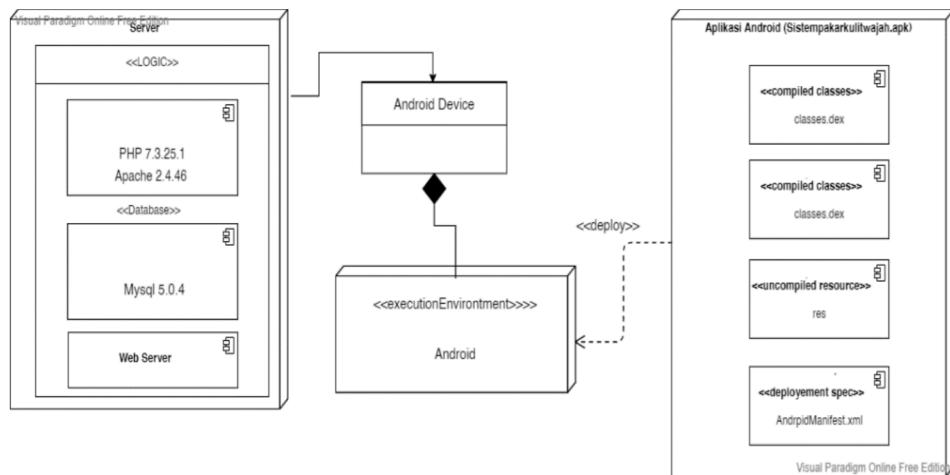


BAB IV

HASIL DAN PEMBAHASAN

4.1 Deployment Diagram

Dalam Pada lingkungan percobaan ini akan dijabarkan semua elemen perangkat yang digunakan dalam pembuatan program memlalui *deployment diagram*. Adapun perangkat lunak dan perangkat keras yang digunakan antara lain adalah sebagai berikut.



Gambar 4.1 Deployment Diagram Sistem Pakar Kulit Wajah

Dari *deployment diagram* gambar 4.1 memperlihatkan bahwa pembuatan *logic* dari aplikasi yang akan dibuat menggunakan *apache 2.4.46* dan *php 7.3*. Sedangkan untuk *database* menggunakan *MySQL 5.0.4*. *PHP* disini juga berfungsi menghubungkan antara *database MySQL* dengan *device android* dengan spesifikasi *smartphone android 11.0*.

Pada pengembangan ini, terdapat beberapa komponen pembangun aplikasi android. Diantaranya adalah *classes*, yaitu komponen utama yang terdiri dari kode-kode program, *classes* ini merupakan file *java*. *Resources* berupa file-file pendukung seperti gambar,suara, dan yang lainnya. Sedangkan *res* berisi *file-file* untuk membuat suatu tampilan *android* dan *Android Manifest* yaitu untuk mendaftarkan semua *file-file classes (java)*.

4.2 Implementasi Metode

Implementasi metode yang digunakan dalam penelitian ini yaitu :

4.2.1. Basis Pengetahuan

Keberhasilan suatu sistem pakar terletak pada pengetahuan dan bagaimana mengolah pengetahuan tersebut agar dapat ditarik suatu kesimpulan. Pengetahuan yang diperoleh dari hasil wawancara pakar, studi pustaka dan analisa lewat buku dikonversi kedalam sebuah tabel penyakit dan gejala guna mempermudah proses pencarian solusi. Tabel jenis penyakit dan gejala ini digunakan sebagai pola pencocokan informasi yang dimasukan oleh pengguna (*user*) dan aturan.

4.2.2. Daftar Penyakit

Daftar penyakit kulit wajah dapat dilihat pada tabel 4.1

Tabel 4.1 Penyakit Kulit Wajah

Kode	Nama Penyakit
P01	<i>Acne Vulgaris</i>
P02	<i>Melasma</i>
P03	<i>Nevus Melanosistik</i>
P04	<i>Syringoma</i>
P05	<i>Milia</i>
P06	<i>Rosacea</i>
P07	<i>Xanthelasma</i>
P08	<i>Keratosis Seboroik</i>
P09	<i>Basalioma</i>
P10	<i>Melanoma</i>

4.2.3. Daftar Gejala

Daftar Gejala pada penyakit kulit wajah dapat dilihat pada tabel 4.2.

Tabel 4.2 Gejala-Gejala Penyakit Kulit Wajah

Kode	Gejala
G01	Muncul bintik- bitnik atau bercak
G02	Terasa nyeri

G03	Bintik atau bercak berwarna gelap
G04	Bintik sewarna dengan warna kulit
G05	Bentuknya semakin membesar saat berkeringat
G06	Muncul dikedua sisi wajah
G07	Muncul dibagian wajah sekitar mata
G08	Muncul pori-pori terbuka seperti komedo putih
G09	Muncul pori-pori tertutup seperti komedo hitam
G10	Muncul papul (bintik merah kecil)
G11	Mengalama Peradangan pada area tersebut
G12	Mengalami Hiperpigmentasi atau perbedaan warna kulit dari sebelumnya.
G13	Sebelumnya pernah memakai obat-obatan kontrasepsi
G14	Bintik atau bercak berada dibagian pipi kanan dan kiri
G15	Muncul Perdangan keringpada area tersebut
G16	Bentuk Bintik ataupun bercak semakin membesar
G17	Berbentuk Bulat dan tegas
G18	Muncul benjolan berbentuk bitnik putih
G19	Mudah iritasi saat bergesekan
G20	Bitnik atau bercak berwarna kemerahan
G21	Pelebaran pembuluh darah dibagin wajah
G22	Kulit wajah terasa kering dan kasar
G23	Kulir wajah berminyak dan sensitif
G24	Hangat ketika diraba
G25	Bintik atau bercak berwarna kekuningan
G26	Mempunyai riwayat kesterol yang tinggi
G27	Terdapat plak warna kuning bagian bawah alis mata
G28	Bintik atau bercak berwarna coklat

G29	Bintik atau bercak berwarna coklat kehitaman
G30	Mudah berdarah
G31	Lesi atau luka yang tidak kunjung sembuh
G32	Membesar dan meluas seiring waktu
G33	Batas pada bagian tersebut tidak tegas dan permukaan tidak merata
G34	Bentuk tepi tidak teratur

4.2.4. Rekomendasi Solusi

Rekomendasi merupakan solusi yang diberikan kepada sistem atas hasil yang telah dilakukan. Solusi bersumber dari pakar dan beberapa studi literatur lainnya. Solusi penyakit kulit wajah dapat dilihat pada Tabel 4.3.

Tabel 4.3 Rekomendasi Solusi

Penyakit	Solusi
1. <i>Acne Vulgaris</i>	Solusi penangan awal adalah Jangan menyingkirkan jerawat dengan memencetnya, karena itu akan menyebarkan bakteri dan menimbulkan jaringan parut. Cara terbaik untuk mengatasinya adalah menjaga area di sekitarnya bebas dari bakteri dan sel kulit mati kurangi aktivitas begadang juga mengurangi konsumsi makanan <i>junk food</i> .
2. <i>Melasma</i>	Solusi pertama penyakit melasma dapat diobati dengan berbagai cara, mengurangi ataupun berhenti pemakaian alat kontrasepsi serta pil KB. Namun jika bercak ini dialami selama bertahun-tahun harus segera melakukan pengobatan dan tindakan medis.

3. <i>Nevus Melanositik</i>	Solusi pertama dari penyakit ini jika ingin diangkat maka perlu melakukan terapi. Tindakan terapi sendiri adalah terapi eksisi sederhana <i>nevus</i> .
4. <i>Syringoma</i>	Solusi penangan dari penyakit ini diantaranya adalah dengan menggunakan obat-obatan dari asam trikloroasetat, operasi laser dengan gelombang cahaya merah, <i>cryotherapy</i> , <i>demabrasi</i> dan bedah eksisi.
5. <i>Milia</i>	Solusi pertama dari penyakit ini adalah dengan rutin membersihkan wajah, mengurangi pemakaian <i>make up</i> berbahan berat, konsumsi secara teratur buah-buahan dan sayur-sayuran.
6. <i>Rosacea</i>	Solusi pertama dari penyakit ini adalah dengan memberisihkan kulit wajah dengan hati-hati secara rutin, menggunakan tabir surya sebelum beraktivitas diluar rumah. Mengkompres bagian penyakit tersebut dengan air dingin serta konsumsi makanan sehat seperti buah-buahan dan sayur-sayuran.
7. <i>Xanthelasma</i>	Solusi pertama dari penyakit ini adalah dengan melakukan tindakan medis berupa eksisi bedah, Kauterisasi kimia, <i>cryotherapy</i> , dan ablasi laser.
8. <i>Keratosis Seboroik</i>	Solusi untuk penyakit ini adalah dengan melakukan tindakan medis diantaranya <i>cryosurgery</i> , bedah listrik, kuretase dan ablasi

9. <i>Basalioma</i>	Solusi untuk penyakit ini adalah dengan melakukan tindakan medis berupa elektrodikasi dan kuretase, pemotongan dengan pisah bedah, krioterapi, dan operasi <i>mohs</i> .
10. <i>Melanoma</i>	Solusi penyakit ini adalah dengan operasi pengangkatan sel-sel melanoma. Tingkatan operasi juga tergantung dengan tingkat stadiumnya

4.2.5. Kaidah Produksi (*Production Rules*)

Aturan atau kaidah produksi yang digunakan untuk melakukan penalaran atau penelusuran basis pengetahuan awal sehingga menghasilkan *knowledge* baru untuk mencapai tujuan disebut dengan *production rules* (Kaidah Produksi). Secara default sistem ini memiliki 10 aturan diagnosa. Daftar aturan atau rules dalam sistem ini ditunjukkan pada tabel 4.4.

Tabel 4.4 Kaidah Produksi

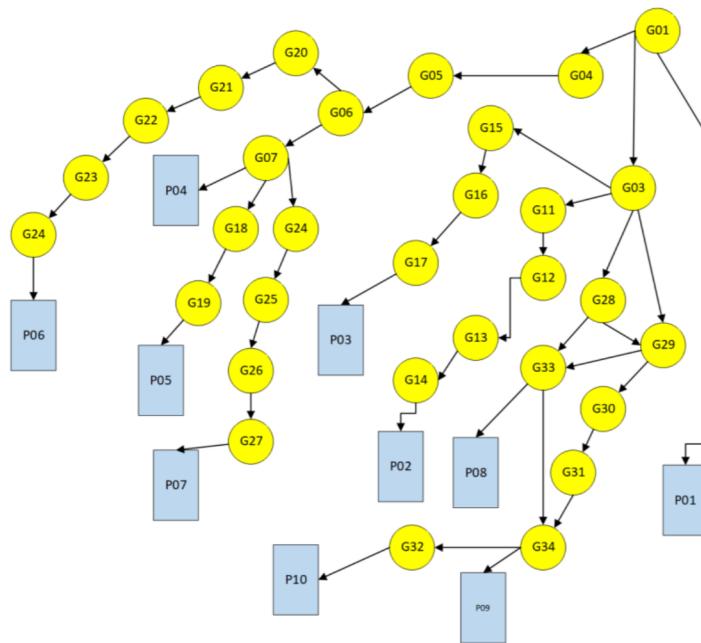
No	Rule
Kaidah 1	JIKA Muncul bitnik-bintik atau bercak(G1) DAN Terasa nyeri (G2) DAN Muncul pori-pori terbuka seperti komedo putih (G08) DAN Muncul pori-pori tertutup seperti komedo hitam(G09) DAN Muncul papul (bintik merah kecil)(G10) MAKA Penyakit <i>Acne Vulgaris</i> (P01)
Kaidah 2	JIKA Muncul bitnik-bintik atau bercak (G1) DAN Bintik atau bercak berwarna gelap (G03) DAN Mengalami Peradangan pada area tersebut (G11) DAN Mengalami hiperpugmentasi atau perbedaan warna kulit dari sebelumnya (G12) DAN Sebelumnya pernah memakai obat-obatan kontrasepsi (G13) DAN Bintik atau bercak berada dibagian pipi kanan dan kiri (G14)

	MAKA Penyakit <i>Melasma</i> (P02)
Kaidah 3	<p>JIKA Muncul bitnik-bintik atau bercak (G1) DAN Bintik atau bercak berwarna gelap (G03) DAN Mengalami peradangan kering pada area tersebut (G15) DAN Bentuk bintik ataupun bercak semakin membesar (G16) DAN Berbentuk bulat dan tegas (G17) MAKA Penyakit <i>Nevus Melanositik</i> (P03)</p>
Kaidah 4	<p>JIKA Muncul bitnik-bintik atau bercak (G1) DAN Bintik sewarna dengan warna kulit (G04) DAN Bentuknya semakin membesar saat berkeringat (G05) DAN Muncul dikedua sisi wajah (G06) DAN Muncul dibagian wajah sekitar mata (G07) MAKA Penyakit <i>Syringoma</i> (P04)</p>
Kaidah 5	<p>JIKA Muncul bitnik-bintik atau bercak (G1) DAN Bintik sewarna dengan warna kulit (G04) DAN Muncul dikedua sisi wajah (G06) DAN muncul dibagian wajah sekitar mata (G07) DAN Muncul benjolan berbentuk bitnik putih (G18) DAN Mudah iritasi saat bergesekan (G19) MAKA Penyakit <i>Milia</i> (P05)</p>
Kaidah 6	<p>JIKA Muncul bitnik-bintik atau bercak (G1) DAN Muncul dikedua sisi wajah (G06) DAN Bintik atau bercak berwarna kemerahan (G20) DAN Pelebaran pembuluh darah dibagian wajah (G21) DAN Kulit wajah terasa kering dan kasar (G22) DAN Kulit wajah berminyak dan sensitif (G23) DAN Hangat Ketika diraba(G24) DAN MAKA Penyakit <i>Rosacea</i> (P06)</p>
Kaidah 7	<p>JIKA Muncul bitnik-bintik atau bercak (G1) DAN muncul dibagian wajah sekitar mata (G07) DAN Hangat ketika diraba(G24) DAN Bintik atau bercak berwarna kekuningan (G25)</p>

	DAN Mempunyai riwayat kolesterol yang tinggi (G26) DAN Terdapat plak warna kuning bagian bawah alis mata (G27) MAKA Penyakit <i>Xanthelasma</i> (P07)
Kaidah 8	JIKA Muncul bitnik-bintik atau bercak (G1) DAN Bintik atau bercak berwarna gelap (G03) DAN Bintik atau bercak berwarna coklat (G28) DAN Batas pada bagian tersebut tidak tegas dan permukaan tidak merata (G33) Penyakit <i>Keratosis Seboroik</i> (P08)
Kaidah 9	JIKA Muncul bitnik-bintik atau bercak (G1) DAN Bintik atau bercak berwarna gelap (G03) DAN Bintik atau bercak berwarna coklat kehitaman (G29) DAN Mudah berdarah (G30) DAN Lesi atau luka yang tidak kunjung sembuh (G31) DAN Bentuk tepi tidak teratur (G34) MAKA Penyakit <i>Basalioma</i> (P09)
Kaidah 10	JIKA Muncul bitnik-bintik atau bercak (G1) DAN Bintik atau bercak berwarna gelap (G03) DAN Bintik atau bercak berwarna coklat (G28) DAN Bintik atau bercak berwarna coklat kehitaman (G29) DAN Membesar dan meluas seiring waktu (G32) DAN Batas pada bagian tersebut tidak tegas dan permukaan tidak merata (G33) DAN Bentuk tepi tidak teratur (G34) Penyakit <i>Melanoma</i> (P10)

4.2.6. Pohon Keputusan

Dengan ditemukannya gejala-gejala penyakit pada kulit wajah serta metode inferensi yang digunakan yaitu *Forward Chaining*, lalu gejala yang timbul atau tampak maka akan mempermudah dalam pembuatan *Decision Tree* atau pohon keputusan tentang penentuan penyakit. Gambar dibawah ini merupakan gambar *Decision Tree* untuk menentukan penyakit berdasarkan gejala-gejala yang muncul.



Gambar 4.2 *Decision Tree (Pohon Keputusan)*

4.2.7. Analisis Forward Chaining

Contoh kasus adalah seorang *user* A ingin melakukan diagnosis tentang penyakit kulit dan wajah dengan menjawab pertanyaan seputar gejala yang dialaminya lalu diberikan oleh sistem dengan jawaban “Ya” atau “Tidak”. Berikut adalah contoh penelusuran tiap-tiap gejala dengan metode *Forward Chaining*:

- 1) Pertanyaan pertama akan muncul, apakah muncul bintik-bintik ? Jika *user* A menjawab “Ya”, maka pertanyaan selanjutnya
- 2) Apakah Terasa nyeri ? Jika *user* menjawab “Tidak”, maka pertanyaan selanjutnya,
- 3) Apakah Bintik berwarna gelap ? Jika *user* menjawab “Tidak”, maka pertanyaan selanjutnya,
- 4) Apakah Bintik Sewarna dengan warna kulit? Jika *user* menjawab “Ya”, maka pertanyaan selanjutnya,
- 5) Apakah Bentuknya semakin membesar saat berkeringat? Jika *user* menjawab “Ya”, maka pertanyaan selanjutnya
- 6) Apakah Muncul di kedua sisi wajah? Jika *user* menjawab “Ya”, maka pertanyaan selanjutnya
- 7) Apakah muncul dibagian mata? Jika *user* menjawab “Ya”, maka aplikasi akan menampilkan hasil diagnosis cepat bahwa *user* A mengalami penyakit *Syringoma*.

4.2.8. Inferensi

Pada aplikasi ini, untuk melakukan diagnosa menggunakan metode inferensi, yaitu runut maju (*forward chaining*). Data yang digunakan dalam inferensi diperoleh dari jawaban yang diberikan oleh pengguna. Aplikasi menampilkan seluruh data gejala, sehingga nantinya jika pengguna mengegunakan menu diagnosa cepat maka akan diarahkan untuk memilih pilihan gejala berdasarkan dirasakan “Ya” atau “Tidak” gejala sesuai dengan gejala yang di alami. Sedangkan jika memilih menu diagnosa akurat maka pengguna akan diarahkan untuk memilih pilihan gejala berdasarkan tingkat keyakinan pengguna akan gejala tersebut yang dimana terdapat opsi pilihan berupa “Tidak”, “Tidak Yakin”, “Sedikit Yakin”, “Cukup Yakin”, “Yakin”, “Sangat Yakin”. Tabel yang digunakan dalam proses inferensi tersebut adalah tabel aturan.

4.2.9. Nilai Tingkat Keyakinan CF (*Rule*)

Berikut adalah nilai tingkat keyakinan CF (*Rule*) yang didapat dari interpretasi *term* dari hasil wawancara dengan pakar yang diubah menjadi nilai CF tertentu sesuai dengan tabel berikut.

Tabel 4.5 Interpretasi (*term*) Certainty Factor

Certainty Term	Certainty Factor
Pasti Tidak	0
Tidak Yakin	0.2
Sedikit Yakin	0.4
Cukup Yakin	0.6
Yakin	0.8
Sangat Yakin	1,0

Tabel 4.5 menjelaskan nilai bobot dari *metode certainty factor* dengan menggunakan 6 bobot yang digunakan yaitu 0, 0.2, 0.4, 0.6, 0.8, dan 1. Semakin besar bobot, semakin besar juga nilai keyakinan dari gejala tersebut.

4.2.10. Teknik Inferensi

Teknik Inferensi pada sistem ini menggunakan teknik *forward chaining* (pelacakan ke depan). Dalam sistem ini, data yang didapat berasal dari pakar langsung, maupun literatur.

Pada tabel 4.6 merupakan tabel aturan, dimana nilai CF *rule* untuk gejala penyakit adalah nilai hipotesis dengan asumsi gejala

(evidence) dari pakar diketahui. Adapun nilai CF Pakar pada masing-masing gejala tersebut di dapat dari pakar yang terkait dengan sistem pakar yang dibuat.

Tabel 4.6 Representasi Aturan Gejala Penyakit Kulit Wajah

Aturan	Penyakit	Gejala	CF Pakar
A01	<i>Acne Vulgaris</i>	G01	0,8
		G02	0,6
		G08	0,8
		G09	0,8
		G10	0,8
A02	<i>Melasma</i>	G01	0,8
		G03	0,8
		G11	0,8
		G12	0,6
		G13	0,8
		G14	0,8
A03	<i>Nevus Melanositik</i>	G01	0,8
		G03	0,8
		G15	0,6
		G16	0,6
		G17	0,8
A04	<i>Syringoma</i>	G01	0,8
		G04	0,6
		G05	0,8
		G06	0,8
		G07	0,8
A05	<i>Milia</i>	G01	0,8
		G04	0,6
		G06	0,8
		G07	0,8
		G18	0,6
		G19	0,8
A06	<i>Rosacea</i>	G01	0,8
		G06	0,8
		G20	0,6
		G21	0,8

		G22	0,6
		G23	0,6
		G24	0,8
A07	<i>Xanthelasma</i>	G01	0,8
		G07	0,8
		G24	0,8
		G25	0,6
		G26	0,8
		G27	0,8
A08	<i>Keratosis Seboroik</i>	G01	0,8
		G03	0,8
		G28	0,6
		G33	0,8
A09	<i>Basalioma</i>	G01	0,8
		G03	0,8
		G29	0,8
		G30	0,6
		G31	0,8
		G34	0,8
A10	<i>Melanoma</i>	G01	0,8
		G03	0,8
		G28	0,6
		G29	0,8
		G32	0,6
		G33	0,8
		G34	0,8

4.2.11. Contoh Kasus dan Proses Perhitungan

- a. Jika *user* mengalami gejala muncul bitnik-bintik atau bercak, terasa nyeri, muncul pori-pori terbuka seperti komedo, Muncul pori-pori tertutup seperti komedo hitam, dan muncul papul (bitnik merah kecil), Untuk menentukan hasil akhir atau kesimpulan penyakit tersebut adalah sebagai berikut:

Tabel 4.7 Perhitungan CF Penyakit *Acne Vulgaris* (Contoh satu)

No	Kode Gejala	Gejala	CF (Pakar)	CF (User)	CF (H,E)
1	G01	muncul bitnik-bintik atau bercak	0,8	0,8	0,64
2	G02	Terasa nyeri	0,6	0,4	0,24
3	G08	Muncul pori-pori terbuka seperti komedo putih	0,8	0,6	0,48
4	G09	Muncul pori-pori tertutup seperti komedo hitam	0,8	0,8	0,64
5	G10	Muncul papul (bintik merah kecil)	0,8	0,8	0,64

Setelah nilai CF diperoleh, langkah selanjutnya adalah menggunakan CF gabungan untuk menghitung nilai CF gabungan karena lebih dari satu gejala. Berikut adalah cara menghitung CF gabungan:

Tabel 4.8 Perhitungan CF Sekuensial Dari Data Contoh Satu

Iterasi/ Hasil	$Cfgabungan = CF1 + CF2 \times (1 - CF1)$
1	$0,64 + 0,24 * (1 - 0,64) = 0,7264$
2	$0,7264 + 0,48 * (1 - 0,7264) = 0,857728$
3	$0,857728 + 0,64 * (1 - 0,857728) = 0,94878208$
4	$0,94878208 + 0,64 * (1 - 0,94878208) = 0,981561549$
Hasil	$0,981561549 * 100\% = 98,15\%$

Hasil CF *combine* terakhir merupakan hasil dari diagnosa yang menentukan jenis kulit wajah dengan menggunakan perhitungan *Certainty factor* dan karena *rule* ke satu yang terpenuhi dengan menggunakan kaidah teknik *Forward chaining*, Maka hasil diagnosa dari data contoh satu menyatakan bahwa *user* memiliki penyakit kulit wajah *Acne Vulgaris* dengan tingkat keyakinan 98.15%.

- b. Jika *user* mengalami gejala muncul bitnik-bintik atau bercak, bintik atau bercak berwarna gelap, mengalami peradangan pada area tersebut, Mengalami hiperpigmentasi atau perbedaan warna kulit dari sebelumnya, Sebelumnya pernah memakai obat-obatan kontrasepsi, Bintik atau bercak berada dibagian pipi kanan dan kiri. Untuk menentukan hasil akhir atau kesimpulan penyakit tersebut adalah sebagai berikut:

Tabel 4.9 Perhitungan CF Penyakit Melasma (Contoh Dua)

No	Kode Gejala	Gejala	CF (Pakar)	CF (User)	CF (H,E)
1	G01	muncul bitnik-bintik atau bercak	0,8	0,8	0,64
2	G03	Bintik atau bercak berwarna gelap	0,8	0,6	0,48
3	G11	Mengalami Peradangan pada area tersebut	0,8	0,4	0,32
4	G12	Mengalami hiperpigmentasi atau perbedaan warna kulit dari sebelumnya	0,6	0,2	0,12
5	G13	Sebelumnya pernah memakai obat-obatan kontrasepsi	0,8	0,8	0,64
5	G14	Bintik atau bercak berada dibagian pipi kanan dan kiri	0,8	0,8	0,64

Setelah nilai CF diperoleh, langkah selanjutnya adalah menggunakan CF gabungan untuk menghitung nilai CF gabungan karena lebih dari satu gejala. Berikut adalah cara

menghitung CF gabungan:

Tabel 4.10 Perhitungan CF Sekuensial Dari Data Contoh Dua

Iterasi / Hasil	$C_{fgabungan} = CF_1 + CF_2 \times (1 - CF_1)$
1	$0,64 + 0,48 * (1 - 0,64) = 0,8128$
2	$0,8128 + 0,32 * (1 - 0,8128) = 0,872704$
3	$0,872704 + 0,12 * (1 - 0,872704) = 0,88797952$
4	$0,88797952 + 0,64 * (1 - 0,88797952) = 0,959672627$
5	$0,959672627 + 0,64 * (1 - 0,959672627) = 0,985482146$
Hasil	$0,985482146 * 100\% = 98,54\%$

Hasil CF *combine* terakhir merupakan hasil dari diagnosa yang menentukan jenis kulit wajah dengan menggunakan perhitungan *Certainty factor* dan karena rule ke satu yang terpenuhi dengan menggunakan kaidah teknik *Forward chaining*, Maka hasil diagnosa dari data contoh dua menyatakan bahwa *user* memiliki penyakit kulit wajah *Melasma* dengan tingkat keyakinan 98.54%.

- c. Jika *user* mengalami gejala muncul bitnik-bintik atau bercak, terasa nyeri, Bintik atau Bercak berwarna gelap, Mengalami Peradangan pada area tersebut, Bentuk Bintik ataupun bercak semakin membesar, dan berbentuk Bulat dan tegas. Untuk menentukan hasil akhir atau kesimpulan penyakit tersebut adalah sebagai berikut.

Tabel 4.11 Perhitungan CF Penyakit *Nevus Melanositik* (Contoh Tiga)

No	Kode Gejala	Gejala	CF (Pakar)	CF (User)	CF (H,E)
1	G01	muncul bitnik-bintik atau bercak	0,8	0,8	0,64
2	G03	Bintik atau Bercak berwarna gelap	0,8	0,6	0,48

3	G15	Mengalami Peradangan pada area tersebut	0,6	0,6	0,36
4	G16	Bentuk Bintik ataupun bercak semakin membesar	0,6	0,4	0,24
5	G17	Berbentuk Bulat dan tegas	0,8	0,4	0,32

Setelah nilai CF diperoleh, langkah selanjutnya adalah menggunakan CF gabungan untuk menghitung nilai CF gabungan karena lebih dari satu gejala. Berikut adalah cara menghitung CF gabungan :

Tabel 4.12 Perhitungan CF Sekuensial Dari Data Contoh Tiga

Iterasi / Hasil	$Cfgabungan = CF1 + CF2 \times (1 - CF1)$
1	$0,64 + 0,48 * (1 - 0,64) = 0,8128$
2	$0,8128 + 0,36 * (1 - 0,8128) = 0,880192$
3	$0,880192 + 0,24 * (1 - 0,880192) = 0,90894592$
4	$0,90894592 + 0,32 * (1 - 0,90894592) = 0,938083226$
Hasil	$0,938083226 * 100\% = 93,80\%$

Hasil CF *combine* terakhir merupakan hasil dari diagnosa yang menentukan jenis kulit wajah dengan menggunakan perhitungan *Certainty factor* dan karena *rule* ke tiga yang terpenuhi dengan menggunakan kaidah teknik *Forward chaining*, Maka hasil diagnosa dari data contoh tiga menyatakan bahwa *user* memiliki penyakit kulit wajah *Nevus Melanositik* dengan tingkat keyakinan 93.80%.

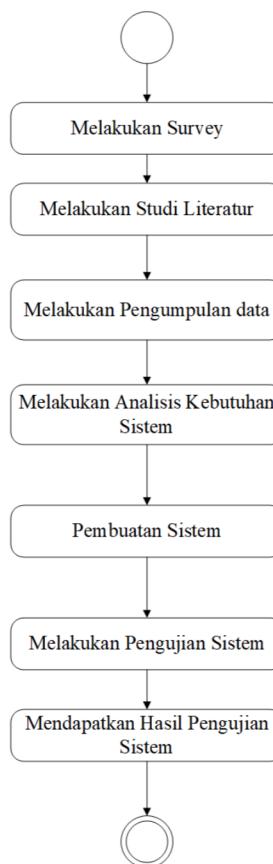
4.3 Activity Diagram

Diagram aktivitas atau *activity diagram* berguna untuk menggambarkan proses atau aktifitas dari sebuah sistem atau proses bisnis atau menu

yang ada pada sistem pakar diagnosa penyakit kulit wajah. Tujuan dengan adanya *activity diagram* adalah menjelaskan sebuah alur program atau kegiatan yang berada di dalam ruang lingkup penggunaan sistem ataupun aplikasi ini. Alur atau aktivitas ini dapat berupa menu-menu pada aplikasi sistem pakar penyakit kulit wajah atau proses bisnis yang terdapat didalam sistem pakar diagnosa penyakit kulit wajah. Pada pembahasan dibawah ini ada beberapa *activity diagram* yang digunakan oleh sistem diantaranya sebagai berikut.

4.3.1. *Activity Diagram* Penelitian

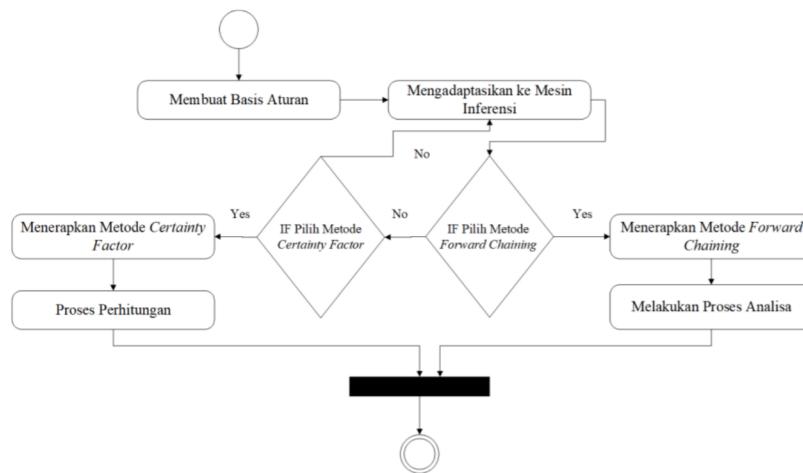
Activity diagram penelitian bertujuan untuk menjelaskan urutan aktivitas dalam penelitian serta memahami proses penelitian secara keseluruhan. Pada *activity diagram* tersebut yang pertama adalah melakukan *survey* kemudian melakukan studi literatur, melakukan pengumpulan data, melakukan analisis sistem, pembuatan sistem, melakukan pengujian sistem, mendapatkan hasil pengujian sistem. Berikut adalah *activity diagram* penelitian dapat dilihat pada gambar 4.3



Gambar 4.3 *Activity Diagram* Penelitian

4.3.2. *Activity Diagram* Analisis Sistem

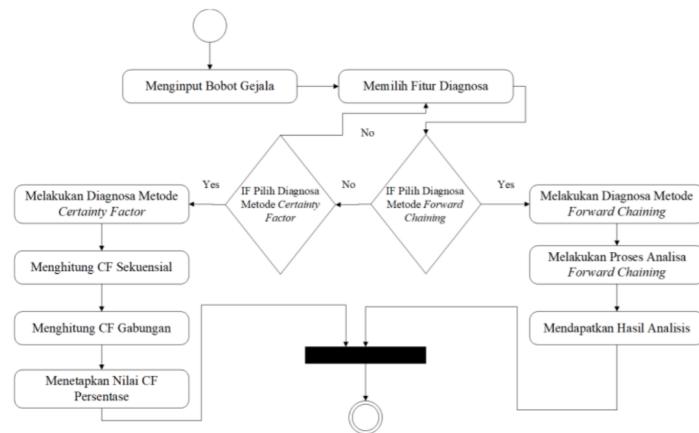
Activity diagram analisis sistem bertujuan untuk menjelaskan urutan aktivitas tentang analisis sistem yang digunakan dalam pembuatan sistem. Berikut adalah *activity diagram* analisis sistem dapat dilihat pada gambar 4.4.



Gambar 4.4 *Activity Diagram* Analisis Sistem

4.3.3. *Activity Diagram* Sistem Pakar

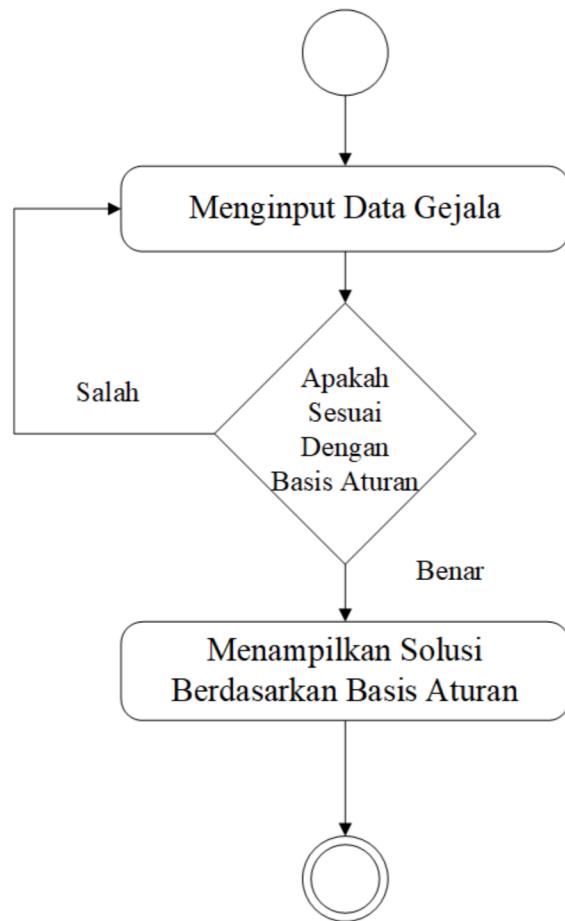
Activity diagram sistem pakar bertujuan untuk menjelaskan urutan aktivitas tentang menjelaskan tentang alur sistem pakar saat *user* melakukan diagnosa. Berikut adalah *activity diagram* sistem pakar dapat dilihat pada gambar 4.5.



Gambar 4.5 *Activity Diagram* Sistem Pakar

4.3.4. *Activity Diagram* Metode Forward Chaining

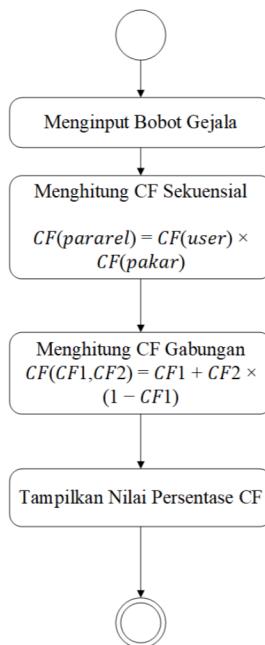
Activity diagram metode *forward chaining* bertujuan untuk menjelaskan urutan aktivitas atau alur metode *forward chaining*. Berikut adalah *activity diagram* daftar dapat dilihat pada gambar 4.6.



Gambar 4.6 *Activity Diagram* Metode Forward Chaining

4.3.5. *Activity Diagram* Metode Certainty Factor

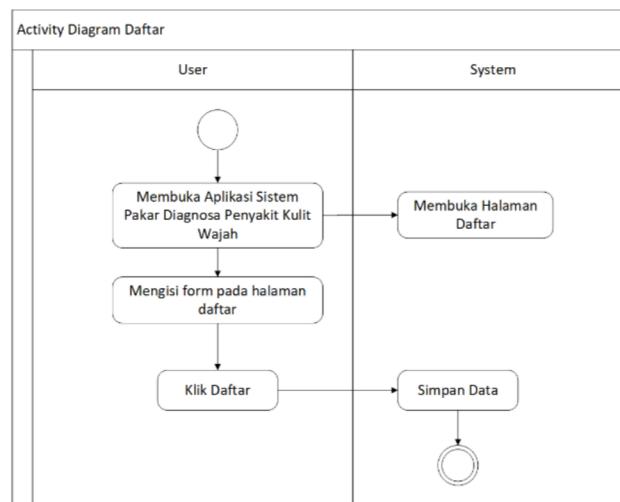
Activity diagram metode *certainty factor* bertujuan untuk menjelaskan urutan aktivitas atau alur metode *certainty factor*. Berikut adalah *activity diagram* daftar dapat dilihat pada gambar 4.7.



Gambar 4.7 Activity Diagram Metode Certainty Factor

4.3.6. Activity Diagram Daftar

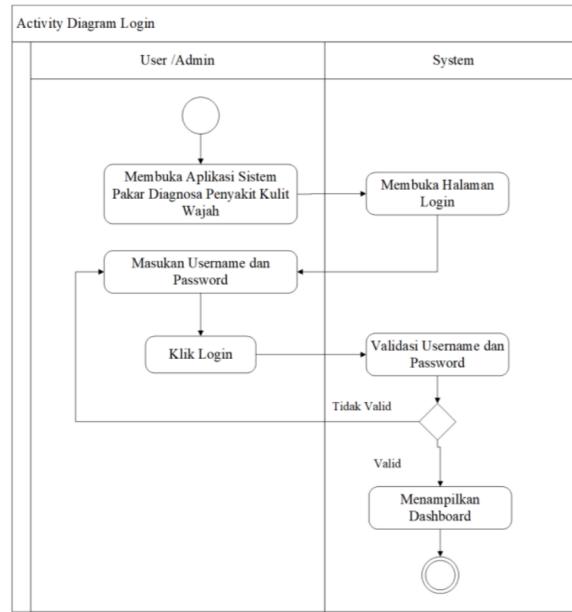
Activity diagram daftar bertujuan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* daftar. *Activity Diagram* daftar memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* daftar dapat dilihat pada gambar 4.8.



Gambar 4.8 Activity Diagram Daftar

4.3.7. Activity Diagram Login User/Admin

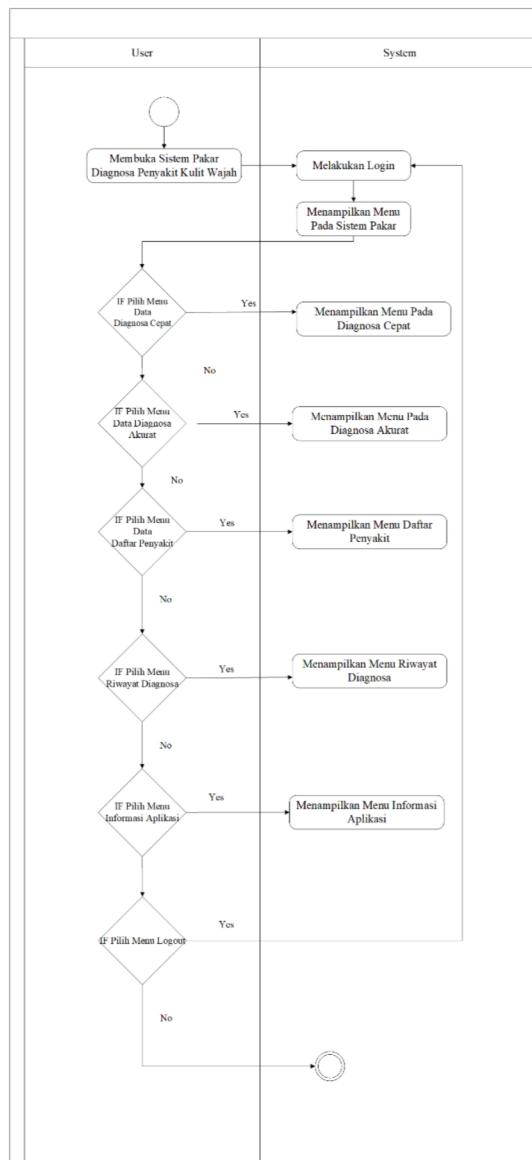
Activity diagram login bertujuan untuk menjelaskan urutan aktivitas atau alur program pada *fitur login* baik disisi *admin* maupun *user*. *Activity diagram* daftar memiliki dua *partition* yaitu *user / admin* dan *system*. Berikut adalah *activity diagram login* dapat dilihat pada gambar 4.9.



Gambar 4.9 Activity Diagram Login

4.3.8. Activity Diagram Menu Utama User

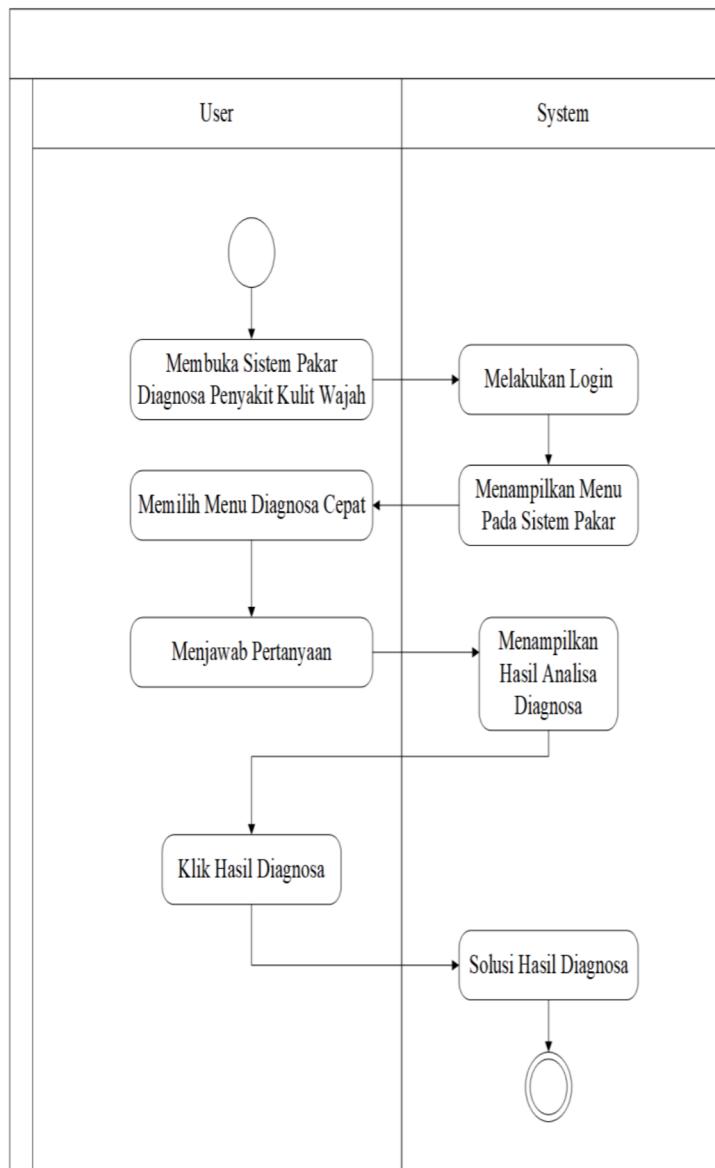
Activity diagram menu utama *user* bertujuan untuk menjelaskan urutan aktivitas atau alur program pada *fitur menu utama user* bagi *user*. *Activity diagram* menu utama *user* memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* menu utama *user* dapat dilihat pada gambar 4.10.



Gambar 4.10 Activity Diagram Menu Utama User

4.3.9. Activity Diagram Diagnosa Cepat

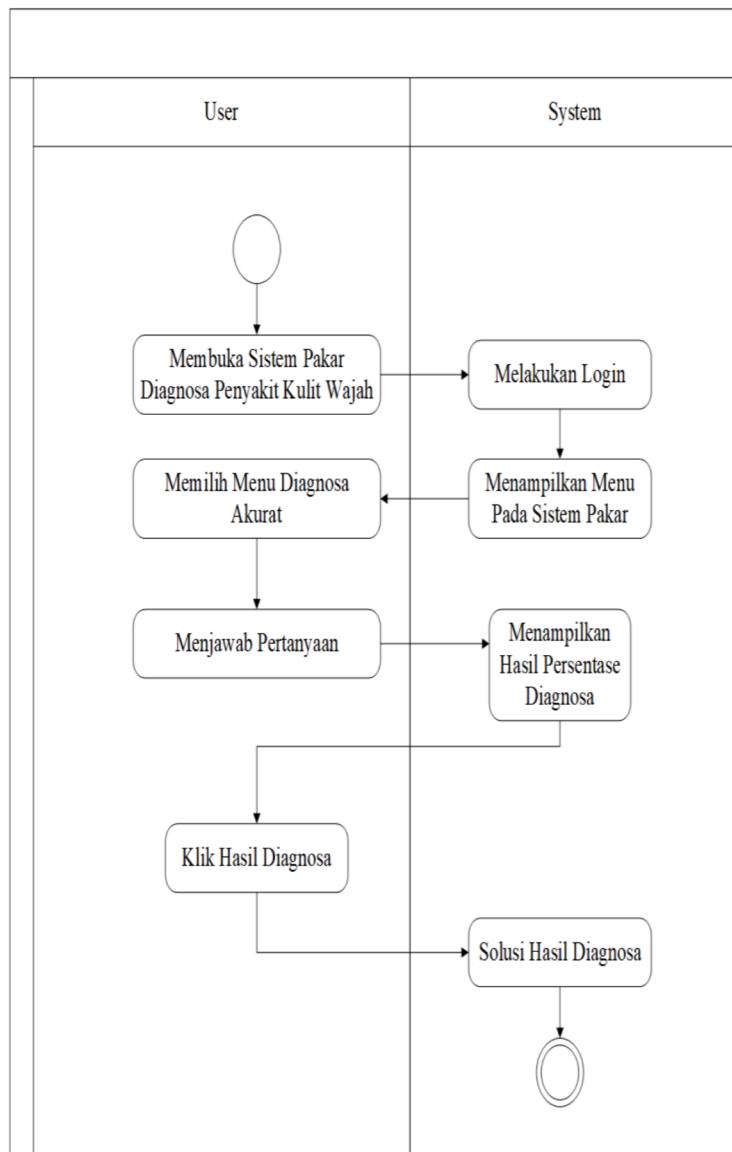
Activity diagram diagnosa bertujuan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* diagnosa cepat bagi *user*. *Activity diagram* diagnosa cepat memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* diagnosa cepat dapat dilihat pada gambar 4.11.



Gambar 4.11 Activity Diagram Diagnosa Cepat

4.3.10. *Activity Diagram* Diagnosa Akurat

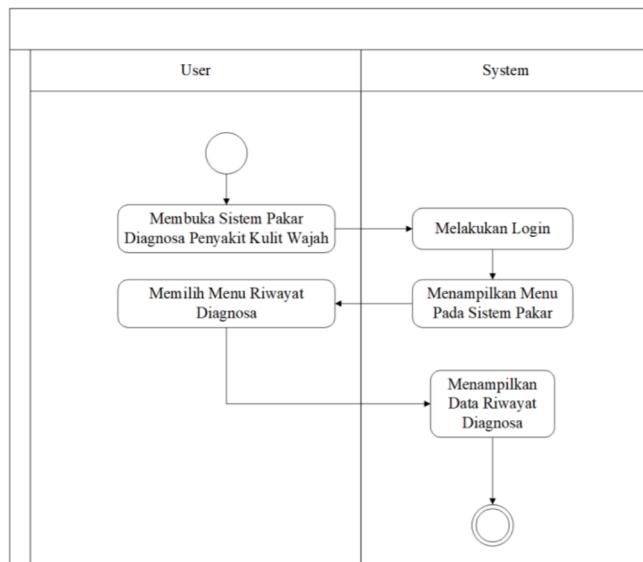
Activity diagram diagnosa akurat bertujuan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* diagnosa cepat bagi *user*. *Activity diagram* diagnosa akurat memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* diagnosa akurat dapat dilihat pada gambar 4.12.



Gambar 4.12 Activity Diagram Diagnosa Akurat

4.3.11. *Activity Diagram* Daftar Riwayat Diagnosa

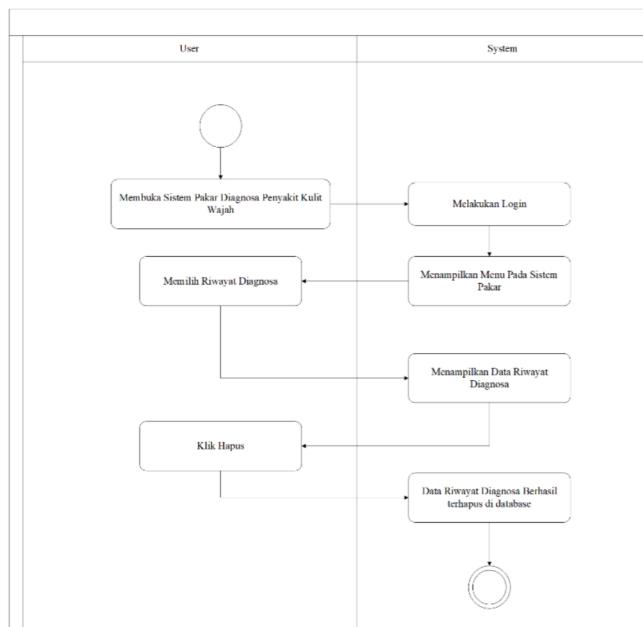
Activity diagram riwayat diagnosa bertujuan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* riwayat diagnosa bagi *user*. *Activity diagram* daftar riwayat diagnosa memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* daftar riwayat diagnosa dapat dilihat pada gambar 4.13.



Gambar 4.13 Activity Diagram Daftar Riwayat Diagnosa

4.3.12. *Activity Diagram* Hapus Daftar Riwayat Diagnosa

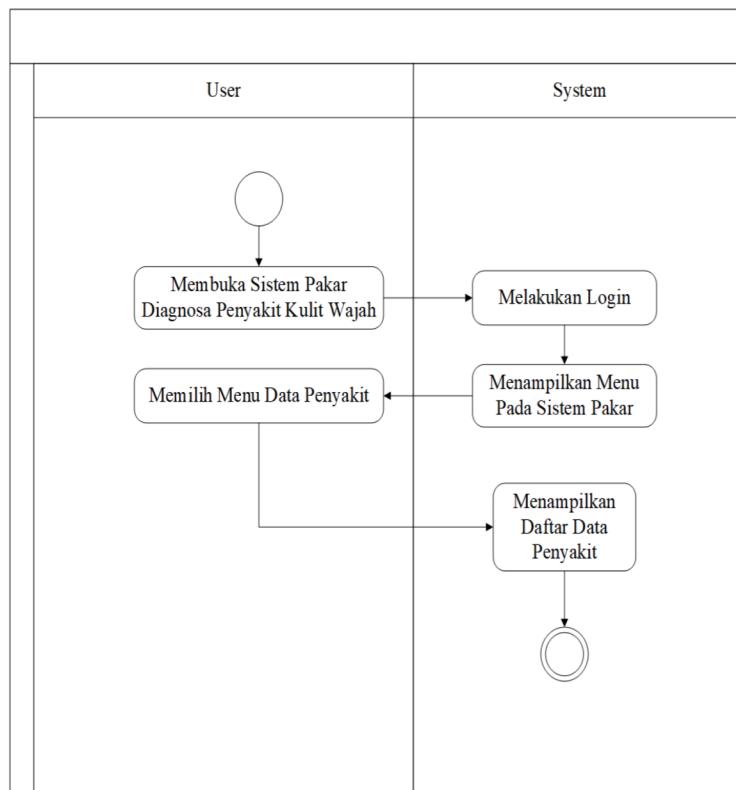
Activity diagram daftar bertujuan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* diagnosa cepat bagi *user*. *Activity diagram* hapus daftar riwayat diagnosa memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* hapus daftar riwayat diagnosa dapat dilihat pada gambar 4.14.



Gambar 4.14 Activity Diagram Hapus Daftar Riwayat Diagnosa

4.3.13. *Activity Diagram* Daftar Penyakit

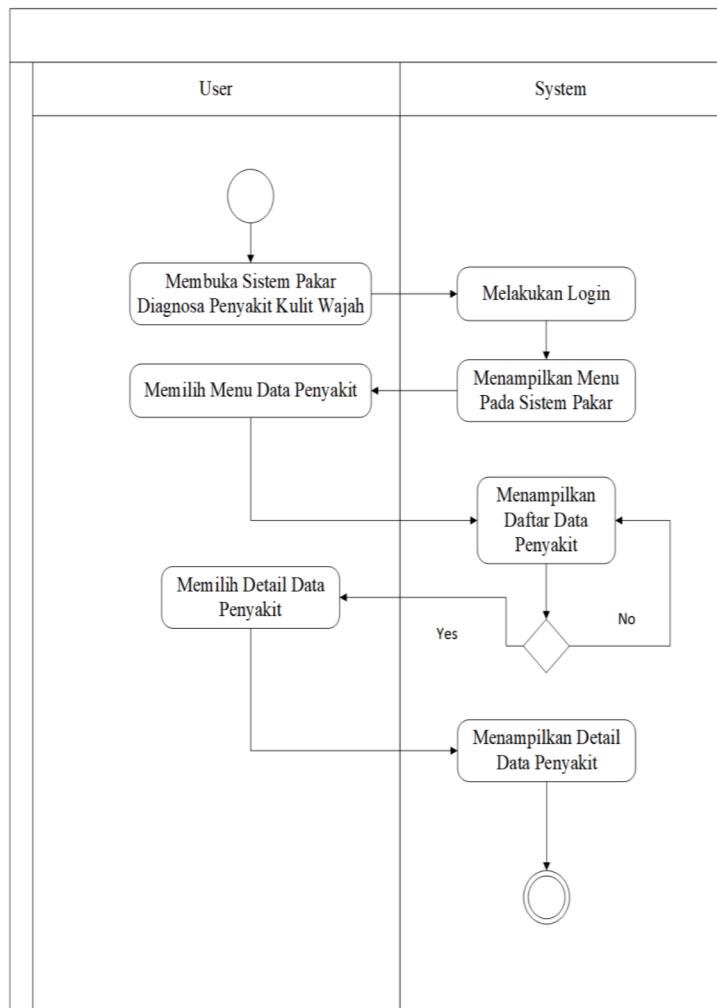
Activity diagram daftar penyakit untuk menjelaskan urutan aktivitas atau alur program pada *fitur* daftar penyakit bagi *user*. *Activity Diagram* daftar penyakit memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* daftar penyakit dapat dilihat pada gambar 4.15.



Gambar 4.15 *Activity Diagram* Daftar Penyakit

4.3.14. *Activity Diagram* Detail Data Penyakit

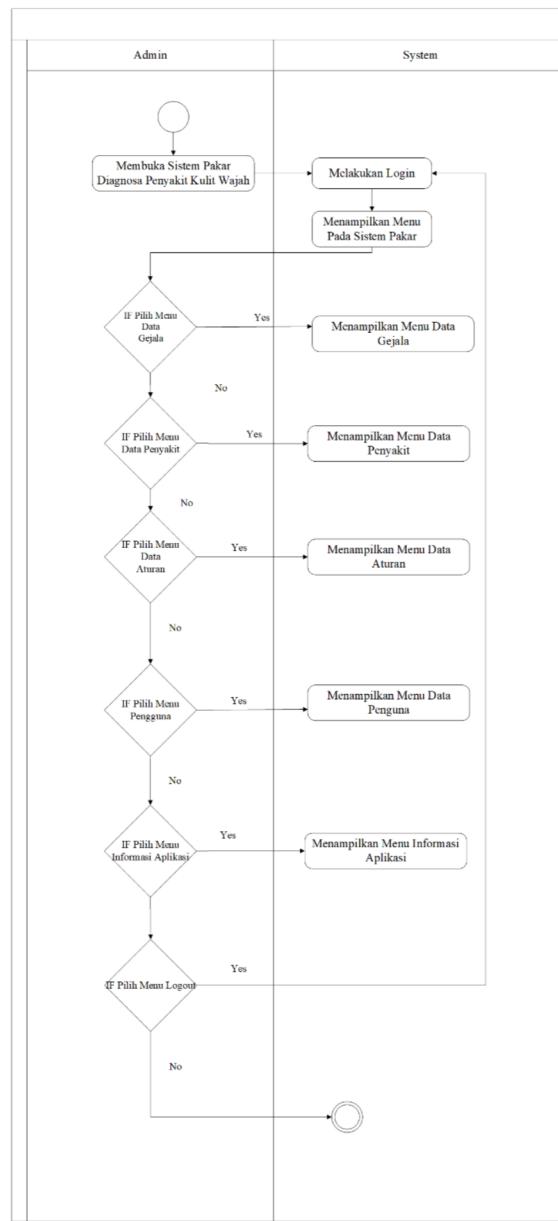
Activity diagram detail penyakit untuk menjelaskan urutan aktivitas atau alur program pada *fitur* detail data penyakit bagi *user*. *Activity Diagram* detail data penyakit memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* detail data penyakit dapat dilihat pada gambar 4.16.



Gambar 4.16 Activity Diagram Detail Data Penyakit

4.3.15. Activity Diagram Menu Utama Admin

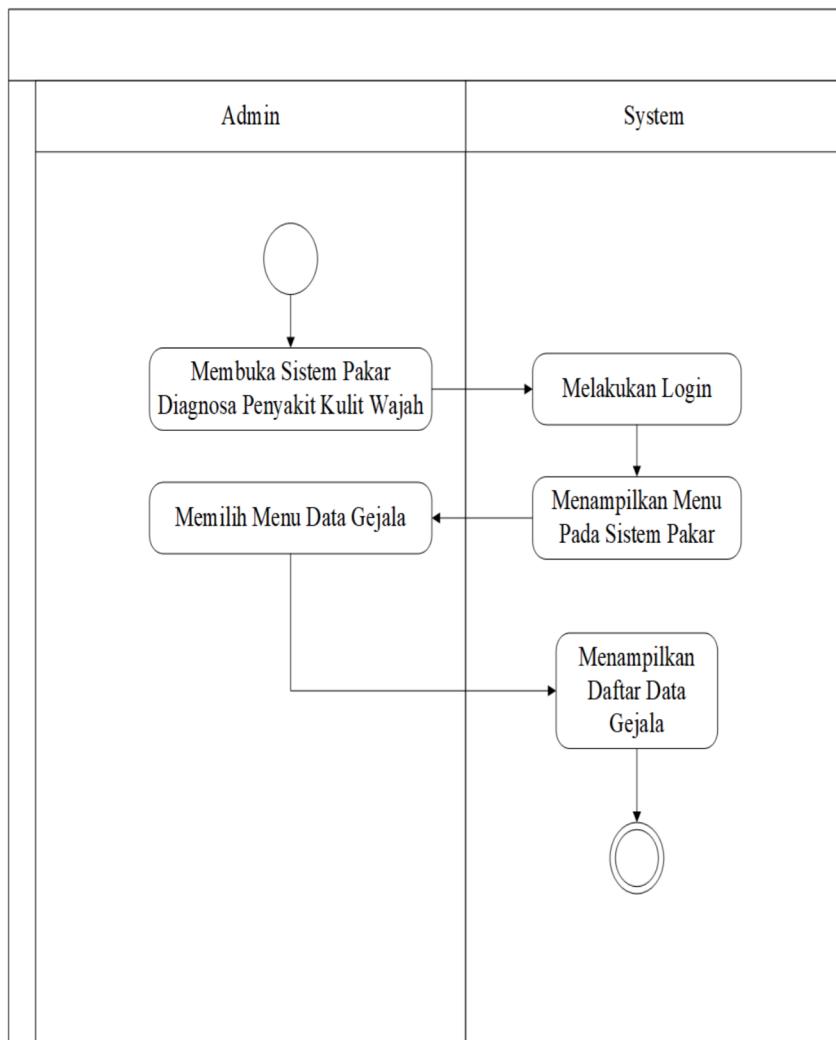
Activity diagram menu utama *admin* bertujuan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* menu utama *admin* bagi *user*. *Activity diagram* menu utama *admin* memiliki dua *partition* yaitu *user* dan *system*. Berikut adalah *activity diagram* menu utama *admin* dapat dilihat pada gambar 4.17.



Gambar 4.17 Activity Diagram Menu Utama Admin

4.3.16. Activity Diagram Daftar Data Gejala

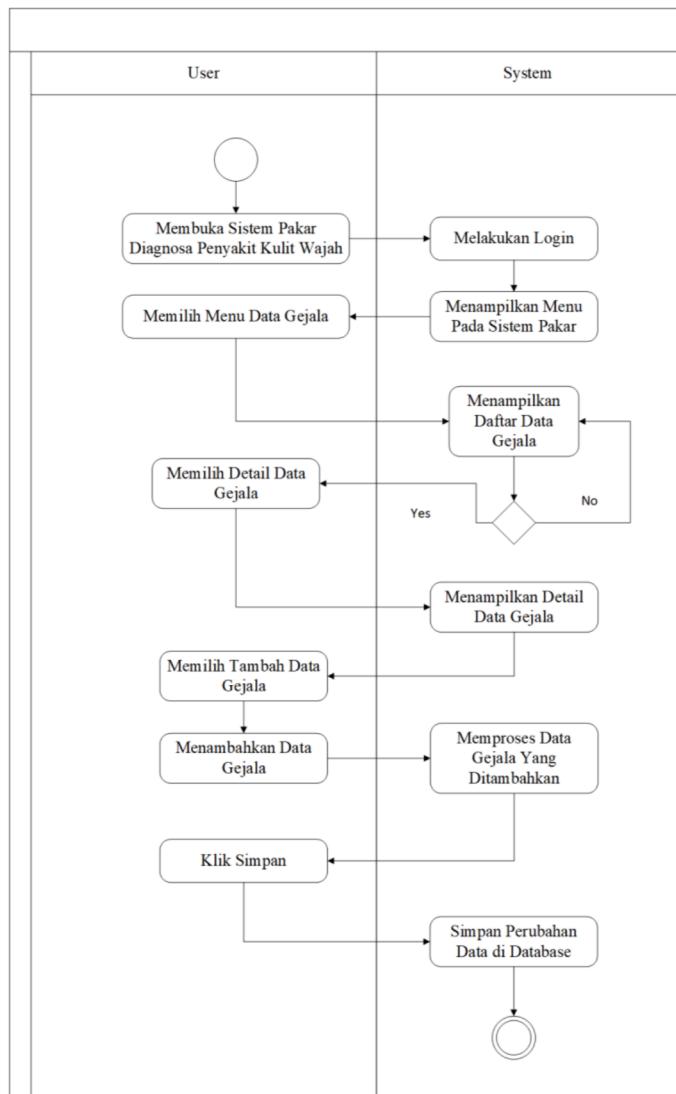
Activity diagram daftar data gejala digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data gejala bagi *admin*. *Activity Diagram* daftar data gejala memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* daftar data gejala dapat dilihat pada gambar 4.18.



Gambar 4.18 Activity Diagram Daftar Data Gejala

4.3.17. Activity Diagram Tambah Data Gejala

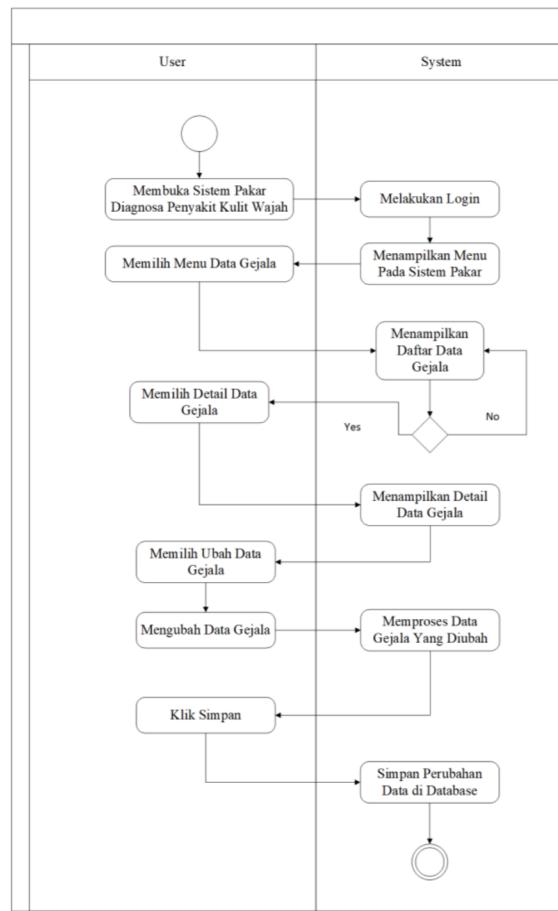
Activity diagram tambah data gejala digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data gejala bagi *admin*. *Activity diagram* tambah data gejala memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* tambah data gejala dapat dilihat pada gambar 4.19.



Gambar 4.19 Activity Diagram Tambah Data Gejala

4.3.18. Activity Diagram Ubah Data Gejala

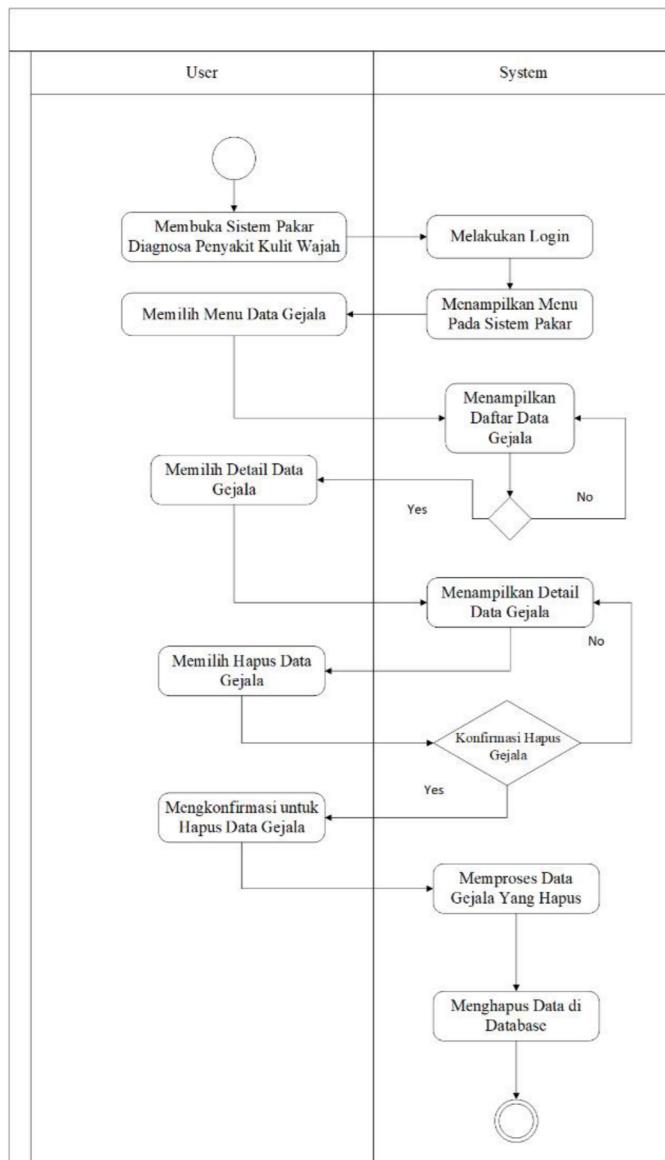
Activity diagram ubah data gejala digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data gejala bagi *admin*. *Activity Diagram* ubah data gejala memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* ubah data gejala dapat dilihat pada gambar 4.20.



Gambar 4.20 Activity Diagram Ubah Data Gejala

4.3.19. Activity Diagram Hapus Data Gejala

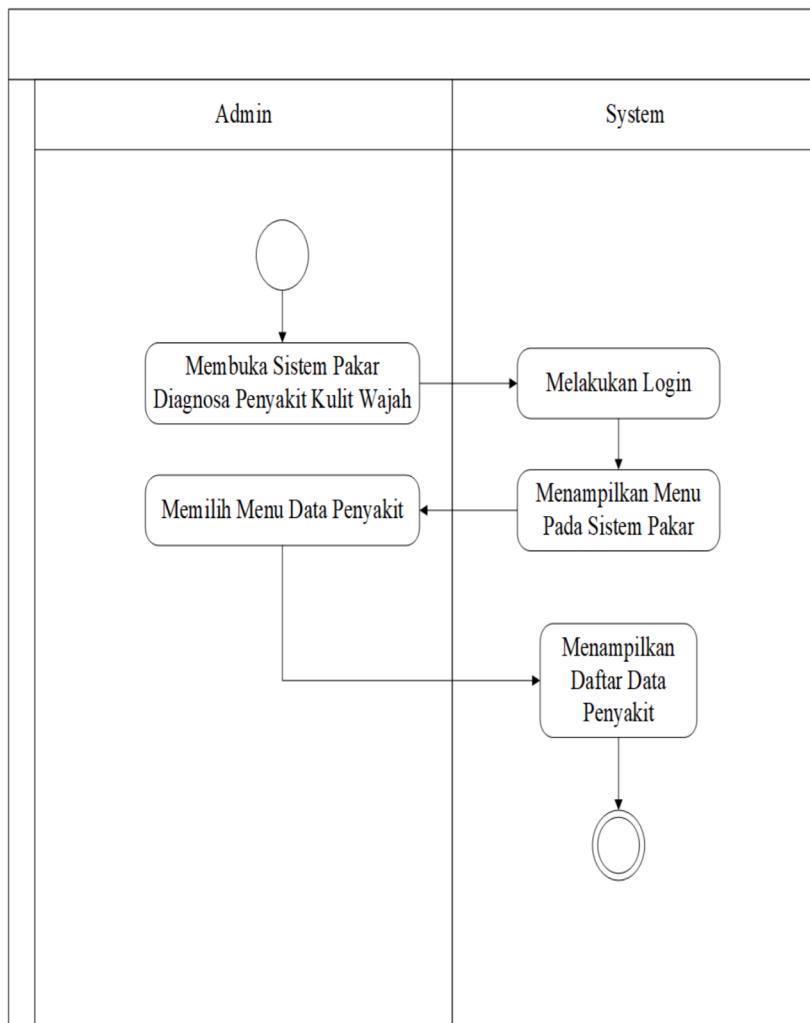
Activity diagram hapus data gejala digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data gejala bagi *admin*. *Activity Diagram* hapus data gejala memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* hapus data gejala dapat dilihat pada gambar 4.2.



Gambar 4.21 Activity Diagram Hapus Data Gejala

4.3.20. Activity Diagram Daftar Data Penyakit

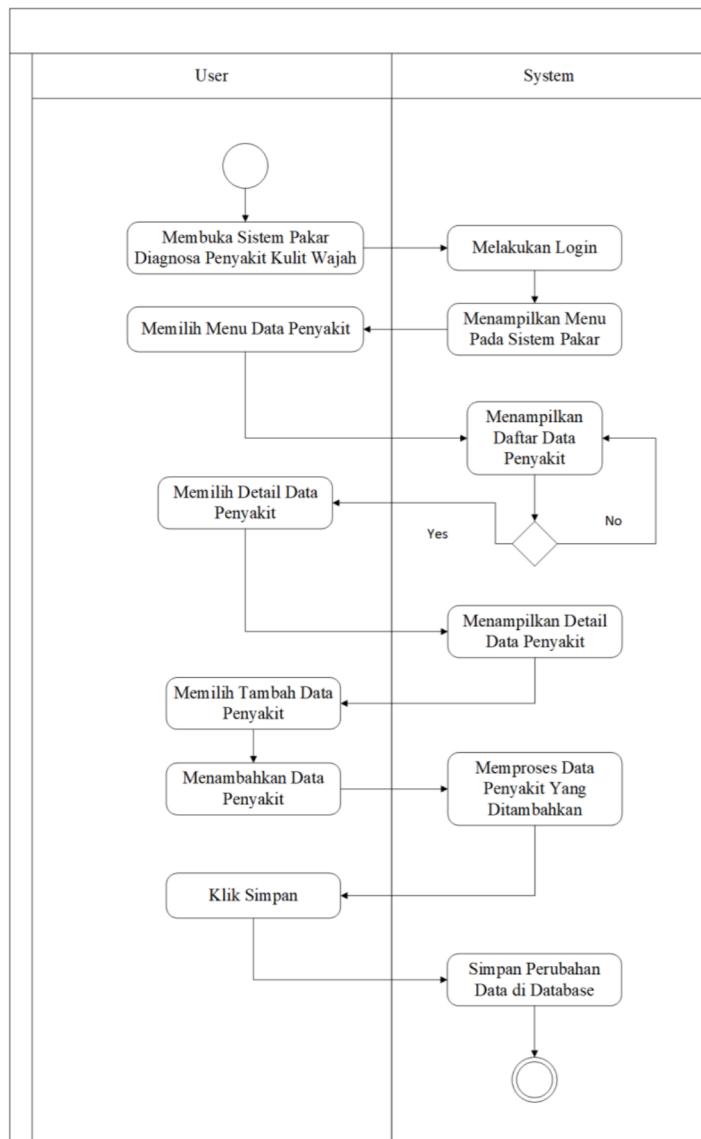
Activity diagram daftar data penyakit digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data penyakit bagi *admin*. *Activity Diagram* daftar data penyakit memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* daftar data penyakit dapat dilihat pada gambar 4.22.



Gambar 4.22 Activity Diagram Daftar Data Penyakit

4.3.21. *Activity Diagram Tambah Data Penyakit*

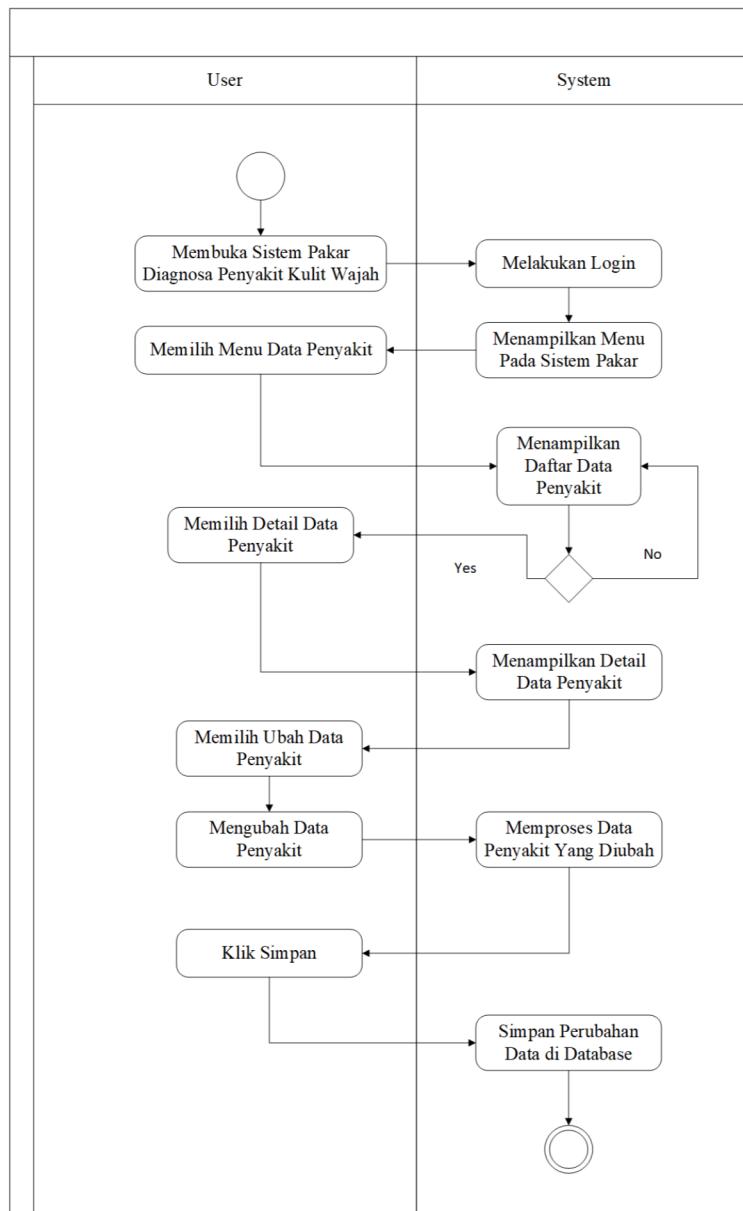
Activity diagram tambah data penyakit digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data penyakit bagi *admin*. *Activity Diagram* tambah data penyakit memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* tambah data penyakit dapat dilihat pada gambar 4.23.



Gambar 4.23 Activity Diagram Tambah Data Penyakit

4.3.22. Activity Diagram Ubah Data Penyakit

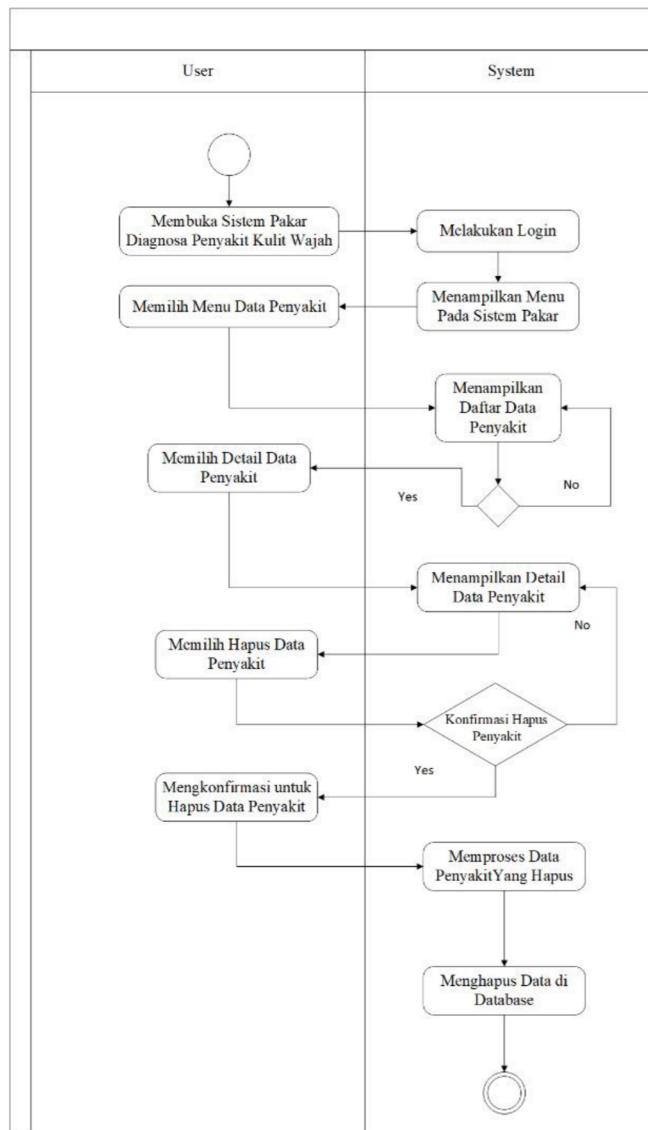
Activity diagram tambah data penyakit digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data penyakit bagi *admin*. *Activity Diagram* ubah data penyakit memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* tambah data penyakit dapat dilihat pada gambar 4.24.



Gambar 4.24 Activity Diagram Ubah Data Penyakit

4.3.23. Activity Diagram Hapus Data Penyakit

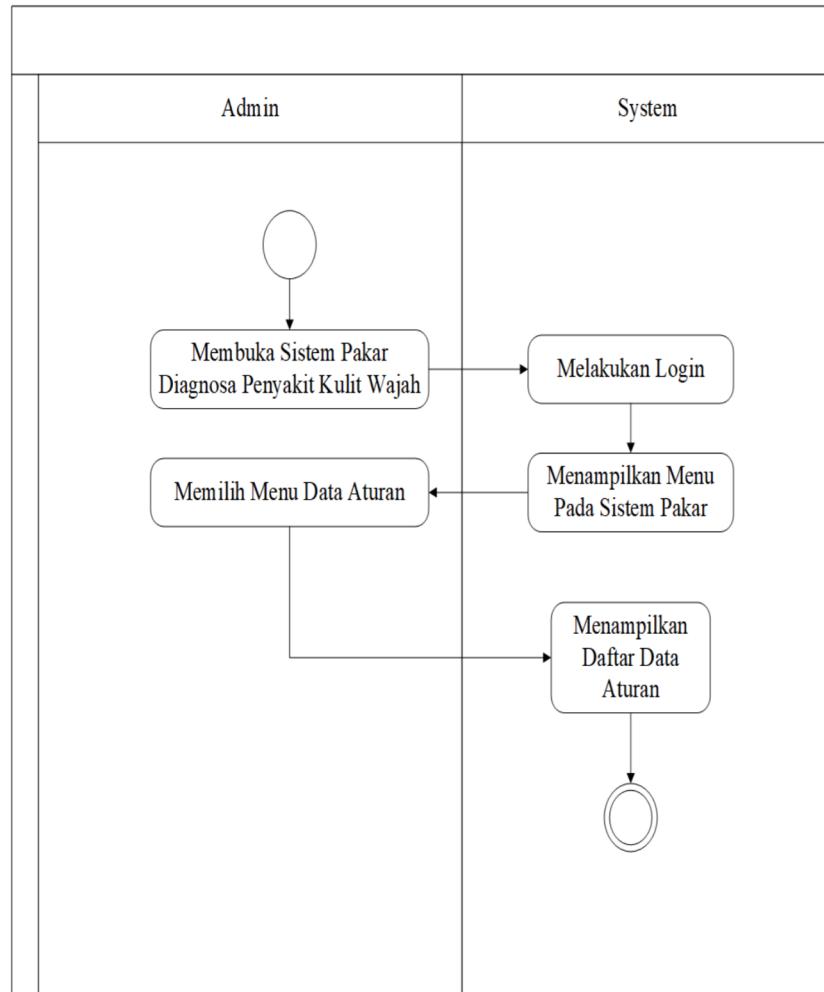
Activity diagram hapus data penyakit digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* hapus data penyakit bagi *admin*. *Activity Diagram* hapus data penyakit memiliki dua partition yaitu *admin* dan *system*. Berikut adalah *activity diagram* hapus data penyakit dapat dilihat pada gambar 4.25.



Gambar 4.25 *Activity Diagram Hapus Data Penyakit*

4.3.24. *Activity Diagram Daftar Data Aturan*

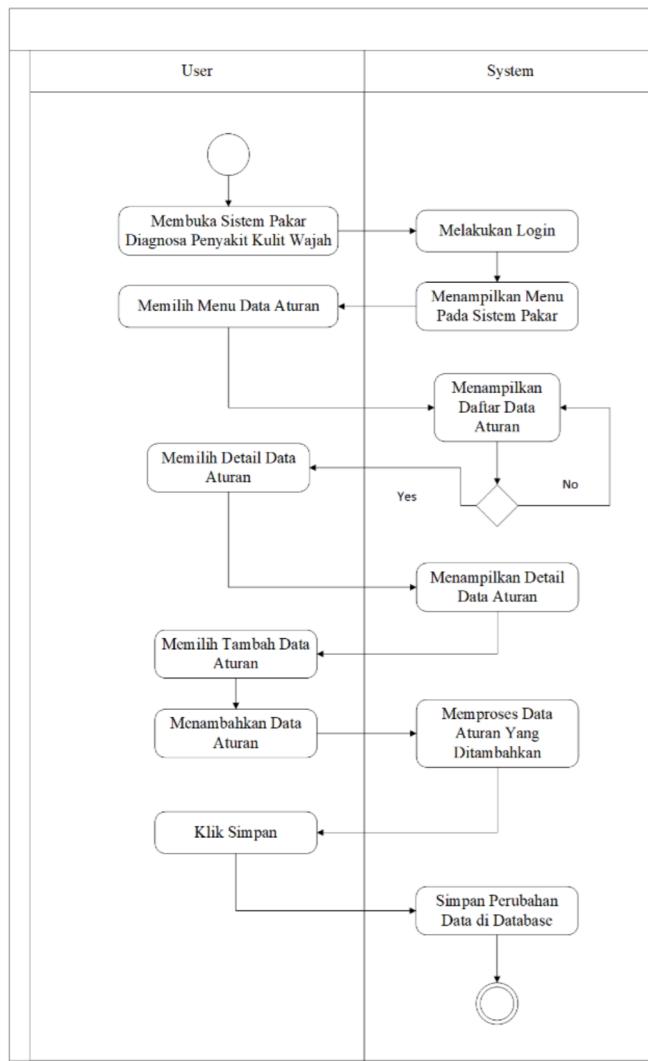
Activity diagram daftar data aturan digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data aturan bagi *admin*. *Activity Diagram* daftar data aturan memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* daftar data aturan dapat dilihat pada gambar 4.26.



Gambar 4.26 Activity Diagram Daftar Data Aturan

4.3.25. Activity Diagram Tambah Data Aturan

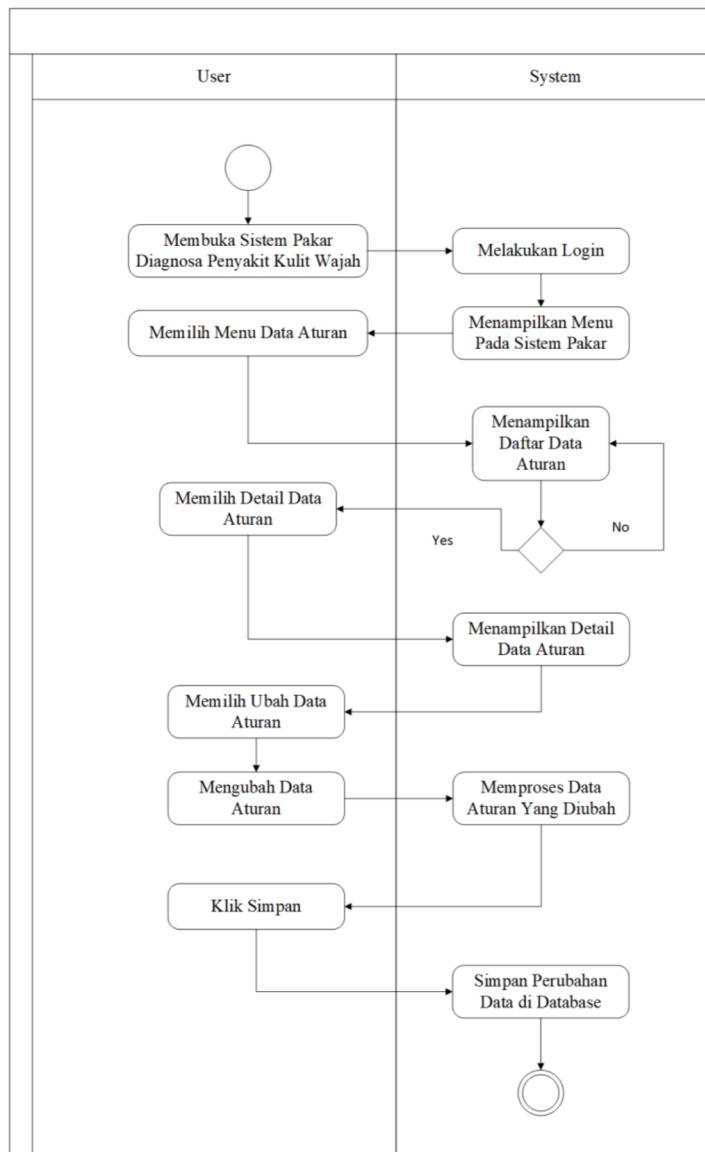
Activity diagram tambah data aturan digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data aturan bagi *admin*. *Activity Diagram* tambah data aturan memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* tambah data aturan dapat dilihat pada gambar 4.27.



Gambar 4.27 Activity Diagram Tambah Data Aturan

4.3.26. Activity Diagram Ubah Data Aturan

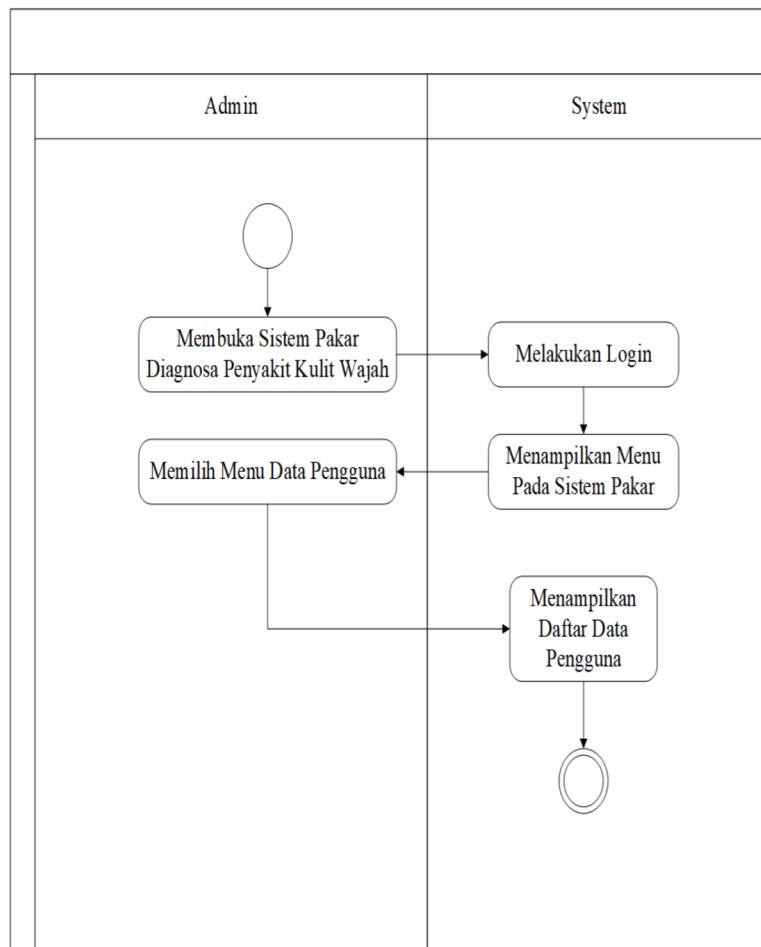
Activity diagram ubah data aturan digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data aturan bagi *admin*. *Activity Diagram* ubah data aturan memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* ubah data aturan dapat dilihat pada gambar 4.28.



Gambar 4.28 Activity Diagram Ubah Data Aturan

4.3.27. Activity Diagram Daftar Data Pengguna

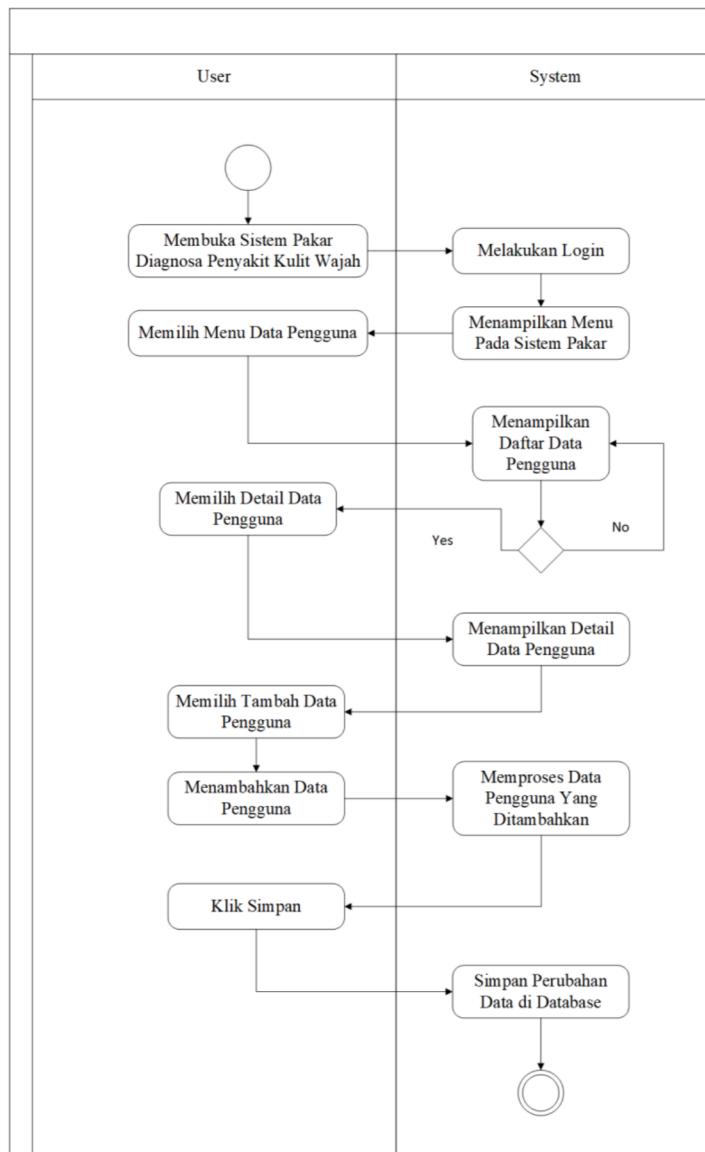
Activity diagram daftar data pengguna digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data pengguna bagi *admin*. *Activity Diagram* daftar memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* daftar data pengguna dapat dilihat pada gambar 4.29.



Gambar 4.29 Activity Diagram Daftar Data Pengguna

4.3.28. *Activity Diagram Tambah Data Pengguna*

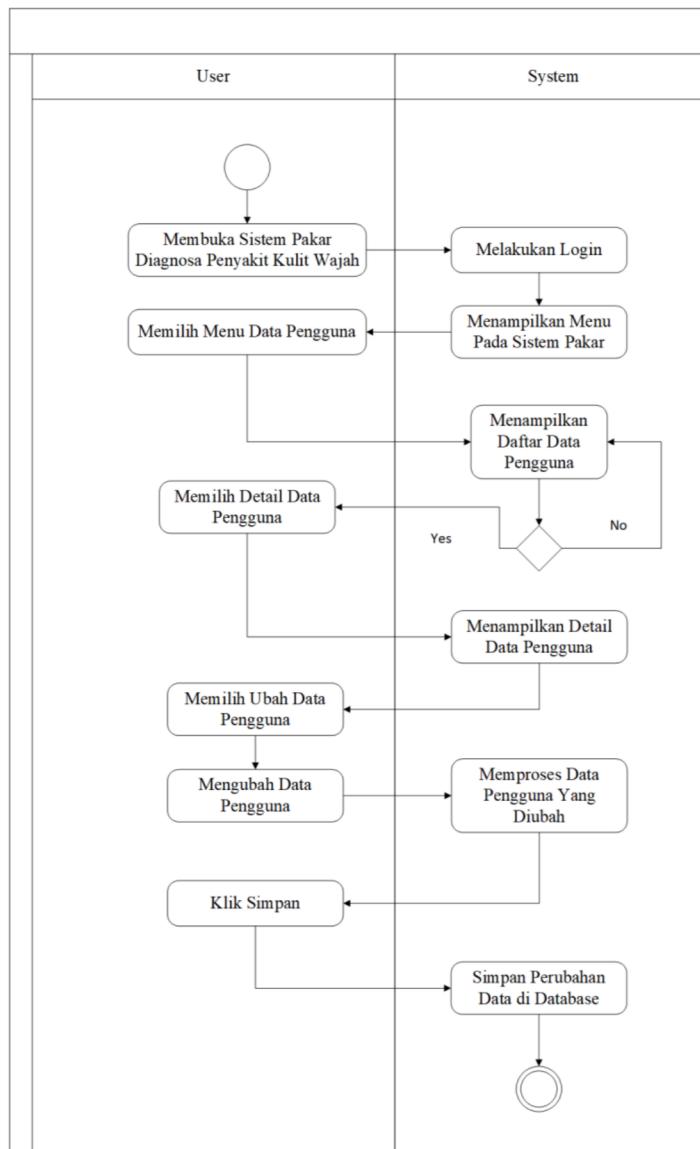
Activity diagram tambah data pengguna digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data pengguna bagi *admin*. *Activity Diagram* daftar memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* tambah data pengguna dapat dilihat pada gambar 4.30.



Gambar 4.30 Activity Diagram Tambah Data Pengguna

4.3.29. Activity Diagram Ubah Data Pengguna

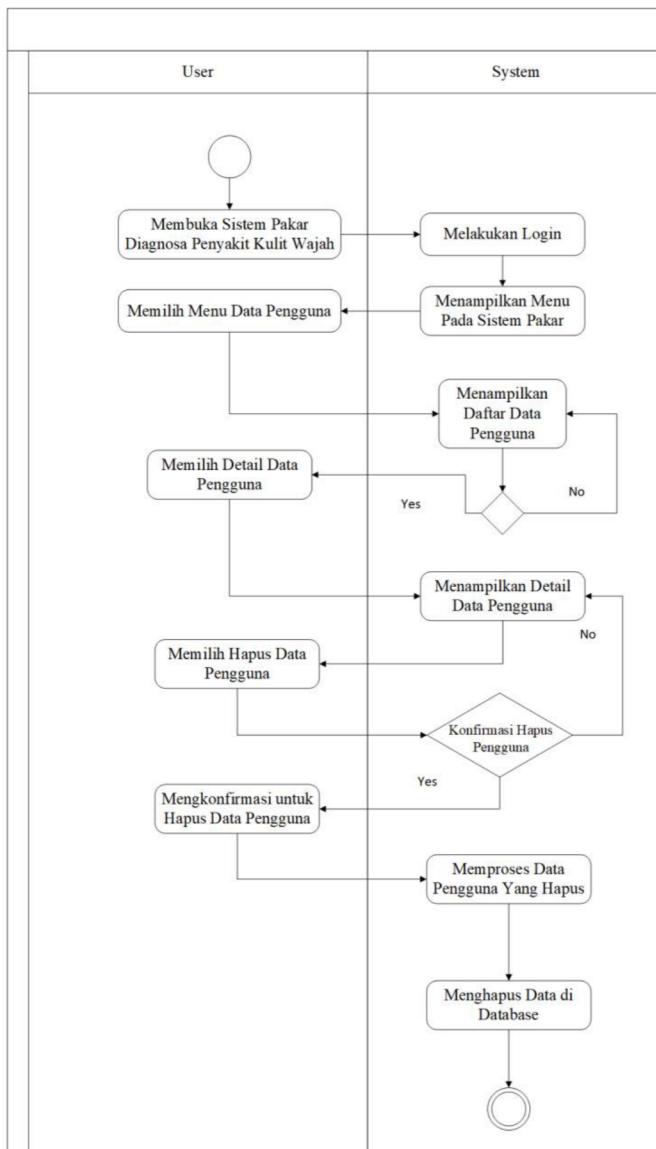
Activity diagram ubah data pengguna digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* data pengguna bagi *admin*. *Activity Diagram* daftar memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* ubah data pengguna dapat dilihat pada gambar 4.31.



Gambar 4.31 *Activity Diagram* Ubah Data Pengguna

4.3.30. *Activity Diagram* Hapus Data Pengguna

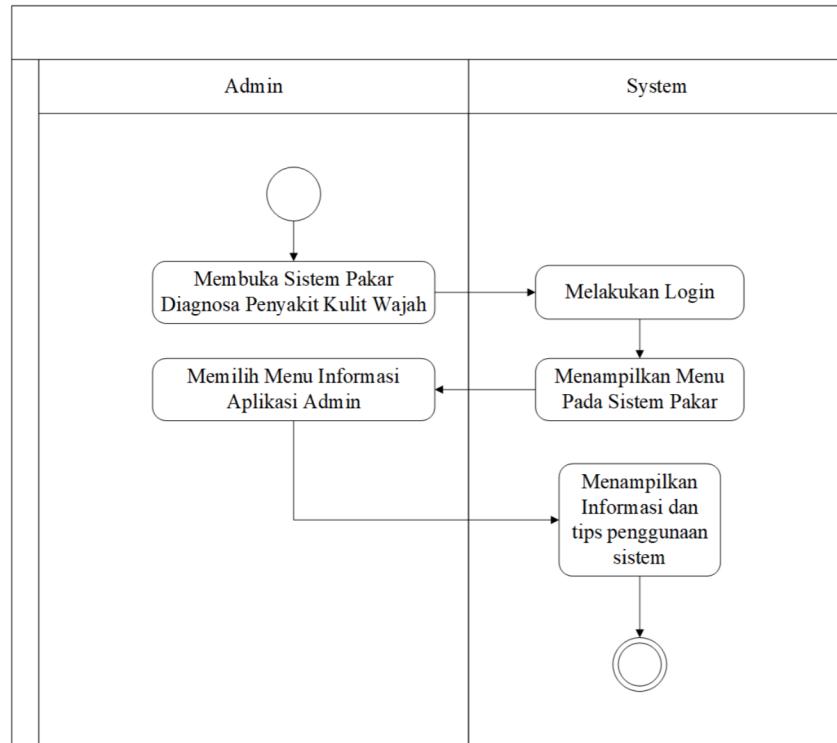
Activity diagram hapus data pengguna digunakan untuk menjelaskan urutan aktivitas atau alur program pada fitur data pengguna bagi admin. *Activity Diagram* daftar memiliki dua partition yaitu admin dan system. Berikut adalah *activity diagram* hapus data pengguna dapat dilihat pada gambar 4.32.



Gambar 4.32 *Activity Diagram Hapus Data Pengguna*

4.3.31. *Activity Diagram* Informasi Aplikasi *Admin*

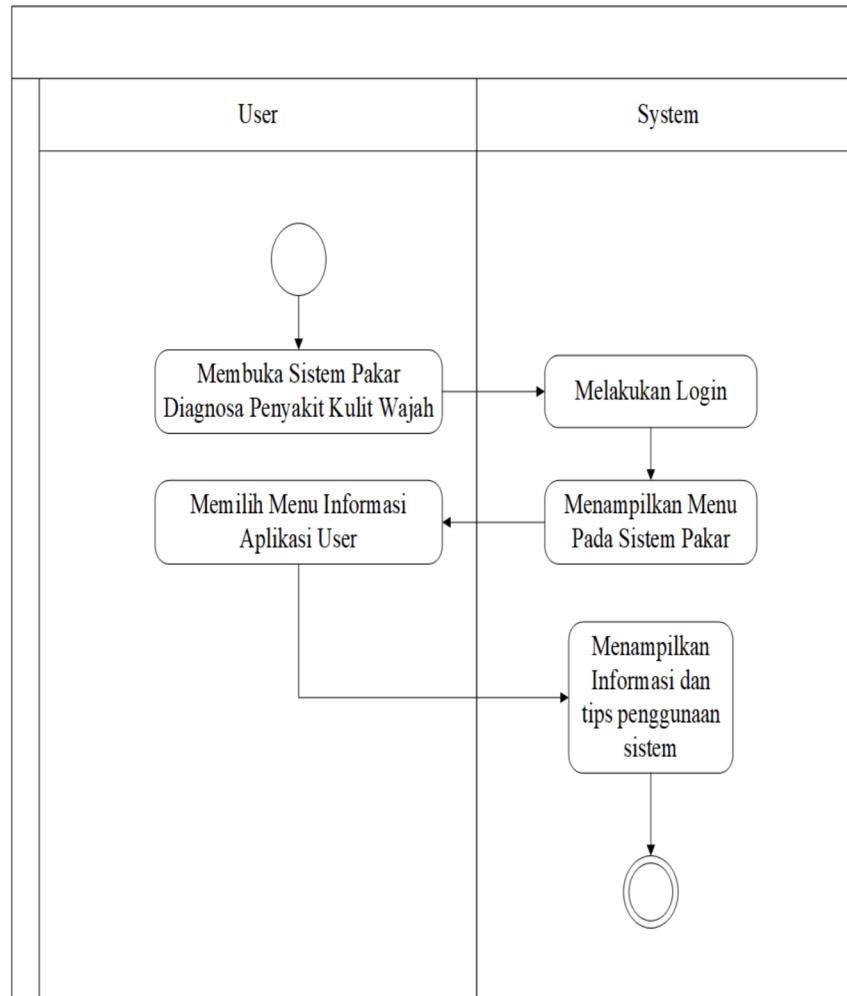
Activity diagram informasi aplikasi *admin* digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* informasi aplikasi bagi *admin*. *Activity Diagram* informasi aplikasi *admin* memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* informasi aplikasi *admin* dapat dilihat pada gambar 4.33.



Gambar 4.33 Activity Diagram Informasi Aplikasi Admin

4.3.32. Activity Diagram Informasi Aplikasi User

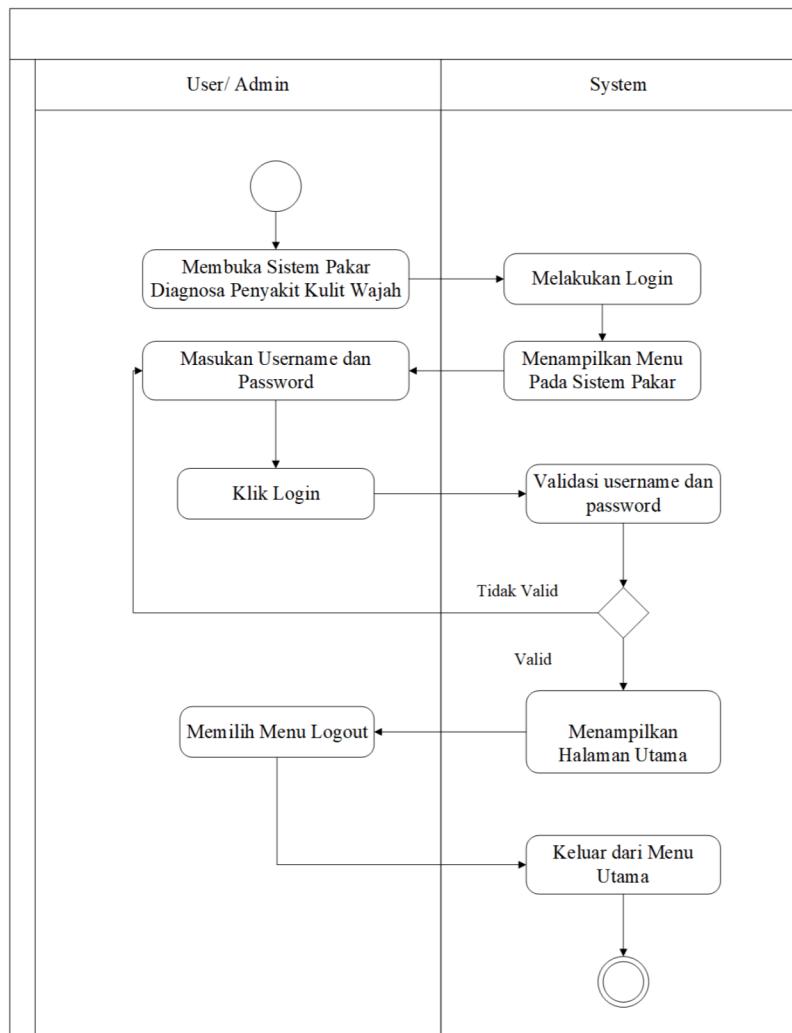
Activity diagram informasi aplikasi *user* digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur* informasi aplikasi bagi *user*. *Activity Diagram* informasi aplikasi *admin* memiliki dua *partition* yaitu *admin* dan *system*. Berikut adalah *activity diagram* informasi aplikasi *admin* dapat dilihat pada gambar 4.34.



Gambar 4.34 Activity Diagram Informasi Aplikasi User

4.3.33. Activity Diagram Logout

Activity diagram logout digunakan untuk menjelaskan urutan aktivitas atau alur program pada *fitur logout* bagi *user* ataupun *admin*. *Activity Diagram logout* memiliki dua *partition* yaitu *user/admin* dan *system*. Berikut adalah *activity diagram logout* dapat dilihat pada gambar 4.35.



Gambar 4.35 Activity Diagram Logout

4.4 Algoritme

4.4.1. Algoritme Analisis Sistem

Algoritme analisis sistem digunakan untuk menjelaskan proses analisis sistem secara terstruktur berdasarkan analisis sistem yang dilakukan. Berikut adalah algoritme analisis sistem dapat dilihat pada algoritme 4.1.

Algoritme 4.1 Analisis Sistem

1. Start
2. Membuat Basis Aturan
3. Analisa Mesin Inferensi
4. If pilih metode *forward chaining*
5. Proses analisa *forward chaining*
6. If pilih metode *certainty factor*
7. Proses perhitungan *certainty factor*
8. End If
9. Finish

4.4.2. Algoritme Sistem Pakar

Algoritme sistem pakar digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program sistem pakar yang diterapkan. Berikut adalah algoritme sistem pakar dapat dilihat pada algoritme 4.2.

Algoritme 4.2 Sistem Pakar

1. Start
2. Menginput bobot gejala
3. If pilih metode *forward chaining*
4. If Input data gejala
5. Analisa metode *forward chaining*
6. Tampil hasil analisa *forward chaining*
7. Else
8. return input data gejala
10. Else If pilih metode *certainty factor*
11. If Input bobot data gejala
12. Menghitung CF Sekuensial
13. Menghitung CF gabungan
14. Menentukan Nilai Persentase
15. Tampil Nilai Hasil Persentase
16. Else
17. return input data gejala
18. End If
19. Finish

4.4.3. Algoritme *Forward Chaining*

Algoritme *forward chaining* digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program sistem pakar yang diterapkan. Berikut adalah algoritme metode *forward chaining* dapat dilihat pada algoritme 4.3.

Algoritme 4.3 Metode *Forward Chaining*

1. Start
2. Menginput bobot gejala
3. If pilih data gejala
4. If data gejala sesuai basis aturan
5. Tampil hasil analisa sesuai basis aturan
6. Else
7. return input data gejala
8. Finish

4.4.4. Algoritme Metode *Certainty Factor*

Algoritme *certainty factor* digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program sistem pakar yang diterapkan. Berikut adalah algoritme metode *certainty factor* dapat dilihat pada algoritme 4.4.

Algoritme 4.4 Metode *Certainty Factor*

1. Program Start
2. Input data gejala
3. If Input bobot gejala
4. Menghitung CF sekuensial
$$CF(pararel) = CF(user) \times CF(pakar)$$
5. Menghitung CF gabungan
$$CF(CF1, CF2) = CF1 + CF2 \times (1 - CF1)$$
6. Tentukan Nilai iterasi perhitungan terakhir
7. Tampilkan Nilai iterasi perhitungan terakhir
8. CF Hasil * 100%
9. Else
10. return input data gejala
11. End If
12. Finish

4.4.5. Algoritme Perhitungan *Certainty Factor*

Algoritme perhitungan *certainty factor* pada dasarnya digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur perhitungan metode tersebut pada aplikasi sistem pakar. Berikut adalah algoritme perhitungan *certainty factor* dapat dilihat pada algoritme 4.5.

Algoritme 4.5 Perhitungan *Certainty Factor*

1. Input CF user, CF gejala
2. Select tabel gejala

```

3. Query tabel gejala
4. Inisialisasi variable Idgejala, CF user, CF gejala , CF Hasil
5. If Id gejala == CF user
6.     CF Hasil = CF gejala * CF_user
7.     If Jumlah gejala >=0
8.         CF Gabungan = CF Gabungan + (CF Hasil *
9.             (1- CF Gabungan ))
10.    Else
11.        CF Gabungan = CF Hasil
12.    End If
13.    Id gejala++
14.    Return Perhitungan
15. Else
16.     Return Input CF user
17. End If
18. CF Persentase = CF Gabungan * 100

```

4.4.6. Algoritme Menu Utama *Admin*

Algoritme menu utama *admin* digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program menu utama pada aplikasi sistem pakar baik disisi *user* maupun *admin*. Berikut adalah algoritma menu utama dapat dilihat pada algoritme 4.6.

Algoritme 4.6 Menu Utama *Admin*

```

1. Program Start
2. Pilih Menu Login
3. Login Aplikasi
4. Masuk Menu Utama
5. If Pilih Menu Data Gejala
6.     Tampil Menu Data Gejala
7. Else If Pilih Menu Data Penyakit
8.     Tampil Menu Data Penyakit
9. Else If Pilih Menu Data Aturan
10.    Tampil Menu Data Aturan
11. Else If Pilih Menu Pengguna
12.    Tampil Menu Data Pengguna
13. Else If Pilih Menu Informasi Aplikasi
14.    Tampil Menu Informasi Aplikasi
15. Else logout
16.    Tampil Menu Login
17. End If
18. Close Connection

```

4.4.7. Algoritme Daftar

Algoritme daftar digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program daftar pada aplikasi sistem pakar. Berikut adalah algoritma daftar dapat dilihat pada algoritme 4.7.

Algoritme 4.7 Daftar

1. Program start
2. Pilih menu daftar
3. Input form daftar
4. If Input !== 0
5. Pilih Daftar
6. Data berhasil disimpan
7. Else
8. Return Input form daftar
9. End If
10. Close connection

4.4.8. Algoritme *Login*

Algoritme *login* digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program *login* pada aplikasi sistem pakar. Berikut adalah algoritma login dapat dilihat pada algoritme 4.8.

Algoritme 4.8 *Login*

1. Program start
2. Pilih menu *login*
3. Input *form login*
4. If Input !== 0
5. Pilih *login*
6. Berhasil masuk *dashboard*
7. Else
8. Return Input *form login*
9. End If
10. Close connection

4.4.9. Algoritme Diagnosa Cepat

Algoritme diagnosa cepat digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program diagnosa cepat pada aplikasi sistem pakar. Berikut adalah algoritma diagnosa cepat dapat dilihat pada algoritme 4.9.

Algoritme 4.9 Diagnosa Cepat

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu diagnosa cepat
5. If Input gejala
6. Tampil hasil diagnosa
7. Data hasil diagnosa tersimpan
8. Else
9. Return input gejala
10. End If
11. Close connection

4.4.10. Algoritme Diagnosa Akurat

Algoritme diagnosa akurat digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program diagnosa akurat pada aplikasi sistem pakar. Berikut adalah algoritme diagnosa akurat dapat dilihat pada algoritme 4.10.

Algoritme 4.10 Diagnosa Akurat

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu diagnosa akurat
5. If Input bobot gejala
6. Tampil hasil persentase diagnosa
7. Data hasil diagnosa tersimpan
8. Else
9. Return input gejala
10. End If
11. Close connection

4.4.11. Algoritme Daftar Riwayat Diagnosa

Algoritme riwayat diagnosa digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program riwayat diagnosa pada aplikasi sistem pakar. Berikut adalah algoritme riwayat diagnosa dapat dilihat pada algoritme 4.11.

Algoritme 4.11 Daftar Riwayat Diagnosa

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu riwayat diagnosa
5. Select tabel riwayat

- | |
|---|
| <ol style="list-style-type: none"> 6. Query tabel riwayat 7. Tampil riwayat diagnosa 8. Close connection |
|---|

4.4.12. Algoritme Hapus Daftar Riwayat Diagnosa

Algoritme hapus riwayat diagnosa digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program hapus riwayat diagnosa pada aplikasi sistem pakar. Berikut adalah algoritme hapus riwayat diagnosa dapat dilihat pada algoritme 4.12.

Algoritme 4.12 Hapus Riwayat Diagnosa

- | |
|---|
| <ol style="list-style-type: none"> 1. Program start 2. Pilih menu <i>login</i> 3. Login aplikasi 4. Pilih menu riwayat diagnosa 5. Select tabel riwayat 6. Query tabel riwayat 7. If Pilih hapus riwayat diagnosa 8. Riwayat diagnosa terhapus 9. Else 10. Tampil riwayat diagnosa 11. End If 12. Close connection |
|---|

4.4.13. Algoritme Daftar Data Gejala

Algoritme daftar data gejala digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program data gejala pada aplikasi sistem pakar. Berikut adalah algoritme data gejala dapat dilihat pada algoritme 4.13.

Algoritme 4.13 Daftar Data Gejala

- | |
|---|
| <ol style="list-style-type: none"> 1. Program start 2. Pilih menu <i>login</i> 3. Login aplikasi 4. Pilih menu data gejala 5. Select tabel gejala 6. Query tabel gejala 7. Tampil data gejala 8. Close connection |
|---|

4.4.14. Algoritme Tambah Data Gejala

Algoritme tambah data gejala digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program

tambah data gejala pada aplikasi sistem pakar. Berikut adalah algoritme tambah data gejala dapat dilihat pada algoritme 4.14.

Algoritme 4.14 Tambah Data Gejala

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu data gejala
5. Select tabel gejala
6. Query tabel gejala
7. If Pilih tambah gejala
8. If Input form data gejala
9. Data gejala berhasil ditambahkan
10. Else
11. Return input form data gejala
12. Else
13. Tampil daftar data gejala
14. End If
15. Close connection

4.4.15. Algoritme Ubah Data Gejala

Algoritme ubah data gejala digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program ubah data gejala pada aplikasi sistem pakar. Berikut adalah algoritme ubah data gejala dapat dilihat pada algoritme 4.15.

Algoritme 4.15 Ubah Data Gejala

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu data gejala
5. Select tabel gejala
6. Query tabel gejala
5. If Pilih detail data gejala
6. Tampil detail data gejala
8. Else If Pilih ubah gejala
9. Gejala berhasil diubah
10. Else
11. Tampil daftar data gejala
12. End If
13. Close connection

4.4.16. Algoritme Hapus Data gejala

Algoritme hapus data gejala digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program hapus data

gejala pada aplikasi sistem pakar. Berikut adalah algoritme hapus data gejala dapat dilihat pada algoritme 4.16.

Algoritme 4.16 Hapus Data Gejala

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu data gejala
5. Select tabel gejala
6. Query tabel gejala
7. Pilih ubah data gejala
8. If Pilih hapus data gejala
9. Data gejala berhasil terhapus
10. Else
11. Tampil detail data gejala
12. End If
13. Close connection

4.4.17. Algoritme Daftar Data Penyakit

Algoritme daftar data penyakit digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program daftar data penyakit pada aplikasi sistem pakar. Berikut adalah algoritme daftar data penyakit dapat dilihat pada algoritme 4.17.

Algoritme 4.17 Daftar Data Penyakit

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu data penyakit
5. Select tabel penyakit
6. Query tabel penyakit
7. Tampil data penyakit
8. Close connection

4.4.18. Algoritme Detail Data Penyakit

Algoritme detail data penyakit digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program detail data penyakit pada aplikasi sistem pakar. Berikut adalah algoritme detail data penyakit dapat dilihat pada algoritme 4.18.

Algoritme 4.18 Daftar Data Penyakit

1. Program start
2. Pilih menu *login*
3. Login aplikasi

4. Pilih menu data penyakit
5. Select tabel penyakit
6. Query tabel penyakit
7. Tampil daftar data penyakit
8. IF pilih salah satu data penyakit
9. Tampil detail data penyakit terpilih
10. Else
11. Tampil Daftar Data Penyakit
12. Close connection

4.4.19. Algoritme Tambah Data Penyakit

Algoritme tambah data penyakit digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program tambah data penyakit pada aplikasi sistem pakar. Berikut adalah algoritme tambah data penyakit dapat dilihat pada algoritme 4.19.

Algoritme 4.19 Tambah Data penyakit

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu data penyakit
5. Select tabel penyakit
6. Query tabel penyakit
5. If Pilih tambah penyakit
6. If Input form data penyakit
7. Data penyakit berhasil ditambahkan
8. Else
9. Return input form data penyakit
10. Else
11. Tampil daftar data penyakit
12. End If
13. Close connection

4.4.20. Algoritme Ubah Data Penyakit

Algoritme ubah data penyakit digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program ubah data penyakit pada aplikasi sistem pakar. Berikut adalah algoritme ubah data penyakit dapat dilihat pada algoritme 4.20.

Algoritme 4.20 Ubah Data Penyakit

1. Program start
2. Pilih menu *login*

3. Login aplikasi
4. Pilih menu data penyakit
5. Select tabel penyakit
6. Query tabel penyakit
5. If Pilih detail data penyakit
 6. Tampil detail data penyakit
8. Else If Pilih ubah penyakit
 9. Penyakit berhasil diubah
10. Else
 11. Tampil daftar data penyakit
12. End If
13. Close connection

4.4.21. Algoritme Hapus Data Penyakit

Algoritme hapus data penyakit digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program hapus data penyakit pada aplikasi sistem pakar. Berikut adalah algoritme hapus data penyakit dapat dilihat pada algoritme 4.21.

Algoritme 4.21 Hapus Data Penyakit

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu data penyakit
5. Select tabel penyakit
6. Query tabel penyakit
7. Pilih ubah data penyakit
8. If Pilih hapus data penyakit
 9. Data penyakit berhasil terhapus
10. Else
 11. Tampil detail data penyakit
12. End If
13. Close connection

4.4.22. Algoritme Daftar Data Aturan

Algoritme daftar data aturan digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program daftar data aturan pada aplikasi sistem pakar. Berikut adalah algoritme daftar data aturan dapat dilihat pada algoritme 4.22.

Algoritme 4.22 Daftar Data Aturan

1. Program start
2. Pilih menu *login*
3. Login aplikasi

4. Pilih menu data aturan
5. Select tabel aturan
6. Query tabel aturan
7. Tampil data aturan
8. Close connection

4.4.23. Algoritme Tambah Data Aturan

Algoritme tambah data aturan digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program tambah data aturan pada aplikasi sistem pakar. Berikut adalah algoritme tambah data aturan dapat dilihat pada algoritme 4.23.

Algoritme 4.23 Tambah Data Aturan

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu data aturan
5. Select tabel aturan
6. Query tabel aturan
5. If Pilih tambah aturan
6. If Input data aturan
 Data aturan berhasil ditambahkan
8. Else
9. Return input data aturan
10. Else
11. Tampil daftar data aturan
12. End If
13. Close connection

4.4.24. Algoritme Ubah Data Aturan

Algoritme ubah data aturan digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program ubah data aturan pada aplikasi sistem pakar. Berikut adalah algoritme ubah data aturan dapat dilihat pada algoritme 4.24.

Algoritme 4.24 Ubah Data Aturan

1. Program start
2. Pilih menu *login*
3. Login aplikasi
4. Pilih menu data aturan
5. Select tabel aturan
6. Query tabel aturan
5. If Pilih detail data aturan
6. Tampil detail data aturan

- ```

8. Else If Pilih ubah aturan
9. Aturan berhasil diubah
10. Else
11. Tampil daftar data aturan
12. End If
13. Close connection

```

#### 4.4.25. Algoritme Daftar Data Pengguna

Algoritme daftar data pengguna digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program daftar data pengguna pada aplikasi sistem pakar. Berikut adalah algoritme daftar data pengguna dapat dilihat pada algoritme 4.25.

**Algoritme 4.25 Daftar Data Pengguna**

- ```

1. Program start
2 Pilih menu login
3. Login aplikasi
4. Pilih menu data pengguna
5. Select tabel pengguna
6. Query tabel pengguna
7. Tampil data pengguna
8. Close connection

```

4.4.26. Algoritme Tambah Data Pengguna

Algoritme tambah data pengguna digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program tambah data pengguna pada aplikasi sistem pakar. Berikut adalah algoritme tambah data pengguna dapat dilihat pada algoritme 4.26.

Algoritme 4.26 Tambah Data Pengguna

- ```

1. Program start
2 Pilih menu login
3. Login aplikasi
4. Pilih menu data pengguna
5. Select tabel pengguna
6. Query tabel pengguna
7. If Pilih tambah pengguna
8. If Input form data pengguna
9. Data pengguna berhasil ditambahkan
10. Else
11. Return input form data pengguna
12. Else
13. Tampil daftar data pengguna
14. End If

```

|                      |
|----------------------|
| 15. Close connection |
|----------------------|

#### 4.4.27. Algoritme Ubah Data Pengguna

Algoritme ubah data pengguna digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program ubah data pengguna pada aplikasi sistem pakar. Berikut adalah algoritme ubah data pengguna dapat dilihat pada algoritme 4.27.

##### **Algoritme 4.27 Ubah Data Pengguna**

- |                                   |
|-----------------------------------|
| 1. Program start                  |
| 2. Pilih menu <i>login</i>        |
| 3. Login aplikasi                 |
| 4. Pilih menu data pengguna       |
| 5. Select tabel pengguna          |
| 6. Query tabel pengguna           |
| 7. If Pilih detail data pengguna  |
| 8. Tampil detail data pengguna    |
| 9. Else If Pilih ubah pengguna    |
| 10. Data pengguna berhasil diubah |
| 11. Else                          |
| 12. Tampil daftar data pengguna   |
| 13. End If                        |
| 14. Close connection              |

#### 4.4.28. Algoritme Hapus Data Pengguna

Algoritme hapus data pengguna digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program hapus data pengguna pada aplikasi sistem pakar. Berikut adalah algoritme hapus data pengguna dapat dilihat pada algoritme 4.28.

##### **Algoritme 4.28 Hapus Data Pengguna**

- |                                    |
|------------------------------------|
| 1. Program start                   |
| 2. Pilih menu login                |
| 3. Login aplikasi                  |
| 4. Pilih menu data pengguna        |
| 5. Select tabel pengguna           |
| 6. Query tabel pengguna            |
| 7. Pilih ubah data pengguna        |
| 8. If Pilih hapus data pengguna    |
| 9. Data pengguna berhasil terhapus |
| 10. Else                           |
| 11. Tampil detail data pengguna    |

- |     |                  |
|-----|------------------|
| 12. | End If           |
| 13. | Close connection |

#### 4.4.29. Algoritme Informasi Aplikasi

Algoritme informasi aplikasi digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program informasi aplikasi pada aplikasi sistem pakar. Berikut adalah algoritme informasi aplikasi dapat dilihat pada algoritme 4.29.

**Algoritme 4.29 Informasi Aplikasi**

- |    |                                |
|----|--------------------------------|
| 1. | Program start                  |
| 2  | Pilih menu <i>login</i>        |
| 3. | Login aplikasi                 |
| 4. | Pilih menu informasi aplikasi  |
| 5. | Tampil menu informasi aplikasi |

#### 4.4.30. Algoritme Menu Utama *User*

Algoritme menu utama digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program menu utama pada aplikasi sistem pakar baik disisi *user*. Berikut adalah algoritma menu utama dapat dilihat pada algoritme 4.30.

**Algoritme 4.30 Menu Utama *User***

- |     |                                       |
|-----|---------------------------------------|
| 1.  | Program Start                         |
| 2.  | Pilih Menu <i>Login</i>               |
| 3.  | Login Aplikasi                        |
| 4.  | Masuk Menu Utama                      |
| 5.  | If Pilih Menu Diagnosa Cepat          |
| 6.  | Tampil Menu Diagnosa Cepat            |
| 7.  | Else If Pilih Menu Diagnosa Akurat    |
| 8.  | Tampil Menu Diagnosa Akurat           |
| 9.  | Else If Pilih Menu Daftar Penyakit    |
| 10. | Tampil Menu Daftar Penyakit           |
| 11. | Else If Pilih Menu Riwayat Diagnosa   |
| 12. | Tampil Menu Riwayat Diagnosa          |
| 13. | Else If Pilih Menu Informasi Aplikasi |
| 14. | Tampil Menu Informasi Aplikasi        |
| 15. | Else <i>logout</i>                    |
| 16. | Tampil Menu <i>Login</i>              |
| 17. | End If                                |
| 18. | Close Connection                      |

#### 4.4.31. Algoritme *Logout*

Algoritme *logout* digunakan untuk menjelaskan proses sistem pakar secara terstruktur berdasarkan alur program *logout* pada aplikasi sistem pakar . Berikut adalah algoritma *logout* dapat dilihat pada algoritme 4.31.

##### **Algoritme 4.31 *Logout***

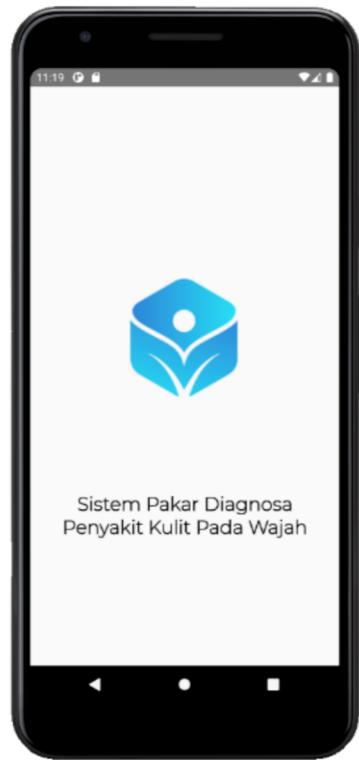
1. Program start
2. Pilih menu *login*
3. Input *form login*
4. If Input != 0
5. Pilih *login*
6. Berhasil masuk *dashboard*
7. If Pilih menu *logout*
8. Berhasil *logout*
9. Else
10. Tetap dihalaman *dashboard*
11. Else
12. Return input *form login*
13. End If
14. Close connection

### 4.5 Implementasi Sistem

Setelah melewati beberapa tahapan yang sudah di jelaskan pada bab sebelumnya. Setelah semua data siap dan dapat digunakan dalam proses pengujian. Berikut langkah-langkah pengujian yang dilakukan guna mengevaluasi kinerja sistem, serta menjelaskan tiap tampilan layar sistem antara lain sebagai berikut.

#### 4.5.1 Tampilan Layar *Splash Screen*

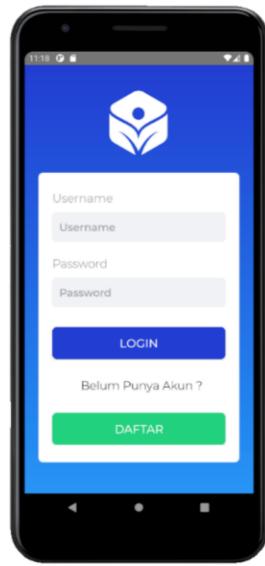
Gambar 4.36 menunjukkan tampilan layar *splash screen*. Tujuan *splash screen* untuk menginformasikan kepada pengguna bahwa aplikasi sudah mulai dijalankan. Saat pertama kali aplikasi dipilih maka *splash screen* akan muncul. Didalam *splash screen* terdapat terdapat *logo* aplikasi dan nama aplikasi ini yaitu “Sistem Pakar Diagnosis Penyakit Kulit Wajah”.



**Gambar 4.36 Tampilan Layar *Splash Screen***

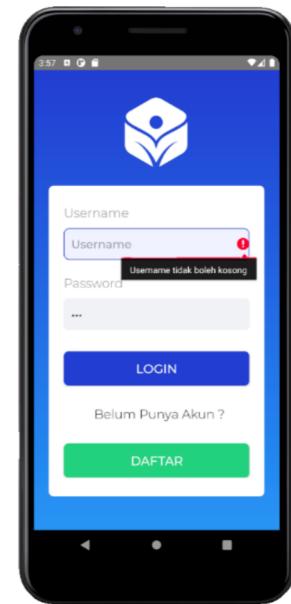
#### 4.5.2 Tampilan Layar *Login*

Pada gambar 4.37 merupakan tampilan layar halaman *login*. Tujuan login adalah agar *admin* dan *user* memiliki hak akses untuk bisa masuk ke menu masing-masing *role*. Didalam *login* terdapat terdapat *form username* dan *password*. Jika *admin* yang *login* maka akan menuju ke *dashboard admin* dan jika *user* yang *login* akan masuk ke *dashboard user*. Jika *user* belum memiliki akun harus melakukan registrasi terlebih dahulu dengan cara klik tombol “Daftar”. Jika sudah memiliki akun silahkan *login* dengan mengetikkan *username* dan *password* yang benar.



Gambar 4.37 Tampilan Layar *Login*

Jika anda tidak menginput *password* ataupun *username* maka akan muncul *error* seperti gambar 4.38.

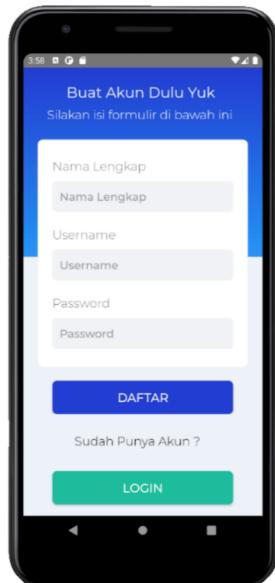


Gambar 4.38 Tampilan Layar *Login Error*

#### 4.5.3

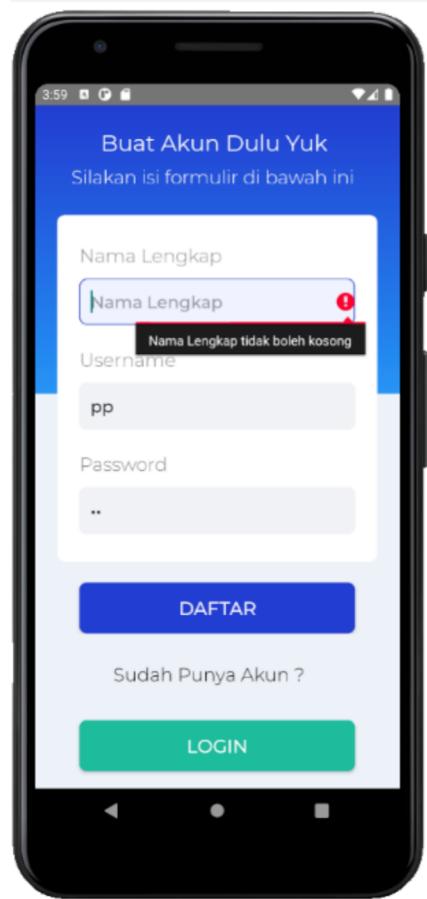
#### Tampilan Layar Daftar

Pada gambar 4.39 merupakan tampilan layar halaman Daftar. Tujuan halaman daftar ini adalah untuk mendapatkan akses sebagai *user* menggunakan aplikasi ini. Didalam halaman daftar terdapat 3 buah *form* diantaranya adalah *form* nama lengkap, *form* *username* dan *form* *password* yang harus diisi guna mendapatkan hak akses sebagai *user* nantinya. Disini kita bisa mendaftar sebagai *user* dengan mengisikan nama, *username* dan *password*. Kemudian terdapat informasi pada layar berupa teks “Buat Akun Dulu Yuk Silahkan isi formulir di bawah ini”. Jika berhasil akan ada *pop up* bahwa anda telah berhasil registrasi dan akan langsung kembali ke halaman *Login*.



**Gambar 4.39 Tampilan Layar Daftar**

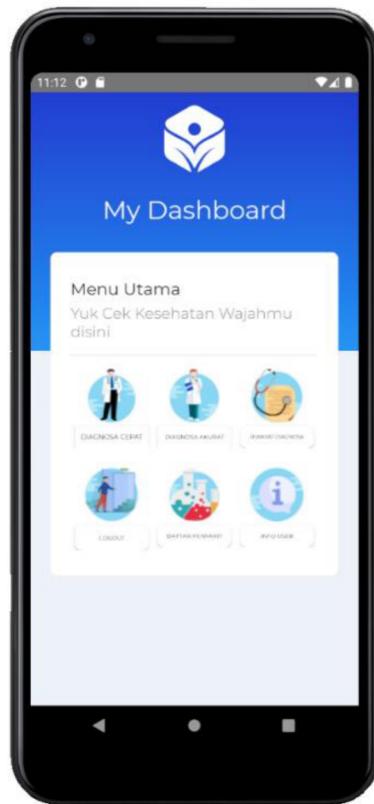
Jika anda tidak mengisi salah satu atau semuanya maka akan muncul *error*. Seperti gambar 3.8 pastikan sudah diisi dengan lengkap serta pastikan juga seluruh identitas yang berada pada tampilan tersebut harus terisi dengan baik dan benar



Gambar 4.40 Tampilan Layar Daftar Error

#### 4.5.4 Tampilan Layar Dashboard User

Pada gambar 4.41 adalah tampilan layar halaman *dashboard user* atau tampilan utama, tampilan ini yang akan muncul pertama kali ketika *user* berhasil *login* terdapat beberapa tombol pilihan untuk pindah ke menu lain seperti, diagnosa cepat, diagnosa akurat, data penyakit, daftar riwayat diagnosa, panduan aplikasi untuk *user* dan *logout*. Tujuan *dashboard user* ini adalah sebagai navigasi bagi *user* diaplikasi ini.



**Gambar 4.41 Tampilan Layar Dashboard User**

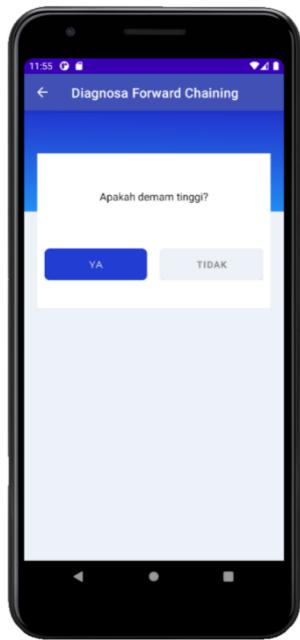
#### 4.5.5

#### Tampilan Layar Diagnosa Cepat

Tampilan layar diagnosa cepat digunakan *user* mendiagnosa kemungkinan penyakit kulit wajah yang dialami pasien dengan metode *forward chaining*. Tampilan layar diagnosa ini terbagi menjadi dua tampilan, yaitu tampilan pemilihan gejala dan hasil diagnosa.

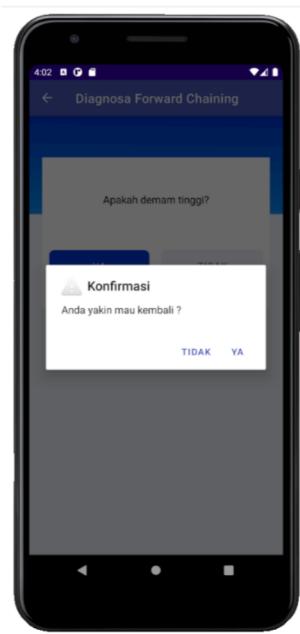
##### a. Tampilan Layar Pemilihan Gejala Diagnosa Cepat

Pada gambar 4.42 merupakan halaman diagnosa atau pemilihan gejala, merupakan proses pemilihan gejala yang dialami oleh pasien. Pada halaman ini akan menampilkan daftar gejala yang telah diinputkan pakar sebelumnya dan kita tinggal memilih gejala tersebut berdasarkan “Ya” atau “Tidak”.



**Gambar 4.42 Tampilan Layar Pemilihan Gejala Diagnosa Cepat**

Jika anda salah dalam mengklik tombol kembali pada *android* maka akan muncul dialog seperti gambar 4.43.



**Gambar 4.43 Tampilan Layar Keluar Pemilihan Diagnosa Cepat**

#### b. Tampilan Layar Hasil Diagnosa Cepat

Gambar 4.44 merupakan tampilan layar halaman hasil diagnosa yang merupakan hasil diagnosa dari pemilihan gejala yang diinputkan sebelumnya.



**Gambar 4.44 Tampilan Layar Hasil Diagnosa Cepat**

Tujuan hasil diagnosa ini adalah sebagai informasi bagi *user* yang telah melakukan diagnosa cepat dengan aplikasi ini. Pada Hasil diagnosa terdapat *button* dan teks dari hasil diagnosa .Pada layar hasil diagnosa cepat juga terdapat *button my dashboard* yang berfungsi untuk kembali ke halaman *dashboard user*

#### 4.5.6 Tampilan Layar Diagnosa Akurat

Tampilan layar diagnosa akurat digunakan *user* mendiagnosa kemungkinan penyakit kulit wajah yang dialami pasien dengan metode *certainty factor* seperti yang tertera pada *action bar* pada tampilan layar. Tampilan layar diagnosa ini terbagi menjadi dua tampilan, yaitu tampilan pemilihan gejala dan hasil diagnosa.

##### a. Tampilan Layar Pemilihan Gejala Diagnosa Akurat

Pada gambar 4.45 merupakan halaman diagnosa atau pemilihan gejala, merupakan proses pemilihan gejala yang dialami oleh pasien.Pada halaman ini akan menampilkan daftar

gejala yang telah diinputkan pakar sebelumnya. Pada halaman ini terdapat *button* pilihan gejala yang akan diisi berdasarkan tingkat kepercayaan gejala *user*.



**Gambar 4.45 Tampilan Layar Pemilihan Gejala Diagnosa Akurat**

Jika anda memilih icon kembali pada *action bar* pada android maka akan muncul *dialog* seperti gambar 4.46.



**Gambar 4.46 Tampilan Layar Keluar Diagnosa Akurat**

### b. Tampilan Layar Hasil Diagnosa Akurat

Tampilan gambar 4.47 merupakan desain halaman hasil diagnosa yang merupakan hasil diagnosa dari pemilihan gejala yang diinputkan sebelumnya.

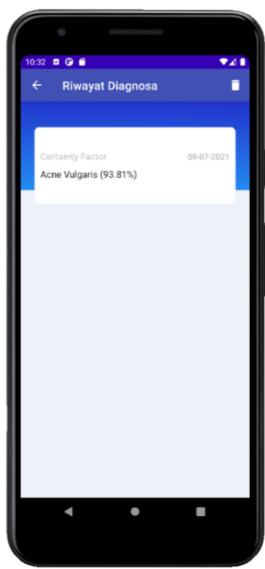


**Gambar 4.47 Tampilan Layar Keluar Diagnosa Akurat**

Tujuan hasil diagnosa ini adalah sebagai informasi bagi *user* yang telah melakukan diagnosa akurat dengan aplikasi ini. Pada Hasil diagnosa terdapat *button* dan teks dari hasil diagnosa berserta persentasenya. Pada layar hasil diagnosa akurat juga terdapat *button my dashboard* yang berfungsi untuk kembali ke halaman *dashboard user*

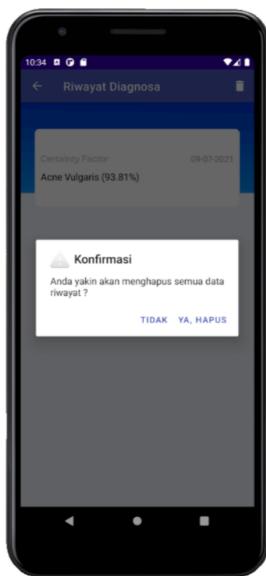
#### 4.5.7 Tampilan Layar Riwayat Hasil Diagnosa

Gambar 4.48 merupakan tampilan halaman riwayat diagnosa yaitu berupa halaman yang memunculkan riwayat hasil diagnosa yang telah dilakukan oleh *user* dari *fitur* diagnosa cepat maupun diagnosa akurat seperti kode penyakit kulit wajah. Penyakit kulit wajah, nilai atau tingkat kepercayaan, solusi dan waktu *user* melakukan diagnosa. Pada tampilan layar riwayat hasil diagnosa ini terdapat tampilan layar daftar riwayat hasil diagnosa dan tampilan layar hapus riwayat hasil diagnosa yang ada pada gambar 4.48 dan gambar 4.49.



**Gambar 4.48 Tampilan Layar Riwayat Hasil Diagnosa**

Jika anda salah mengklik tombol hapus pada *action bar* maka akan muncul diaglog seperti gambar 4.49.



**Gambar 4.49 Tampilan Layar Hapus Riwayat Hasil Diagnosa**

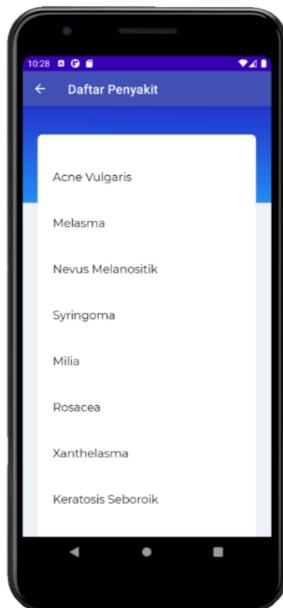
#### 4.5.8 Tampilan Layar Daftar Penyakit

Tampilan layar Daftar Penyakit berisi seluruh daftar penyakit yang berkaitan dengan aplikasi sistem pakar diagnosis penyakit kulit wajah yang digunakan *user* untuk mengetahui penyakit dan

gejala yang berkaitan serta solusi awal untuk menangani penyakit tersebut. Tampilan layar diagnosa ini terbagi menjadi dua tampilan, yaitu tampilan daftar data penyakit dan detail penyakit yang berisi solusi dan penanganannya.

a. Tampilan Layar Daftar Data Penyakit *User*

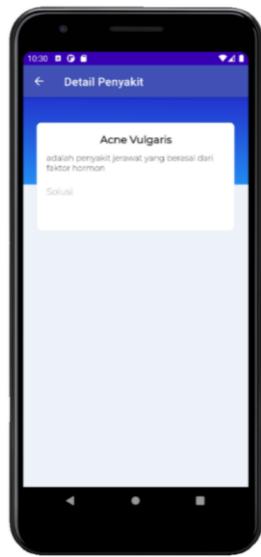
Pada gambar 4.50 merupakan tampilan layar daftar data penyakit, merupakan daftar data penyakit berserta gejala yang berkaitan berdasarkan aplikasi sistem pakar ini. Kemudian juga terdapat kode gejala disetiap penyakit yang berkaitan. Pada halaman ini akan menampilkan daftar penyakit dan gejala yang berkaitan sesuai arahan dan inputan dari pakar .



**Gambar 4.50 Tampilan Layar Daftar Data Penyakit *User***

b. Tampilan Layar Detail Data Penyakit

Pada gambar 4.51 merupakan halaman detail data penyakit yang berisi informasi data penyakit berserta gejala dan kode gejalanya berserta deskripsi dari penyakit tersebut yang terdapat solusi atau penanganan awal dari penyakit tersebut.



**Gambar 4.51 Tampilan Layar Detail Data Penyakit**

#### 4.5.9 Tampilan Layar Informasi *User*

Gambar 4.52 merupakan tampilan layar informasi aplikasi yaitu berupa halaman yang memberikan informasi aplikasi berupa *text*. Tujuan halaman informasi aplikasi ini adalah untuk memberikan arahan kepada *user* dalam menggunakan aplikasi.

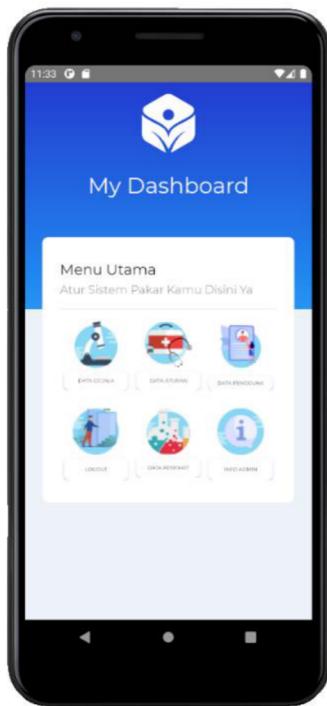


**Gambar 4.52 Tampilan Layar Informasi *User***

#### 4.5.10 Tampilan Layar *Dashboard Admin*

Pada gambar 4.53 merupakan tampilan layar isi dari menu pakar atau *admin* yaitu *Dashboard admin*, data gejala, data penyakit, data aturan, data pengguna serta menu panduan pengguna yang ditunjukkan untuk *admin*. Pada halaman *dashboard admin* berisi informasi berupa teks “Atur Sistem Pakar Kamu Disini Ya”.

Di halaman gejala, kelompok gejala, data penyakit, data aturan juga data pengguna bisa melakukan CRUD atau *Create, Read, Update, dan Delete* pada masing-masing halaman menunya.



Gambar 4.53 Tampilan Layar *Dashboard Admin*

#### 4.5.11 Tampilan Layar Daftar Data Gejala

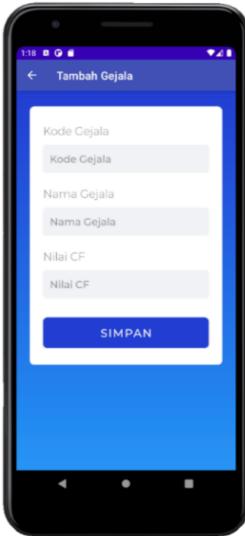
Pada tampilan layar halaman ini akan ditampilkan gejala yang diinputkan, *admin* dapat melihat dan meninjau data-data gejala yang ada. Tujuan halaman data gejala adalah untuk memfasilitasi *admin* dalam mengelola data-data gejala pada sistem pakar ini. Pada halaman data gejala terdapat tampilan daftar data-data gejala apa saja yang telah diinputkan oleh *admin*. Berikut gambar 4.54.



Gambar 4.54 Tampilan Layar Daftar Data Gejala

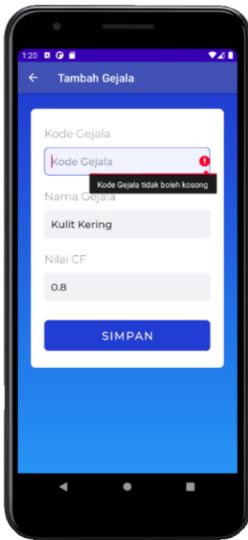
#### 4.5.12 Tampilan Layar Tambah Data Gejala

Pada tampilan layar ini *admin* dapat menambahkan data-data gejala pada aplikasi ini. Tujuan halaman tambah data gejala adalah untuk menambahkan data-data gejala yang berhubungan dengan penyakit. sehingga nantinya bisa membentuk *rule* aplikasi pada sistem pakar ini. Berikut gambar 4.55.



Gambar 4.55 Tampilan Layar Tambah Data Gejala

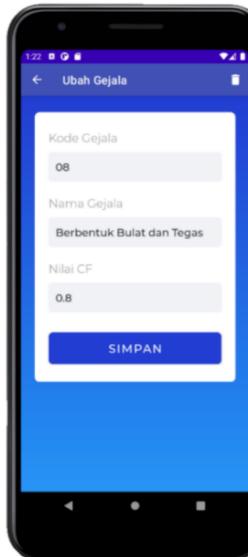
Jika anda tidak menginput salah satu *form* tersebut maka akan muncul *error* seperti gambar 4.56.



**Gambar 4.56 Tampilan Layar Error Tambah Data Gejala**

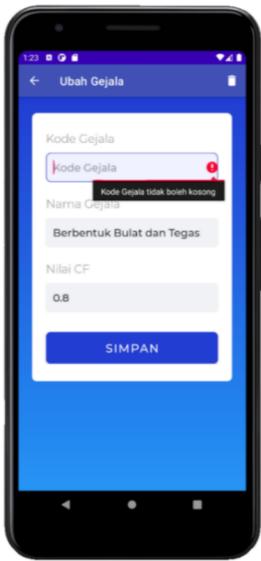
#### 4.5.13 Tampilan Layar Ubah Data Gejala

Pada halaman ini *admin* dapat mengubah data gejala dengan mengganti data *form* kode gejala, nama gejala. nilai CF yang tersedia pada halaman data gejala.Berikut gambar 4.57.



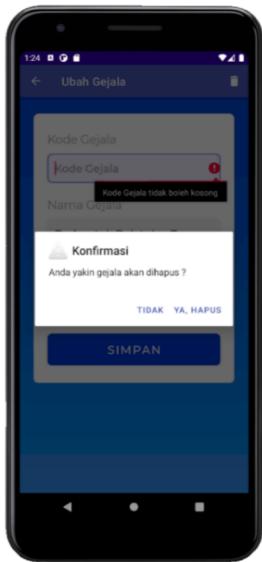
**Gambar 4.57 Tampilan Layar Ubah Data Gejala**

Jika anda tidak menginput salah satu *form* tersebut maka akan muncul error seperti gambar 4.58.



Gambar 4.58 Tampilan Layar Ubah Data Gejala *Error*

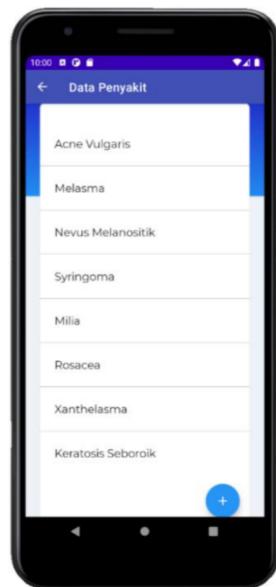
Jika anda memilih *icon* pada *action bar* hapus salah satu *form* tersebut maka akan muncul konfirmasi seperti gambar 4.59.



Gambar 4.59 Tampilan Layar Konfirmasi Hapus Gejala

#### 4.5.14 Tampilan Layar Daftar Data Penyakit

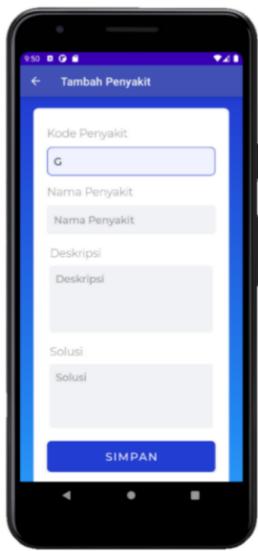
Pada tampilan layar ini akan ditampilkan penyakit yang diinputkan, *admin* dapat melihat dan meninjau data-data penyakit yang ada. Tujuan halaman data penyakit adalah untuk memfasilitasi *admin* dalam mengelola data-data penyakit pada sistem pakar ini. Pada halaman data penyakit terdapat tampilan daftar data-data penyakit apa saja yang telah diinputkan oleh *admin* berdasarkan arahan pakar/dokter. Pada tampilan layar daftar data penyakit terdapat *list view* data-data penyakit yang telah diinputkan dan jika untuk memuat lebih banyak penyakit yang telah diinputkan bisa melakukan *scrolling* pada layar *smartphone* sedangkan untuk menambahkan data penyakit pada tampilan layar ini bisa dengan memilih *button* berbentuk *plus*. Berikut adalah gambar 4.60.



Gambar 4.60 Tampilan Layar Daftar Data Penyakit

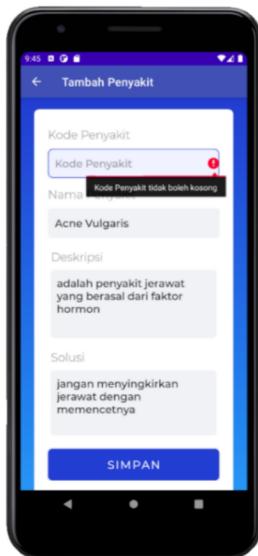
#### 4.5.15 Tampilan Layar Tambah Data Penyakit

Pada tampilan layar ini *admin* dapat menambahkan data-data penyakit pada aplikasi ini. Tujuan halaman tambah data gejala adalah untuk menambahkan data penyakit yang berhubungan dengan data-data gejala dan kemudian menghubungkannya satu sama lain. sehingga nantinya bisa membentuk *rule* aplikasi pada sistem pakar ini. Berikut gambar 4.61.



Gambar 4.61 Tampilan Layar Tambah Data Penyakit

Jika anda tidak menginput salah satu *form* tersebut maka akan muncul *error* seperti gambar 4.62.



Gambar 4.62 Tampilan Layar Error Tambah Data Penyakit

#### 4.5.16 Tampilan Layar Ubah Data Penyakit

Pada tampilan layar ini *admin* dapat mengubah data penyakit dengan mengganti data *form* kode penyakit, nama penyakit, deskripsi serta solusi penyakit tersebut. Berikut gambar 4.63.



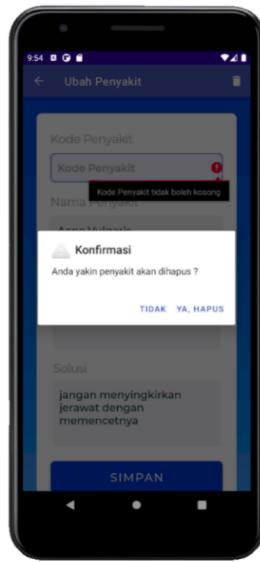
**Gambar 4.63 Tampilan Layar Ubah Data Penyakit**

Jika anda tidak menginput salah satu *form* tersebut maka akan muncul *error* seperti gambar 4.64.



**Gambar 4.64 Tampilan Layar Error Ubah Data Penyakit**

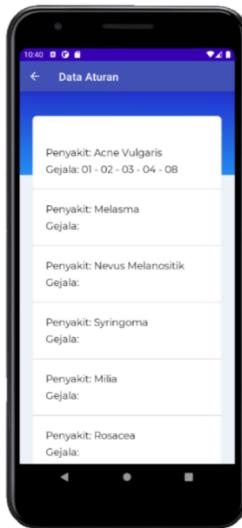
Jika anda memilih *icon* pada *action bar* hapus salah satu *form* tersebut maka akan muncul konfirmasi seperti gambar 4.65.



**Gambar 4.65 Tampilan Layar Konfirmasi Hapus Data Penyakit**

#### 4.5.17 Tampilan Layar Daftar Data Aturan

Pada tampilan layar ini akan ditampilkan aturan yang diinputkan, *admin* dapat melihat dan meninjau data-data aturan yang ada. Tujuan halaman data aturan adalah untuk memfasilitasi *admin* dalam mengelola data-data aturan berserta menentukan *rule* berdasarkan keterkaitan gejala dan penyakit pada sistem pakar ini. Pada halaman data aturan terdapat tampilan daftar data-data aturan apa saja yang telah diinputkan oleh *admin*. Berikut gambar 4.66.



**Gambar 4.66 Tampilan Layar Detail Data Aturan**

#### 4.5.18 Tampilan Layar Detail Data Aturan

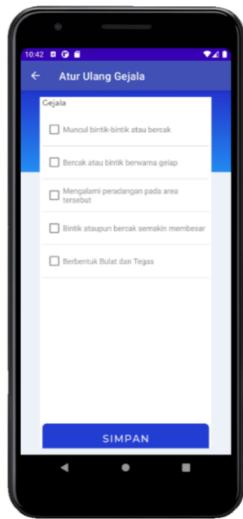
Pada tampilan layar ini akan ditampilkan aturan yang telah dipilih dari halaman daftar data aturan, *admin* dapat melihat dan meninjau data-data aturan tersebut. Tujuan halaman data aturan adalah untuk memfasilitasi *admin* dalam melihat data-data aturan berserta *rule* sudah yang keterkaitan antara gejala dan penyakit pada sistem pakar ini. Pada halaman data aturan terdapat tampilan data aturan yang terpilih dan telah diinputkan oleh *admin* serta terdapat tombol atur ulang gejala. Berikut gambar 4.67.



Gambar 4.67 Tampilan Layar Detail Data Aturan

#### 4.5.19 Tampilan Layar Ubah Data Aturan

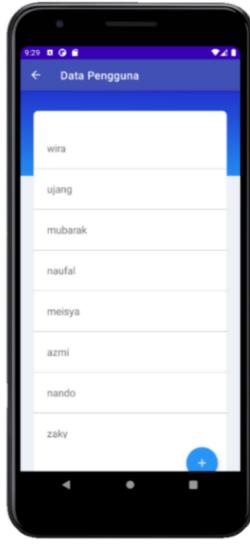
Pada halaman ini *admin* dapat mengubah data aturan dengan mengganti data gejala tersebut dan memilih atur ulang gejala untuk mengubah data .Berikut tampilan layar 4.68.



Gambar 4.68 Tampilan Layar Detail Data Aturan

#### 4.5.20 Tampilan Layar Daftar Data Pengguna

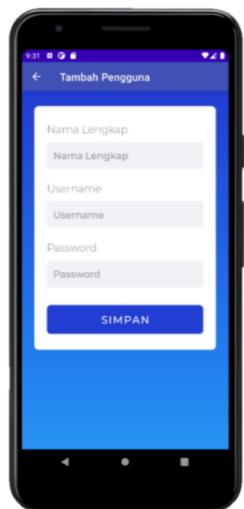
Pada tampilan layar ini akan ditampilkan pengguna yang telah terdaftar di aplikasi, *admin* dapat melihat dan meninjau data-data pengguna yang ada. Tujuan halaman data pengguna adalah untuk memfasilitasi *admin* dalam mengelola data-data pengguna pada sistem pakar ini. Pada halaman data pengguna terdapat tampilan daftar data-data pengguna yang telah menggunakan aplikasi ini. Sebagaimana gambar 4.69.



Gambar 4.69 Tampilan Layar Daftar Data Pengguna

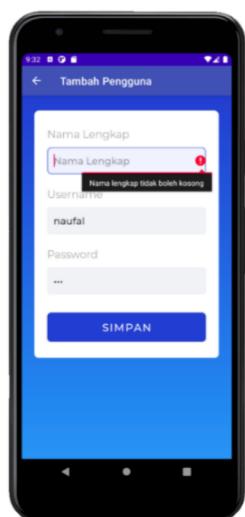
#### 4.5.21 Tampilan Layar Tambah Data Pengguna

Pada tampilan layar ini *admin* dapat menambahkan data-data pengguna pada aplikasi ini. Tujuan halaman tambah data gejala adalah untuk menambahkan data pengguna. sehingga nantinya bisa meninjau kembali pengguna diaplikasi pada sistem pakar ini. Sebagaimana gambar 4.70 berikut ini.



**Gambar 4.70 Tampilan Layar Tambah Data Pengguna**

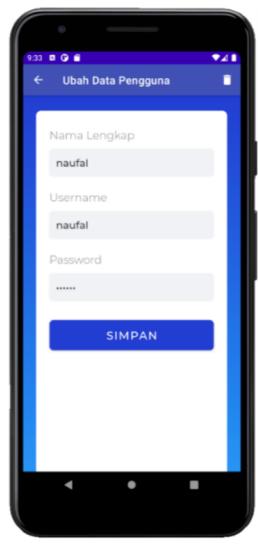
Jika anda tidak menginput salah satu *form* tersebut maka akan muncul *error* seperti gambar 4.71.



**Gambar 4.71 Tampilan Layar Error Tambah Data Pengguna**

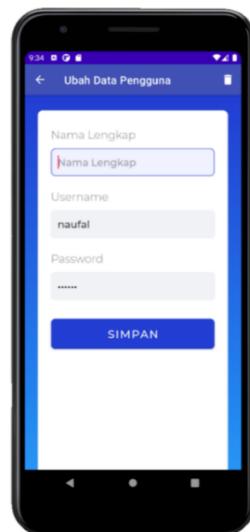
#### 4.5.22 Tampilan Layar Ubah Data Pengguna

Pada halaman ini *admin* dapat mengubah data pengguna dengan mengganti data *form* nama lengkap, *username*, password. Seperti gambar 4.72.



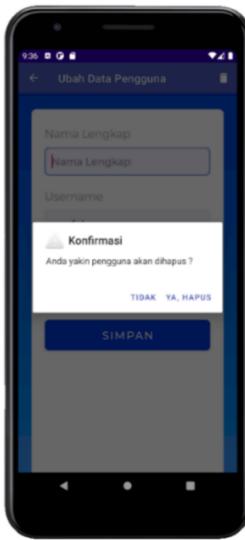
**Gambar 4.72 Tampilan Layar Ubah Data Pengguna**

Jika anda tidak menginput salah satu *form* tersebut maka akan muncul *error* seperti gambar 4.73.



**Gambar 4.73 Tampilan Layar Error Ubah Data Pengguna**

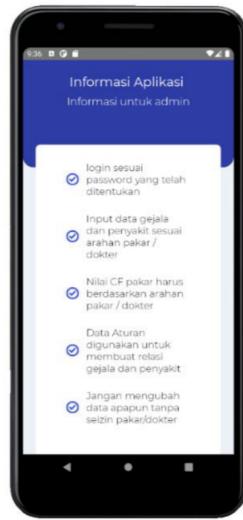
Jika anda memilih *icon* pada *action bar* hapus salah satu *form* tersebut maka akan muncul konfirmasi seperti gambar 4.74.



Gambar 4.74 Tampilan Layar Konfirmasi Hapus Data Pengguna

#### 4.5.23 Tampilan Layar Informasi Aplikasi *Admin*

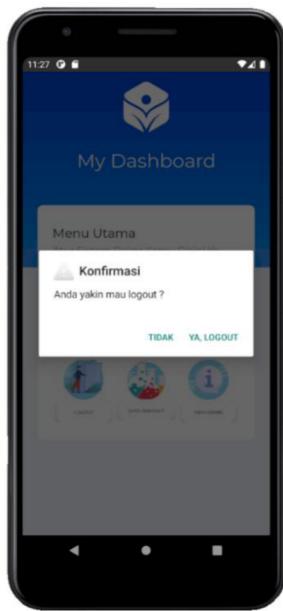
Gambar 4.75 merupakan desain informasi aplikasi yaitu berupa halaman yang memberikan informasi aplikasi berupa *text*. Tujuan halaman informasi aplikasi ini adalah untuk memberikan arahan kepada *admin* dalam menggunakan aplikasi.



Gambar 4.75 Tampilan Informasi Aplikasi *Admin*

#### 4.5.24 Tampilan Layar *Logout*

Pada gambar 4.76 merupakan desain halaman *Logout*. Dimana *admin* dan *user* memiliki hak akses untuk bisa *logout* dari menu masing-masing *role*. Jika *admin* ataupun *user* yang *logout* maka akan menuju ke halaman *login*.



Gambar 4.76 Tampilan Layar *Logout*

### 4.6 Pengujian

Berikut adalah hasil pengujian menggunakan metode *Black Box* dan Akurasi, antara lain sebagai berikut:

#### 4.6.1. *Black Box*

Pengujian dengan memakai metode *Black Box* testing adalah tahap pengujian yg memfokuskan persyaratan fungsional pada perangkat lunak. Pengujian *Black Box* Ini dirancang untuk mendeteksi kesalahan persyaratan fungsional tanpa mempengaruhi perilaku internal program asli.

Dari serangkaian uji coba membuat hasil yang cukup baik. Semua fungsi menu berfungsi seperti yang diharapkan. Secara umum sistem ini dapat diujicobakan kepada *user*.

a. Pengujian *Black Box* pada halaman *Splash*

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman *splash*. Pengujian dan hasil uji *Black Box* halaman *login* pada Tabel 4.13.

**Tabel 4.13 Hasil Pengujian *Black Box* Halaman *Login***

| Aktifitas Pengujian                                       | Output yang diharapkan                                          | Hasil  |
|-----------------------------------------------------------|-----------------------------------------------------------------|--------|
| Klik aplikasi Sistem Pakar Diagnosa Penyakit Kulit Wajah. | Menampilkan <i>Splash Screen</i> , lalu ke halaman <i>login</i> | Sukses |

b. Pengujian *Black Box* Pada Halaman *Login*

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman *login*. Pengujian dan hasil uji *Black Box* halaman *login* pada Tabel 4.14.

**Tabel 4.14 Hasil Pengujian *Black Box* Halaman *Login***

| Aktifitas Pengujian                                                                                     | Output yang diharapkan                                                                                                          | Hasil  |
|---------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------|--------|
| Menginputkan data <i>login</i> yang benar, lalu klik tombol ‘ <i>Login</i> ’.                           | Sistem menerima akses <i>login</i> dan kemudian langsung menampilkan <i>form admin/user</i> . Sesuai <i>role</i> masing-masing. | Sukses |
| Menginputkan dengan kondisi <i>password</i> yang salah, lalu mengklik tombol ‘ <i>Login</i> ’.          | Sistem akan menolak akses <i>login</i> dan menampilkan pesan “ <i>Password Salah!</i> ”.                                        | Sukses |
| Menginputkan dengan kondisi salah satu <i>form username</i> saja lalu mengklik tombol ‘ <i>Login</i> ’. | Sistem akan menolak akses <i>login</i> dan menampilkan pesan “ <i>Username tidak boleh kosong!</i> ”.                           | Sukses |
| Menginputkan dengan kondisi salah satu <i>form password</i> saja lalu                                   | Sistem akan menolak akses <i>login</i> dan menampilkan pesan “ <i>Password tidak boleh kosong!</i> ”.                           | Sukses |

|                          |                          |        |
|--------------------------|--------------------------|--------|
| mengklik tombol ‘Login’. |                          |        |
| Klik Daftar              | Menuju ke halaman Daftar | Sukses |

c. Pengujian *Black Box* Halaman Daftar

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman daftar. Pengujian dan hasil uji *Black Box* halaman daftar pada Tabel 4.15.

**Tabel 4.15 Hasil Pengujian *Black Box* Halaman Daftar**

| Aktifitas Pengujian                                                                     | Output yang diharapkan                                                                                                                   | Hasil  |
|-----------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|--------|
| Menginputkan data pendaftaran akun baru yang benar, lalu klik tombol ‘Registrasi’.      | Sistem menerima data pendaftaran akun baru dan menampilkan pesan “Pendaftaran berhasil” dan Otomatis menuju halaman <i>Login</i> .       | Sukses |
| Menginputkan dengan kondisi salah satu data kosong, lalu langsung klik tombol ‘Daftar’. | Sistem akan menolak pendaftaran dan menampilkan pesan “Nama harus di isi”, “Username tidak boleh kosong”, “Password tidak boleh kosong”. | Sukses |
| Klik tombol <i>login</i>                                                                | Menuju ke halaman <i>Login</i>                                                                                                           | Sukses |

d. Pengujian *Black Box* Pada Halaman *Dashboard User*

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman *dashboard user*. Pengujian dan hasil uji *Black Box* halaman *dashboard user* pada Tabel 4.16.

**Tabel 4.16 Hasil Pengujian Black Box Halaman Dashboard user**

| Aktifitas Pengujian          | Output yang diharapkan                                                             | Hasil  |
|------------------------------|------------------------------------------------------------------------------------|--------|
| Klik menu Informasi Aplikasi | Petunjuk penggunaan sistem pakar, dan tips penggunaan sistem.                      | Sukses |
| Klik menu Diagnosa Cepat     | Menuju ke halaman diagnosa cepat.                                                  | Sukses |
| Klik menu Diagnosa Akurat    | Menuju ke halaman diagnosa akurat.                                                 | Sukses |
| Klik menu Daftar Penyakit    | Menuju ke halaman daftarpenyakit , untuk menampilkan detail pilihan data penyakit. | Sukses |
| Klik menu logout             | Menampilkan konfirmasi “Anda Yakin Mau Logout ?”                                   | Sukses |
| Klik “Ya Logout”             | Menampilkan halaman <i>login</i> atau keluar dari <i>dashboard</i> awal            | Sukses |
| Klik “Tidak”                 | Tetap menampilkan halaman <i>dashboard</i> awal                                    | Sukses |

e. Pengujian *Black Box* pada halaman Diagnosa Cepat

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman diagnosa cepat. Pengujian dan hasil uji *Black Box* halaman diagnosa cepat pada Tabel 4.17.

**Tabel 4.17 Hasil Pengujian Black Box Halaman Diagnosa Cepat**

| Aktifitas Pengujian           | Output yang diharapkan                                        | Hasil  |
|-------------------------------|---------------------------------------------------------------|--------|
| Klik tombol “Ya” atau “Tidak” | Menampilkan pertanyaan lain dan hasil diagnosa cepat penyakit | Sukses |
| Klik Tombol Hasil diagnosa    | Menampilkan detail data penyakit terdiagnosis.                | Sukses |
| Klik Tombol Mydasboard        | Menampilkan kembali ke halaman awal <i>dashboard user</i>     | Sukses |
| Klik tombol                   | Memunculkan Konfirmasi                                        | Sukses |

|                                                                        |                                                           |        |
|------------------------------------------------------------------------|-----------------------------------------------------------|--------|
| <i>back</i> pojok kiri atas                                            | untuk kembali ke halaman <i>dashboard user</i>            |        |
| Klik tombol ‘Ya’ untuk konfirmasi Keluar dari proses diagnosa cepat    | Menampilkan kembali ke halaman awal <i>dashboard user</i> | Sukses |
| Klik tombol “Tidak” untuk konfirmasi tetap dalam proses diagnosa cepat | Tetap Menampilkan halaman diagnosa cepat                  | Sukses |

f. Pegujian *Black Box* Pada Halaman Diagnosa Akurat

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman diagnosa akurat. Pengujian dan hasil uji *Black Box* halaman diagnosa akurat pada Tabel 4.18.

**Tabel 4.18 Hasil Pengujian *Black Box* Halaman Diagnosa Akurat**

| Aktifitas Pengujian                                                 | Output yang diharapkan                                                               | Hasil  |
|---------------------------------------------------------------------|--------------------------------------------------------------------------------------|--------|
| Klik tombol opsi tingkat gejala yang dialami                        | Opsi tingkat gejala yang dialami terisi.                                             | Sukses |
| Klik Lanjut                                                         | Menampilkan pertanyaan lain dan hasil diagnosa akurat penyakit dengan persentasenya. | Sukses |
| Klik Tombol Hasil diagnosa                                          | Menampilkan detail data penyakit terdiagnosa.                                        | Sukses |
| Klik Tombol <i>Mydasboard</i>                                       | Menampilkan kembali ke halaman awal <i>dashboard user</i>                            | Sukses |
| Klik tombol <i>back</i> pojok kiri atas                             | Memunculkan konfirmasi untuk kembali ke halaman <i>dashboard user</i>                | Sukses |
| Klik tombol “Ya” untuk konfirmasi Keluar dari proses diagnosa cepat | Menampilkan kembali ke halaman awal <i>dashboard user</i>                            | Sukses |

|                                                                        |                                           |        |
|------------------------------------------------------------------------|-------------------------------------------|--------|
| Klik tombol “Tidak” untuk konfirmasi tetap dalam proses diagnosa cepat | Tetap menampilkan halaman diagnosa akurat | Sukses |
|------------------------------------------------------------------------|-------------------------------------------|--------|

g. Pengujian *Black Box* Pada Halaman Daftar Penyakit *User*

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman daftar penyakit *user*. Pengujian dan hasil uji *Black Box* halaman penyakit *user* pada Tabel 4.19.

**Tabel 4.19 Hasil Pengujian *Black Box* Halaman Daftar Penyakit *User***

| Aktifitas Pengujian                                                  | Output yang diharapkan                                    | Hasil  |
|----------------------------------------------------------------------|-----------------------------------------------------------|--------|
| Klik tombol data penyakit terpilih                                   | Menampilkan detail data penyakit tersebut.                | Sukses |
| Klik tombol <i>back</i> pojok kiri atas pada daftar penyakit         | Menampilkan kembali ke halaman awal <i>dashboard user</i> | Sukses |
| Klik tombol <i>back</i> pojok kiri atas pada halaman daftar penyakit | Menampilkan kembali ke halaman awal <i>dashboard user</i> | Sukses |

h. Pengujian *Black Box* Pada Halaman Riwayat Diagnosa

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman riwayat diagnosa. Pengujian dan hasil uji *Black Box* halaman riwayat diagnosa pada Tabel 4.20.

**Tabel 4.20 Hasil Pengujian *Black Box* Halaman Riwayat Diagnosa**

| Aktifitas Pengujian                                      | Output yang diharapkan                                                                                | Hasil  |
|----------------------------------------------------------|-------------------------------------------------------------------------------------------------------|--------|
| Klik tombol hapus riwayat diagnosa pada pojok kanan atas | Memunculkan Konfirmasi untuk menghapus seluruh riwayat diagnosa “Anda yakin riwayat diagnosa dihapus” | Sukses |

|                                                                       |                                                           |        |
|-----------------------------------------------------------------------|-----------------------------------------------------------|--------|
| halaman ini.                                                          |                                                           |        |
| Klik tombol konfirmasi hapus riwayat diagnosa “Ya”                    | Menghapus seluruh riwayat diagnosa                        | Sukses |
| Klik tombol konfirmasi hapus riwayat diagnosa “Tidak”                 | Menampilkan kembali seluruh riwayat diagnosa              | Sukses |
| Klik tombol <i>back</i> pojok kiri atas pada detail penyakit          | Menampilkan kembali ke halaman daftar penyakit.           | Sukses |
| Klik tombol <i>back</i> pojok kiri atas pada halaman riwayat diagnosa | Menampilkan kembali ke halaman awal <i>dashboard user</i> | Sukses |

i. Pengujian *Black Box* Pada Halaman *Dashboard Admin*

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman *dashboard admin*. Pengujian dan hasil uji *Black Box* halaman *dashboard admin* pada Tabel 4.21.

**Tabel 4.21 Hasil Pengujian *Black Box* Halaman *Dashboard Admin***

| Aktifitas Pengujian          | Output yang diharapkan                                        | Hasil  |
|------------------------------|---------------------------------------------------------------|--------|
| Klik menu informasi aplikasi | Petunjuk penggunaan Sistem Pakar, dan tips penggunaan sistem. | Sukses |
| Klik menu data penyakit      | Menuju ke halaman data gejala                                 | Sukses |
| Klik menu data penyakit      | Menuju ke halaman data penyakit                               | Sukses |
| Klik menu data pengguna      | Menuju ke halaman data pengguna                               | Sukses |
| Klik menu data aturan        | Menuju ke halaman data aturan                                 | Sukses |
| Klik menu <i>logout</i>      | Menampilkan konfirmasi “Anda Yakin Mau <i>Logout</i> ?”       | Sukses |

|                  |                                                                  |        |
|------------------|------------------------------------------------------------------|--------|
| Klik “Ya Logout” | Menampilkan halaman login atau keluar dari <i>dashboard</i> awal | Sukses |
| Klik “Tidak”     | Tetap menampilkan halaman <i>dashboard</i> awal                  | Sukses |

j. Pengujian *Black Box* Pada Halaman Data Gejala

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman data gejala. Pengujian dan hasil uji *Black Box* halaman data gejala pada Tabel 4.22.

**Tabel 4.22 Hasil Pengujian *Black Box* Halaman Data Gejala**

| Aktifitas Pengujian                                                                                                            | Output yang diharapkan                                                                           | Hasil  |
|--------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------|--------|
| Masuk ke halaman daftar gejala                                                                                                 | Menampilkan daftar data gejala                                                                   | Sukses |
| Klik tombol ‘Tambah’                                                                                                           | Menampilkan halaman tambah data gejala                                                           | Sukses |
| Mengisi <i>form</i> tambah data gejala berupa kode gejala, nama gejala, nilai CF lalu klik simpan                              | Data tersimpan dan tampil di Daftar gejala                                                       | Sukses |
| Klik gejala yang dipilih lalu mengisi <i>form</i> ubah data gejala berupa kode gejala, nama gejala , nilai CF lalu klik simpan | Data berhasil diubah dan tersimpan serta langsung tampil di daftar data gejala                   | Sukses |
| Klik gejala terpilih lalu klik hapus data gejala dan mengkonfirmasi “Ya hapus”                                                 | Data berhasil terhapus dari <i>database</i> serta gejala tersebut hilang dari daftar data gejala | Sukses |

k. Pengujian *Black Box* Pada Halaman Data Penyakit

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman data penyakit. Pengujian dan hasil uji *Black Box* halaman data penyakit pada Tabel 4.23.

**Tabel 4.23 Hasil Pengujian *Black Box* Halaman Data Penyakit**

| Aktifitas Pengujian                                                                                                                                 | Output yang diharapkan                                                                               | Hasil  |
|-----------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|--------|
| Masuk ke halaman daftar penyakit                                                                                                                    | Menampilkan daftar data penyakit                                                                     | Sukses |
| Klik tombol ‘Tambah’                                                                                                                                | Menampilkan halaman tambah data penyakit                                                             | Sukses |
| Mengisi <i>form</i> tambah data penyakit berupa kode penyakit, nama penyakit, lalu klik simpan                                                      | Data tersimpan dan tampil di daftar penyakit                                                         | Sukses |
| Klik penyakit yang dipilih lalu mengisi <i>form</i> ubah data penyakit berupa kode penyakit, nama penyakit , deskripsi, dan solusi lalu klik simpan | Data berhasil diubah dan tersimpan serta langsung tampil di daftar data penyakit                     | Sukses |
| Klik penyakit terpilih lalu klik hapus data penyakit dan mengkonfirmasi “Ya hapus”                                                                  | Data berhasil terhapus dari <i>database</i> serta penyakit tersebut hilang dari daftar data penyakit | Sukses |

l. Pengujian *Black Box* Pada Halaman Data Aturan

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman data aturan. Pengujian dan hasil uji *Black Box* halaman data aturan pada Tabel 4.24.

**Tabel 4.24 Hasil Pengujian *Black Box* Halaman Data Aturan**

| Aktifitas Pengujian                                                                                                    | Output yang diharapkan                                                         | Hasil  |
|------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------|--------|
| Masuk ke halaman daftar aturan                                                                                         | Menampilkan daftar data aturan                                                 | Sukses |
| Klik aturan yang dipilih lalu isi opsi dengan gejala yang berkaitan dengan rule penyakit lalu klik ‘atur ulang gejala’ | Data berhasil diubah dan tersimpan serta langsung tampil di daftar data aturan | Sukses |
| Klik tombol kembali pada <i>action bar</i>                                                                             | Menampilkan kembali halaman <i>dashboard</i> awal                              | Sukses |

m. Pengujian *Black Box* Pada Halaman Data Pengguna

Pengujian *Black Box* dilakukan dengan menguji setiap aktivitas yang terdapat pada halaman data pengguna. Pengujian dan hasil uji *Black Box* halaman data pengguna pada Tabel 4.25.

**Tabel 4.25 Hasil Pengujian *Black Box* Halaman Data Pengguna**

| Aktifitas Pengujian                                                                      | Output yang diharapkan                            | Hasil  |
|------------------------------------------------------------------------------------------|---------------------------------------------------|--------|
| Masuk ke halaman data pengguna                                                           | Menampilkan daftar data penyakit                  | Sukses |
| Klik tombol ‘Tambah’                                                                     | Menampilkan halaman tambah data pengguna          | Sukses |
| Mengisi <i>form</i> tambah data pengguna berupa, namalengkap, <i>use rname, password</i> | Data tersimpan dan tampil di daftar pengguna      | Sukses |
| Klik pengguna yang dipilih lalu                                                          | Data berhasil diubah dan tersimpan serta langsung | Sukses |

|                                                                                                                  |                                                                                                      |        |
|------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------|--------|
| mengisi <i>form</i> ubah data gejala berupa nama lengkap, <i>username</i> , dan <i>password</i> lalu klik simpan | tampil di daftar data pengguna                                                                       |        |
| Klik pengguna terpilih lalu klik hapus data pengguna dan mengkonfirmasi “Ya hapus”                               | Data berhasil terhapus dari <i>database</i> serta pengguna tersebut hilang dari daftar data pengguna | Sukses |
| Masuk ke halaman daftar pengguna                                                                                 | Menampilkan daftar data pengguna                                                                     | Sukses |
| Klik tombol ‘Tambah’                                                                                             | Menampilkan halaman tambah data pengguna                                                             | Sukses |

Berdasarkan seluruh pengujian *blackbox* di atas dapat disimpulkan yaitu hasil uji fungsional sistem pakar untuk mendiagnosis penyakit kulit wajah adalah 100%. Dari hasil pengujian *black box* juga dinyatakan bahwa aplikasi sistem pakar penyakit kulit wajah dapat berjalan tanpa ada kendala atau tidak terjadi *bug*.

#### 4.6.2. Pengujian Akurasi

Pengujian akurasi bertujuan untuk mengetahui tingkat akurasi sistem pakar dalam memberikan hasil diagnosa penyakit kulit wajah. Data yang diuji berjumlah 10 sampel data analisa pakar. Hasil rekomendasi yang diperoleh dari sistem akan dibandingkan dengan hasil analisa pakar. Hasil pengujian akurasi sistem pakar dari 10 sampel yang telah diuji ditunjukkan pada Tabel 4.26. Pada tabel tersebut terdapat seluruh data sampel pengujian berdasarkan inputan gejala berdasarkan *CF user*.

**Tabel 4.26 Pengujian Akurasi**

| No. | Gejala dan Nilai CF User                                                                              | Hasil Diagnosa            |        |        | Kecocokan |
|-----|-------------------------------------------------------------------------------------------------------|---------------------------|--------|--------|-----------|
|     |                                                                                                       | Pakar                     | Sistem | Manual |           |
| 1   | (G01[0,6]);<br>(G02[0,8]);<br>(G08[0,4]);<br>(G09[0,8]);<br>(G10[0,8]);                               | <i>Acne Vulgaris</i>      | 93,38% | 93,38% | 1         |
| 2   | (G01[0,8]);<br>(G03[0,6]);<br>(G11[0,2]);<br>(G12[0,4]);<br>(G13[0,6]);<br>(G14[0,6]);                | <i>Melasma</i>            | 95,5%  | 95,5%  | 1         |
| 3   | (G01[0,8]);<br>(G03[0,6]);<br>(G15[0,6]);<br>(G16[0,8]);<br>(G17[0,8]);                               | <i>Nevus Melano sitik</i> | 95,69% | 95,69% | 1         |
| 4   | (G01[0,6]);<br>(G04[0,8]);<br>(G05[0,6]);<br>(G06[0,8]);<br>(G07[0,6]);                               | <i>Syringoma</i>          | 97,37% | 97,37% | 1         |
| 5   | (G01[0,6]);<br>(G04[0,8]);<br>(G06[0,6]);<br>(G07[0,6]);<br>(G18[0,8]);<br>(G19[0,8]);                | <i>Milia</i>              | 98,63% | 98,63% | 1         |
| 6   | (G01[0,6]);<br>(G06[0,8]);<br>(G20[0,8]);<br>(G21[0,6]);<br>(G22[0,6]);<br>(G23[0,6]);<br>(G24[0,6]); | <i>Rosacea</i>            | 97,93% | 97,93% | 1         |
| 7   | (G01[0,6]);<br>(G07[0,4]);<br>(G24[0,8]);<br>(G25[0,8]);<br>(G26[0,6]);                               | <i>Xanthe lasma</i>       | 96,56% | 96,56% | 1         |

|           |                                                                                                       |                           |        |        |   |
|-----------|-------------------------------------------------------------------------------------------------------|---------------------------|--------|--------|---|
|           | (G27[0,6]);                                                                                           |                           |        |        |   |
| <b>8</b>  | (G01[0,6]);<br>(G03[0,6]);<br>(G28[0,8]);<br>(G33[0,8]);                                              | <i>Keratosis Seboroik</i> | 93,77% | 93,77% | 1 |
| <b>9</b>  | (G01[0,6]);<br>(G03[0,8]);<br>(G29[0,8]);<br>(G30[0,6]);<br>(G31[0,8]);<br>(G34[0,6]);                | <i>Basalioma</i>          | 99,19% | 99,19% | 1 |
| <b>10</b> | (G01[0,4]);<br>(G03[0,8]);<br>(G28[0,6]);<br>(G29[0,4]);<br>(G32[0,8]);<br>(G33[0,6]);<br>(G34[0,4]); | <i>Melanoma</i>           | 98,04% | 98,04% | 1 |

Hasil dari akurasi pada penyakit kulit wajah bernali 1 artinya diagnosa sistem sama dengan diagnosa pakar dan sesuai dengan perhitungan manual. Sebaliknya, jika hasil akurasi bernali 0 artinya keluaran dari sistem tidak sama dengan keluaran dari pakar. Berdasarkan tabel 4.27 telah dilakukan pengujian akurasi dengan 10 sampel data diagnosa penyakit kulit wajah dan menghasilkan nilai akurasi sesuai perhitungan berikut.

$$\text{Nilai akurasi} = \frac{\text{Jumlah data akurasi}}{\text{Keseluruhan data}} \times 100\%$$

$$\text{Nilai akurasi} = \frac{10}{10} \times 100\% = 100\%$$

Hasil dari nilai akurasi sistem pakar diagnosa kerusakan komputer dengan metode *certainty factor* dan *forward chaining* adalah 100%.

#### 4.7 Evaluasi Pengujian

Berdasarkan serangkaian pengujian yang telah dilakukan, ditemukan kelebihan dan kekurangan yang masih ada pada sistem, antara lain :

##### 4.7.1 Kelebihan Sistem

- a. Pengguna dapat memilih jenis diagnosa sesuai kebutuhannya.

- b. Setiap gejala yang ditemukan pada saat proses diagnosa, memiliki solusi yang dapat membantu pengguna dalam mendiagnosis penyakit kulit pada wajah.
- c. Pengguna dapat melihat riwayat hasil diagnosa, yaitu diagnosa yang telah dilakukan *user* akan tercatat informasinya seperti metode diagnosa, nilai, solusi dan waktu *user* melakukan diagnosa.
- d. *Admin* bisa mengelola aturan yang akan ditetapkan diaplikasi ini.

#### 4.7.2 Kelemahan Sistem

- a. Penyajian Informasi penyakit sangat sederhana karena tidak disertai dengan gambar ataupun foto terkait penyakit tersebut.
- b. Penyakit yang ada pada sistem pakar ini masih tergolong sangat umum.
- c. Penentuan nilai CF Pakar hanya sebatas sumber dari seorang pakar