

## BAB III METODE PENELITIAN

### 3.1 Tahap Pengumpulan Data

Pengumpulan data diperlukan untuk mendapatkan data yang dibutuhkan dalam membangun sistem perancangan sistem pakar diagnosa penyakit COVID-19 dengan metode *fuzzy logic*. Metode yang digunakan untuk mendapatkan data tersebut yaitu :

#### 3.1.1 Studi Literatur

Studi literatur dibutuhkan untuk menunjang pemahaman dan pengetahuan penulis tentang materi, konsep, teori, dan metode apa yang diperlukan dalam proses pengerjaan tugas akhir ini. Studi literatur yang dilakukan meliputi penelitian-penelitian terdahulu di berbagai jurnal, buku, dan *e-book*. Adapun hal-hal yang harus dipahami melalui studi literatur adalah beberapa penelitian terdahulu mengenai metode *fuzzy logic* untuk mendapatkan alternatif terbaik dalam diagnosa penyakit covid-19. **Tabel 3.1** berikut ini merupakan penelitian terdahulu yang terkait dengan metode *fuzzy logic*.

**Tabel 3.1** Tabel Penelitian Terdahulu.

<b>Nama</b>	<b>Judul</b>	<b>Terbit/ Tahun</b>	<b>Keterangan</b>
Immanuel Sinuraya, Agung Triayudi, Ira Diana Sholihati	Sistem Pakar Diagnosis Covid-19 Berbasis Mobile Application Android Dengan Metode Certainty Factor	Universitas Nasional , 2020;	Sistem ini dapat menganalisis jenis Covid-19 berdasarkan keluhan atau gejala yang di input oleh client. Sistem ini juga mampu menyimpan representasi pengetahuan pakar berdasarkan nilai kepercayaan (MB) dan nilai tidak kepercayaan (MD) berupa nilai akhir CF, dan dapat bekerja layaknya seorang pakar dalam mengdiagnosa Covid- 19 .

**Tabel 3.1** (lanjutan).

Sunardi dan Rara Sri Artati	Aplikasi Metode Fuzzy Sugeno untuk Sistem Informasi Ketinggian Air dan Ketinggian Pintu Air Suatu Bendungan	Universitas Stikubank Semarang , 2014;	User yang akan menangani sistem ini adalah user admin, user ini memiliki kewenangan untuk mensetup preferensi dan memasukan aturan fuzzy, serta mencoba pintu air secara manual.
Dany Suktiawan Irman Fiano , Agus Sidiq Purnomo	Sistem Pakar Untuk Mendeteksi Tingkat Resiko Penyakit Jantung Dengan Fuzzy Inferensi	Universitas Mercu Buana, 2017	Berdasarkan penelitian dan pengujian untuk mendeteksi tingkat resiko penyakit jantung menggunakan metode fuzzy inferensi (Mamdani), disarankan untuk penelitian selanjutnya yang berkaitan mengenai penentuan deteksi tingkat resiko penyakit jantung dapat disempurnakan lagi dengan mengelompokkan berdasarkan jenis kelamin dan usia
Putu Ananta Dama Putra, I Ketut Adi Purnawan, Desy Purnami Singgih Putri	Sistem Pakar Diagnosa Penyakit Mata dengan Fuzzy Logic dan Naïve Bayes	Universitas Udayana, 2018	Implementasi Fuzzy Logic pada aplikasi adalah untuk membantu menghasilkan jawaban dengan nilai pasti dari nilai tidak pasti yang diberikan oleh pasien.

### 3.1.2 Wawancara

Teknik wawancara merupakan metode pengumpulan data dengan cara mengajukan pertanyaan-pertanyaan atau tanya jawab langsung dengan narasumber yang berhubungan dengan masalah-masalah yang dibahas. Dalam hal ini peneliti yaitu melakukan tanya jawab antara penulis dengan orang pakar (ahli) dalam hal ini adalah dokter paru-paru, sehingga mendapatkan data yang akurat.

### 3.2 Analisis Kebutuhan Sistem

Pada tahap ini yang dilakukan adalah menganalisis kebutuhan sistem untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*. Analisis kebutuhan perangkat lunak dalam penelitian ini yaitu dengan cara mengidentifikasi permasalahan yang ada untuk

dicatat dan dijadikan bahan untuk mulai membangun sistem pakar diagnosa penyakit COVID-19.

Analisis kebutuhan yang dilakukan meliputi pengumpulan data kebutuhan *fungsional* dan kebutuhan *non-fungsional*.

### **3.2.1 Analisis Kebutuhan *Fungsional***

Kebutuhan fungsional merupakan layanan yang harus disediakan oleh sistem, yaitu sebagai berikut :

1. Sistem mampu menampilkan *form* gejala.
2. Sistem dapat mengelola data gejala, data penyakit, dan data alternatif
3. Sistem dapat menampilkan solusi dari diagnosa.
4. Sistem mampu melakukan diagnosa dengan metode *fuzzy logic*.

### **3.2.2 Analisis Kebutuhan *Non Fungsional***

Analisis kebutuhan *non fungsional* yang dimaksud adalah spesifikasi dari perangkat keras dan perangkat lunak yang digunakan dalam sistem yang akan dibangun.

#### **1. Analisis Kebutuhan Perangkat Lunak (*Software*)**

Perangkat lunak yang dikembangkan dalam sistem pakar diagnosa penyakit covid-19 ini berbasis *website* yang dijalankan secara *online*. Perangkat lunak yang disarankan untuk menjalankan aplikasi ini sebagai berikut :

- a. Sistem Operasi yang digunakan adalah *Microsoft Windows 10*
- b. *Web Server* menggunakan XAMPP
- c. *Web Browser* Internet (*Mozilla Firefox* atau *Google Chrome*)
- d. *Editor Interface* menggunakan *Sublime Text*
- e. *Provider* yang mempunyai koneksi stabil.

#### **2. Analisis Kebutuhan Perangkat Keras (*Hardware*)**

Spesifikasi *hardware* yang digunakan untuk membangun sistem pakar diagnosa penyakit COVID-19 sebagai berikut :

- a. *Processor* Intel(R) Core(TM) i3-4005U 1.70GHz
- b. Monitor 14"

- c. RAM 2 GB
- d. *Harddisk* 500 GB
- e. *Keyboard* dan *mouse* Optik.

### **3.3 Tahapan Perancangan Perangkat Lunak**

Model yang digunakan dalam perancangan sistem pada penelitian ini adalah menggunakan model *prototype* dan *fuzzy logic sugeno*.

#### **3.3.1 *Prototype***

Peneliti menggunakan metode *protype* karena merupakan salah satu jenis metode pengembangan sistem yang sifatnya sangat cepat dan dapat menghemat waktu. Berikut ini merupakan langkah – langkah dari metode *prototype* yang dilakukan :

##### **3.3.1.1 Komunikasi**

Pada tahap ini peneliti mengumpulkan data menggunakan metode wawancara dan studi literatur.

##### **3.3.1.2 Perencanaan Secara Cepat**

Pada tahap ini peneliti mengolah data hasil wawancara dan studi literatur (Proses Fuzzifikasi), peneliti juga melakukan analisis data dengan menggunakan metode *fuzzy logic sugeno*.

##### **3.3.1.3 Pemodelan Perancangan Secara Cepat**

Peneliti melakukan perancangan secara cepat menggunakan DFD (*Data Flow Diagram*) sebagai rancangan sistem, ERD (*Entity Relationship Diagram*) sebagai rancangan *database* dan membuat desain tampilan *website* (*User Interface*).

##### **3.3.1.4 Pembentukan *Prototype***

Pada tahap ini peneliti membuat *website* menggunakan bahasa pemrograman *HTML, PHP, CSS, Java Script* dan menggunakan *Query* untuk membuat *Database*.

### **3.3.1.5 Penyerahan Sistem Pengguna, Pengirim dan Umpan Balik**

Peneliti melakukan ujicoba program kepada *user* kemudian melakukan umpan balik terhadap program yang telah diujicoba.

### **3.3.2 Metode *Fuzzy Logic***

Pada tahap ini akan dijelaskan tentang sumber data yang digunakan dan langkah dalam mengimplementasikan sistem pakar diagnosa penyakit COVID-19 dengan metode *fuzzy logic*.

#### **3.3.2.1 Proses Data Penelitian Dalam Sistem Pakar Diagnosa Penyakit COVID-19 dengan Metode *Fuzzy Logic***

Representasi pengetahuan bertujuan membuat struktur yang akan digunakan dalam sistem untuk membantu pengadaan pengetahuan ke dalam program. Pengetahuan direpresentasikan ke dalam format tertentu dan akan disimpan dalam suatu basis pengetahuan. Langkah-langkah atau base rule yang perlu dilakukan untuk membuat representasi pengetahuan sistem pakar ini adalah :

1. Pembuatan tabel keputusan (decision table) yang berguna untuk mendokumentasikan dan mendeskripsikan pengetahuan.
2. Pembuatan pohon keputusan (decision tree) yang berguna untuk menghilangkan kaidah-kaidah dengan tujuan untuk meniadakan terjadinya perulangan pertanyaan.
3. Pembuatan basis aturan untuk menentukan jenis penyakit.

Sistem pakar ini terdiri dari sebuah pohon keputusan untuk diagnosa penyakit COVID-19, serta himpunan kaidah untuk mendiagnosa COVID-19. Proses representasi pengetahuan untuk sistem pakar ini diawali dengan proses masuk ke dalam sistem dan kemudian sistem ini akan melakukan diagnosa penyakit COVID-19. Untuk mendiagnosa penyakit COVID-19 diawali dengan menjawab pertanyaan pertanyaan yang dirasakan oleh pasien maka setelah dilakukan diagnosa oleh sistem maka akan menghasilkan sebuah diagnosa.

Berdasarkan survey dan pengumpulan data yang diambil oleh penulis didapatkan pengetahuan-pengetahuan yang akan menjadi dasar rekayasa pengetahuan. Diantaranya pengetahuan yang didapatkan dari hasil survey sebagai berikut:

### 1. COVID-19

*Deskripsi* : Penyakit ini terjadi akibat infeksi virus yang menyerang saluran pernapasan. Masa inkubasi flu termasuk singkat. Anda akan mengalami gejala hanya dalam beberapa hari setelah pertama kali terinfeksi. Masa di mana flu paling menular adalah sehari sebelum gejala muncul dan sekitar enam hari berikutnya. Tabel Gejala dapat dilihat pada **Tabel 3.2** berikut.

**Tabel 3.2** Gejala.

KODE	GEJALA	SOLUSI
G-01	Sakit Tenggorokan	Beberapa langkah pengobatan yang bisa dilakukan adalah istirahat yang cukup, banyak minum, serta menjaga tubuh agar tetap hangat. Anda tidak dianjurkan mengonsumsi antibiotik karena obat ini berfungsi untuk membunuh bakteri sedangkan flu disebabkan oleh virus.
G-02	Sesak Nafas	
G-03	Pilek	
G-04	Batuk yang berkelanjutan	
G-05	Demam	
G-06	Menggigil	

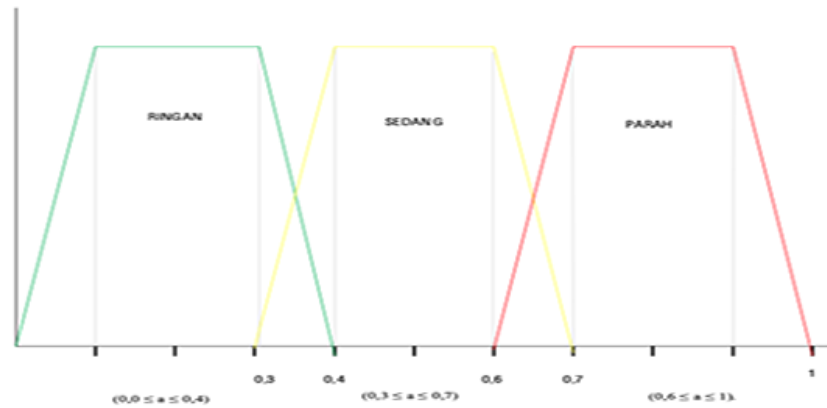
### 1. Proses Fuzzifikasi

Proses fuzzifikasi terhadap variabel Sakit Tenggorokan, Sesak Nafas, Pilek, Batuk yang Berkelanjutan, Demam, Menggigil. Adapun penilaian dari setiap variabel adalah sebagai berikut :

#### a. Variabel Sakit Tenggorokan ( $V_1$ )

Dalam variabel (sakit tenggorokan) terbagi atas 4 (empat) bilangan *fuzzy* yaitu Tidak Tahu (T), Mungkin (M), Kemungkinan Besar (K), dan Pasti (P) seperti

terlihat pada **Gambar 3.1**. Dimana bilangan-bilangan fuzzy tersebut dikonversikan ke bilang crisp seperti terlihat pada **Tabel 3.2**.



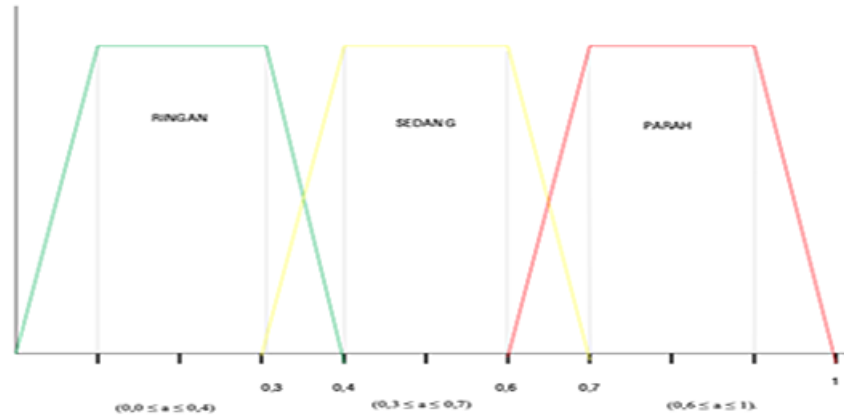
**Gambar 3.1** Bilangan *Fuzzy* Sakit Tenggorokan

**Tabel 3.3** Keterangan Bilangan *Fuzzy* Untuk Sakit Tenggorokan.

Bilangan <i>fuzzy</i>	Bilangan <i>crisp</i>
Tidak Tahu (T)	0
Mungkin (M)	0,15
Kemungkinan Besar (K)	0,5
Pasti (P)	0,8

b. Variabel Sesak Nafas ( $V_2$ )

Dalam variabel (sesak nafas) terbagi atas 4 (empat) bilangan *fuzzy* yaitu Tidak Tahu (T), Mungkin (M), Kemungkinan Besar (K), dan Pasti (P) seperti terlihat pada **Gambar 3.2**. Dimana bilangan-bilangan fuzzy tersebut dikonversikan ke bilang crisp seperti terlihat pada **Tabel 3.3**.



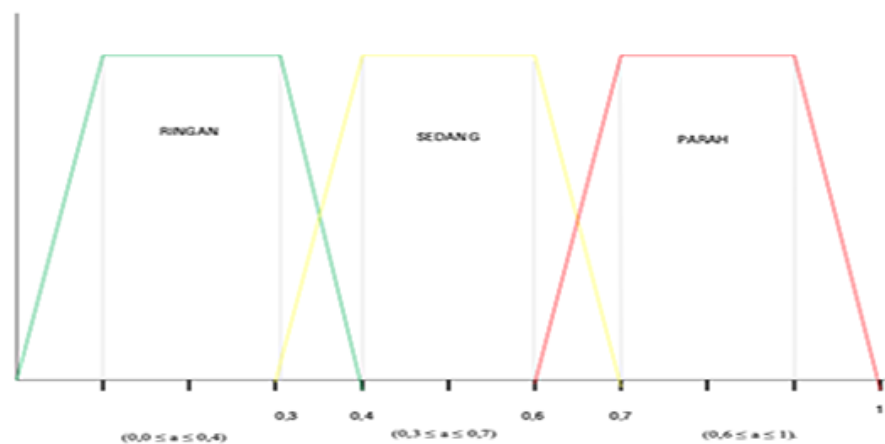
**Gambar 3.2** Bilangan *Fuzzy* Sesak Nafas

**Tabel 3.4** Keterangan Bilangan *Fuzzy* Untuk Sesak Nafas.

Bilangan <i>fuzzy</i>	Bilangan <i>crisp</i>
Tidak Tahu (T)	0
Mungkin (M)	0,15
Kemungkinan Besar (K)	0,5
Pasti (P)	0,8

c. Variabel Pilek ( $V_3$ )

Dalam variabel (pilek) terbagi atas 4 (empat) bilangan *fuzzy* yaitu Tidak Tahu (T), Mungkin (M), Kemungkinan Besar (K), dan Pasti (P) seperti terlihat pada **Gambar 3.3**. Dimana bilangan-bilangan fuzzy tersebut dikonversikan ke bilang crisp seperti terlihat pada **Tabel 3.4**.



**Gambar 3.3** Bilangan *Fuzzy* Pilek

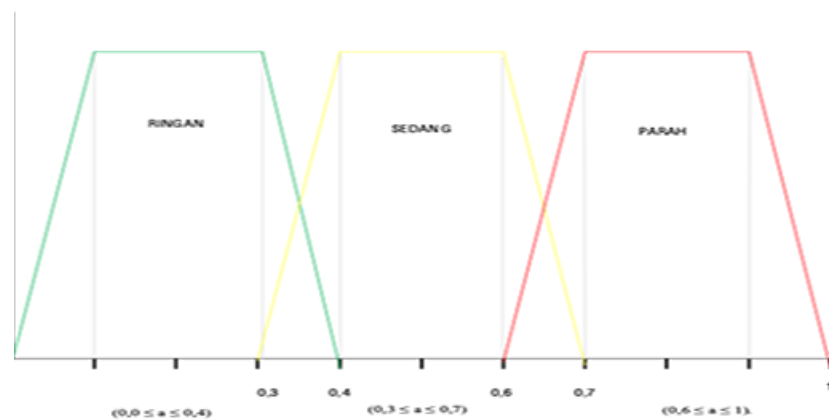


**Tabel 3.5** Keterangan Bilangan *Fuzzy* Untuk Pilek.

Bilangan <i>fuzzy</i>	Bilangan <i>crisp</i>
Tidak Tahu (T)	0
Mungkin (M)	0,15
Kemungkinan Besar (K)	0,5
Pasti (P)	0,8

d. Variabel Batuk yang Berkelanjutan ( $V_4$ )

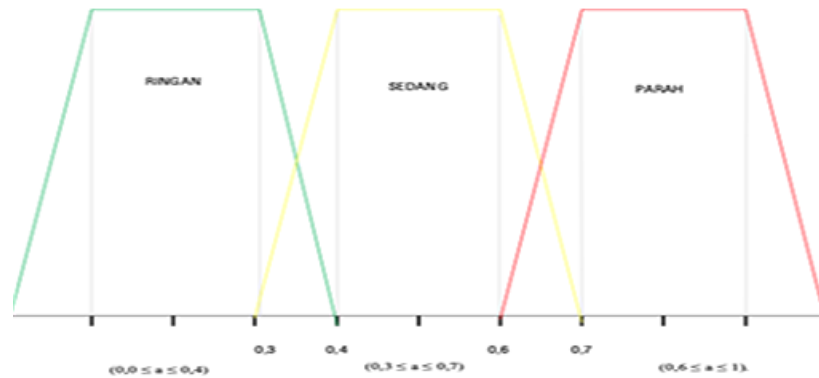
Dalam variabel (batuk yang berkelanjutan) terbagi atas 4 (empat) bilangan *fuzzy* yaitu Tidak Tahu (T), Mungkin (M), Kemungkinan Besar (K), dan Pasti (P) seperti terlihat pada **Gambar 3.4**. Dimana bilangan-bilangan fuzzy tersebut dikonversikan ke bilangan crisp seperti terlihat pada **Tabel 3.5**.

**Gambar 3.4** Bilangan *Fuzzy* Batuk yang Berkelanjutan**Tabel 3.6** Keterangan Bilangan *Fuzzy* Untuk Batuk yang Berkelanjutan.

Bilangan <i>fuzzy</i>	Bilangan <i>crisp</i>
Tidak Tahu (T)	0
Mungkin (M)	0,15
Kemungkinan Besar (K)	0,5
Pasti (P)	0,8

e. Variabel Demam ( $V_5$ )

Dalam variabel (demam) terbagi atas 4 (empat) bilangan *fuzzy* yaitu Tidak Tahu (T), Mungkin (M), Kemungkinan Besar (K), dan Pasti (P) seperti terlihat pada **Gambar 3.5**. Dimana bilangan-bilangan fuzzy tersebut dikonversikan ke bilang crisp seperti terlihat pada **Tabel 3.6**.



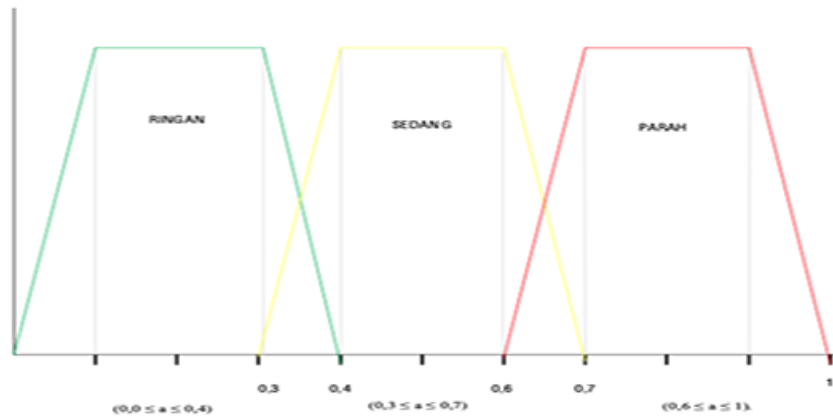
**Gambar 3.5** Bilangan *Fuzzy* Demam

**Tabel 3.7** Keterangan Bilangan *Fuzzy* Untuk Demam.

Bilangan <i>fuzzy</i>	Bilangan <i>crisp</i>
Tidak Tahu (T)	0
Mungkin (M)	0,15
Kemungkinan Besar (K)	0,5
Pasti (P)	0,8

f. Variabel Menggigil ( $V_6$ )

Dalam variabel (menggigil) terbagi atas 4 (empat) bilangan *fuzzy* yaitu Tidak Tahu (T), Mungkin (M), Kemungkinan Besar (K), dan Pasti (P) seperti terlihat pada **Gambar 3.6**. Dimana bilangan-bilangan fuzzy tersebut dikonversikan ke bilang crisp seperti terlihat pada **Tabel 3.7**.



**Gambar 3.6** Bilangan *Fuzzy* Menggigil

**Tabel 3.8** Keterangan Bilangan *Fuzzy* Untuk Menggigil.

Bilangan <i>fuzzy</i>	Bilangan <i>crisp</i>
Tidak Tahu (T)	0
Mungkin (M)	0,15
Kemungkinan Besar (K)	0,5
Pasti (P)	0,8

### 3.3.2.2 Penerapan Metode *Fuzzy Sugeno*

Berikut contoh penyelesaian identifikasi penyakit menggunakan metode *fuzzy sugeno*. dengan gejala – gejala sebagai berikut :

**Tabel 3.9** Gejala-gejala yang telah di pilih

Kode Gejala	Nama Gejala	Kategori	Nilai Bobot
G-01	Sakit Tenggorokan	Mungkin	0,15
G-03	Pilek	Kemungkinan Besar	0,5
G-02	Sesak Nafas	Pasti	0,8
G-06	Menggigil	Mungkin	0,15
G-05	Demam	Pasti	0,8

Menghitung proses *fuzzifikasi* dengan rumus sebagai berikut:

Rumus umum untuk *fuzzifikasi* metode *Fuzzy Sugeno* yaitu sebagai berikut:

$$\mu[x, a, b, c] = \begin{cases} 0 & x \leq a \text{ atau } x \geq c \\ \frac{(x-a)}{(b-a)} & ; a \leq x \leq b \\ \frac{(c-x)}{(c-b)} & ; b \leq x \leq c \end{cases} \dots\dots\dots(1)$$

$$b = \frac{\sum a \text{ sampai } b}{n}$$

Keterangan:

x = Bobot nilai yang sudah ditentukan pada setiap gejala yang dipilih a = Batas nilai minimum pada setiap gejala

b = Nilai tengah dari batas minimum dan maksimum

c = Batas nilai maksimum pada setiap gejala

1. Selanjutnya, menghitung nilai b (nilai tengah dari batas minimum dan maksimum) dari setiap kategori tingkat tidak tahu, mungkin, kemungkinan besar, pasti. Dengan rumus sebagai berikut:

a) Ringan dengan *range* interval  $(0,0 \leq a \leq 0,4)$

$$b = \frac{0,0 + 0,1 + 0,2 + 0,3 + 0,4}{5} = \frac{1}{5} = 0,2$$

Jadi nilai tengah dari batas minimum dan maksimum pada kategori ringan adalah 0,2

b) Ringan dengan *range* interval  $(0,3 \leq a \leq 0,7)$

$$b = \frac{0,0 + 0,1 + 0,2 + 0,3 + 0,4}{5} = \frac{1}{5} = 0,5$$

Jadi nilai tengah dari batas minimum dan maksimum pada kategori sedang adalah 0,5

c) Ringan dengan *range* interval  $(0,6 \leq a \leq 0,1)$

$$b = \frac{0,0 + 0,1 + 0,2 + 0,3 + 0,4}{5} = \frac{1}{5} = 0,8$$

Jadi nilai tengah dari batas minimum dan maksimum pada kategori berat adalah 0,8

2. Menghitung nilai *fuzzifikasi* dari G-01, G-03, G-02, G-06, G-05.

$$F = \frac{(x - a)}{(b - a)}$$

- a. Menghitung F (G-01)

$$b = \frac{(0,15 - 0,0)}{(0,2 - 0,0)} = 0,75$$

Jadi hasil proses *fuzzyfikasi* G-01 adalah 0,75

- b. Menghitung F (G-02 dan G-05) gejala ini memiliki bobot nilai 0,8.

$$b = \frac{(0,8 - 0,6)}{(0,8 - 0,6)} = 1$$

Jadi hasil proses *fuzzyfikasi* G-02 dan G-05 adalah 1.

- c. Menghitung F (G-03) gejala ini memiliki bobot nilai 0,5.

$$b = \frac{(0,5 - 0,3)}{(0,5 - 0,3)} = 1$$

Jadi hasil proses *fuzzyfikasi* G-03 adalah 1

3. Menghitung Proses *Defuzzifikasi*

Rumus umum untuk *defuzzifikasi* metode *Fuzzy Sugeno* yaitu sebagai berikut:

Keterangan:

WA = (*Weighted Average*) Nilai rata-rata

$a_n$  = Nilai predikat aturan ke-n

$z_n$  = Indeks nilai *input(konstanta)* ke-n

$$WA = \frac{\alpha_1 z_1 + \alpha_2 z_2 + \alpha_3 z_3 + \dots + \alpha_n z_n}{\alpha_1 + \alpha_2 + \alpha_3 + \dots + \alpha_n}$$

$$\begin{aligned} WA &= (F.G-01 \times BN.G-01) + (F.G-03 \times BN.G-01) + (F.G-02 \times BN.G001) + (F.G- \\ &05 \times BN.G-01) + (F.G-06 \times BN.G0-01) / F.G001 + F.G003 + F.G004 + F.G005 + \\ &F.G006 = ((0,75 \times 0,15) + (1 \times 0,8) + (1 \times 0,5) + (1 \times 0,8) + (1 \times 0,5)) / 0,75 + 1 + 1 + 1 + 1 \\ &= (0,1125 + 0,8 + 0,5 + 0,8 + 0,5) / (4,75) \\ &= 2,7125 / 4,75 \\ &= 0,571 \end{aligned}$$

Maka, nilai *defuzzifikasi* (Tingkat kepengaruh) adalah  $0,571 \times 100\% = 57,1\%$   
Maka dapat disimpulkan bahwa hasil *defuzzifikasi* **57,1%** masuk dalam kategori **Berat**.

### 3.4 Desain Sistem

Desain sistem merupakan proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean.

#### 3.4.1 Perancangan Sistem

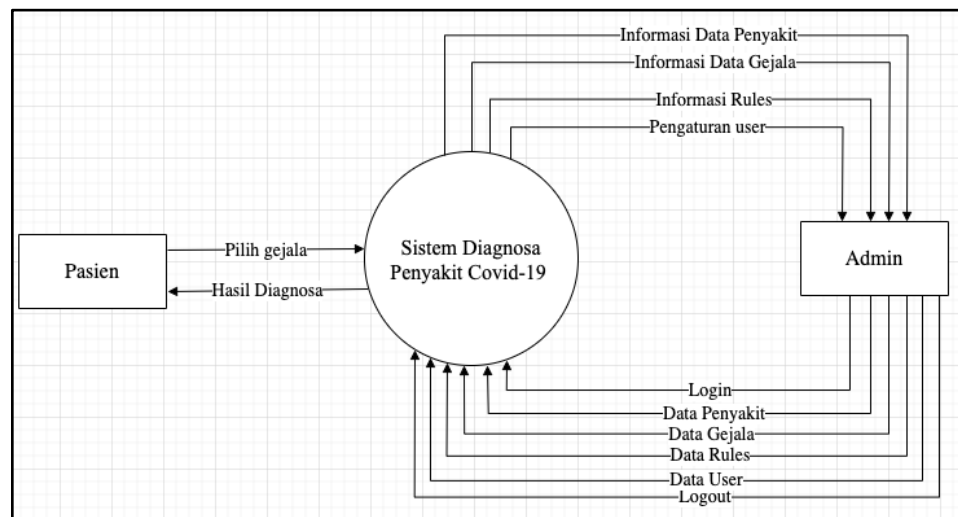
Perancangan sistem digunakan untuk memodelkan perancangan yang telah ditetapkan berdasarkan analisis sehingga menghasilkan informasi yang dibutuhkan dan dapat diimplementasikan menjadi program pada tahap selanjutnya. Urutan perancangan sistem pada penelitian ini adalah berupa *Data Flow Diagram* (DFD), *Entity Relation Diagram* (ERD), *Physical Data Model* (PDM), dan Struktur (Rancangan) *Database*.

### 3.4.1.1 Data Flow Diagram (DFD)

Data flow diagram (DFD) merupakan representasi grafik yang menggambarkan aliran informasi dan transformasi informasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*). DFD dibagi menjadi beberapa level yang lebih detail, dimana diawali dengan *Context Diagram* yang menjelaskan hubungan atau interaksi sistem dengan entitas-entitas yang mempunyai keterkaitan dengan sistem.

#### 1. Diagram Konteks (*Context Diagram*)

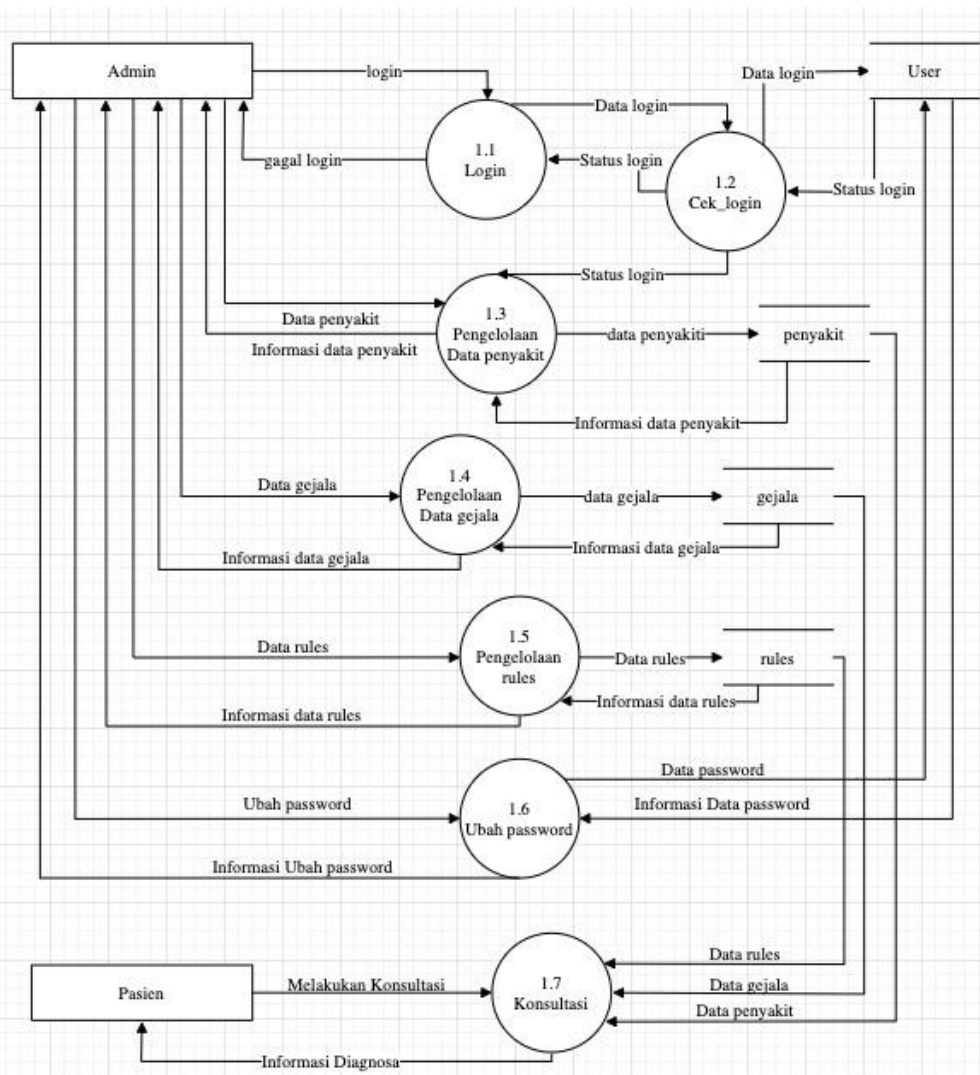
*Context Diagram* sering disebut juga dengan DFD Level 0, *Context Diagram* digunakan untuk menggambarkan interaksi antara sistem yang akan dibangun dengan entitas luar. Pada *Context Diagram* sistem seleksi pemilihan asisten terdapat 2 entitas, yaitu Admin dan pasien. Masing-masing entitas tersebut memberikan *input* dan diberikan *output* berupa informasi berkaitan dengan sistem pakar diagnosa penyakit COVID-19 seperti terlihat pada **Gambar 3.7** berikut.



**Gambar 3.7** *Context Diagram* Sistem Pakar Diagnosa Penyakit COVID-19.

## 2. Data Flow Diagram (DFD) Level 1

DFD Level 1 digunakan untuk menggambarkan modul-modul yang ada dalam sistem. Level 1 membentuk semua aliran proses *input* dan *output* yang ada pada *context diagram* sebelumnya. Tiap-tiap proses tersebut membentuk aliran proses yang menggambarkan proses dari rancangan sistem pakar diagnosa penyakit COVID-19. Seperti terlihat pada **Gambar 3.8** berikut.



**Gambar 3.8** DFD Level 1 Sistem Pakar Diagnosa Penyakit COVID-19.



Guna menjelaskan fungsi-fungsi aliran data (*input/output*) pada **Gambar 3.8** sistem tersebut, maka pada **Tabel 3.9** berikut dijelaskan mengenai proses-proses yang terkait pada DFD Level 1.

**Tabel 3.10** Proses-Proses Pada DFD Level 1.

No. 1	Nama Proses	:	<i>Login</i>
	Digunakan di	:	Proses <i>login</i> (masukkan/ <i>input</i> )
	Deskripsi	:	<ol style="list-style-type: none"> <li>1. Masukkan <i>username</i> dan <i>password</i></li> <li>2. Apabila berhasil, maka dapat mengakses halaman utama sistem. Sedangkan jika gagal (<i>username</i> atau <i>password</i> salah) maka akan menampilkan pesan gagal <i>login</i>.</li> </ol>
No.2	Nama Proses	:	Cek status <i>Login</i>
	Digunakan di	:	<ol style="list-style-type: none"> <li>1. Proses memeriksa status <i>login</i></li> <li>2. Proses mengelola data penyakit</li> <li>3. Proses mengelola data gejala</li> <li>4. Proses mengelola data rules</li> <li>5. Proses mengubah <i>password</i></li> </ol>
	Deskripsi	:	<ol style="list-style-type: none"> <li>1. Jika <i>login</i> berhasil dilakukan maka <i>user</i> dapat mengakses fitur-fitur yang tersedia di dalam sistem</li> <li>2. Jika tidak berhasil melakukan <i>login</i> maka <i>user</i> (admin) tidak bisa mengakses fitur-fitur yang ada.</li> </ol>
No. 3	Nama Proses	:	Input Data Penyakit
	Digunakan di	:	<ol style="list-style-type: none"> <li>1. Proses menambah data penyakit dalam sistem pakar diagnosa penyakit covid-19</li> <li>2. Proses mengubah data penyakit</li> <li>3. Proses menghapus data penyakit</li> </ol>
	Deskripsi	:	<i>User</i> (admin) dapat mengelola data penyakit yang digunakan dalam melakukan sistem pakar diagnosa penyakit covid-19.

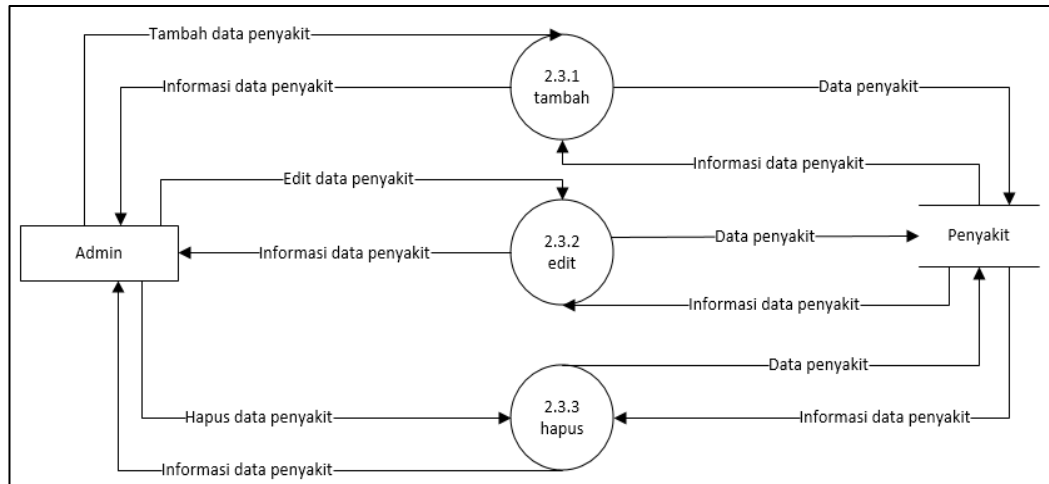
**Tabel 3.10** (lanjutan).

No.4	Nama Proses	:	Input Data Gejala
	Digunakan di	:	<ol style="list-style-type: none"> <li>1. Proses menambah data penyakit dalam sistem pakar diagnosa penyakit covid-19</li> <li>2. Proses mengubah data penyakit</li> <li>3. Proses menghapus data penyakit</li> </ol>
	Deskripsi	:	<i>User</i> (admin) dapat mengelola data gejala yang digunakan dalam melakukan sistem pakar diagnosa penyakit covid-19
No.5	Nama Proses	:	Input Data Rules
	Digunakan di	:	<ol style="list-style-type: none"> <li>1. Proses menambah data rules dalam sistem pakar diagnosa penyakit covid-19</li> <li>2. Proses menghapus data rules</li> </ol>
	Deskripsi	:	<i>User</i> (admin) dapat melakukan penambahan data rules dengan memilih data penyakit dan memasukan gejala-gejala yang telah ditentukan.
No.6	Nama Proses	:	Ubah <i>Password</i>
	Digunakan di	:	Proses mengubah data password user (admin)
	Deskripsi	:	<i>User</i> (admin) dapat mengubah password dengan mengisi form password baru yang akan diganti.
No.7	Nama Proses	:	Proses Diagnosa
	Digunakan di	:	Proses melakukan sistem pakar diagnosa penyakit covid-19
	Deskripsi	:	<i>User</i> (petani) memilih pertanyaan sesuai dengan gejala-gejala yang telah di alami pada Sistem Pakar Diagnosa Penyakit COVID-19.

### 3. Data Flow Diagram (DFD) Level 2

Merupakan hasil *breakdown* dari level sebelumnya (level 1), dimana dalam mem-*breakdown* bergantung pada tingkat kedetailan modul (proses dalam

sistem). Dalam sistem pakar diagnosa penyakit covid-19, proses input data penyakit dilakukan *breakdown* karena dalam proses tersebut terdapat proses menambah, menyimpan, mengedit, dan menghapus data penyakit seperti terlihat pada **Gambar 3.9** berikut.



**Gambar 3.9** DFD Level 2 Proses Input Data Penyakit

Untuk lebih jelas, DFD level 2 proses input data penyakit dapat dilihat pada **Tabel 3.10** berikut.

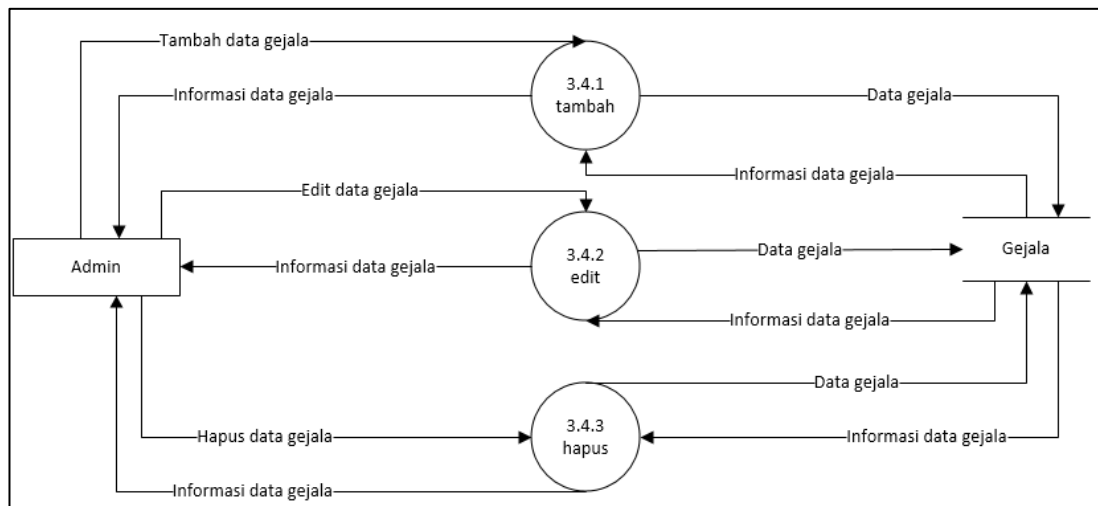
**Tabel 3.11** Penjelasan DFD Level 2 Proses Input Data Penyakit.

No 3.1	Nama Proses	:	Tambah
	Digunakan di	:	Proses menambah data penyakit
	Deskripsi	:	Admin bisa melakukan penambahan penyakit apabila perlu adanya data penyakit tambahan
No 3.2	Nama Proses	:	Edit
	Digunakan di	:	Proses mengubah data penyakit
	Deskripsi	:	Admin dapat mengubah data penyakit apabila perlu adanya perubahan data dalam data penyakit

**Tabel 3.11** (lanjutan).

No 3.3	Nama Proses	:	Hapus
	Digunakan di	:	Proses menghapus penyakit
	Deskripsi	:	Admin dapat menghapus data penyakit apabila data penyakit tersebut hendak dihilangkan

Selain itu, pada proses pengelolaan data gejala pada DFD level 1 juga dilakukan *breakdown* karena dalam proses tersebut terdapat proses menambah, menyimpan, mengedit, dan menghapus data gejala seperti terlihat pada **Gambar 3.10** berikut.

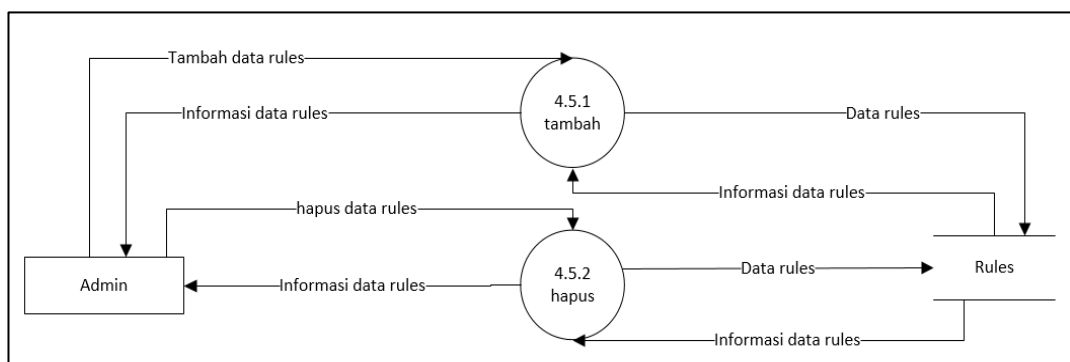
**Gambar 3.10** DFD Level 2 Proses Input data Gejala

Pada **Tabel 3.11** berikut adalah penjelasan mengenai DFD level 2 proses input data gejala.

**Tabel 3.12** Penjelasan DFD Level 2 Proses Input Data Gejala.

No 4.1	Nama Proses	:	Tambah
	Digunakan di	:	Proses menambah data gejala
	Deskripsi	:	Admin bisa melakukan penambahan gejala apabila perlu adanya data penyakit tambahan
No 4.2	Nama Proses	:	Edit
	Digunakan di	:	Proses mengubah data gejala
	Deskripsi	:	Admin dapat mengubah data gejala apabila perlu adanya perubahan data dalam data gejala
No 4.3	Nama Proses	:	Hapus
	Digunakan di	:	Proses menghapus gejala
	Deskripsi	:	Admin dapat menghapus data gejala apabila data gejala tersebut hendak dihilangkan

Kemudian, pada proses pengelolaan data rules juga dilakukan *breakdown* karena pada proses tersebut terdapat proses lain seperti terlihat pada **Gambar 3.11** berikut.



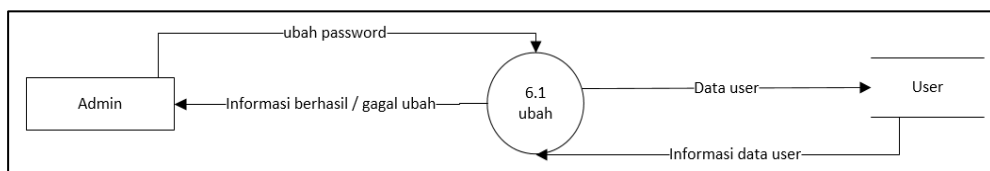
**Gambar 3.11** DFD Level 2 Proses Input Data Rules

Pada **Gambar 3.11** tersebut, alur aliran data dijelaskan pada **Tabel 3.12** berikut.

**Tabel 3.13** Penjelasan DFD Level 2 Proses Input Data Rules

No 5.1	Nama Proses	:	Tambah
	Digunakan di	:	Proses tambah data rules
	Deskripsi	:	Admin bisa melakukan penambahan rules apabila perlu adanya data rules tambahan.
No. 5.2	Nama Proses	:	Hapus
	Digunakan di	:	Proses menghapus rules
	Deskripsi	:	Admin dapat menghapus data rules apabila data rules tersebut hendak dihilangkan.

Selanjutnya, pada proses ubah *password* juga dilakukan *breakdown* karena pada proses tersebut terdapat proses lain seperti terlihat pada **Gambar 3.12** berikut.



**Gambar 3.12** DFD Level 2 Proses Ubah *Password*

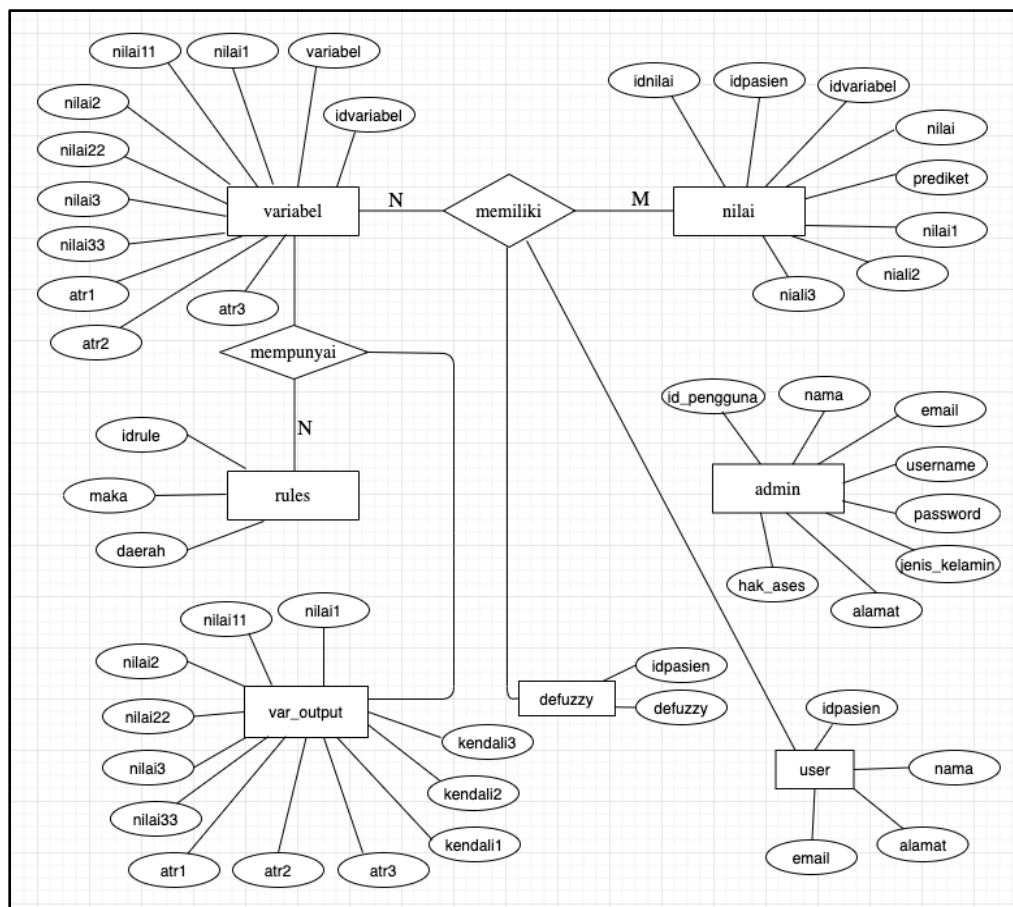
Pada **Gambar 3.12** tersebut, alur aliran data dijelaskan pada **Tabel 3.13** berikut.

**Tabel 3.14** Penjelasan DFD Level 2 Proses Ubah Password.

No 6.1	Nama Proses	: Ubah
	Digunakan di	: Proses mengubah password
	Deskripsi	: Admin dapat mengubah password apabila perlu adanya perubahan data dalam data user

**3.4.1.2 Entity Relational Diagram (ERD)**

Pada model ERD berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atributnya. ERD ini merupakan bentuk perancangan basis data relasional. ERD dalam sistem pakar diagnosa penyakit covid-19 dapat dilihat pada **Gambar 3.13** berikut.



**Gambar 3.13** ERD Sistem Pakar Diagnosa Penyakit Covid-19

### 3.4.1.3 Struktur Database

Adapun struktur *database* dari sistem pakar diagnosa penyakit covid-19 adalah sebagai berikut :

1. Nama *database* : dbfuzzysugeno  
 Nama Tabel : user  
 Fungsi : Untuk menyimpan data user  
*Primary key* : idpasien

**Tabel 3.15** Tabel User.

No	Field	Type	Length	Constraint
1	<i>idpasien</i>	Int	10	<i>Primary key</i>
2	nama	Varchar	30	
3	alamat	Varchar	50	
4	email	Varchar	30	

2. Nama *database* : dbfuzzysugeno  
 Nama Tabel : Variabel  
 Fungsi : Untuk menyimpan data variabel  
*Primary key* : *idvariabel*

**Tabel 3.16** Tabel Variabel.

No	Field	Type	Length	Constraint
1	<i>Idvariabel</i>	int	3	<i>Primary key</i>
2	variabel	Varchar	100	
3	<i>nilai1</i>	Varchar	10	
4	<i>Nilai11</i>	Varchar	10	
5	<i>Nilai2</i>	Varchar	10	
6	<i>Nilai22</i>	Varchar	10	
7	<i>Nilai3</i>	Varchar	10	
8	<i>Nilai33</i>	Varchar	10	
9	<i>Atr1</i>	Varchar	30	
10	<i>Atr2</i>	Varchar	30	
11	<i>Atr3</i>	Varchar	30	



3. Nama *database* : dbfuzzysugeno  
 Nama Tabel : rule  
 Fungsi : Untuk menyimpan data rule  
*Primary key* : *idrule*

**Tabel 3.17** Tabel Rule.

No	Field	Type	Length	Constraint
1	<i>idrule</i>	Varchar	10	<i>Primary key</i>
2	maka	Varchar	50	
3	daerah	Int	1	

4. Nama *database* dbfuzzysugeno  
 Nama Tabel : *admin*  
 Fungsi : Untuk menyimpan data admin

**Tabel 3.18** Tabel Admin.

No	Field	Type	Length	Constraint
1	Idpengguna	int	10	<i>Primary Key</i>
2	Nama	varchar	30	
3	Email	Varchar	50	
4	Username	varchar	30	
5	Password	varchar	30	
6	Jenis_kelamin	varchar	15	
7	Alamat	Varcahr	60	
8	Hak_ases	varchar	15	

5. Nama *database* : dbfuzzysugeno  
 Nama Tabel : var\_output  
 Fungsi : Untuk menyimpan data variabel  
*Primary key* : -

**Tabel 3.19** Tabel Var\_output.

No	Field	Type	Length	Constraint
1	<i>nilai1</i>	Varchar	10	
2	<i>Nilai11</i>	Varchar	10	
3	<i>Nilai2</i>	Varchar	10	
4	<i>Nilai22</i>	Varchar	10	
5	<i>Nilai3</i>	Varchar	10	
6	<i>Nilai33</i>	Varchar	10	
7	<i>Atr1</i>	Varchar	30	
8	<i>Atr2</i>	Varchar	30	
9	<i>Atr3</i>	Varchar	30	
10	<i>Kendali1</i>	Text		
11	<i>Kendali2</i>	Text		
12	<i>Kendali3</i>	text		

6. Nama *database* : dbfuzzysugeno

Nama Tabel : Defuzzy

Fungsi : Untuk menyimpan data devuzzy

*Primary key* : *idvariabel*

**Tabel 3.20** Tabel Defuzzy.

No	Field	Type	Length	Constraint
1	<i>Idpasien</i>	int	10	<i>Primary key</i>
2	defuzzy	Float		

7. Nama *database* : dbfuzzysugeno

Nama Tabel : Nilai

Fungsi : Untuk menyimpan data nilai

*Primary key* : *idnilai*

**Tabel 3.21** Tabel Nilai.

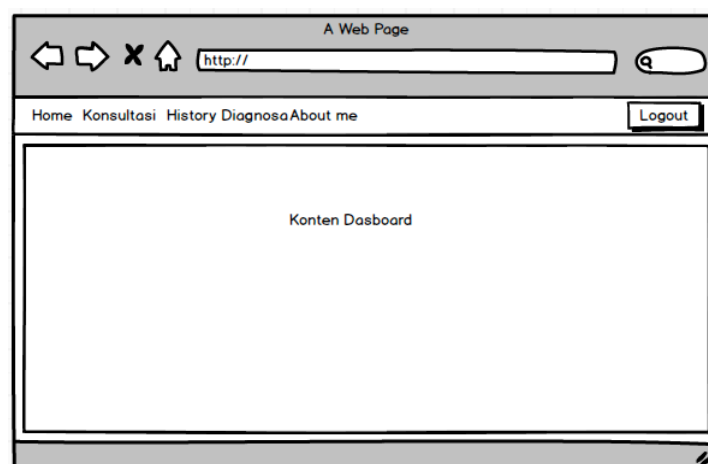
No	Field	Type	Length	Constraint
1	<i>Idnilai</i>	int	3	<i>Primary key</i>
2	idpasien	int	10	
3	<i>idvariabel</i>	int	10	
4	<i>Nilai</i>	Varchar	10	
5	<i>Prediket</i>	Varchar	30	
6	<i>Nilai1</i>	Varchar	10	
7	<i>Nilai2</i>	Varchar	10	
8	<i>Nilai3</i>	Varchar	10	

### 3.4.2 Perancangan Antar muka (*Desain Interface*)

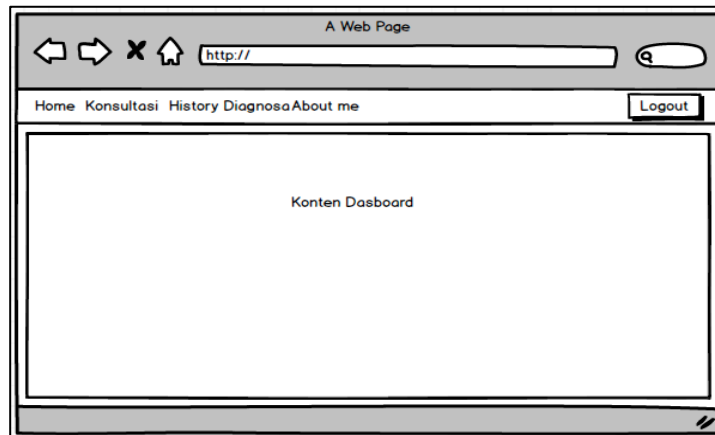
Perancangan antar muka merupakan rancangan dari *website* yang hendak dibangun dan dalam proses perancangan ini pengembang membagi kebutuhan-kebutuhan perangkat lunak sehingga menghasilkan sebuah arsitektur perangkat lunak dan dapat diterjemahkan kedalam kode-kode program dan *interface*.

Percancangan antar muka sistem pakar diagnosa penyakit covid-19 digambarkan pada gambar-gambar berikut ini :

#### 1. Rancangan Tampilan Halaman Utama (*Home*)

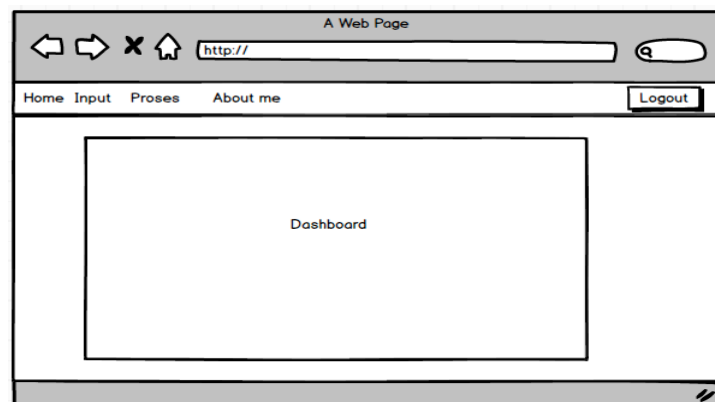
**Gambar 3.14** Rancangan Tampilan Halaman Utama

## 2. Rancangan Tampilan Halaman *Login Admin*



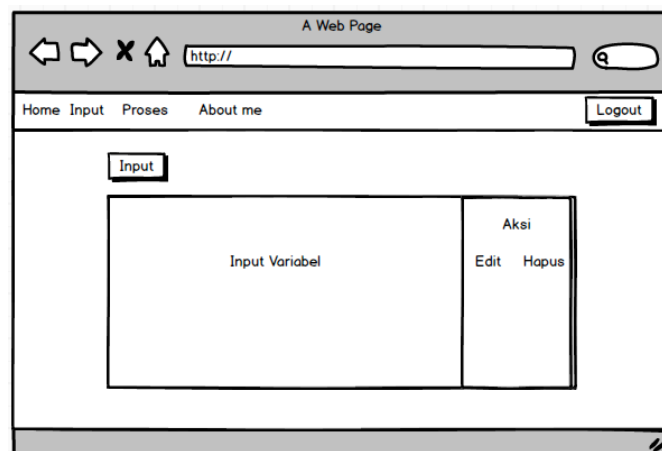
**Gambar 3.15** Rancangan Tampilan Form *Login Admin*

## 3. Rancangan Tampilan Halaman Utama (*Home*) Admin



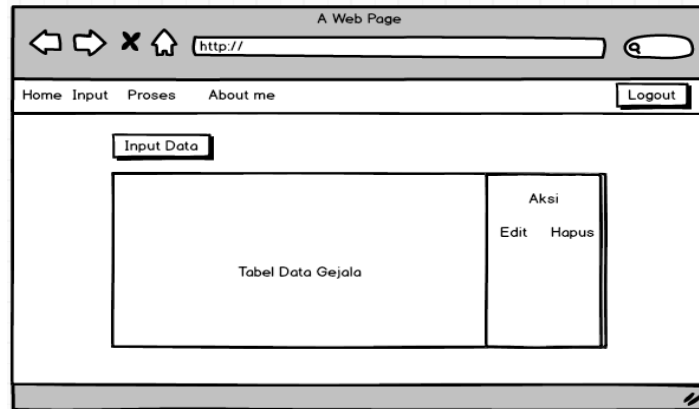
**Gambar 3.16** Rancangan Tampilan Halaman Utama Admin

## 4. Rancangan Tampilan Halaman Input



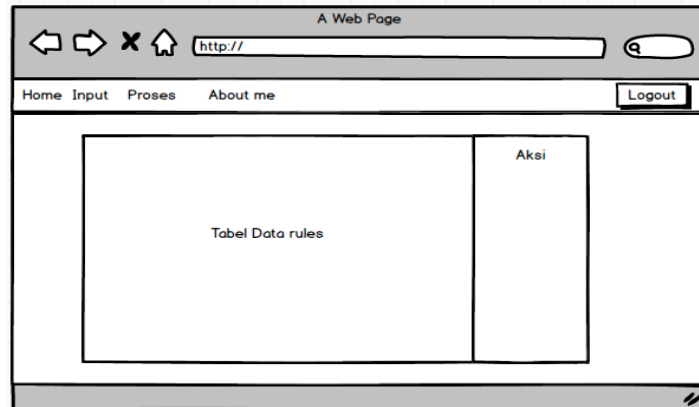
**Gambar 3.17** Rancangan Tampilan Halaman Input

### 5. Rancangan Tampilan Halaman Gejala



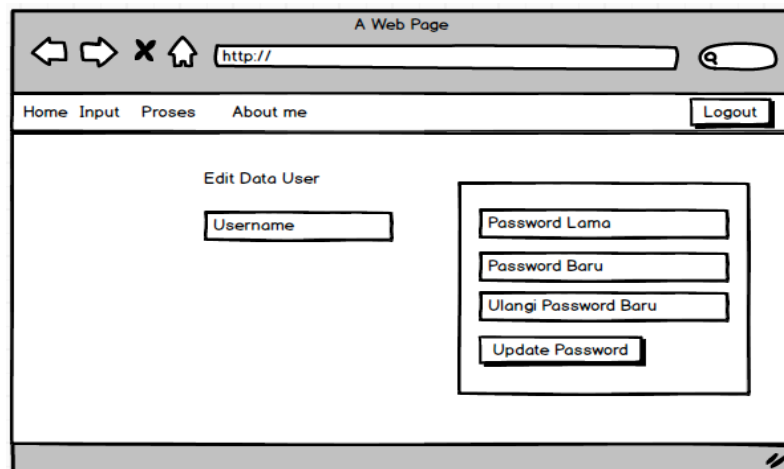
**Gambar 3.18** Rancangan Tampilan Halaman Gejala

### 6. Rancangan Tampilan Mapping Rules



**Gambar 3.19** Rancangan Tampilan Mapping Rules

### 7. Rancangan Tampilan Hasil Pengaturan



**Gambar 3.20** Rancangan Tampilan Pengaturan