



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Dalam penelitian ini, metodologi penelitian yang dilakukan adalah sebagai berikut.

1. Studi literatur

Dalam proses ini, penulis mempelajari teori-teori yang berhubungan dengan aplikasi yang dibuat, seperti *Fuzzy Logic*, *Expert System*, *Particle Swarm Optimization*, pengaturan pola makan, pembuatan aplikasi dengan basis android, dan sebagainya.

2. Wawancara dan konsultasi

Wawancara dan konsultasi dilakukan untuk mendapatkan data-data serta cara-cara perhitungan yang digunakan untuk menentukan bagaimana pola makan yang benar untuk mendapatkan berat badan yang ideal. Wawancara dilakukan dengan ahli nutrisi yang sesungguhnya yaitu ahli nutrisi *Gold Gym*

Summarecon Mall Serpong yang sudah memiliki pengalaman dalam menyusun menu makan *member-member gold gym* untuk mendapatkan berat badan yang ideal.

3. Analisis dan perancangan

Dari hasil studi literatur dan wawancara tersebut, maka peneliti menganalisis data-data yang sudah didapat dan merancang sistem. Pada tahap analisis, atribut-atribut dari *Fuzzy Logic* dan batasan-batasan yang digunakan

untuk menyusun menu makanan menggunakan PSO ditetapkan. Pada tahap perancangan, peneliti merancang alur kerja dari aplikasi seperti data-data yang diperlukan sebagai basis pengetahuan sistem pakar, pengaturan *user privilege*, dan tampilan dari aplikasi. Pembuatan diagram-diagram seperti *Data Flow Diagram*, *Flow Chart*, *Entity Relationship Diagram* dan *User Interface* dilakukan pada tahap ini.

4. Implementasi

Pada tahap ini, dilakukan implementasi *coding* berdasarkan hasil rancangan yang telah dibuat sebelumnya. *Coding* dilakukan dengan menggunakan aplikasi Eclipse, bahasa pemrograman Java berbasis Android, serta menggunakan *database* MySQL.

5. Melakukan pengujian (*testing*).

Setelah pembuatan dan implementasi selesai, maka dilakukan pengujian untuk mengetahui kehandalan (*reliability*) dan kebenaran *output* dari aplikasi yang telah dibangun. *Testing* dilakukan dalam 2 tahap yaitu pengujian ketepatan *output* aplikasi dan pengujian dengan kuesioner untuk menilai kemudahan serta tampilan dari aplikasi.

6. Dokumentasi

Setelah data-data terkumpul dan dilakukan pengujian secara memadai, maka dibuatlah dokumentasi berdasarkan data tersebut.

3.2 Analisis Sistem

Melalui hasil wawancara dan konsultasi dengan ahli nutrisi *gold gym* yaitu Lutfi Utami, didapatkan beberapa cara untuk menghitung kalori dan membuat menu makanan dengan penjelasan sebagai berikut.

1. Perhitungan Kalori

Perhitungan Kalori dilakukan dengan beberapa tahap yaitu

- a. Menghitung AMB dengan menggunakan Rumus Harris Benedict.
- b. Mencari konstanta yang akan dikalikan dengan hasil AMB sesuai dengan tingkat aktivitas *user*.
- c. Menghitung penambahan atau pengurangan kalori sesuai dengan kategori berat badan sekarang. (Kekurangan, normal atau kelebihan berat badan).

2. Penyusunan menu makanan.

Beberapa contoh makanan diberikan oleh pakar gizi beserta dengan kalori dan waktu memakannya. Beberapa makanan ini dikombinasikan untuk mendapatkan menu makanan harian bagi pengguna. Kombinasi dilakukan dengan menggunakan algoritma *Particle Swarm Optimization*. *Constraint* yang diterapkan dalam algoritma ini yaitu :

- Total kalori dari kombinasi makanan diberi toleransi sebesar 90% sampai dengan kalori yang dihasilkan oleh *Fuzzy Logic*.
- Kombinasi makanan harus sesuai dengan tipe waktu memakan makanan tersebut. Misalnya jus alpokat harus keluar di menu makanan selingan pagi, tidak boleh keluar pada menu makanan malam.

- Makanan yang sama tidak boleh keluar dalam satu set waktu makan yang sama.
- Makanan pokok hanya boleh satu dalam satu set waktu makan. Misalnya pada menu makanan siang sudah terdapat nasi putih, maka nasi merah tidak boleh pada set menu makanan siang.

3.3 Perancangan Sistem

Dalam membuat aplikasi ini terdapat beberapa perancangan yang dilakukan, misalnya perancangan *Inferensi Fuzzy*, Basis Pengetahuan, dan Perancangan Diagram. Diagram-diagram yang dibuat dalam proses perancangan adalah *Flow Chart*, *Data Flow Diagram* (DFD), dan *Entity Relationship Diagram* (ERD), dan perancangan *User Interface* (UI) beserta struktur tabel. Perancangan yang dibuat untuk aplikasi sistem pakar ini dijelaskan sebagai berikut.

3.3.1 Perancangan Inferensi Fuzzy

Sebelum melakukan perancangan Mesin Inferensi *Fuzzy*, perlu dilakukan penentuan terhadap himpunan *fuzzy* dari tiap-tiap variabel *fuzzy*. Variabel *fuzzy* yang digunakan adalah IMT dan tingkat aktivitas. Batasan variabel dan himpunan *fuzzy* dari hasil IMT yang berfungsi sebagai *input* dalam Mesin Inferensi *Fuzzy* dapat dilihat pada tabel 3.1.

Tabel 3.1 Variabel dan himpunan *Fuzzy* IMT

No	Variabel <i>Fuzzy</i>	Himpunan <i>fuzzy</i>		
		Kurus	Ideal	Gemuk
1	IMT	≤ 19	$18 \leq x \leq 25.5$	≥ 24.5

Selain IMT, variabel dan himpunan *fuzzy* juga menggunakan tingkat aktivitas yang di-*input* oleh *user*.

Tabel 3.2 Variabel dan himpunan *Fuzzy* tingkat aktivitas

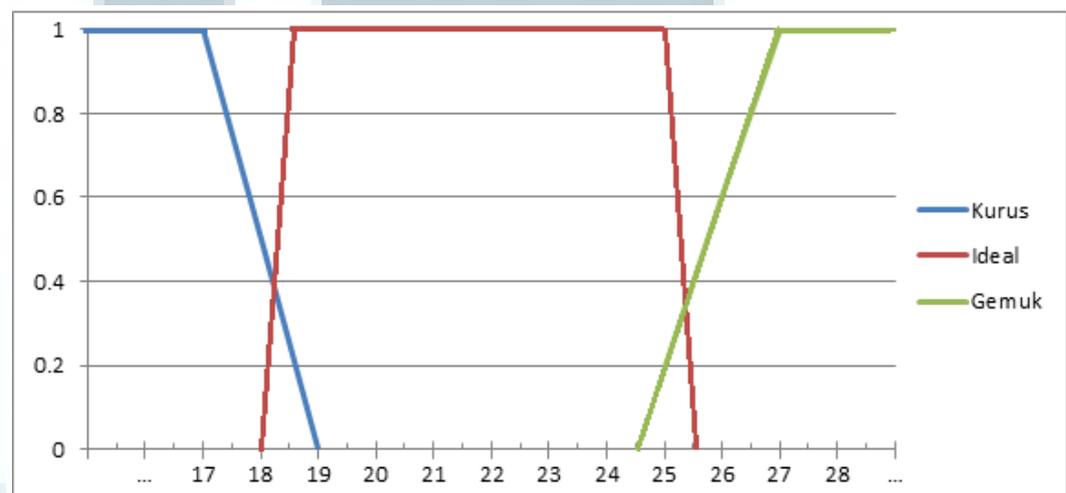
No	Variabel <i>Fuzzy</i>	Himpunan <i>fuzzy</i>		
		Ringan	Sedang	Berat
1	Tingkat Aktivitas	≤ 3.5	$2 \leq x \leq 6$	≥ 4.5

Berdasarkan kedua tabel tersebut, variabel-variabel *fuzzy* yang digunakan dengan penjelasan sebagai berikut.

1. Variabel Indeks Masa Tubuh (IMT)

Variabel IMT terbagi menjadi tiga kategori, yaitu kurus ($x < 19$), ideal ($18 \leq x \leq 25.5$), dan gemuk ($x > 24.5$). Fungsi keanggotaan pada setiap himpunan *fuzzy* dapat diketahui melalui pembagian kategori tersebut.

Penjelasannya dapat diuraikan seperti dalam grafik pada gambar 3.1.



Gambar 3.1 Himpunan *fuzzy* variabel IMT

Dari grafik pada gambar 3.1, derajat keanggotaan *fuzzy* dapat diuraikan sebagai berikut.

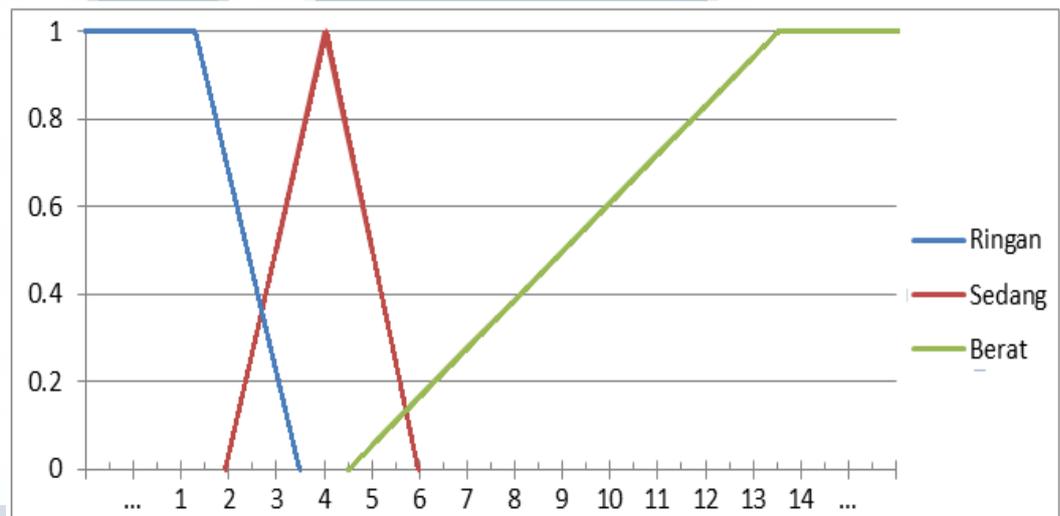
$$\mu_{\text{kurus}} [x] = \begin{cases} 1; & x < 17 \\ \frac{19-x}{2}; & 17 \leq x \leq 19 \\ 0; & x > 19 \end{cases}$$

$$\mu_{\text{ideal}} [x] = \begin{cases} 0; & x < 18 \\ \frac{x-18}{0.5}; & 18 \leq x < 18.5 \\ 1; & 18.5 \leq x \leq 25 \\ \frac{25.5-x}{0.5}; & 25 < x \leq 25.5 \\ 0; & x > 25.5 \end{cases}$$

$$\mu_{\text{gemuk}} [x] = \begin{cases} 1; & x > 27 \\ \frac{x-24.5}{2.5}; & 24.5 \leq x \leq 27 \\ 0; & x < 24.5 \end{cases}$$

2. Variabel tingkat aktivitas

Variabel tingkat aktivitas terbagi menjadi 3 kategori yaitu ringan ($x < 3.5$), sedang ($2 \leq x \leq 6$), dan berat ($x > 4.5$). Fungsi keanggotaan pada setiap himpunan *fuzzy* dapat diketahui melalui pembagian kategori tersebut. Penjelasannya dapat diuraikan seperti dalam grafik pada gambar 3.2.



Gambar 3.2 Himpunan *fuzzy* variabel tingkat aktivitas

Dari grafik pada gambar 3.2, derajat keanggotaan *fuzzy* dapat diuraikan sebagai berikut.

$$\mu \text{ ringan } [x] = \begin{cases} 1; x < 1.3 \\ \frac{3.5 - x}{2.2}; 1.3 \leq x \leq 3.5 \\ 0; x > 3.5 \end{cases}$$

$$\mu \text{ sedang } [x] = \begin{cases} 0; x < 2 \\ \frac{x - 2}{2}; 2 \leq x \leq 4 \\ \frac{6 - x}{2}; 4 < x \leq 6 \\ 0; x > 6 \end{cases}$$

$$\mu \text{ berat } [x] = \begin{cases} 1; x > 13.5 \\ \frac{x - 4.5}{9}; 4.5 \leq x \leq 13.5 \\ 0; x < 4.5 \end{cases}$$

3.3.2 Basis Pengetahuan

Basis pengetahuan diperlukan dalam perancangan sistem ini karena berguna dalam pengambilan keputusan sebagai *output* dari *Fuzzy Logic*. Setelah pembentukan himpunan *fuzzy* selesai, maka tahap selanjutnya adalah perancangan basis pengetahuan yang berupa aturan-aturan di dalam sistem *fuzzy*.

Dalam menentukan AMB peneliti menggunakan Rumus Harris Benedict (1919).

$$\text{Laki-laki} = 66 + (13,7 \times \text{BB}) + (5 \times \text{TB}) - (6,8 \times \text{U})$$

$$\text{Perempuan} = 655 + (9,6 \times \text{BB}) + (1,8 * \text{TB}) - (4,7 \times \text{U})$$

Keterangan : BB = Berat badan dalam kg

TB = Tinggi badan dalam cm

U = Umur dalam tahun

Setelah AMB berhasil didapatkan, selanjutnya dikalikan dengan faktor koefisien tingkat aktivitas yang terdapat pada tabel 3.3.

Tabel 3.3 Konstanta Tingkat Aktivitas (Penuntun Diet, 2005)

Aktivitas	Pria	Wanita
Ringan	1,3	1,3
Sedang	1,65	1,55
Berat	1,76	1,7

Kemudian, peneliti menghitung IMT untuk menentukan penambahan atau pengurangan kalori setiap *user*. Dari hasil IMT, dikategorikan sebagai berikut.

- Kurus (*Underweight*) : Kekurangan berat badan, sehingga kalori yang harus dikonsumsi akan ditambahkan. $\text{Kalori} = \text{Kalori} + 500$.
- Normal : Berat badan ideal, tidak perlu penyesuaian kebutuhan kalori.
- Gemuk (*Overweight*) : Kelebihan berat badan sehingga kalori yang harus dikonsumsi akan dikurangi. $\text{Kalori} = \text{Kalori} - 500$

Berdasarkan data-data yang telah dijelaskan, maka peneliti membuat aturan-aturan *fuzzy* pada tabel 3.4

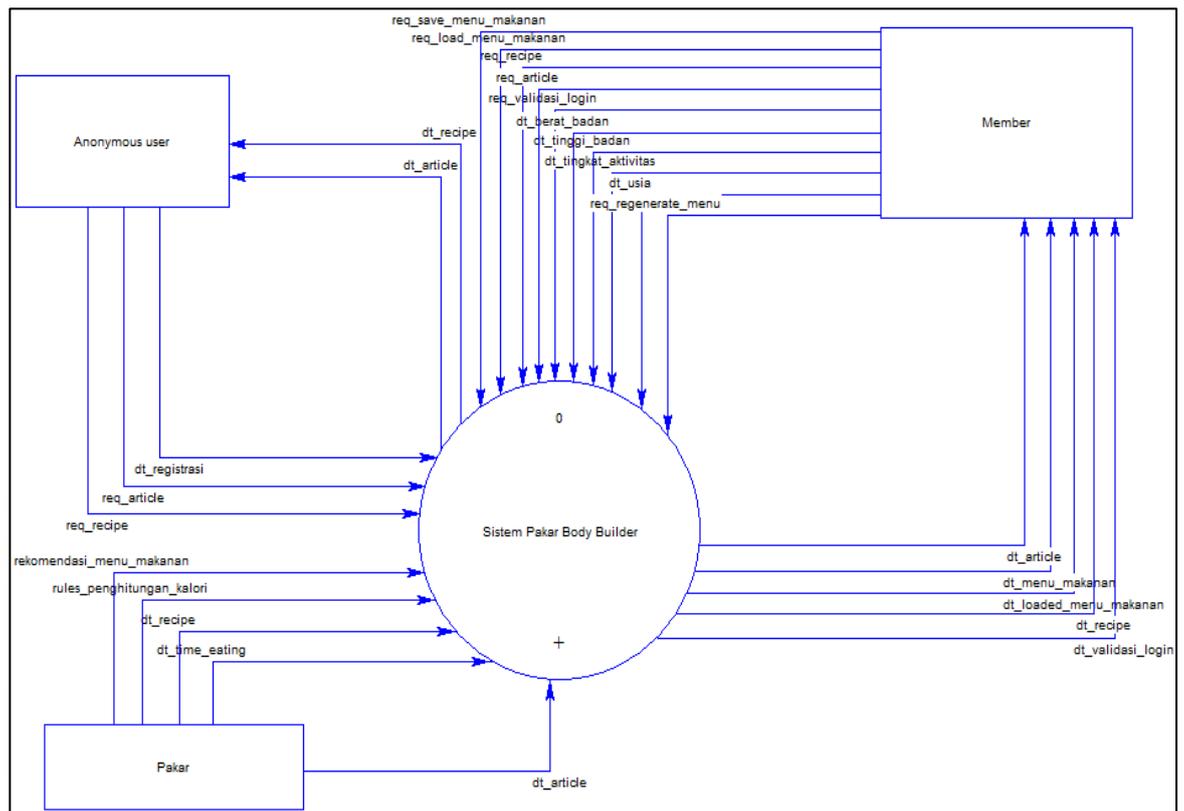
Tabel 3.4 Tabel aturan untuk perancangan sistem *fuzzy*

ID	Kondisi (IF)	Rekomendasi (THEN)
R1	kurus dan aktivitas ringan dan wanita	$500 + (1.3 * \text{AMB})$
R2	ideal dan aktivitas ringan dan wanita	$0 + (1.3 * \text{AMB})$
R3	gemuk dan aktivitas ringan dan wanita	$-500 + (1.3 * \text{AMB})$
R4	kurus dan aktivitas ringan dan pria	$500 + (1.3 * \text{AMB})$
R5	ideal dan aktivitas ringan dan pria	$0 + (1.3 * \text{AMB})$
R6	gemuk dan aktivitas ringan dan pria	$-500 + (1.3 * \text{AMB})$
R7	kurus dan aktivitas sedang dan wanita	$500 + (1.55 * \text{AMB})$
R8	ideal dan aktivitas sedang dan wanita	$0 + (1.55 * \text{AMB})$
R9	gemuk dan aktivitas sedang dan wanita	$-500 + (1.55 * \text{AMB})$
R10	kurus dan aktivitas sedang dan pria	$500 + (1.65 * \text{AMB})$
R11	ideal dan aktivitas sedang dan pria	$0 + (1.65 * \text{AMB})$
R12	gemuk dan aktivitas sedang dan pria	$-500 + (1.65 * \text{AMB})$
R13	kurus dan aktivitas berat dan wanita	$500 + (1.7 * \text{AMB})$
R14	ideal dan aktivitas berat dan wanita	$0 + (1.7 * \text{AMB})$
R15	gemuk dan aktivitas berat dan wanita	$-500 + (1.7 * \text{AMB})$
R16	kurus dan aktivitas berat dan pria	$0 + (1.76 * \text{AMB})$
R17	ideal dan aktivitas berat dan pria	$500 + (1.76 * \text{AMB})$
R18	gemuk dan aktivitas berat dan pria	$-500 + (1.76 * \text{AMB})$

3.3.3 Data Flow Diagram

Untuk mempermudah memahami sistem dan aliran data pada subsistem yang dibangun maka dibuat *Data Flow Diagram* (DFD) pada gambar 3.3 dengan penjelasan sebagai berikut.

- Terdapat 3 buah entitas yang berkaitan dengan sistem yaitu pakar, *anonymous user* dan *member*.
- Pakar memberikan aturan-aturan yang akan diterapkan pada *Fuzzy* dan *PSO*. Pakar juga memberikan rekomendasi makanan-makanan berserta dengan resepnya.
- *Anonymous user* dapat melihat artikel kesehatan, resep makanan dan melakukan registrasi untuk menjadi *member*.
- *Member* yang telah *login* dapat melihat artikel kesehatan, resep makanan, memasukkan/meng-*update* data personal yang digunakan oleh sistem untuk menentukan kalori, mengambil menu makanan terakhir yang disimpan dan men-*generate* ulang menu makanan hingga sesuai dengan selera.
- Data personal yang dimasukkan dan dapat diubah yaitu data berat badan, data tinggi badan, data tingkat aktivitas dan usia.



Gambar 3.3 Context Diagram

Pada pembuatan sistem pakar *Body Builder* dapat dibagi menjadi 3 proses yakni

1. Registrasi

Proses ini digunakan oleh *anonymous user* untuk melakukan pendaftaran sehingga dapat menjadi *member* dan menggunakan semua fitur yang berada di dalam sistem.

2. Proses menu makanan

Proses ini digunakan oleh *member* yang telah *login* untuk mengambil menu makanan harian yang terakhir kali disimpan. *Member* juga dapat *generate* ulang menu makanan hingga sesuai dengan selernya. Setelah

mendapatkan menu makanan yang diinginkan, *member* juga dapat menyimpannya. Penurunan dari proses ini dapat dilihat pada gambar 3.5.

3. Ambil artikel/resep

Proses ini digunakan oleh *anonymous user*, *member* yang belum *login* maupun sudah *login* untuk melihat artikel-artikel kesehatan dan resep-resep makanan yang ada di dalam sistem.

4. Simpan artikel/resep

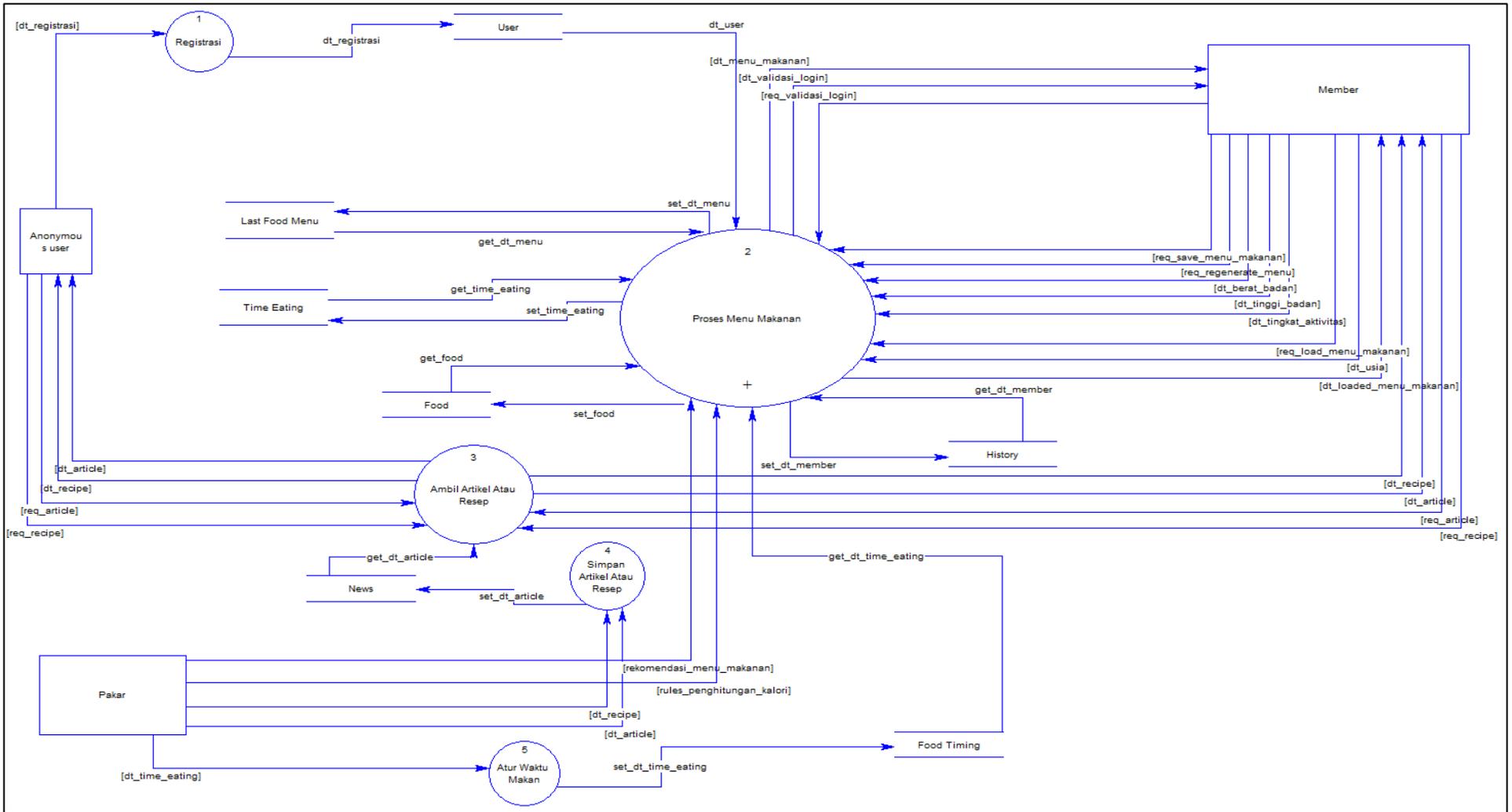
Proses ini digunakan oleh pakar untuk menambahkan artikel-artikel kesehatan / resep-resep makanan yang baru agar dapat digunakan oleh *member*.

5. Atur Waktu Makan

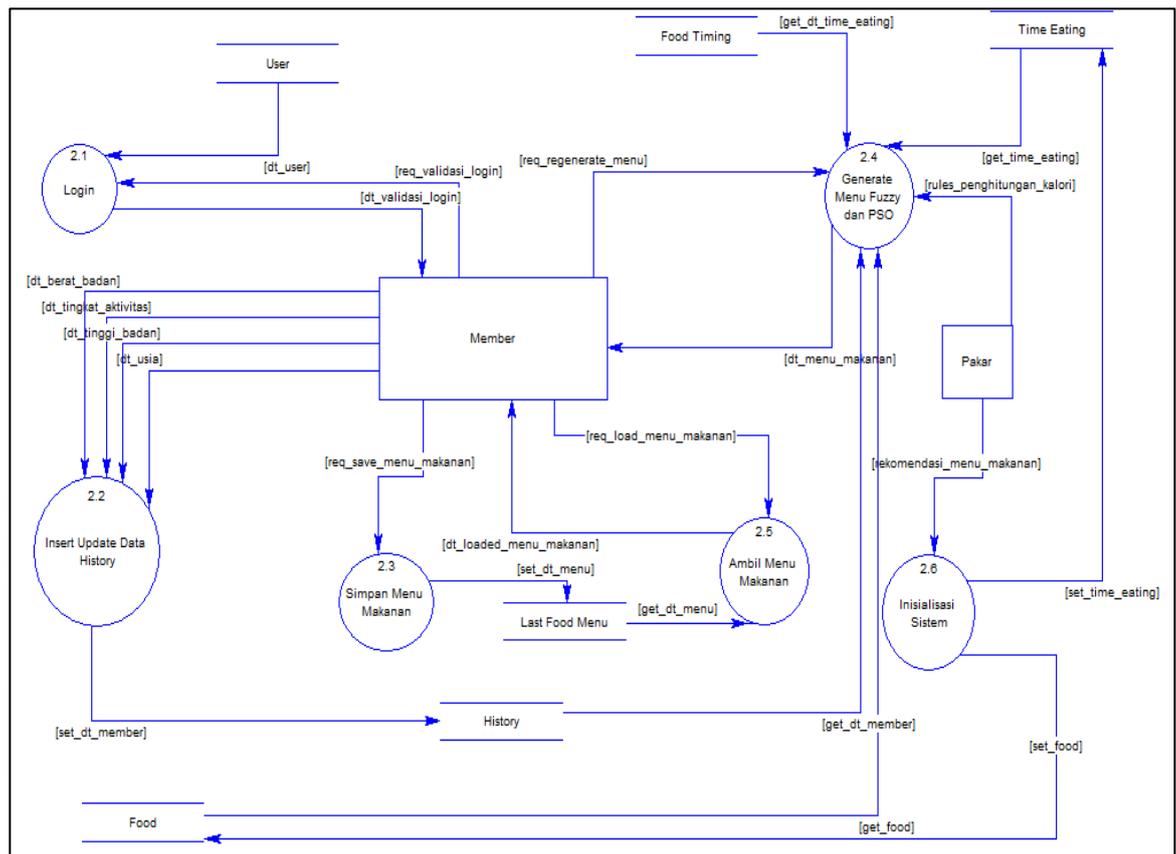
Proses ini digunakan oleh pakar untuk menentukan jumlah waktu makan dalam satu hari.

Berikut akan digambarkan *Diagram Level 1* yang merupakan penurunan dari *Diagram Context* pada halaman selanjutnya. (Gambar 3.4)

U
M
N



Gambar 3.4 Diagram Level 1



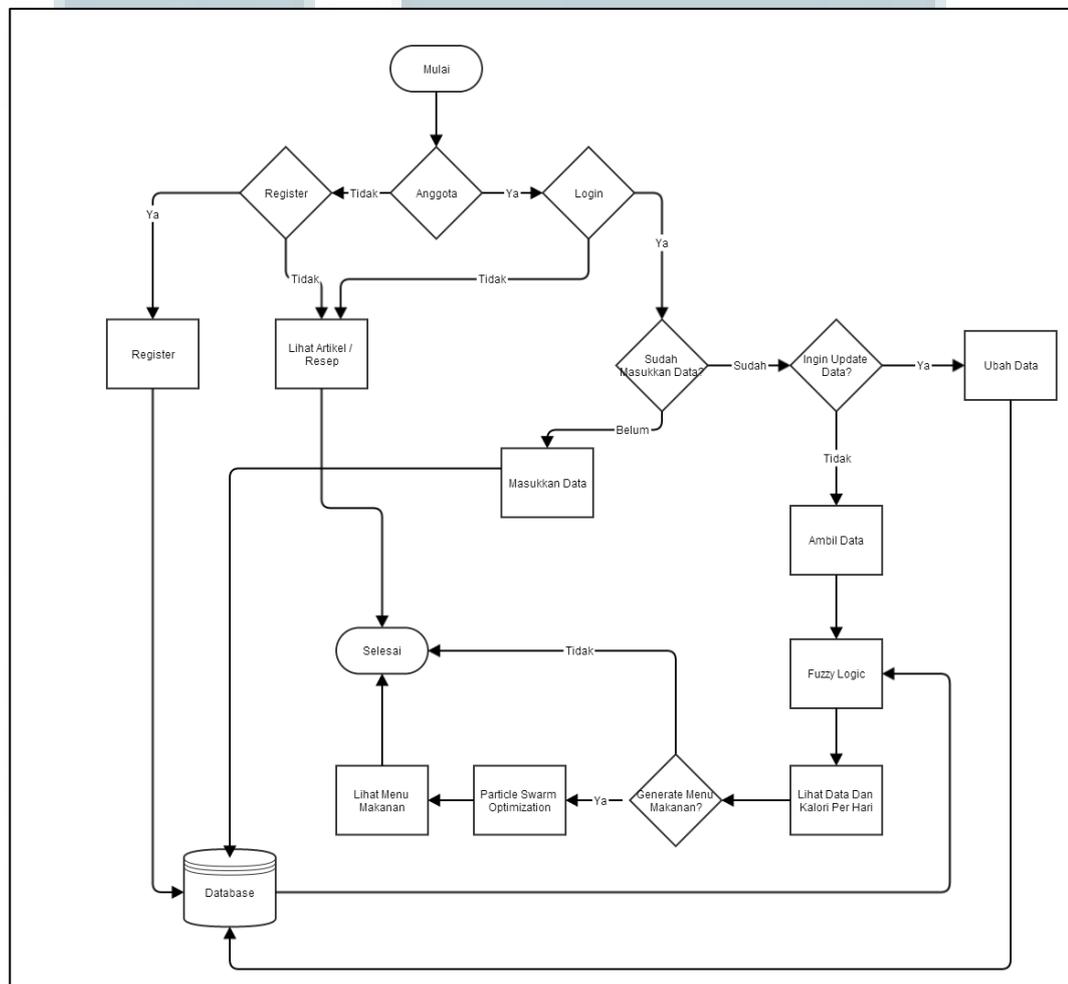
Gambar 3.5 Diagram proses menu makanan

3.3.4 Flow Chart Sistem Pakar

Pada sistem pakar ini, *user* dibagi menjadi 2 yaitu *anonymous user* dan *member*. *Anonymous user* dan *member* dapat melihat artikel-artikel kesehatan dan juga resep-resep makanan. Namun untuk perhitungan kalori dan penyusunan menu makanan hanya dapat dilakukan oleh *member*. *Anonymous user* dapat menjadi *member* dengan melakukan proses registrasi. Setelah menjadi *member*, *user* dapat melihat artikel kesehatan dan juga resep-resep makanan tanpa harus *login* ke dalam sistem. *Login* diperlukan untuk memasukkan data-data seperti berat badan, tinggi badan, usia, tingkat aktivitas. Saat pertama kali mencoba aplikasi, *member* diharuskan untuk mengisi data-data tersebut. Data-data itu akan

disimpan ke dalam *database*, sehingga ada kunjungan selanjutnya, *member* tetap bisa menggunakan data lama atau mengubah data sesuai perkembangannya.

Data-data yang dimasukkan menentukan berapa jumlah kalori yang diperlukan oleh *user* dalam satu hari. Jumlah kalori dihitung dengan menggunakan *Fuzzy Logic*. Selanjutnya jika *user* ingin melihat rekomendasi makanan, maka sistem akan *generate* menu makanan dengan menggunakan *Particle Swarm Optimization* berdasarkan kalori yang telah didapat dari proses *Fuzzy*. Berdasarkan proses yang telah dijelaskan, maka digambarkan *Flowchart* pada gambar 3.6.



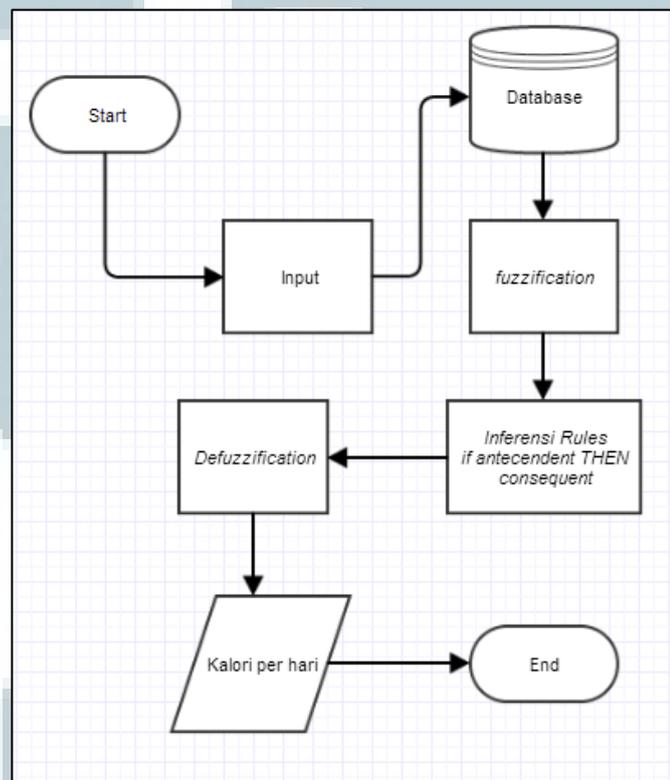
Gambar 3.6 *Flowchart* sistem pakar untuk mencapai berat badan ideal

3.3.5 Flowchart Algoritma

Aplikasi sistem pakar ini menggunakan algoritma *Fuzzy Logic* dalam menentukan kalori yang diperlukan oleh *user* dalam satu hari dan algoritma *Particle Swarm Optimization* dalam menyusun menu makanan yang direkomendasikan.

a. Flowchart Fuzzy Logic

Uraian mengenai proses *fuzzy logic* dapat dilihat pada gambar 3.7.



Gambar 3.7 Flowchart Fuzzy Logic

Tahap yang pertama dilakukan yaitu *fuzzy* akan mengambil data-data yang diperlukan dari *database* seperti berat badan, tinggi badan, jenis kelamin, usia dan tingkat aktivitas. Kemudian dilakukan proses inferensi dengan metode Sugeno yang memiliki aturan seperti pada rumus 3.1.

$$\text{IF } x_1 \text{ is } A_1 \text{ AND } \dots \text{ AND } x_n \text{ is } A_n \text{ THEN } y = f(x_1, \dots, x_n) \dots \dots \text{Rumus 3.1}$$

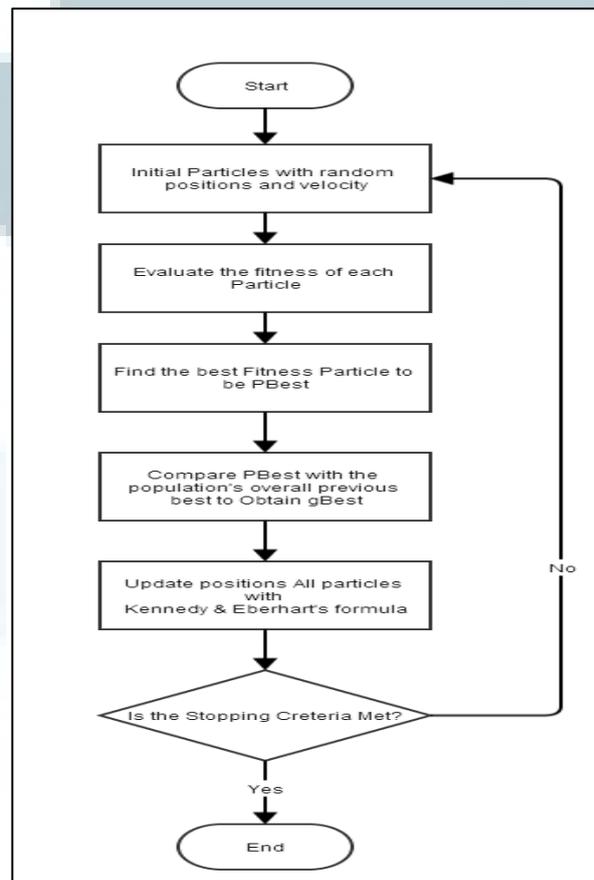
Setelah proses inferensi selesai, maka selanjutnya dilakukan proses *defuzzification* dengan metode *weighted average* yang mengambil nilai rata-rata menggunakan pembobotan berupa derajat keanggotaan. Metode tersebut didefinisikan dalam rumus 3.2.

$$y = \sum \frac{\mu(y)y}{\mu(y)} \dots \dots \dots \text{Rumus 3.2}$$

Proses *defuzzification* tersebut akan memberikan *output* berupa rekomendasi kalori harian yang dibutuhkan.

b. Flowchart Particle Swarm Optimization

Uraian mengenai proses yang dilakukan oleh *particle swarm optimization* dapat dilihat pada gambar 3.8.



Gambar 3.8 Flowchart Particle Swarm Optimization

Proses dimulai dengan melakukan inisialisasi terhadap partikel-partikel secara *random*. Dari partikel-partikel yang telah didapat, dipilih partikel yang paling baik berdasarkan *fitness value*-nya dan ditetapkan sebagai *pBest*. Selanjutnya bandingkan antara *pBest* dengan *gBest*. *gBest* merupakan *pBest* yang terbaik dari awal sampai saat iterasi berlangsung. Untuk semua partikel *update velocity* atau kecepatan yang digunakan sebagai perpindahan indeks agar dapat mendekati atau lebih baik dari partikel yang terbaik. Rumus perpindahan mengikuti rumus dari Kennedy & Eberhart (1995) dengan rumus sebagai berikut.

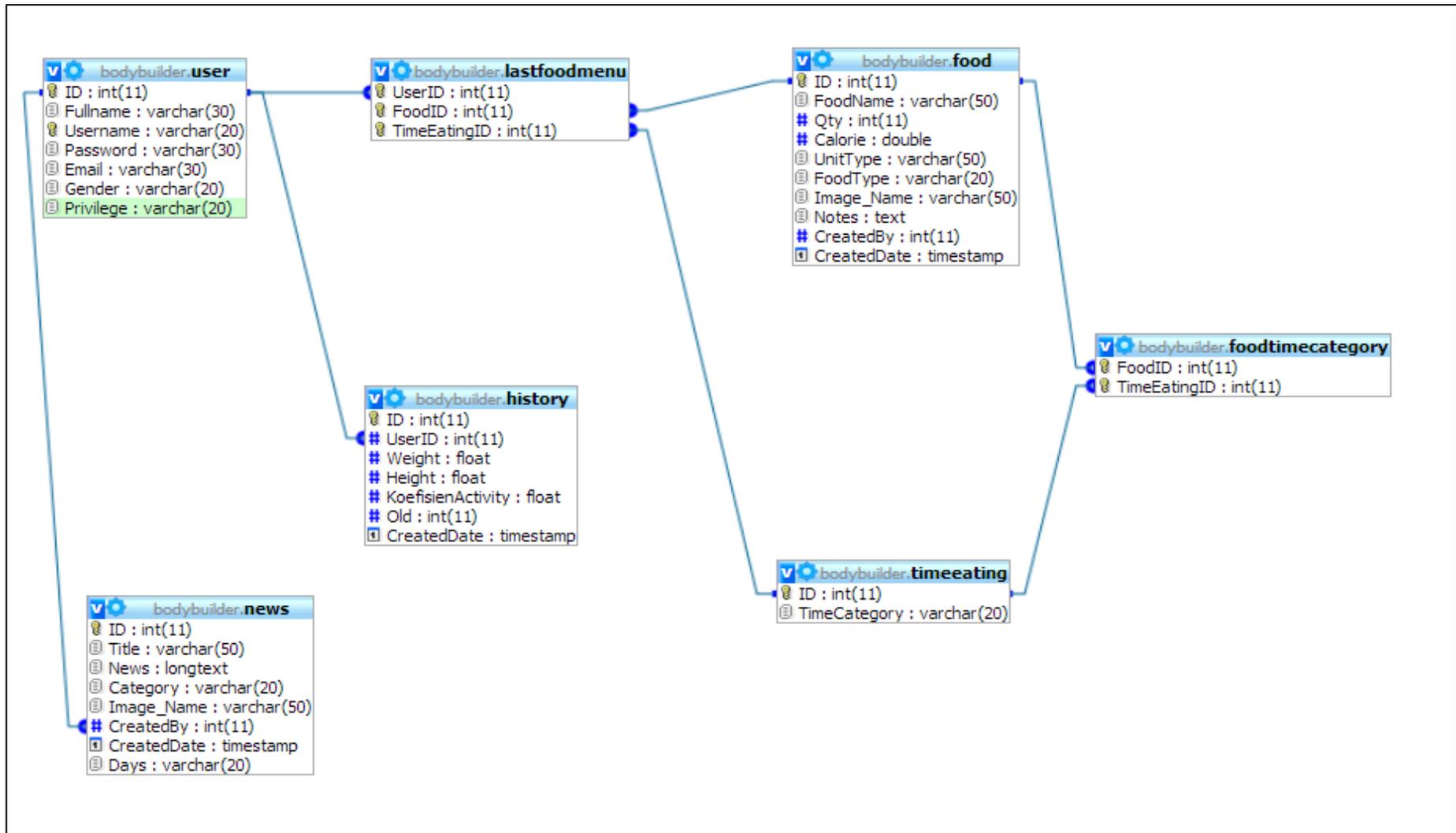
$$v_i = v_i + 2 * rand() * (Pbest - x_i) + 2 * rand() * (Gbest - x_i) \dots\dots\dots \text{Rumus 3.3}$$

$$x_i = x_i + v_i \dots\dots\dots \text{Rumus 3.4}$$

Proses yang telah dijelaskan sebelumnya, terus dilakukan secara berulang-ulang hingga mendapatkan partikel terbaik yang sesuai dengan target yang diinginkan.

3.3.6 Entity Relationship Diagram

Untuk menjelaskan hubungan antar objek data dalam *database* dapat digambarkan dengan *Entity Relationship Diagram* (ERD). Terdapat tujuh buah entitas yakni User, News, Food, History, Time Eating, Food Timing dan Last Food Menu. Hubungan antar entitas dan atributnya dapat dilihat pada gambar 3.9.



Gambar 3.9 Entity Relationship Diagram

3.3.7 Struktur Tabel

Dalam membangun aplikasi, dibutuhkan beberapa Tabel untuk mendukung penyimpanan data. Tabel-Tabel yang terdapat dalam aplikasi ini dapat diuraikan sebagai berikut.

1. Nama Tabel : User

Fungsi :

Primary Key : ID

Foreign Key : -

Tabel 3.5 Tabel User

No	Nama	Tipe Data	Panjang Data	Keterangan
1	ID	Int	11	ID pengguna
2	Fullname	Varchar	30	Nama lengkap
3	Username	Varchar	20	<i>Username</i> pengguna
4	Password	Varchar	30	<i>Password</i> pengguna
5	Email	Varchar	30	Alamat email
6	Gender	Varchar	20	Jenis Kelamin
7	Privilege	Varchar	20	Jenis pengguna

2. Nama Tabel : Timeeating

Fungsi :

Primary Key : ID

Foreign Key : Food (ID)

Tabel 3.6 Tabel Timeeating

No	Nama	Tipe Data	Panjang Data	Keterangan
1	ID	Int	11	ID waktu makan
2	TimeCategory	Varchar	20	Waktu Makan

3. Nama Tabel : News

Fungsi :

Primary Key : ID

Foreign Key : User (ID)

Tabel 3.7 Tabel News

No	Nama	Tipe Data	Panjang Data	Keterangan
1	ID	Int	11	ID artikel
2	Title	Varchar	50	Judul artikel
3	News	Longtext	-	Artikel
4	Category	Varchar	20	Kategori artikel
5	Image	Varchar	50	<i>Path</i> gambar artikel
6	CreatedBy	Int	11	ID pengguna yang meng- <i>input</i> artikel
7	CreatedDate	Timestamp	-	Tanggal artikel di- <i>input</i>
8	Days	Varchar	20	Hari artikel di- <i>input</i>

4. Nama Tabel : FoodTimeCategory

Fungsi :

Primary Key : FoodID,TimeEatingID

Foreign Key : Food (ID),timeeating(ID)

Tabel 3.8 Tabel FoodTimeCategory

No	Nama	Tipe data	Panjang data	Keterangan
1	FoodID	Int	11	ID makanan
2	TimeEatingID	Int	11	ID waktu makan

5. Nama Tabel : History

Fungsi :

Primary Key : ID

Foreign Key : User (ID)

Tabel 3.9 Tabel History

No	Nama	Tipe Data	Panjang Data	Keterangan
1	ID	Int	11	ID Histori
2	UserID	Int	11	ID pengguna
3	Weight	Float	-	Berat badan pengguna
4	Height	Float	-	Tinggi Badan pengguna
5	KoefisienActivity	Float	-	Tingkat aktivitas pengguna
6	Old	Int	11	Usia pengguna
7	CreatedDate	Datetime	-	Tanggal <i>input</i>

6. Nama Tabel : LastFoodMenu

Fungsi :

Primary Key : UserID,FoodID

Foreign Key : Food (ID)

Tabel 3.10 Tabel LastFoodMenu

No	Nama	Tipe	Panjang Data	Keterangan
1	UserID	Int	11	ID pengguna
2	FoodID	Int	11	ID makanan
3	TimeEatingID	Int	11	ID Waktu Makan

7. Nama Tabel : Food

Fungsi :

Primary Key : ID

Foreign Key : -

Tabel 3.11 Tabel Food

No	Nama	Tipe	Panjang Data	Keterangan
1	ID	Int	11	ID makanan
2	FoodName	Varchar	50	Nama makanan
3	Qty	Int	11	Banyaknya makanan
4	Calorie	Double	-	Jumlah kalori
5	UnitType	Varchar	50	Satuan makanan
6	FoodType	Varchar	20	Tipe Makanan
7	ImageName	Varchar	50	Nama Image
7	Notes	Text	-	Keterangan
8	CreatedBy	Int	11	ID user yang meng- <i>input</i> makanan
9	CreatedDate	Timestamp	-	Tanggal makanan di- <i>input</i>

3.3.8 Rancangan Tampilan Antarmuka

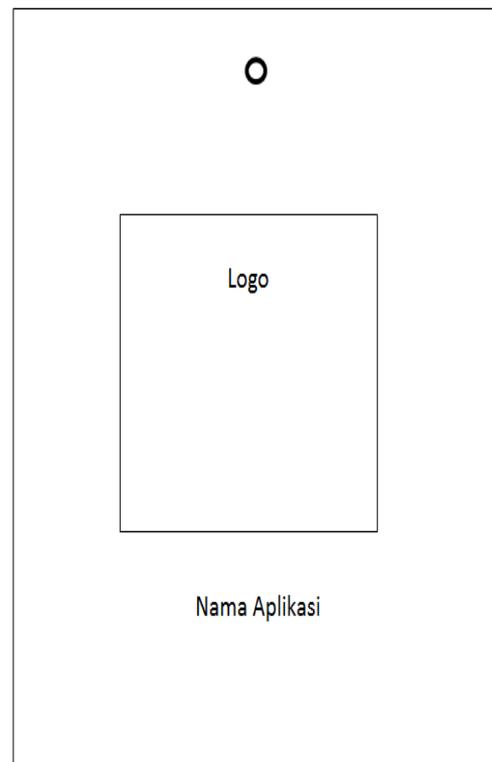
Berikut akan dijelaskan rancangan tampilan antarmuka pada aplikasi yang dibangun. Tampilan antarmuka dibagi menjadi 8 bagian.

1. Rancangan *splash screen*

Halaman ini berisi logo dan nama dari aplikasi yang dibangun.

Halaman ini dijalankan pada saat pertama kali aplikasi dibuka atau ketika aplikasi sedang men-*download resource* dari *web service*.

Rancangan tampilan dari *splash screen* dapat dilihat pada gambar 3.10.



Gambar 3.10 Rancangan *splash screen*

2. Rancangan halaman utama

Halaman utama merupakan halaman pertama setelah *splash screen*.

Pada halaman ini berisi pilihan-pilihan fitur yang dapat diakses oleh *user* antara lain, melihat artikel, melihat resep-resep makanan, melakukan perhitungan kalori dan melihat menu makanan yang pernah disimpan/akan di-generate. Rancangan tampilan dari halaman utama dapat dilihat pada gambar 3.11.



Gambar 3.11 Rancangan halaman utama

3. Rancangan halaman artikel kesehatan/resep makanan.

Ketika *user* mengklik pilihan artikel kesehatan atau resep makanan dari halaman utama maka akan dialihkan ke halaman yang berisi judul-judul dari artikel kesehatan atau resep makanan. Rancangan tampilan dari halaman ini dapat dilihat pada gambar 3.12.

Nama Halaman	↑
Tanggal update artikel	
<input type="checkbox"/> Gambar Artikel / Resep 1	
<input type="checkbox"/> Gambar Artikel / Resep 2	
<input type="checkbox"/> Gambar Artikel / Resep 3	
<input type="checkbox"/> Gambar Artikel / Resep 4	
<input type="checkbox"/> Gambar Artikel / Resep 5	↓

Gambar 3.12 Rancangan halaman artikel/resep

4. Rancangan halaman detail artikel kesehatan/resep makanan

Halaman ini ditampilkan ketika *user* mengklik salah satu judul artikel atau resep makanan. Halaman ini berisi detail dari artikel kesehatan atau resep makanan yang diklik. Rancangan tampilan dari halaman ini dapat dilihat pada gambar 3.13

The wireframe shows a page layout with a header section containing three fields: 'Nama Halaman', 'Judul artikel/Resep makanan', and 'Tanggal artikel/resep dibuat'. Below the header is a 'Gambar' (Image) placeholder, followed by a large 'Isi Artikel/Resep' (Article/Recipe Content) area. A vertical scrollbar is positioned on the right side of the content area, with an upward arrow at the top and a downward arrow at the bottom.

Gambar 3.13 Rancangan halaman detail artikel/resep

5. Rancangan halaman *Login*

Halaman *login* tampil ketika *user* yang belum *login* ingin mengakses menu yang membutuhkan data dari *database* seperti kalkulator kalori dan *generate* menu makanan. Rancangan tampilan dari halaman ini dapat dilihat pada gambar 3.14.

Nama Halaman	↑
judul	
User Name	
Password	
Login	
New to bodyBuilder? Register Here	
Footer	↓

Gambar 3.14 Rancangan halaman *login*.

6. Rancangan halaman kalkulator kalori

Halaman ini tampil ketika *user* yang sudah *login* mengklik menu kalkulator kalori atau ketika *user* yang belum *login* berhasil *login*. Halaman ini berisi data-data yang terakhir di-*input* oleh *user* dan hasil perhitungan kalori yang dilakukan *Fuzzy Logic*. Rancangan tampilan dari halaman ini dapat dilihat pada gambar 3.15

Nama Halaman	↑
Judul	
<input type="text" value="Berat badan"/>	
<input type="text" value="Tinggi badan"/>	
<input type="text" value="Tingkat Aktivitas"/>	
<input type="text" value="Usia"/>	
<input type="text" value="Kebutuhan Kalori per Hari"/>	
<input type="button" value="Button Update data"/> <input type="button" value="Button Generate Menu"/>	
Footer	↓

Gambar 3.15 Rancangan halaman kalkulator kalori

7. Rancangan halaman *update/insert* data

Halaman ini akan tampil ketika *user* baru melakukan registrasi atau ketika *user* ingin meng-*update* data yang digunakan *Fuzzy Logic* seperti berat badan, tinggi badan, usia dan tingkat aktivitas. Rancangan tampilan dari halaman ini dapat dilihat pada gambar 3.16

Nama Halaman	↑
Judul	
↓ Pagi	
↓ Menjelang Siang	
↓ Siang	
↓ Sore	
↓ Malam	
↓ Penutupan	
<input type="button" value="Save Menu"/> <input type="button" value="Regenerate New Menu"/>	
	↓

Gambar 3.16 Rancangan halaman *update/insert* data.

8. Rancangan halaman menu makanan

Halaman ini tampil ketika *user* yang telah *login* mengklik *generate* menu dari menu utama atau ketika *user* mengklik *generate* menu dari halaman kalkulator kalori. Halaman ini berisi menu makanan yang merupakan *output Particle Swarm Optimization*. Rancangan tampilan dari halaman ini dapat dilihat pada gambar 3.17

Nama Halaman	↑
Judul	
↓ Pagi	
↓ Menjelang Siang	
↓ Siang	
↓ Sore	
↓ Malam	
↓ Penutupan	
Save Menu	
Regenerate New Menu	↓

Gambar 3.17 Rancangan halaman menu makanan

UMMN