm/yyyy	AKSI
4	ddd	ddd	dd/mm/yyyy	AKSI

Gambar 3. 23 Perancangan Halaman Kriteria

### 3.7.6. Perancangan Halaman Rule Kriteria

Dengan melihat kondisi "*if-then*" yang ada di website, admin dapat mengakses aturan logika *forward chaining*. Ini akan membantu mereka memahami proses penalaran sistem secara lebih baik.

Daftar Kriteria/Rules Penyakit Tomat

RULE 01 IF G1 IF G2 IF G3 THEN P1

RULE 02 IF G1 IF G2 THEN P2

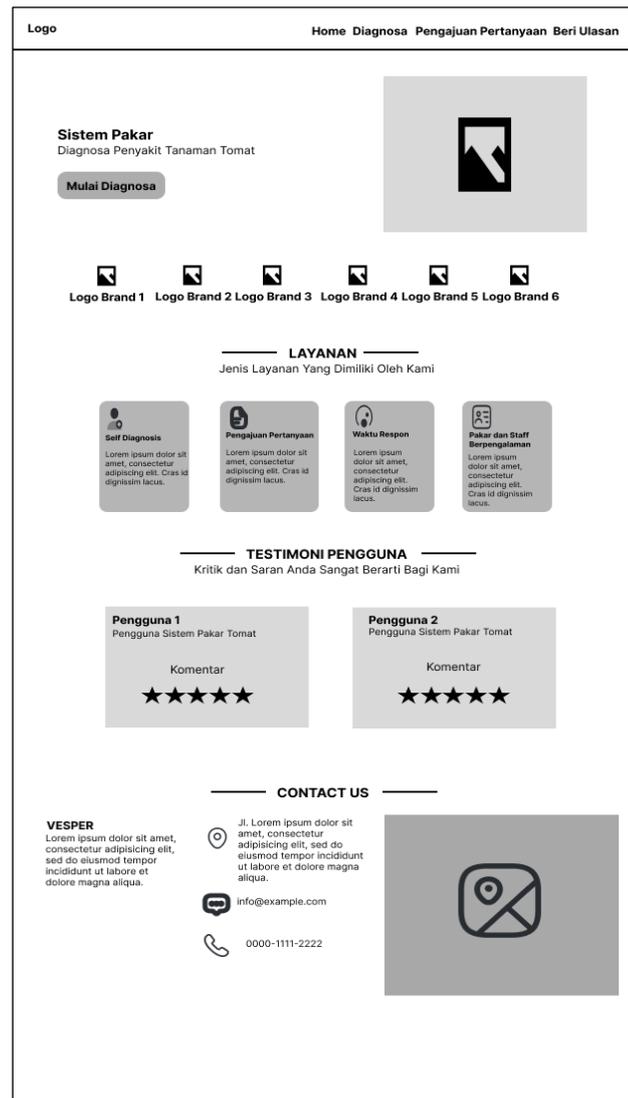
RULE 03 IF G1 THEN P3

Gambar 3. 24 Perancangan Halaman Rule Kriteria

### 3.7.7. Perancangan Halaman Utama

*Layout* halaman ini dirancang untuk menjadi interaktif dan mudah dipahami oleh pengguna. Pengguna dapat mengakses berbagai fitur pada

halaman ini dengan menu navigasi. Selain itu, histori penilaian juga tersedia, yang memungkinkan pengguna melihat ulasan sebelumnya tentang layanan atau hasil sistem. Informasi lokasi juga disajikan untuk memberikan konteks geografis yang relevan. Menu utama halaman ini memungkinkan pengguna melakukan diagnosa langsung dan menemukan solusi masalah.



Gambar 3. 25 Perancangan Halaman Utama

### 3.7.8. Perancangan Halaman Melakukan Diagnosa

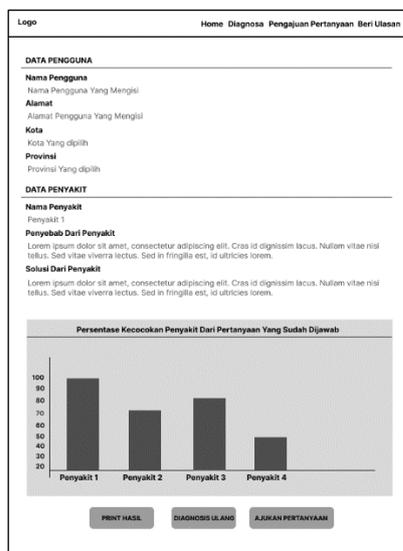
Halaman diagnosa ini dirancang untuk memungkinkan pengguna memilih jawaban "ya" atau "tidak" berdasarkan karakteristik tanaman yang diamati. Halaman ini akan menyajikan serangkaian pertanyaan dengan jawaban

yang sesuai, yang membantu sistem menemukan gejala yang mungkin dialami tanaman. Dengan memilih jawaban yang tepat, pengguna dapat memberikan informasi yang diperlukan untuk proses diagnosa yang lebih akurat dan relevan.

Gambar 3. 26 Perancangan Halaman Melakukan Diagnosa

### 3.7.9. Perancangan Halaman Hasil Diagnosa

Halaman hasil diagnosa dirancang untuk menampilkan hasil dari pertanyaan dan jawaban yang telah dipilih pengguna. Halaman ini akan menyajikan informasi tentang kondisi tanaman berdasarkan gejala yang telah diidentifikasi selama proses diagnosa. Hasil ini akan membantu pengguna memahami masalah yang mungkin dihadapi tanaman mereka dan memberikan saran untuk solusi atau langkah-langkah yang dapat diambil untuk memperbaiki kondisi tanaman mereka.



Gambar 3. 27 Perancangan Halaman Hasil Diagnosa