

BAB III

ANALISA PERANCANGAN

III.1. Analisa Sistem Yang Berjalan

Analisa sistem yang berjalan bertujuan untuk mengidentifikasi serta melakukan evaluasi terhadap sistem pakar diagnosa gejala virus *Covid-19* menggunakan metode *Forward Chaining*, analisis dilakukan agar dapat menemukan masalah-masalah dalam pengolahan sistem pakar diagnosa gejala virus *Covid-19* menggunakan metode *Forward Chaining*, dalam menentukan gejala-gejala dari setiap resiko agar mudah dalam menentukan jenis penyakit yang diderita pengguna. Analisis sistem ini meliputi *input*, proses dan *output* yang akan dijabarkan. Sebelum itu, penting untuk kita mengetahui bahwa hingga saat ini, penyebaran *Covid-19* di Indonesia sudah melebihi angka 4 juta dengan penambahan kasus setiap harinya.

Tabel III.1. Sebaran Covid-19 di Indonesia

Terkonfirmasi	Kasus Aktif	Sembuh	Meninggal
4,319,175 Juta (+ 9,905 kasus)	43,574 (+7,879 kasus aktif) Dengan persentase 1,0%	4,131,333 (+ 2,028 kasus sembuh) Dengan persentase 95,7%	144,268 (+ 7 kasus meninggal) Dengan persentase 3,3%

Tabel III.2. Sebaran Covid-19 Berdasarkan Provinsi di Indonesia

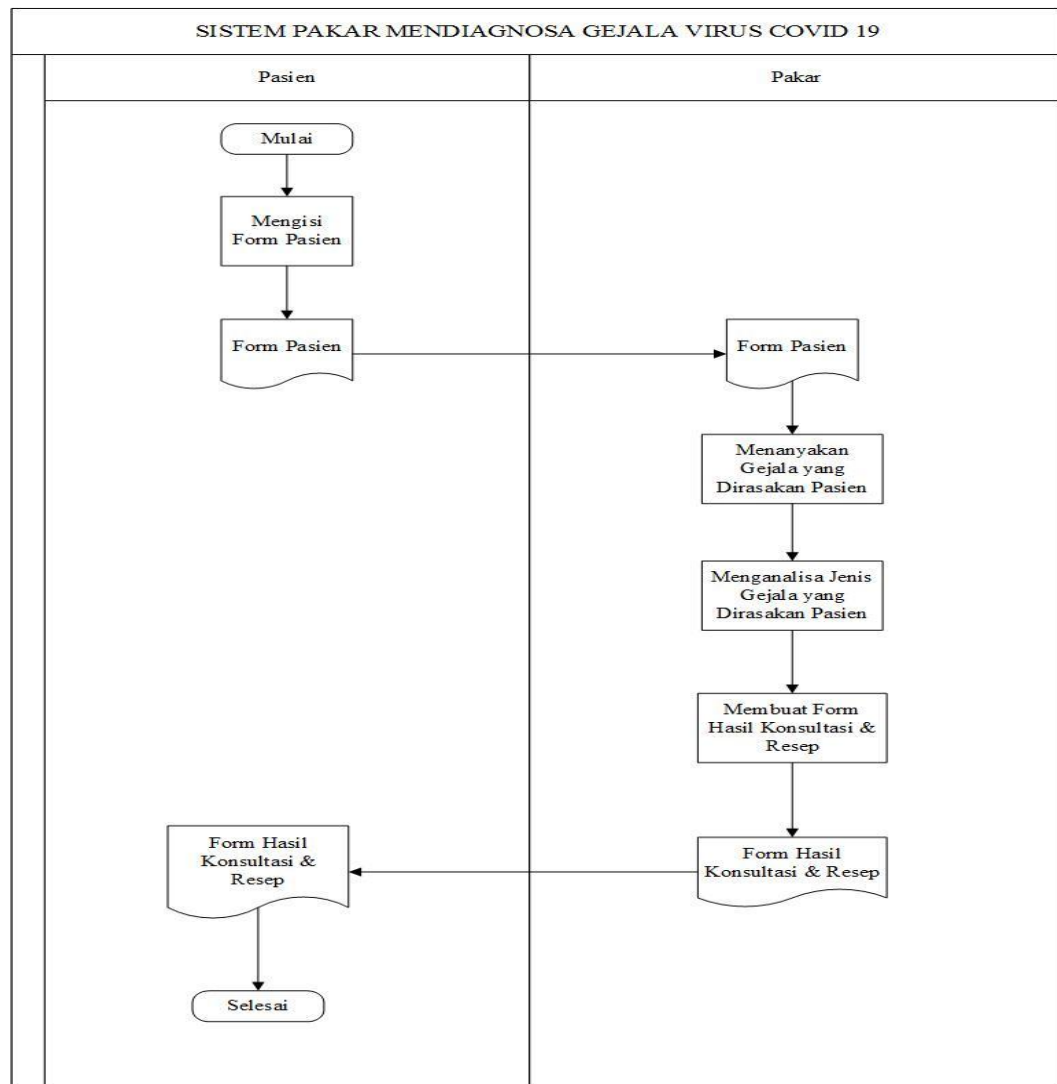
No	Provinsi	Jumlah Kasus	Persentase
1	DKI Jakarta	891,148	20.7%
2	Jawa Barat	717,339	16,6%
3	Jawa Tengah	487,656	11,3%
4	Jawa Timur	401,558	9,3%
5	Kalimantan Timur	158,562	3,7%
6	Daerah Istimewa Yogyakarta	157,212	3,6%
7	Banten	139,005	3,2%
8	Riau	128,981	3,0%
9	Bali	115,204	2,7%
10	Sulawesi Selatan	110,181	2,6%
11	Sumatera Utara	106,296	2,5%
12	Sumatera Barat	89,918	2,1%
13	Kalimantan Selatan	70,025	2,6%
14	Nusa Tenggara Timur	64,547	1,5%
15	Sumatera Selatan	60,051	1,4%
16	Kepulauan Riau	54,863	1,3%
17	Kepulauan Bangka Belitung	52,429	1,2%
18	Lampung	49,806	1,2%
19	Sulawesi Tengah	47,320	1,1%
20	Kalimantan Tengah	45,696	1,1%
21	Kalimantan Barat	41,861	1.0%
22	Aceh	38,445	0,9%
23	Kalimantan Utara	35,993	0,8%
24	Sulawesi Utara	34,895	0,8%
25	Papua	34,499	0,8%
26	Jambi	29,803	0,7%
27	Nusa Tenggara Barat	27,863	0,7%
28	Papua Barat	23,787	0,6%
29	Bengkulu	23,161	0,5%
30	Sulawesi Tenggara	20,180	0,5%
31	Maluku	14,624	0,3%
32	Sulawesi Barat	12,377	0,3%
33	Maluku Utara	12,116	0,3%
34	Gorontalo	11,861	0,3%

III.1.1. Analisa Input

Masukan sistem (*input*) adalah merupakan data gejala-gejala yang nantinya pengguna dapat memilih gejala-gejala yang dialami kemudian dimasukkan ke dalam sistem untuk diproses. Pada bagian ini, tidak ada yang menjadi masukan sistem karena sistem yang digunakan adalah dengan cara manual. Biasanya dokter mengandalkan pengetahuannya tentang gejala-gejala yang dialami pasien kemudian mengambil kesimpulan.

III.1.2. Analisa Proses

Proses diagnosa gejala virus *Covid-19* yaitu dengan melihat gejala-gejala yang dirasakan oleh pasien. Setelah data diketahui maka dokter dapat menyimpulkan apakah pasien terinfeksi virus *Covid-19* atau tidak. Analisa proses yang dilakukan dapat dilihat pada gambar sebagai berikut:



Gambar III.1. FOD Sistem Pakar Mendiagnosa Gejala Virus Covid-19

Menggunakan Metode *Forward Chaining*

III.1.3. Analisa Output

Terdapat analisa output dalam sistem pakar diagnosa gejala virus Covid-19 menggunakan metode *Forward Chaining* yaitu, hasilnya berupa data hasil konsultasi.

III.2. Kaidah Produksi (*IF-THEN*)

Algoritma *Forward Chaining* bekerja dengan dua data yang tersedia dan menggunakan aturan-aturan inferensi untuk mendapatkan hasil keputusan atau kesimpulan. Kaidah produksinya biasanya dituliskan dalam bentuk jika-maka (*IF-THEN*). Kaidah ini dapat dikatakan sebagai hubungan implikasi dua bagian, yaitu *premise* (jika) dan bagian *konklusi* (maka). Apabila bagian *premise* terpenuhi, maka bagian *konklusi* juga akan bernilai benar. Sebuah kaidah terdiri dari *klausa-klausa* yang mirip dengan sebuah kalimat subjek, kata kerja, dan objek yang mengatakan suatu fakta. Ada sebuah *klausa premise* dan *klausa konklusi* pada sebuah kaidah. Suatu kaidah juga terdiri atas beberapa *premise* dan lebih dari satu *konklusi*. Antara *premise* dan *konklusi* dapat berhubungan dengan “OR” atau “AND”. Berikut adalah kaidah produksi dalam menganalisis gejala virus *Covid-19*.

1. Tabel Gejala

Tabel III.3. Tabel Data Gejala

No	Kode Gejala	Nama Gejala
1.	G1	Demam
2.	G2	Pilek
3.	G3	Batuk
4.	G4	Malaise (Badan Lemas)
5.	G5	Anosmia (Gangguan Indra Penciuman)
6.	G6	Diare (Gangguan Pencernaan)
7.	G7	Sakit Tenggorokan
8.	G8	Sesak Nafas
9.	G9	Nyeri Dada
10.	G10	Sakit Kepala
11.	G11	Bersin
12.	G12	Mialgia (Nyeri Otot)
13.	G13	Sakit Perut
14.	G14	Batuk Berkepanjangan

15.	G15	Mual Muntah
16.	G16	Hilang Nafsu Makan
17.	G17	Gangguan Pendengaran

2. Tabel Keputusan & Solusi

Tabel III.4. Tabel Keputusan & Solusi

No	Kode	Nama	Solusi
1.	P1	Demam/Batuk Biasa	Banyak beristirahat dan minum obat pereda demam serta batuk.
2.	P2	<i>Covid-19</i> ringan	Konsumsi suplemen peningkat daya tahan tubuh, misalnya vitamin C dan vitamin D.
3.	P3	<i>Covid-19</i> sedang	Lakukan pemeriksaan ke Dokter, Dokter akan memberikan obat antivirus. Gejala sedang dapat dirawat di rumah asalkan telah mendapatkan obat dari Dokter.
4.	P4	<i>Covid-19</i> berat	Segera lakukan pemeriksaan Dokter dan lakukan perawatan di rumah sakit. Dokter akan memberikan terapi infus, pemberian oksigen, dan pemberian obat-obatan.
5.	P5	<i>Covid-19</i> varian Omicron	Lakukan isolasi mandiri. Jika memerlukan obat, dapat dibeli dari Apotik.
6.	P6	<i>Covid-19</i> varian Alpha	Lakukan pemeriksaan ke Dokter agar mendapatkan penanganan yang tepat.
7.	P7	<i>Covid-19</i> varian Beta	Lakukan pemeriksaan ke Dokter agar mendapatkan penanganan yang tepat.
8.	P8	<i>Covid-19</i> varian Delta	Lakukan pemeriksaan ke Dokter agar mendapatkan penanganan yang tepat.

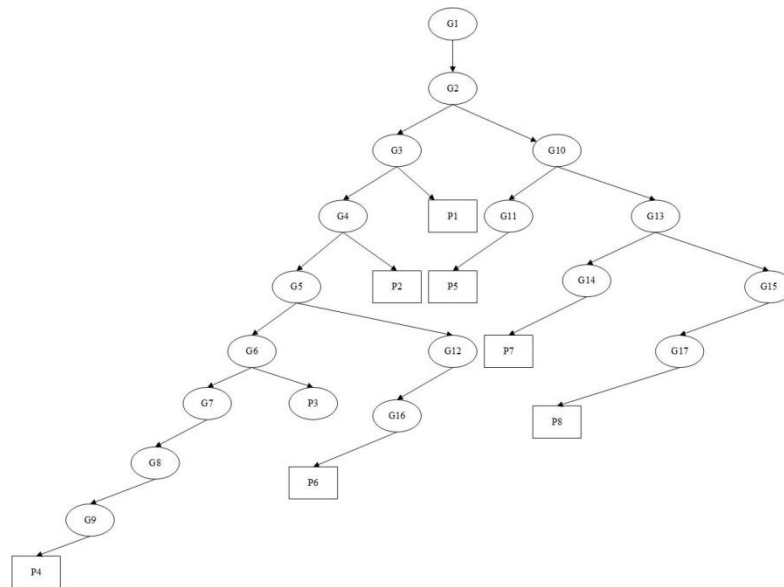
3. Tabel Rule (Aturan)

Tabel III.5. Tabel Rule (Aturan)

Rule	IF	Then
1.	G1&G2&G3	P1
2.	G1&G2&G3&G4	P2
3.	G1& G2&G3&G4&G5&G6	P3
4.	G1& G2&G3&G4&G5&G6&G7&G8&G9	P4
5.	G1&G10&G11	P5
6.	G1&G12&G16	P6
7.	G1&G13&G14	P7

8.	G1&G15&G17	P8
----	------------	----

4. Pohon Keputusan



Gambar III.2. Pohon Keputusan Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

Berdasarkan pohon keputusan, maka kaidah aturan gejala Covid-19 dengan metode *forward chaining* terdapat beberapa rule sebagai berikut :

Kaidah 1 : <i>IF</i> (Demam <i>OR</i> Pilek <i>OR</i> Batuk) <i>Then</i> Demam/Batuk biasa (P1).
Kaidah 2 : <i>IF</i> (Demam <i>OR</i> Pilek <i>OR</i> Batuk) <i>AND</i> (Malaise <i>OR</i> Anosmia <i>OR</i> Diare) <i>Then</i> Covid-19 ringan (P2).
Kaidah 3 : <i>IF</i> (Demam <i>OR</i> Pilek <i>OR</i> Batuk) <i>AND</i> (Malaise <i>OR</i> Anosmia <i>OR</i> Diare) <i>AND</i> (Sakit Tenggorokan <i>OR</i> Sesak Nafas <i>OR</i> Nyeri Dada) <i>Then</i> Covid-19 sedang (P3).
Kaidah 4 : <i>IF</i> ((Demam <i>OR</i> Pilek <i>OR</i> Batuk)) <i>OR</i> ((Malaise <i>OR</i> Anosmia <i>OR</i> Diare)) <i>AND</i> (Sakit Tenggorokan <i>OR</i> Sesak Nafas <i>OR</i> Nyeri Dada) <i>Then</i> Beresiko (P4).
Kaidah 5 : <i>IF</i> (Demam <i>OR</i> Sakit Kepala <i>OR</i> Bersin) <i>Then</i> Covid-19 varian Omicron (P5).
Kaidah 6 : <i>IF</i> (Demam <i>OR</i> Mialgia <i>OR</i> Hilang Nafsu Makan) <i>Then</i> Covid-19 varian Alpha (P6).
Kaidah 7 : <i>IF</i> (Demam <i>OR</i> Sakit Perut <i>OR</i> Batuk Berkepanjangan) <i>Then</i> Covid-19 varian Betha (P7).

Kaidah 8 : *IF* (Demam *OR* Mual Muntah *OR* Gangguan Pendengaran) *Then Covid-19* varian Delta (P8).

III.3. Desain Sistem

Untuk membantu membangun sistem pakar dalam mendiagnosa gejala virus *Covid-19* menggunakan metode *Forward Chaining*, penulis mengusulkan pembuatan sebuah sistem dengan menggunakan aplikasi yang lebih akurat dan lebih mudah dalam penggunaannya. Pada perancangan sistem ini terdiri dari tahap perancangan yaitu :

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

III.3.1. Use Case Diagram

Dalam penyusunan suatu program diperlukan suatu model data yang berbentuk diagram yang dapat menjelaskan suatu alur proses sistem yang akan di bangun. Dalam penulisan skripsi ini penulis menggunakan metode UML yang dalam metode itu penulis menerapkan diagram *Use Case*. Maka digambarlah suatu bentuk diagram *Use Case* yang dapat dilihat pada gambar berikut:



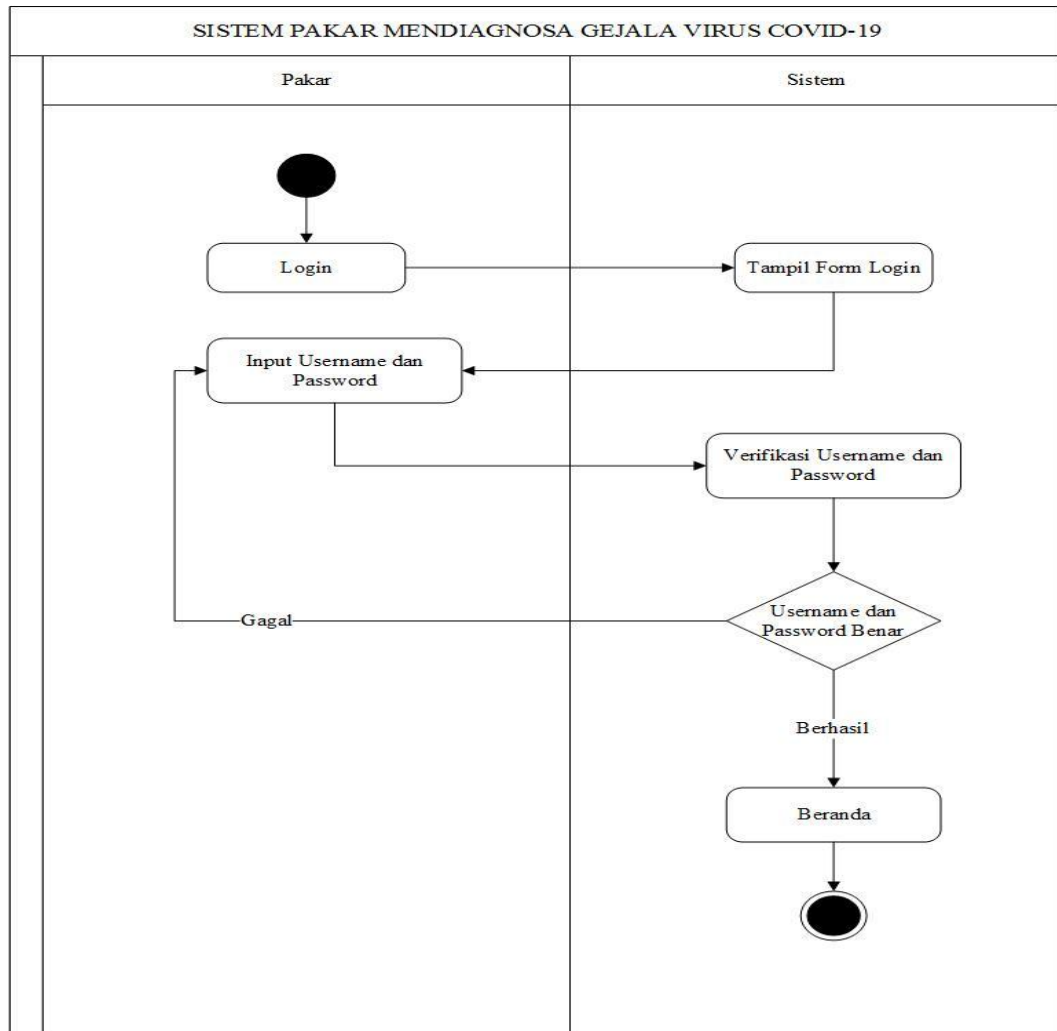
Gambar III.3. Uce Case Diagram Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

III.3.2. Activity Diagram

Activity diagram menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing – masing alir berawal, *Decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

1. Activity Diagram Login

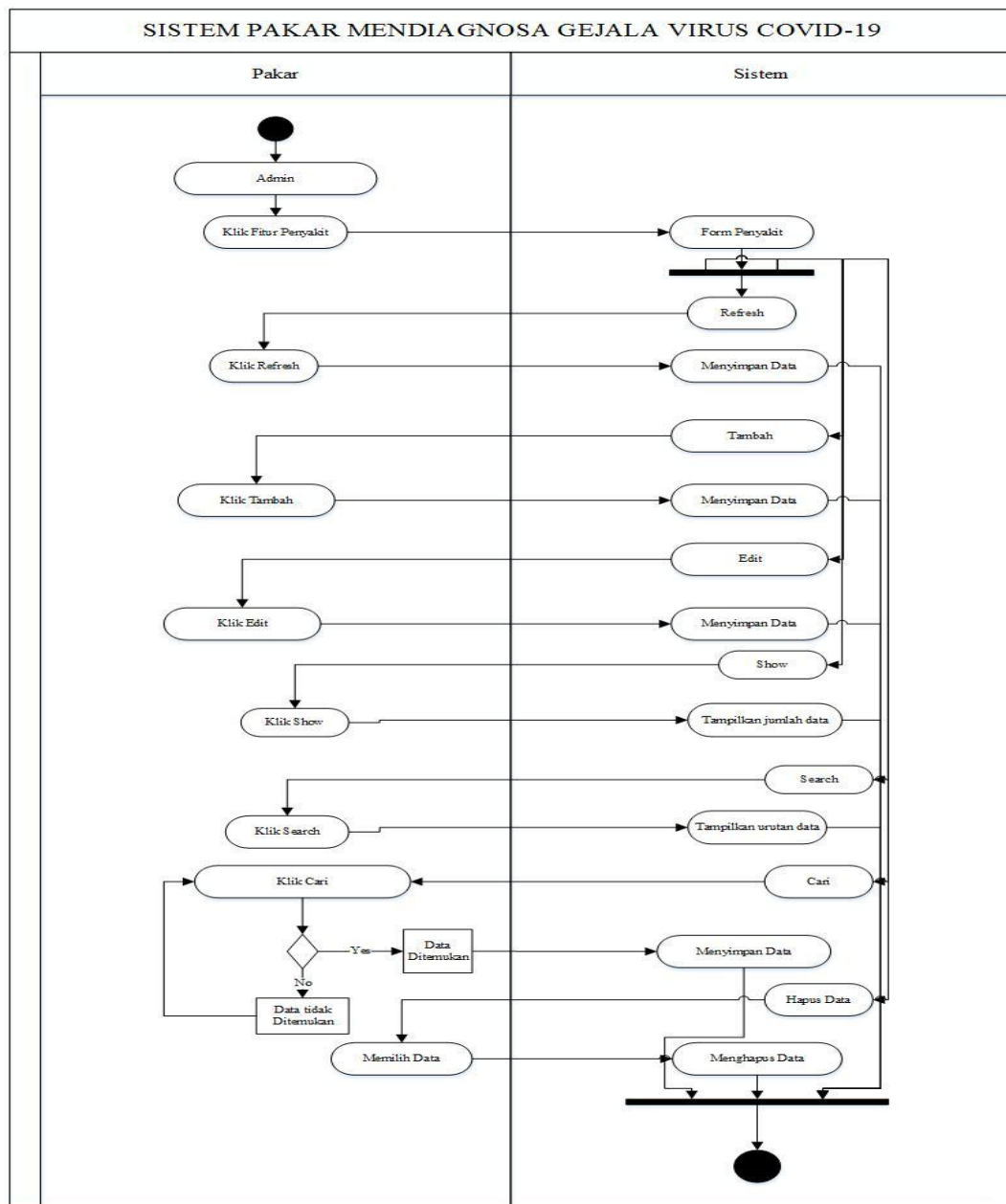
Activity diagram login dapat dilihat pada gambar dibawah ini, sebagai berikut :



Gambar III.4. Acticity Diagram Login Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

2. Activity Diagram Penyakit

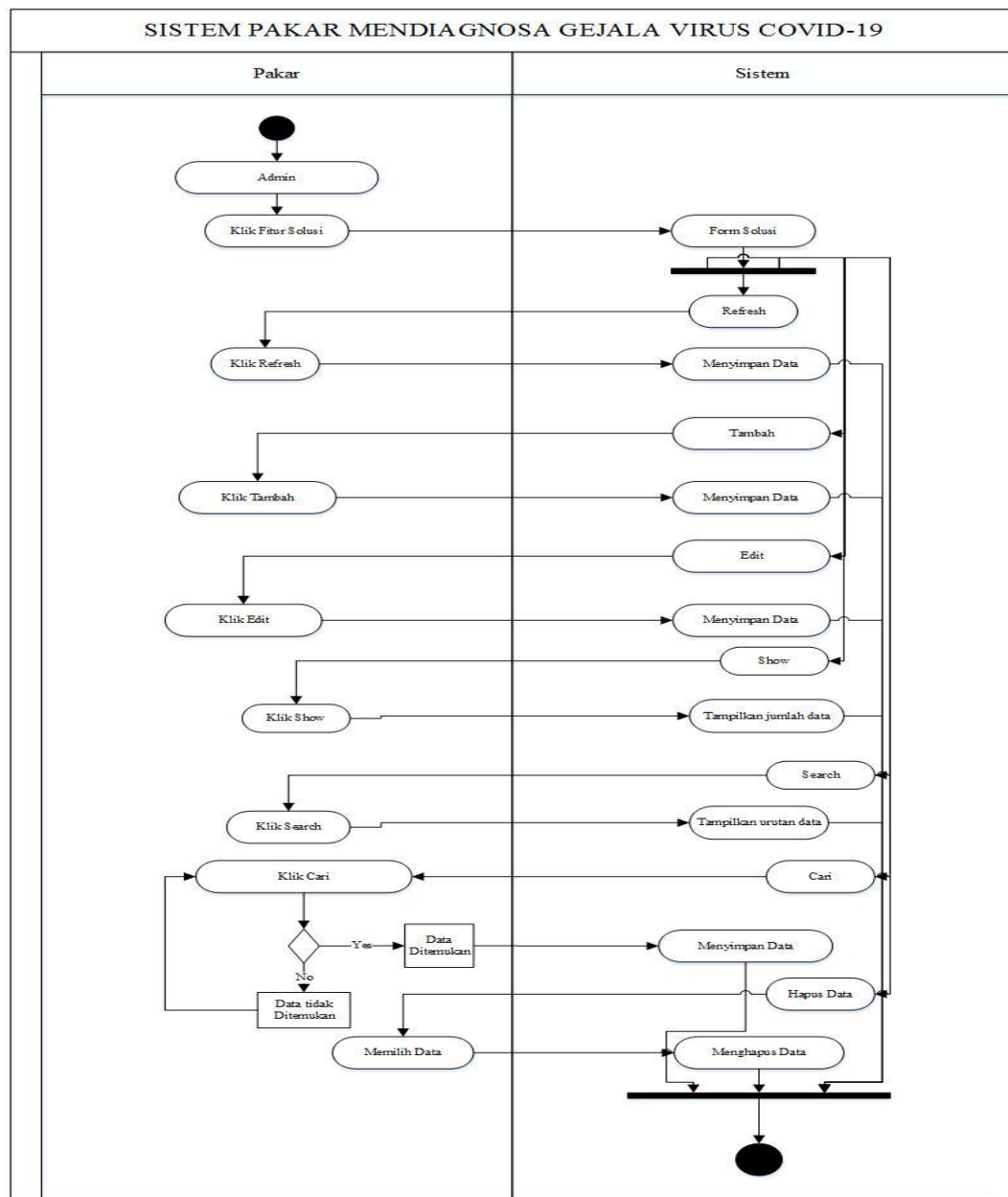
Activity diagram data penyakit dapat dilihat pada gambar dibawah ini, sebagai berikut :



Gambar III.5. Activity Diagram Data Penyakit Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

3. Activity Diagram Solusi

Activity diagram solusi konsultasi dapat dilihat pada gambar dibawah ini, sebagai berikut :

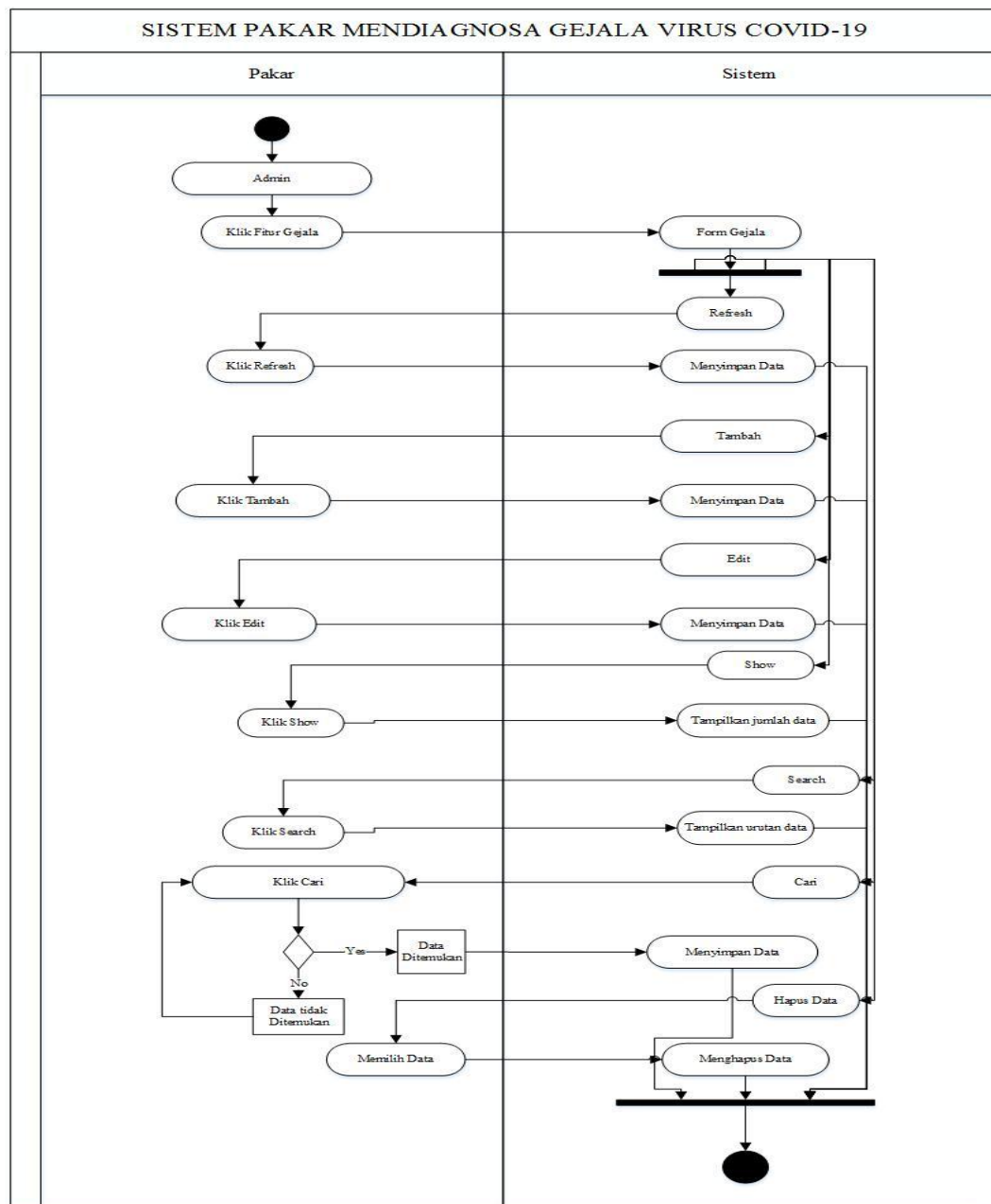


Gambar III.6. Activity Diagram Solusi Sistem Pakar Mendiagnosa Gejala

Virus Covid-19 Menggunakan Metode *Forward Chaining*

4. Activity Diagram Data Gejala

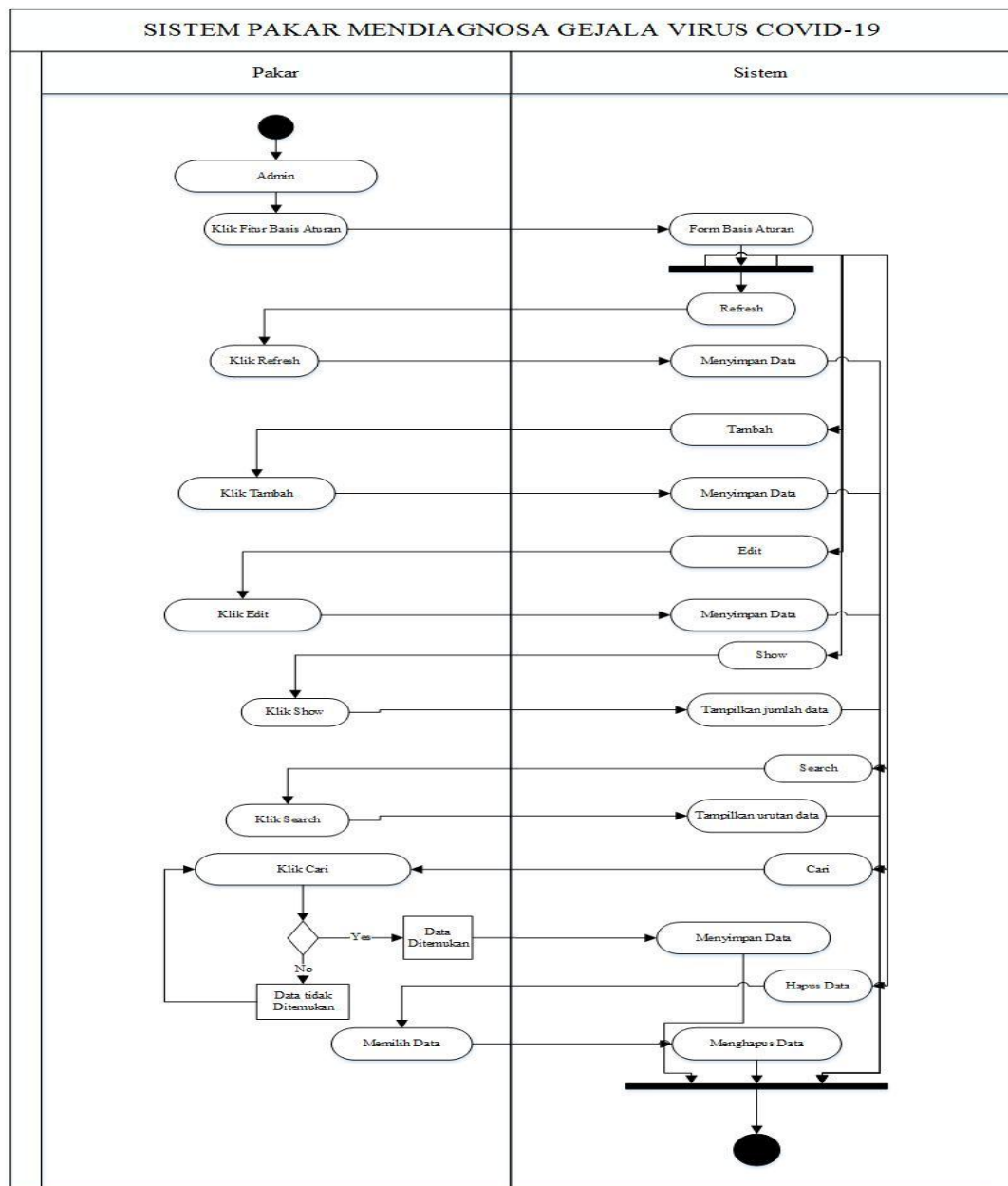
Activity diagram data gejala dapat dilihat pada gambar dibawah ini, sebagai berikut :



Gambar III.7. Activity Diagram Data Gejala Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

5. Activity Diagram Basis Aturan

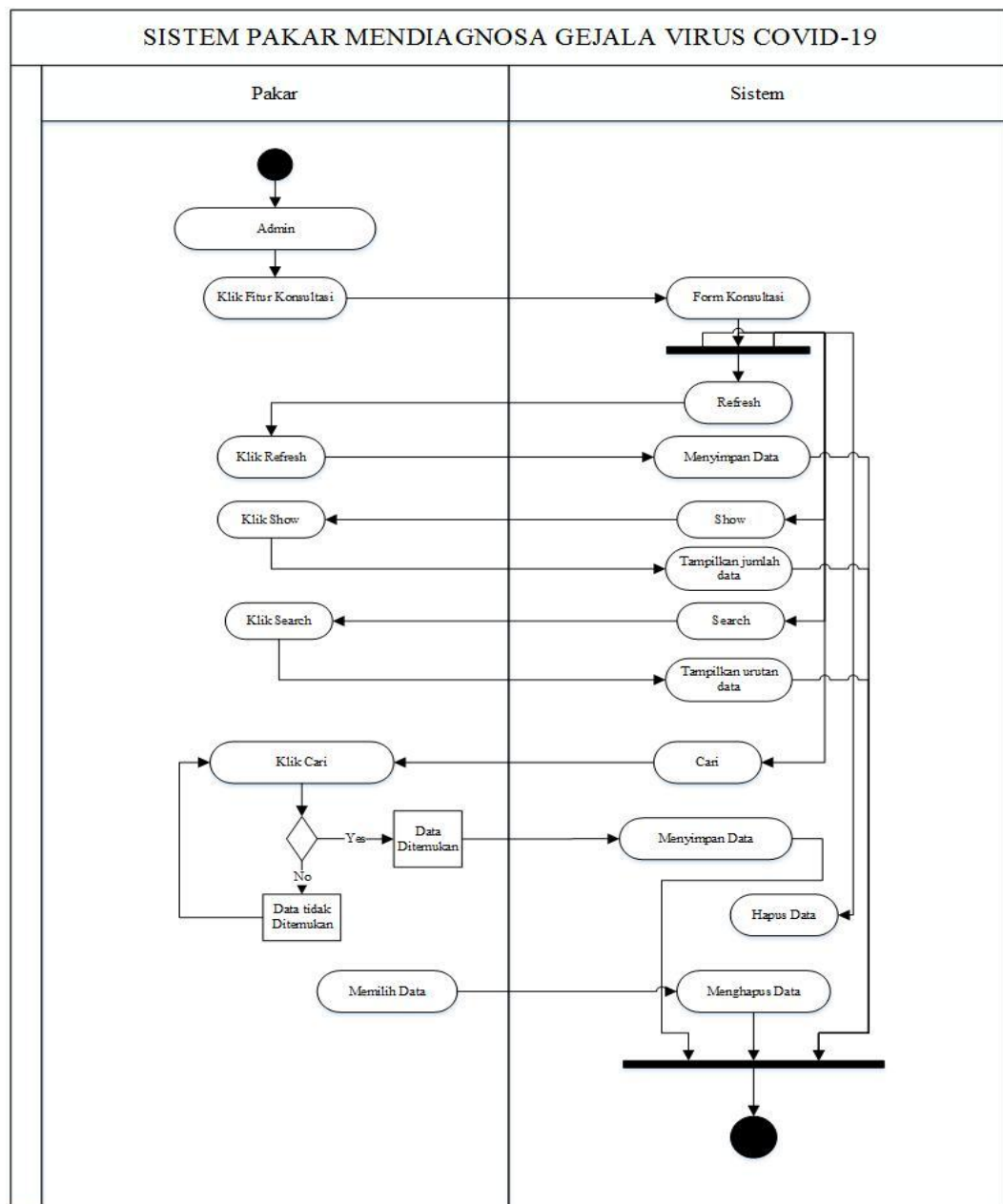
Activity diagram basis aturan dapat dilihat pada gambar dibawah ini, sebagai berikut :



Gambar III.8. Activity Diagram Basis Aturan Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

6. Activity Diagram Daftar Konsultasi

Activity diagram daftar konsultasi dapat dilihat pada gambar dibawah ini, sebagai berikut :



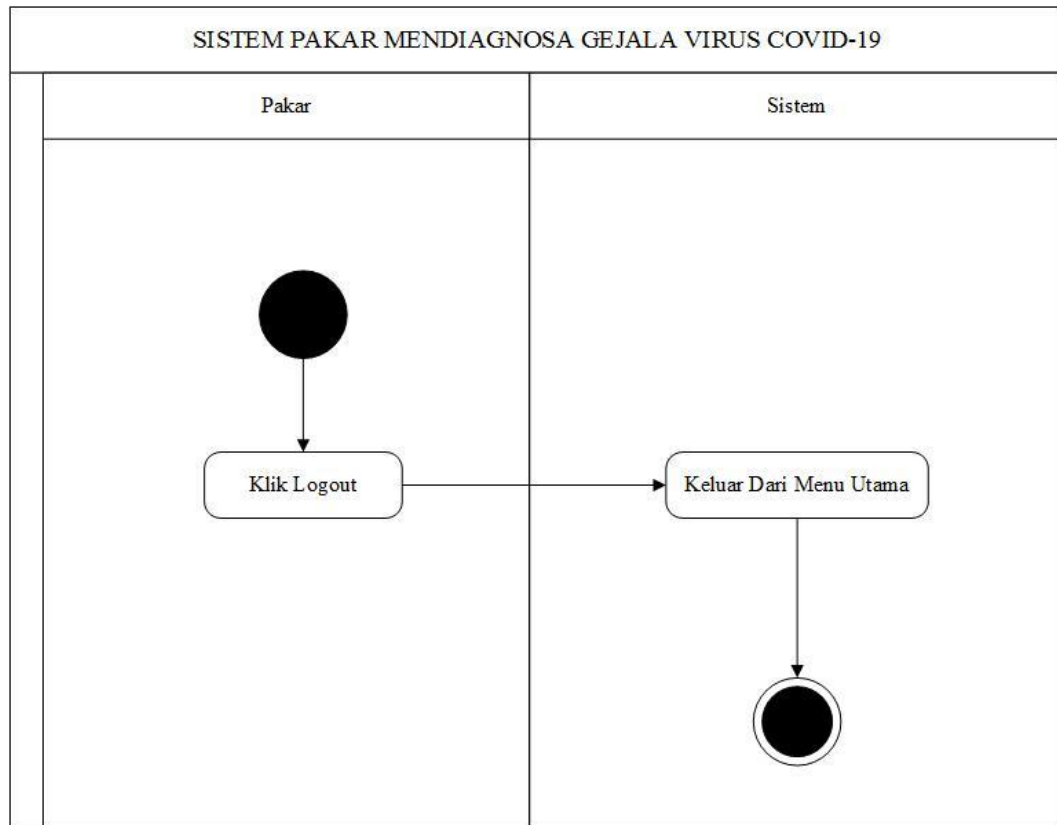
Gambar III.9. Activity Diagram Daftar Konsultasi Sistem Pakar

Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward*

Chaining

7. Activity Diagram Logout

Activity diagram logout dapat dilihat pada gambar dibawah ini, sebagai berikut :



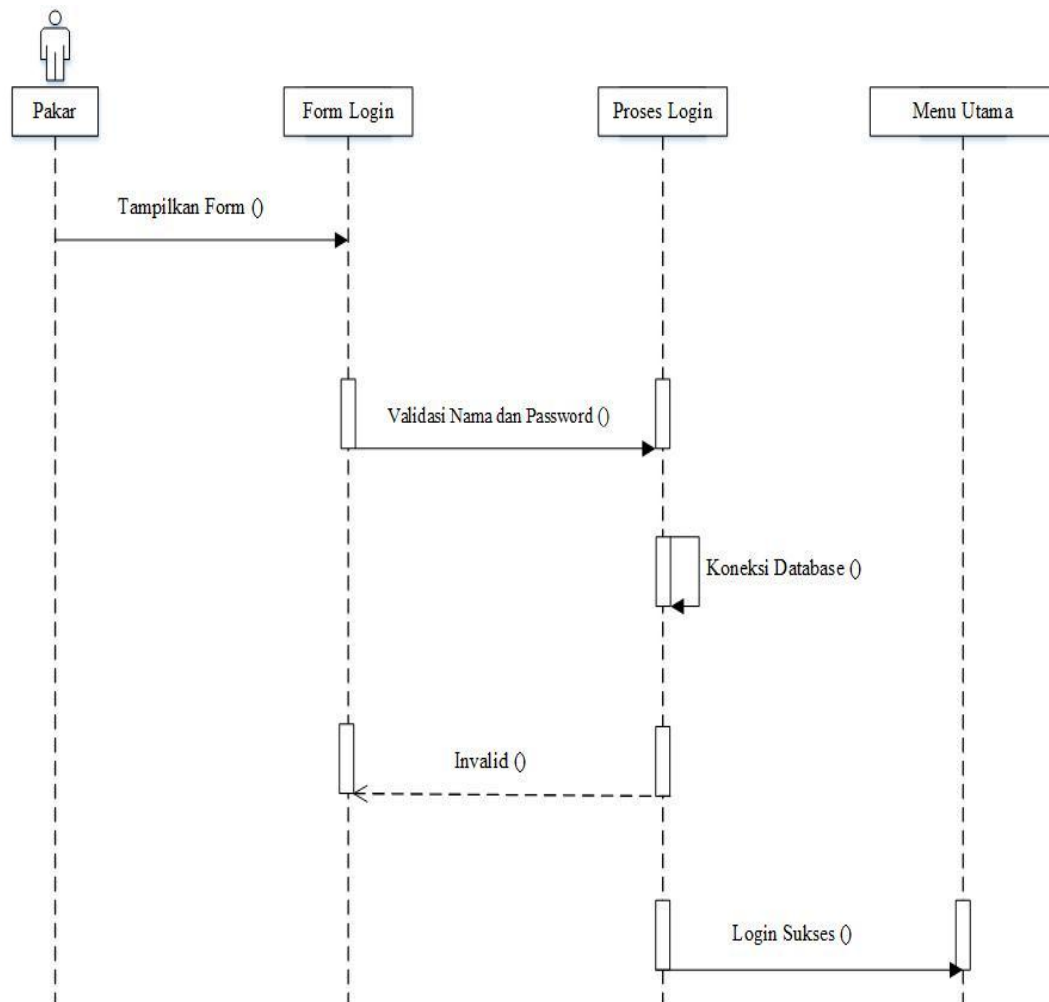
Gambar III.10. Activity Diagram Logout Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

III.3.3. Sequence Diagram

Sequence diagram menggambarkan perilaku pada sebuah skenario, diagram ini menunjukkan sejumlah contoh objek dan *message* (pesan) yang diletakkan diantara objek – objek ini di dalam *use case*.

1. Sequence Diagram Login

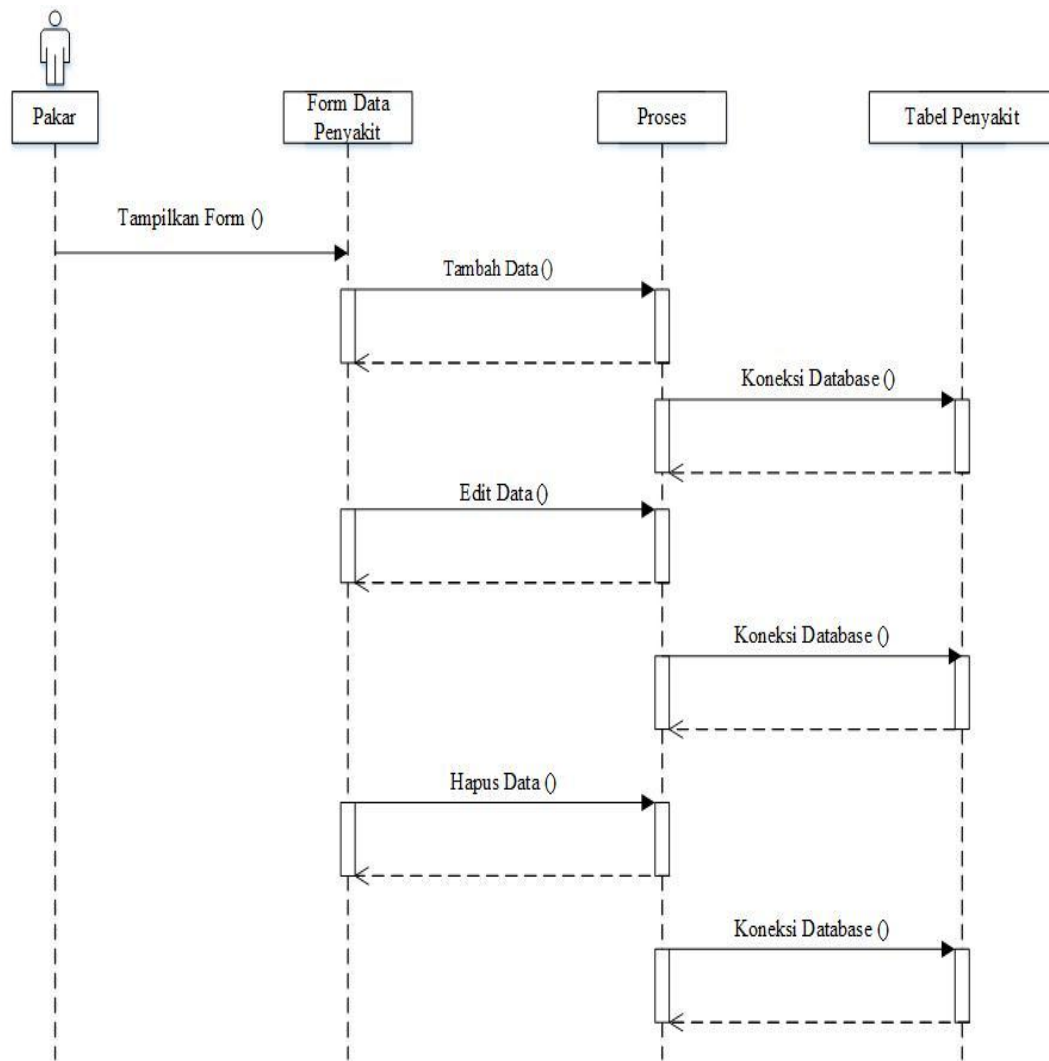
Sequence diagram login dapat dilihat pada gambar sebagai berikut :



Gambar III.11. Sequence Diagram Login Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

2. Sequence Diagram Data Penyakit

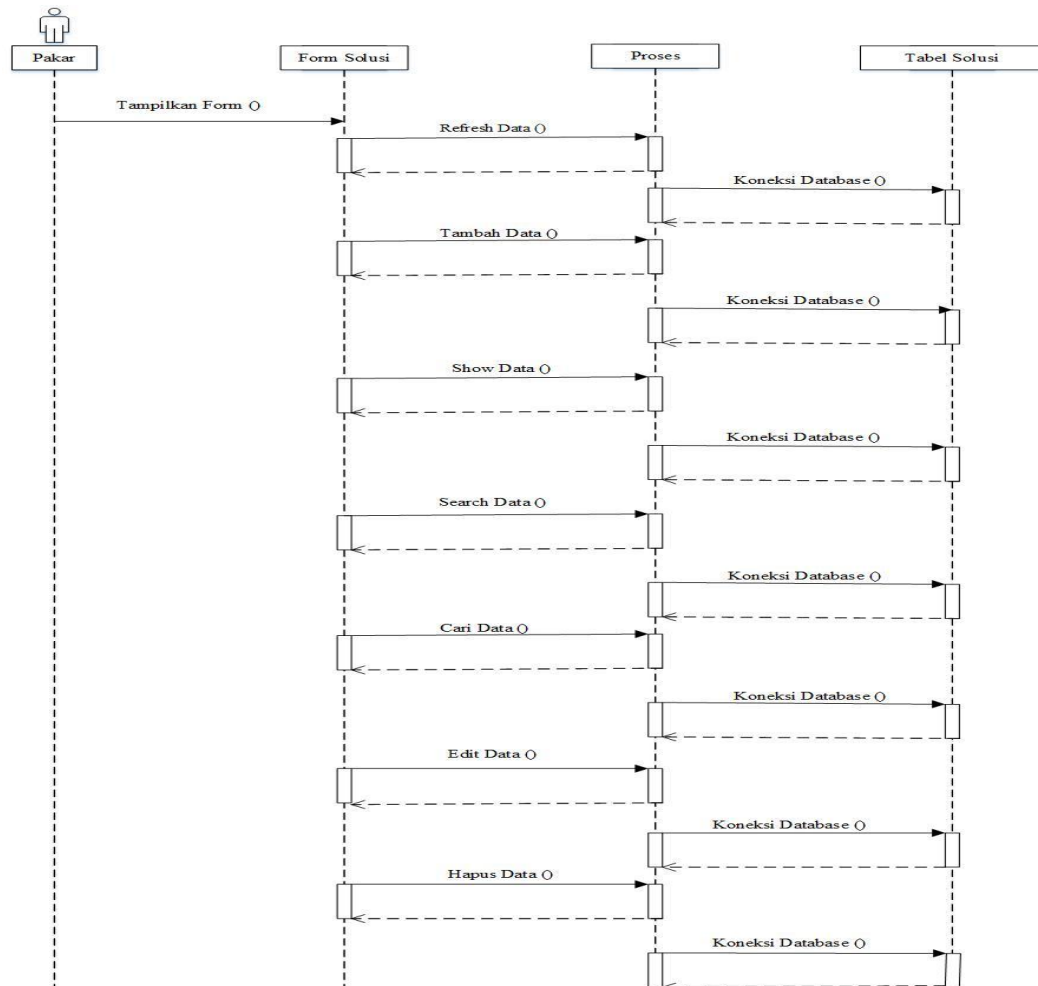
Sequence diagram data penyakit dapat dilihat pada gambar sebagai berikut :



Gambar III.12. Sequence Diagram Data Penyakit Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

3. Sequence Diagram Solusi

Sequence diagram solusi aturan dapat dilihat pada gambar sebagai berikut :

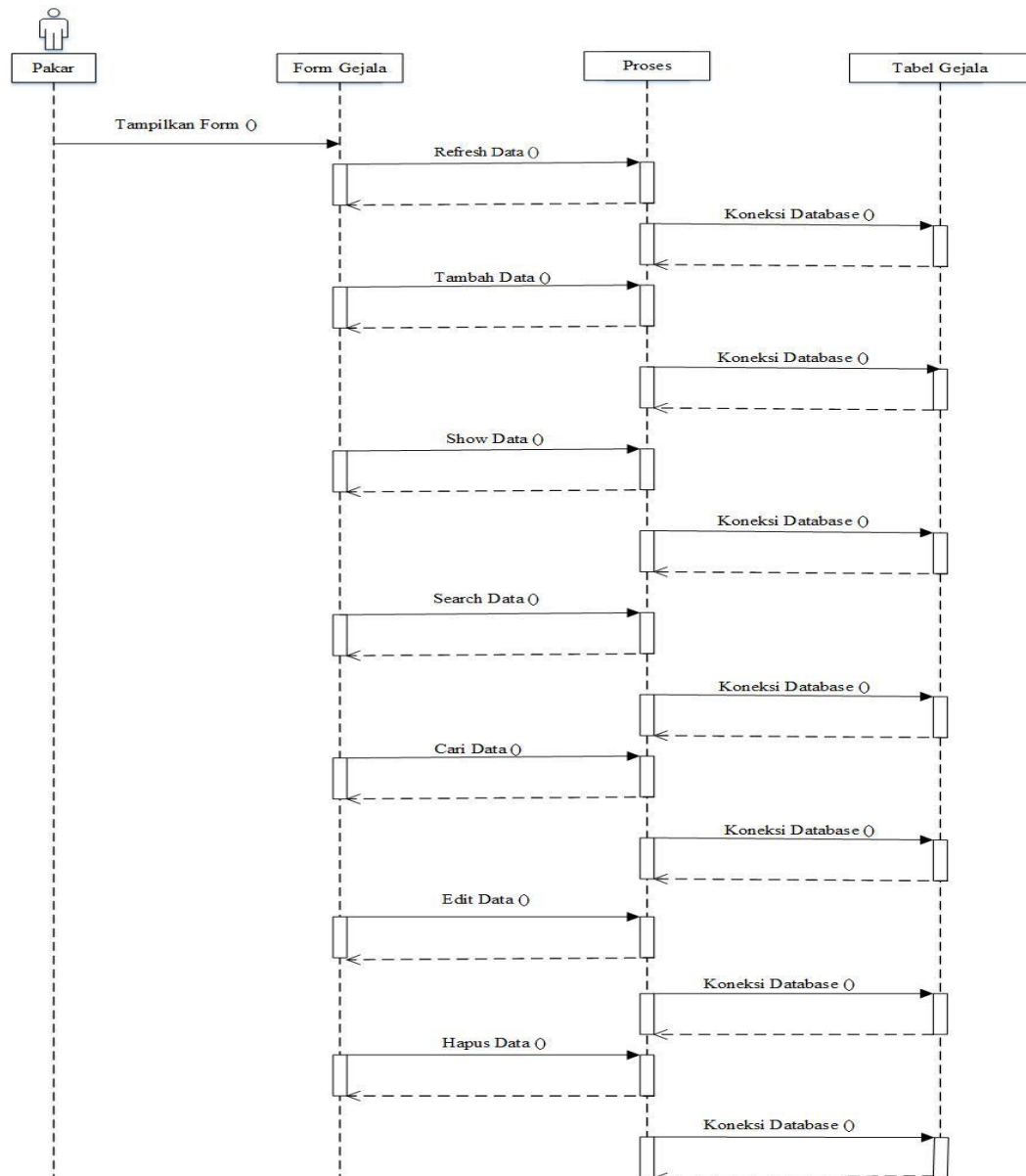


Gambar III.13. Sequence Diagram Solusi Sistem Pakar Mendiagnosa Gejala

Virus Covid-19 Menggunakan Metode *Forward Chaining*

4. Sequence Diagram Data Gejala

Sequence diagram data gejala dapat dilihat pada gambar sebagai berikut :

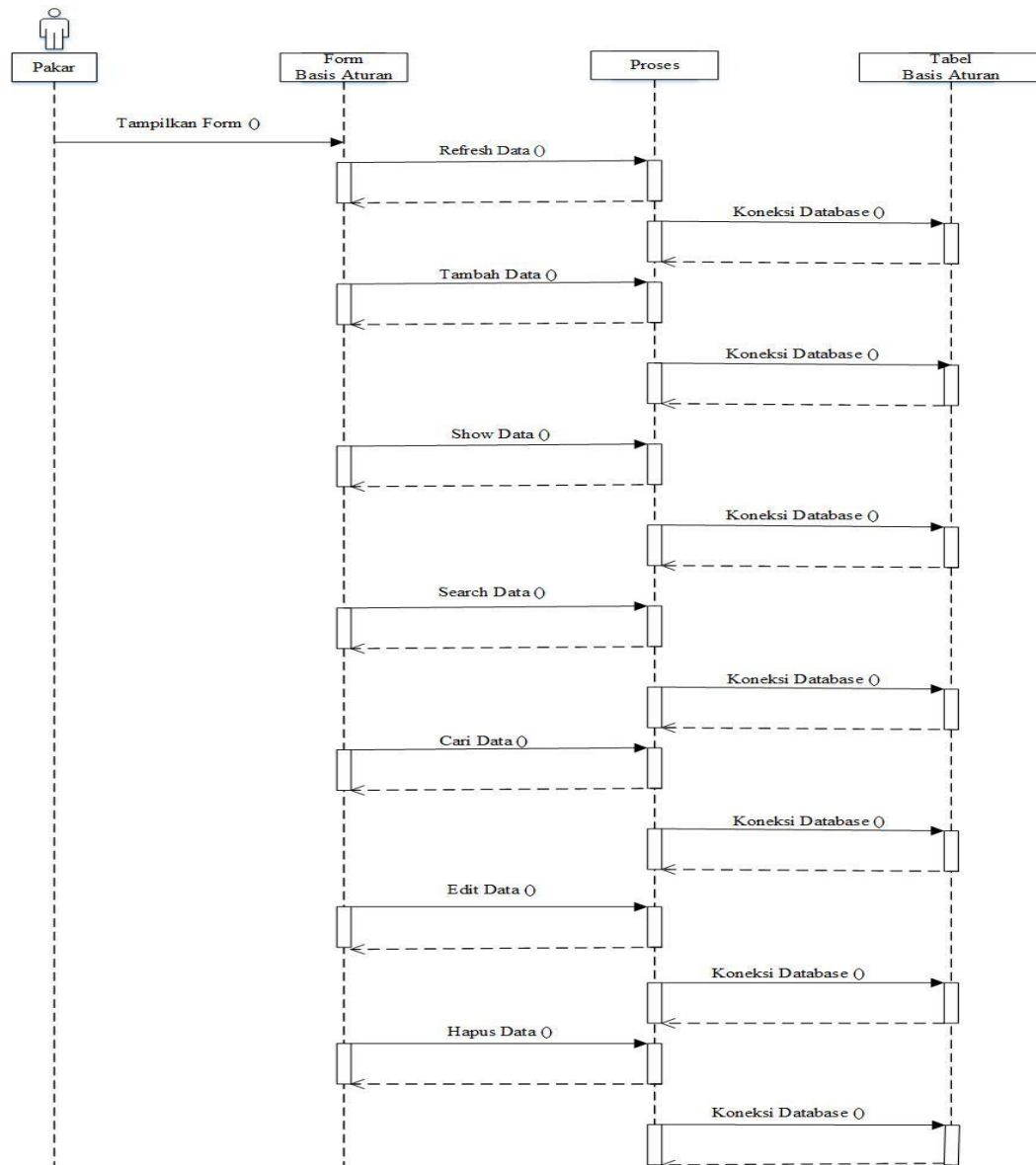


Gambar III.14. Sequence Diagram Data Gejala Sistem Pakar Mendiagnosa

Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

5. Sequence Diagram Basis Aturan

Sequence diagram basis aturan dapat dilihat pada gambar sebagai berikut :

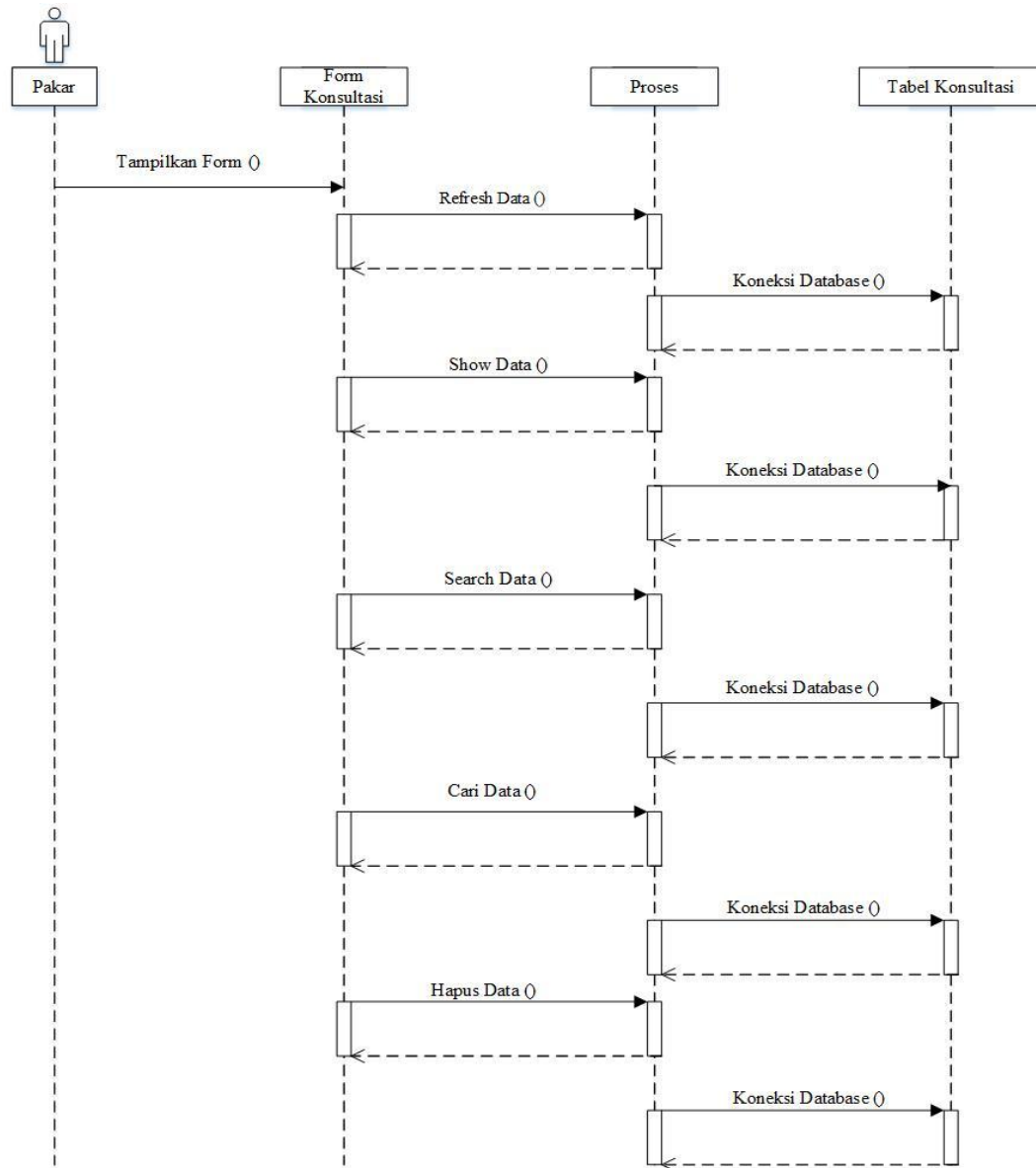


Gambar III.15. Sequence Diagram Basis Aturan Sistem Pakar Mendiagnosa

Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

6. Sequence Diagram Daftar Konsultasi

Sequence diagram daftar konsultasi dapat dilihat pada gambar sebagai berikut :



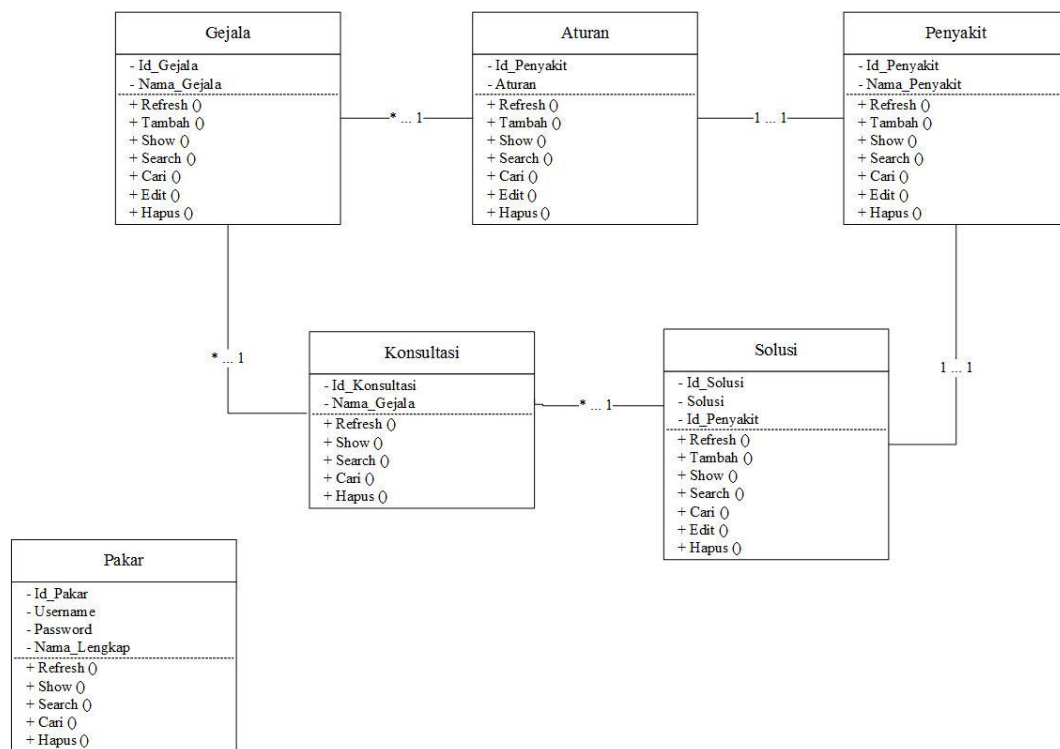
Gambar III.16. Sequence Diagram Daftar Konsultasi Sistem Pakar

Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward*

Chaining

III.3.4. Class Diagram

Class Diagram adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).



Gambar III.17. Class Diagram Sistem Pakar Mendiagnosa Gejala Virus

Covid-19 Menggunakan Metode *Forward Chaining*

III.4. Desain Sistem Detail

Desain sistem detail dari sistem pakar mendiagnosa gejala virus *Covid-19* menggunakan metode *forward chaining* adalah sebagai berikut:

III.4.1. Desain Output

Desain sistem secara *detail* mencakup desain output, desain *input*, dan desain *database*. Rancangan *output* laporan hasil konsultasi berfungsi menampilkan data hasil konsultasi pengguna. Adapun rancangan *output* laporan hasil konsultasi dapat dilihat pada gambar sebagai berikut :

1. Rancangan Hasil Diagnosa

The image shows a web-based interface for displaying diagnostic results. At the top right, there are two buttons: 'Kembali' (Back) and 'Logout'. Below these, the main content area is divided into two sections. The first section, titled 'Kondisi' (Condition), contains a large rectangular box with the text 'Keterangan Gejala yang telah dipilih' (Description of the selected symptoms). The second section, titled 'Hasil Diagnosa' (Diagnostic Result), contains a rounded rectangular box with the text 'Diagnosa & Solusi' (Diagnosis & Solution). Below this box are two buttons: 'Simpan' (Save) and 'Buang' (Discard).

Gambar III.18. Rancangan Hasil Diagnosa Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

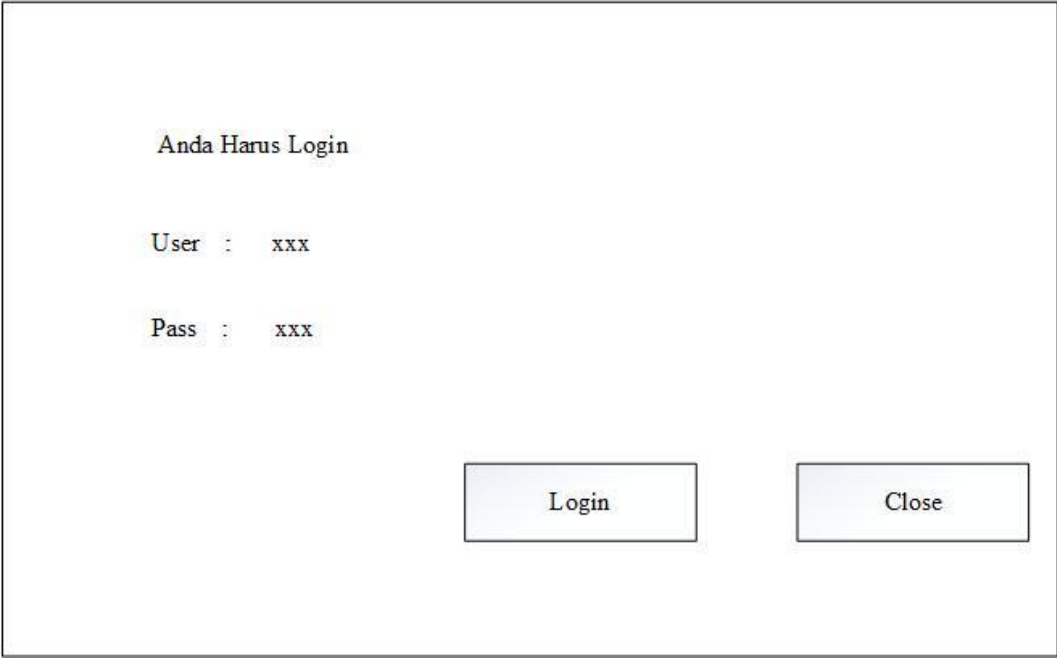
III.4.2. Desain Input

Perancangan *input* merupakan masukan yang penulis rancang guna lebih memudahkan dalam *entry* data. *Entry* data yang dirancang akan lebih mudah dan cepat dan meminimalisir kesalahan penulisan dan memudahkan perubahan.

Perancangan *input* tampilan yang dirancang adalah sebagai berikut:

1. Perancangan Form *Login*.

Perancangan form *login* berfungsi untuk verifikasi admin yang berhak menggunakan sistem. Adapun rancangan *form login* dapat dilihat pada gambar sebagai berikut :

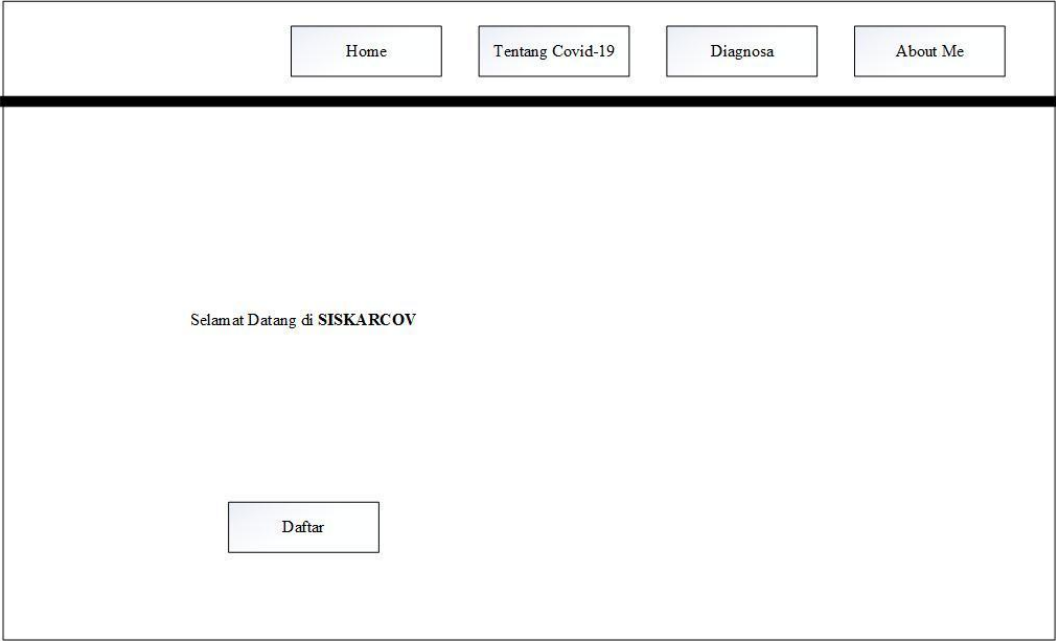


The image shows a login form with the title "Anda Harus Login". Below the title, there are two input fields: "User" and "Pass". The "User" field contains the text "xxx" and the "Pass" field also contains the text "xxx". At the bottom of the form, there are two buttons: "Login" and "Close".

Gambar III.19. Rancangan Form Login Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

2. Rancangan Form Menu Utama

Rancangan *input* form utama berfungsi untuk menampilkan tampilan utama dari *user interface*. Adapun rancangan menu utama dapat dilihat pada gambar sebagai berikut :



The image shows a web form layout for a COVID-19 expert system. At the top, there is a horizontal navigation bar containing four buttons: "Home", "Tentang Covid-19", "Diagnosa", and "About Me". Below this bar, the main content area displays the text "Selamat Datang di SISKARCOV" in the center. At the bottom of the main content area, there is a single button labeled "Daftar".

Gambar III.20. Rancangan Form Menu Utama Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

3. Rancangan Form Daftar

Rancangan *input* form daftar berfungsi untuk melakukan pendaftaran sebelum melakukan diagnosa mandiri. Adapun rancangan form daftar dapat dilihat pada gambar sebagai berikut :

Silahkan masukkan data Anda.

Nama :

Password :

Jenis Kelamin :

Tanggal lahir :

Login Close

Gambar III.21. Rancangan Form Daftar Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

4. Rancangan Form Pertanyaan

Rancangan *input* form penyakit berfungsi untuk menampilkan gejala-gejala yang akan dipilih. Adapun rancangan form penyakit dapat dilihat pada gambar sebagai berikut :

The image shows a rectangular window with a thin black border. Inside the window, the text "Pertanyaan" is at the top left. Below it, the question "Apakah anda mengalami ...?" is centered. At the bottom, there are two rectangular buttons: "Benar" on the left and "Tidak" on the right. Both buttons have a light blue background and a thin black border.

Gambar III.22. Rancangan Form Pertanyaan Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

5. Rancangan Form Diagnosa

Rancangan *input* form diagnosa berfungsi untuk menampilkan hadir dari gejala yang dipilih sebelumnya. Adapun rancangan form diagnosa dapat dilihat pada gambar sebagai berikut :

Nomor	Kondisi Anda
1	Gejala 1
2	Gejala 2

Hasil Diagnosa

Gambar III.23. Rancangan Form Diagnosa Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

6. Rancangan Form *Login* Admin

Rancangan *input* form *login* admin berfungsi untuk pengolahan data penyakit, data gejala, basis aturan, dan konsultasi. Adapun rancangan form *login* admin dapat dilihat pada gambar sebagai berikut :

Sistem Pakar
Covid-19

Username

Password

Login

Gambar III.24. Rancangan Form *Login* Admin Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

7. Rancangan Form Data Penyakit

Perancangan *form* data penyakit merupakan *form* untuk penyimpanan data-data penyakit. Adapun bentuk *form* data penyakit dapat dilihat pada gambar sebagai berikut:

Data Penyakit

Refresh

Tambah

Search

Pilih

No	ID Penyakit	Nama Penyakit	Aksi
x	xxx	xxx	(Edit) (Hapus)
Show			

Gambar III.25. Rancangan Form Data Penyakit Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

8. Rancangan Form Data Gejala

Perancangan *form* data gejala merupakan form yang berfungsi untuk mengolah data gejala. Adapun bentuk *form* gejala dapat dilihat pada gambar sebagai berikut:

Data Gejala

Refresh

Tambah

Search

Pilih

No	KD Gejala	Nama Gejala	Aksi
x	xxx	xxx	(Edit) (Hapus)

Show

Gambar III.26. Rancangan Form Data Gejala Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

9. Rancangan Form Data Basis Aturan

Perancangan *form* aturan merupakan *form* untuk mengolah data basis aturan.

Adapun bentuk *form* basis aturan dapat dilihat pada gambar sebagai berikut:

Data Rule / Basis Aturan

Refresh

Tambah

Search

Pilih

No	ID Penyakit	Nama Penyakit	Aturan	Aksi
x	xxx	xxx	xxx	(Edit) (Hapus)

Show

Gambar III.27. Rancangan Form Basis Aturan Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

10. Rancangan Form Solusi

Perancangan *form* daftar konsultasi merupakan form yang berfungsi untuk menampilkan daftar solusi dari gejala yang dipilih pengguna. Adapun bentuk *form* solusi dapat dilihat pada gambar sebagai berikut:

Data Solusi

Refresh

Tambah

Search

Pilih

No	KD Solusi	Solusi	Penyakit	Aksi
x	xxx	xxx	xxx	(Edit) (Hapus)

Show

Gambar III.28. Rancangan Form Solusi Sistem Pakar Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward Chaining*

11. Rancangan Form Data Konsultasi

Perancangan *form* daftar konsultasi merupakan form yang berfungsi untuk menampilkan daftar konsultasi pengguna. Adapun bentuk *form* konsultasi dapat dilihat pada gambar sebagai berikut:

Data Konsultasi

Refresh

Tambah

Search

Pilih

ID Konsultasi	TGL Konsultasi	Nama Pasien	TGL Lahir	Jenis Kelamin	Penyakit	Password	Aksi
x	xxx	xxx	xxx	xxx	xxx	xxx	(Edit) (Hapus)

Show

Gambar III.29. Rancangan Form Data Konsultasi Sistem Pakar
Mendiagnosa Gejala Virus Covid-19 Menggunakan Metode *Forward*
Chaining

III.5. Perancangan Database

III.5.1. Kamus Data (*Data Dictionaries*)

Kamus data merupakan suatu daftar terorganisasi tentang komposisi elemen data, aliran data dan data store yang digunakan. Pengisian data dictionary dilakukan setiap saat selama proses pengembangan berlangsung, ketika diketahui adanya data atau saat diperlukan penambahan data item ke dalam sistem. Berikut kamus data dari sistem yang penulis rancang yaitu :

1. penyakit = **id_penyakit** + nama_penyakit
2. solusi = **id_solusi** + solusi + id_penyakit

3. gejala = **id_gejala** + nama_gejala

4. basis aturan = **id_aturan** + aturan

5. konsultasi = **id_konsultasi** + tgl_konsultasi + nama_pasien + tgl_lahir +
jenis_kelamin + penyakit + password

III.5.2. Desain Tabel / Field

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

Berikut adalah desain *database* dan tabel dari sistem yang dirancang :

1. Tabel Penyakit

Nama Database : forchacovid

Nama Tabel : daftar_penyakit

Primary Key : id_penyakit

Foreign Key : -

Tabel III.6. Tabel Penyakit

Nama Field	Tipe Data	Ukuran	Keterangan
Id_penyakit	Varchar	10	*ID PENYAKIT
Nama_penyakit	Varchar	50	Nama Penyakit

2. Tabel Solusi

Nama Database : forchacovid

Nama Tabel : daftar_solusi

Primary Key : id_solusi

Foreign Key : -

Tabel III.7. Tabel Solusi

Nama Field	Tipe Data	Ukuran	Keterangan
Id_solusi	Varchar	19	*ID SOLUSI
Solusi	Varchar	300	Solusi
Id_penyakit	Varchar	10	ID PENYAKIT

3. Tabel Gejala

Nama Database : forchacovid

Nama Tabel : daftar_gejala

Primary Key : id_gejala

Tabel III.8. Tabel Gejala

Nama Field	Tipe Data	Ukuran	Keterangan
Id_gejala	Varchar	10	*ID Gejala
Nama_gejala	Varchar	150	Nama Gejala

4. Tabel Data Basis Aturan

Nama Database : forchacovid

Nama Tabel : basis_aturan

Primary Key : id_penyakit

Foreign Key : -

Tabel III.9. Tabel Basis Aturan

Nama Field	Tipe Data	Ukuran	Keterangan
Id_penyakit	Varchar	4	*ID PENYAKIT
Aturan	Varchar	200	Basis Aturan

5. Tabel Data Konsultasi

Nama Database : forchacovid

Nama Tabel : daftar_konsultasi

Primary Key : id_konsultasi

Foreign Key : -

Tabel III.10. Tabel Daftar Konsultasi

Nama Field	Tipe Data	Ukuran	Keterangan
Id_user	Varchar	15	*ID PENYAKIT
Nama	Varchar	35	Nama Lengkap
Password	Varchar	13	Password
Jenis_kelamin	Varchar	10	Jenis Kelamin

Tgl_lahir	Date	-	Tanggal Lahir
Tgl_diagnosa	Date	-	Tanggal Diagnosa
Keterangan	Varchar	35	Keterangan

III.5.3 Normalisasi

Normalisasi merupakan cara pendekatan dalam membangun desain logika basis data relasional yang tidak secara langsung berkaitan dengan model data, tetapi dengan menerapkan sejumlah aturan dan kriteria standart untuk menghasilkan struktur tabel yang normal. Bentuk – bentuk normalisasi pada rancangan *database* adalah sebagai berikut :

1. Tabel Penyakit

Tabel penyakit memiliki atribut: id_penyakit dan nama_penyakit. Melihat struktur tabel tersebut tidak ada redundansi sehingga sudah memenuhi bentuk normalisasi pertama (1NF).

2. Tabel Solusi

Tabel solusi memiliki atribut: id_solusi, solusi, dan ida_penyakit. Melihat struktur tabel tersebut tidak ada redundansi sehingga sudah memenuhi bentuk normalisasi pertama (1NF).

3. Tabel Gejala

Tabel gejala memiliki atribut: id_gejala dan nama_gejala. Melihat struktur tabel tersebut tidak ada redundansi sehingga sudah memenuhi bentuk normalisasi pertama (1NF).

4. Tabel Basis Aturan

Tabel aturan memiliki atribut: id_penyakit dan aturan. Melihat struktur tabel tersebut tidak ada redundansi sehingga sudah memenuhi bentuk normalisasi pertama (1NF).

6. Tabel Konsultasi

Tabel konsultasi memiliki atribut : id_user, nama, password, jenis_kelamin, tgl_lahir, tgl_diagnosa, dan keterangan. Melihat struktur tabel tersebut tidak ada redundansi sehingga sudah memenuhi bentuk normalisasi pertama (1NF).

Jadi, dalam hal ini dapat dinyatakan bahwa rancangan tabel pada *database* sudah normal. Artinya sistem akan melakukan aktifitasnya sesuai dengan yang telah ditargetkan sebelumnya karena tidak ada redundansi atau duplikasi data.