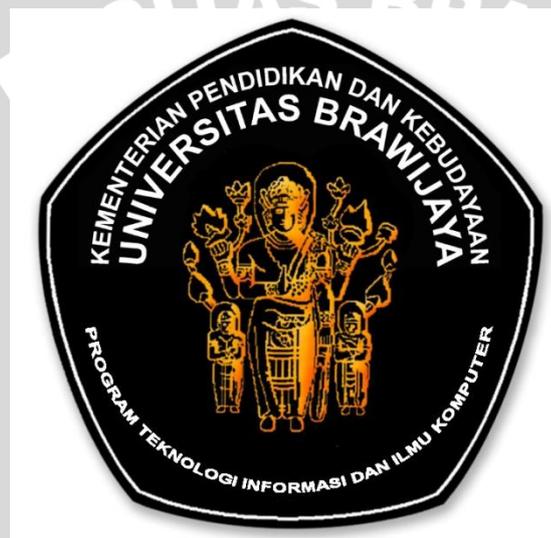


**PENERAPAN ALGORITMA GENETIKA UNTUK OPTIMASI FUNGSI
KEANGGOTAAN *FUZZY INFERENCE SYSTEM* MODEL SUGENO
PADA PERHITUNGAN ANGKA METABOLISME BASAL (AMB)**

SKRIPSI

**Diajukan untuk memenuhi persyaratan
memperoleh gelar Sarjana Komputer**



Disusun Oleh:

HENDRY ANGGARIAWAN

NIM. 105060801111010

KEMENTERIAN PENDIDIKAN DAN KEBUDAYAAN

UNIVERSITAS BRAWIJAYA

PROGRAM TEKNOLOGI INFORMASI DAN ILMU KOMPUTER

PROGRAM STUDI INFORMATIKA/ILMU KOMPUTER

MALANG

2014

LEMBAR PERSETUJUAN

**PENERAPAN ALGORITMA GENETIKA UNTUK OPTIMASI FUNGSI
KEANGGOTAAN *FUZZY INFERENCE SYSTEM* MODEL SUGENO
PADA PERHITUNGAN ANGKA METABOLISME BASAL (AMB)**

SKRIPSI

Diajukan untuk memenuhi persyaratan
Memperoleh gelar Sarjana Komputer



Disusun Oleh :

HENDRY ANGGARIAWAN

NIM. 105060801111010

Skripsi ini telah diperiksa dan disetujui oleh :

Dosen Pembimbing I

Dosen Pembimbing II

Candra Dewi, S.Kom., M.Sc

NIP. 19771114 200312 2 001

Budi Darma Setiawan, S.Kom., M.Cs

NIP. 841015 06 1 1 0090

LEMBAR PENGESAHAN

**PENERAPAN ALGORITMA GENETIKA UNTUK OPTIMASI FUNGSI
KEANGGOTAAN *FUZZY INFERENCE SYSTEM* MODEL SUGENO
PADA PERHITUNGAN ANGKA METABOLISME BASAL (AMB)**

SKRIPSI

Diajukan untuk memenuhi persyaratan memperoleh gelar Sarjana Komputer

Disusun oleh :

HENDRY ANGGARIAWAN

NIM. 105060801111010

Skripsi ini telah diuji dan dinyatakan lulus pada
tanggal 3 Juli 2014

Penguji I

Penguji II

Edy Santoso, S.Si., M.Kom
NIP. 19740414 200312 1 004

Indriati, ST., M.Kom
NIK. 831013 06 1 2 0035

Penguji III

Ir. Heru Nurwasito, M.Kom
NIP. 19650402 199002 1 001

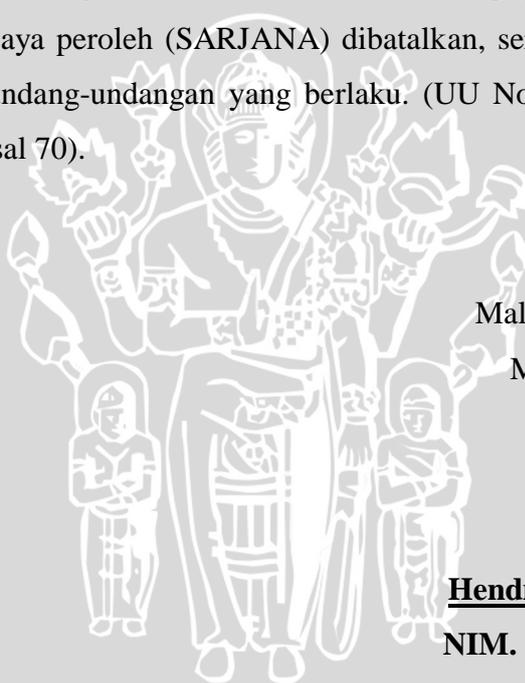
Mengetahui,
Ketua Program Studi Informatika / Ilmu Komputer

Drs. Marji, M.T.
NIP. 19670801 199203 1 001

**PERNYATAAN
ORISINALITAS SKRIPSI**

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah SKRIPSI ini tidak menjiplak karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata didalam naskah SKRIPSI ini dapat dibuktikan terdapat unsur-unsur PLAGIASI, saya bersedia SKRIPSI ini digugurkan dan gelar akademik yang telah saya peroleh (SARJANA) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku. (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).



Malang, 3 Juli 2014

Mahasiswa,

Hendry Anggariawan

NIM. 105060801111010

KATA PENGANTAR

Puji syukur penulis panjatkan kehadiran Tuhan Yang Maha Esa karena hanya dengan rahmat dan karunia-Nya, penulis dapat menyelesaikan skripsi dengan judul “Penerapan Algoritma Genetika Untuk Optimasi Fungsi Keanggotaan *Fuzzy Inference System* Model Sugeno Pada Perhitungan Angka Metabolisme Basal (AMB)”.

Melalui kesempatan ini, penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan dan dukungan selama penulisan skripsi, diantaranya:

1. Candra Dewi, S.Kom., M.Sc. selaku dosen pembimbing I yang telah memberikan ilmu dan saran untuk menyelesaikan skripsi ini.
2. Budi Darma Setiawan, S.Kom., M.Cs. selaku dosen pembimbing II yang juga memberikan ilmu dan saran untuk menyelesaikan skripsi ini.
3. Ibunda Titiek Intiyas, Ayahanda Nono Suharmono, Kakak Wenty Yulia P., Adik Danny Rahardyan dan seluruh keluarga besar atas segala nasehat, kasih sayang, perhatian dan kesabarannya dalam mendidik penulis serta memberikan doa dan semangat secara terus-menerus demi terselesaikannya skripsi ini.
4. Ir. Sutrisno, M.T., Ir. Heru Nurwasito, M.Kom., Himawat Aryadita, S.T., M.Sc., dan Edy Santoso, S.Si., M.Kom selaku Ketua, Wakil Ketua 1, Wakil Ketua 2 dan Wakil Ketua 3 Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
5. Drs. Marji, M.T. dan Issa Arwani, S.Kom., M.Sc. selaku Ketua dan Sekretaris Program Studi Teknik Informatika Universitas Brawijaya.
6. Issa Arwani, S.Kom., M.Sc. selaku dosen pembimbing akademik yang telah memberikan pengarahan selama penulis menempuh pendidikan di Program Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya.
7. Seluruh Dosen Teknik Informatika Universitas Brawijaya atas kesediaan membagi ilmunya kepada penulis.
8. Seluruh Civitas Akademika Teknik Informatika Universitas Brawijaya yang telah banyak memberi bantuan dan dukungan selama penulis menempuh

studi di Teknik Informatika Universitas Brawijaya dan selama penyelesaian skripsi ini.

9. Siti Kurniawati F yang selalu menemani, membantu, memberikan semangat serta mendoakan penulis dalam menyelesaikan skripsi ini.
10. Sahabat-sahabat penulis yaitu Rezki Trianto, Rendra Pranadipa, Mutiara Arinda Putri, Irine Octaviana, Maya Kurniawati, Vika Novitasari, Rizkie Purnama, Dita Nur Yuni Astiti, Ni Nengah Dewi M dan teman-teman TIF 2010 maupun kakak tingkat TIF 2009 yang telah memberikan doa, bantuan serta semangat kepada penulis selama menempuh studi di Teknik Informatika Universitas Brawijaya.
11. Semua pihak yang tidak dapat penulis sebutkan satu per satu yang terlibat baik secara langsung maupun yang tidak langsung demi terselesaikannya skripsi ini.

Hanya doa yang bisa penulis berikan semoga Allah SWT memberikan pahala serta balasan kebaikan yang berlipat. Penulis menyadari bahwa skripsi ini masih banyak kekurangan. Oleh karena itu, kritik dan saran yang bersifat membangun sangat diharapkan untuk menyempurnakan skripsi ini. Semoga skripsi ini membawa manfaat bagi penyusun maupun pihak lain yang menggunakannya.

Malang, 3 Juli 2014

Penulis

ABSTRAK

Hendry Anggariawan. 2014. *Penerapan Algoritma Genetika Untuk Optimasi Fungsi Keanggotaan Fuzzy Inference System Model Sugeno Pada Perhitungan Angka Metabolisme Basal (AMB)*. Skripsi. Program Studi Informatika/Ilmu Komputer, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya. Pembimbing : Candra Dewi, S.Kom., M.Sc. dan Budi Darma Setiawan, S.Kom., M.Cs.

Di dalam melakukan pembentukan fungsi keanggotaan pada metode Logika *Fuzzy* tidak harus dilakukan secara statis, dimana penentuan batas-batas domain pada fungsi keanggotaan dilakukan secara manual. Pembentukan batas-batas domain pada fungsi keanggotaan dapat dilakukan secara otomatis dengan menerapkan metode yang dapat melakukan pembentukan dan optimasi fungsi keanggotaan. Dari berbagai macam metode untuk melakukan pembentukan dan optimasi fungsi keanggotaan, di dalam penelitian ini dipilih menggunakan Algoritma Genetika, karena Algoritma Genetika merupakan salah satu metode yang dapat digunakan untuk memecahkan masalah optimasi sehingga hasil yang dicapai bisa maksimal. Metode yang dipakai di dalam Algoritma Genetika antara lain *Simple Arithmetic Crossover*, *Random Gen Mutasi* dan *Rank-based Fitness Selection*.

Hasil pengujian dan analisis sistem menunjukkan bahwa parameter Algoritma Genetika yang terbaik untuk menentukan fungsi keanggotaan IMT diperoleh dari probabilitas *crossover* 0.9, probabilitas mutasi 0.9, jumlah generasi 75 dan jumlah populasi 50 sedangkan untuk fungsi keanggotaan Umur diperoleh dari probabilitas *crossover* 0.9 dan probabilitas mutasi 0.7, jumlah generasi 100 dan jumlah populasi 50. Hasil perhitungan akurasi sistem dalam melakukan perhitungan Angka Metabolisme Basal (AMB) pada 100 data uji menunjukkan nilai sebesar 94%. Hasil akurasi ini mengalami kenaikan daripada penelitian sebelumnya yang tidak menggunakan optimasi pada fungsi keanggotaannya yakni sebesar 75% [FEB-12].

Kata kunci : Algoritma Genetika, Optimasi Fungsi Keanggotaan, Perhitungan AMB

ABSTRACT

Hendry Anggariawan. 2014. *Penerapan Algoritma Genetika Untuk Optimasi Fungsi Keanggotaan Fuzzy Inference System Model Sugeno Pada Perhitungan Angka Metabolisme Basal (AMB).* Skripsi. Program Studi Informatika/Ilmu Komputer, Program Teknologi Informasi dan Ilmu Komputer, Universitas Brawijaya. Pembimbing : Candra Dewi, S.Kom., M.Sc. dan Budi Darma Setiawan, S.Kom., M.Cs.

Make the membership function in Fuzzy Logic Method should not be done with statically, which domain boundaries on membership functions has been done manually. Formation of domain boundaries on the membership functions can be performed automatically by applying the method to make the domain boundaries and optimization of membership functions. Of the various methods to make the domain boundaries and optimization of membership functions, in this study were selected using a Genetic Algorithm. Genetic Algorithm is one method that can be used to solve optimization problems that can achieve maximum results. The method used in the Genetic Algorithm are Simple Arithmetic Crossover, Random Gene Mutation and Selection Rank-based Fitness.

From the results of the testing and analysis of the system, the best results of the Genetic Algorithm parameters for the membership function of BMI obtained from crossover probability 0.9, mutation probability of 0.9, generation size 75 and population size 50, while for the membership function of age was obtained from the crossover probability 0.9 and mutation probability of 0.7, the number of generations 100 and population size 50. The calculation accuracy of the system in calculating Basal Metabolic rate (AMB) on 100 test data showed a value of 94%. The result of accuracy increased than previous studies, that did not use the optimization of membership function which is equal to 75% [FEB-12].

Keywords : *Genetic Algorithms, Optimization of Membership Function, Calculation of AMB*

DAFTAR ISI

LEMBAR PERSETUJUAN.....	i
LEMBAR PENGESAHAN	ii
KATA PENGANTAR	iii
ABSTRAK.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xiii
DAFTAR <i>SOURCE CODE</i>	xvi
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Tujuan Penelitian.....	3
1.4 Batasan Masalah.....	3
1.5 Manfaat.....	4
1.6 Metodologi Penelitian.....	4
1.7 Sistematika Penulisan.....	5
BAB II KAJIAN PUSTAKA DAN DASAR TEORI.....	7
2.1 Kajian Pustaka	7
2.2 Angka Metabolisme Basal (AMB).....	8
2.3 Indeks Massa Tubuh (IMT).....	9
2.4 Logika <i>Fuzzy</i>	10
2.5 <i>Fuzzy Inference System</i>	13
2.6 Metode Sugeno.....	15
2.7 Algoritma Genetika	17
2.8 <i>Diabetes Melitus</i>	25
BAB III METODE DAN PERANCANGAN SISTEM.....	27
3.1 Metode Penelitian.....	27
3.2 Teknik Pengumpulan Data	28



3.2.1	Menentukan Variabel	28
3.2.2	Menentukan <i>Sample</i>	29
3.3	Analisis Kebutuhan Sistem.....	29
3.3.1	<i>Use Case Diagram</i>	31
3.3.2	Skenario <i>Use Case</i>	32
3.3.3	<i>Sequence Diagram</i>	32
3.3.4	<i>Class Diagram</i>	35
3.4	Basis Pengetahuan	36
3.5	Perancangan Sistem.....	37
3.5.1	Perancangan Fungsi Keanggotaan dengan Algoritma Genetika	39
3.5.1.1	Inisialisasi Populasi Awal.....	40
3.5.1.2	Menghitung Nilai <i>Fitness</i>	43
3.5.1.3	Pendefinisian Penalti	47
3.5.1.4	<i>Crossover</i>	47
3.5.1.5	Mutasi	49
3.5.1.6	Evaluasi <i>Fitness</i>	51
3.5.1.7	Seleksi.....	51
3.5.1.8	Hasil Fungsi Keanggotaan	52
3.5.2	Perancangan Perhitungan AMB dengan <i>Fuzzy Model Sugeno</i>	53
3.5.2.1	Pembentukan Himpunan <i>Fuzzy</i>	54
3.5.2.2	Fuzzyfikasi.....	54
3.5.2.3	Inferensi	58
3.5.2.4	Defuzzyfikasi.....	59
3.6	Perhitungan Manual.....	60
3.6.1	Pembentukan Fungsi Keanggotaan dengan Algoritma Genetika	61
3.6.1.1	Inisialisasi Populasi Awal.....	61
3.6.1.2	Menghitung Nilai <i>Fitness</i>	62
3.6.1.3	Melakukan Penalti	62
3.6.1.4	<i>Crossover</i>	65
3.6.1.5	Mutasi	68
3.6.1.6	Evaluasi <i>Fitness</i>	70

3.6.1.7	Seleksi.....	71
3.6.1.8	Hasil Fungsi Keanggotaan.....	74
3.6.2	Perancangan Perhitungan AMB dengan <i>Fuzzy Model Sugeno</i>	76
3.6.2.1	Fuzzyfikasi.....	76
3.6.2.2	Inferensi.....	78
3.6.2.3	Defuzzyfikasi.....	79
3.7	Perancangan Antarmuka Pengguna.....	80
3.7.1	<i>Interface Login</i>	81
3.7.2	<i>Interface Proses Fungsi Keanggotaan</i>	81
3.7.3	<i>Interface Hasil Akurasi Sistem</i>	82
3.7.4	<i>Interface Prediksi AMB Pasien</i>	83
3.7.5	<i>Interface Help</i>	84
3.7.6	<i>Interface About</i>	84
3.8	Mekanisme Pengujian.....	85
3.9	Penarikan Kesimpulan.....	87
BAB IV	IMPLEMENTASI SISTEM	88
4.1	Spesifikasi Sistem.....	88
4.1.1	Spesifikasi Perangkat Keras.....	88
4.1.2	Spesifikasi Perangkat Lunak.....	88
4.2	Implementasi Program.....	89
4.2.1	Inisialisasi Populasi Awal.....	90
4.2.2	Perhitungan <i>Fitness</i>	91
4.2.3	Proses <i>Crossover</i>	92
4.2.4	Mutasi.....	94
4.2.5	Seleksi.....	96
4.2.6	Fuzzyfikasi.....	97
4.2.7	Inferensi.....	98
4.2.8	Defuzzyfikasi.....	100
4.3	Implementasi Antarmuka Aplikasi.....	101
4.3.1	Tampilan Halaman <i>Login</i>	101
4.3.2	Tampilan Halaman Pembentukan Fungsi Keanggotaan.....	101

4.3.3	Tampilan Halaman Akurasi Sistem.....	102
4.3.4	Tampilan Halaman Prediksi AMB Pasien.....	103
4.3.5	Halaman <i>About</i>	104
4.3.6	Halaman <i>Help</i>	105
BAB VI PENGUJIAN DAN ANALISIS		106
5.1	Pengujian Validasi.....	106
5.2	Pengujian parameter Algoritma Genetika	108
5.2.1	Pengujian probabilitas crossover dan probabilitas mutasi terhadap nilai <i>fitness</i>	108
5.2.2	Pengujian jumlah generasi terhada nilai <i>fitness</i>	112
5.2.3	Pengujian jumlah populasi terhada nilai <i>fitness</i>	115
5.3	Pengujian Data Latih	118ss
5.4	Analisa Hasil Pengujian.....	120
BAB VII PENUTUP.....		122
7.1	Kesimpulan.....	122
7.2	Saran	123
DAFTAR PUSTAKA		124
LAMPIRAN 1		L-1
LAMPIRAN 2		L-6
LAMPIRAN 3.....		L-12
LAMPIRAN 4.....		L-13



DAFTAR GAMBAR

Gambar 2.1 Diagram *Fuzzy Inference System*.....14

Gambar 2.2 Simple Arithmetic Crossover dengan $\alpha=0,5$ 23

Gambar 3.1 Metode Penelitian Optimasi Fungsi Keanggotaan pada Perhitungan AMB dengan Algoritma Genetika.27

Gambar 3.2 Diagram *use case* sistem.....31

Gambar 3.3 *Sequence diagram* Login33

Gambar 3.4 *Sequence diagram* pembentukan fungsi keanggotaan33

Gambar 3.5 *Sequence diagram* perhitungan nilai AMB34

Gambar 3.6 *Sequence diagram* pengujian akurasi data.....34

Gambar 3.7 *Sequence diagram* Logout35

Gambar 3.8 *Class diagram* sistem.....36

Gambar 3.9 Diagram alir perancangan sistem.....39

Gambar 3.10 Diagram alir proses Algoritma Genetika40

Gambar 3.11 Diagram alir proses Inisialisasi Populasi Awal41

Gambar 3.12 Diagram fungsi keanggotaan Umur.....42

Gambar 3.13 Diagram fungsi keanggotaan IMT.....43

Gambar 3.14 Diagram alir proses perhitungan nilai *fitness*44

Gambar 3.15 Diagram alir proses *Crossover*49

Gambar 3.16 Diagram alir proses Mutasi Gen51

Gambar 3.17 Diagram alir proses Seleksi *Rank-based Fitness*.....52

Gambar 3.18 Diagram alir proses perhitungan AMB dengan Fuzzy Model Sugeno.....53

Gambar 3.19 Diagram alir proses Fuzzyfikasi55

Gambar 3.20 Grafik fungsi keanggotaan Variabel IMT.....56

Gambar 3.21 Grafik fungsi keanggotaan Variabel Umur.....58



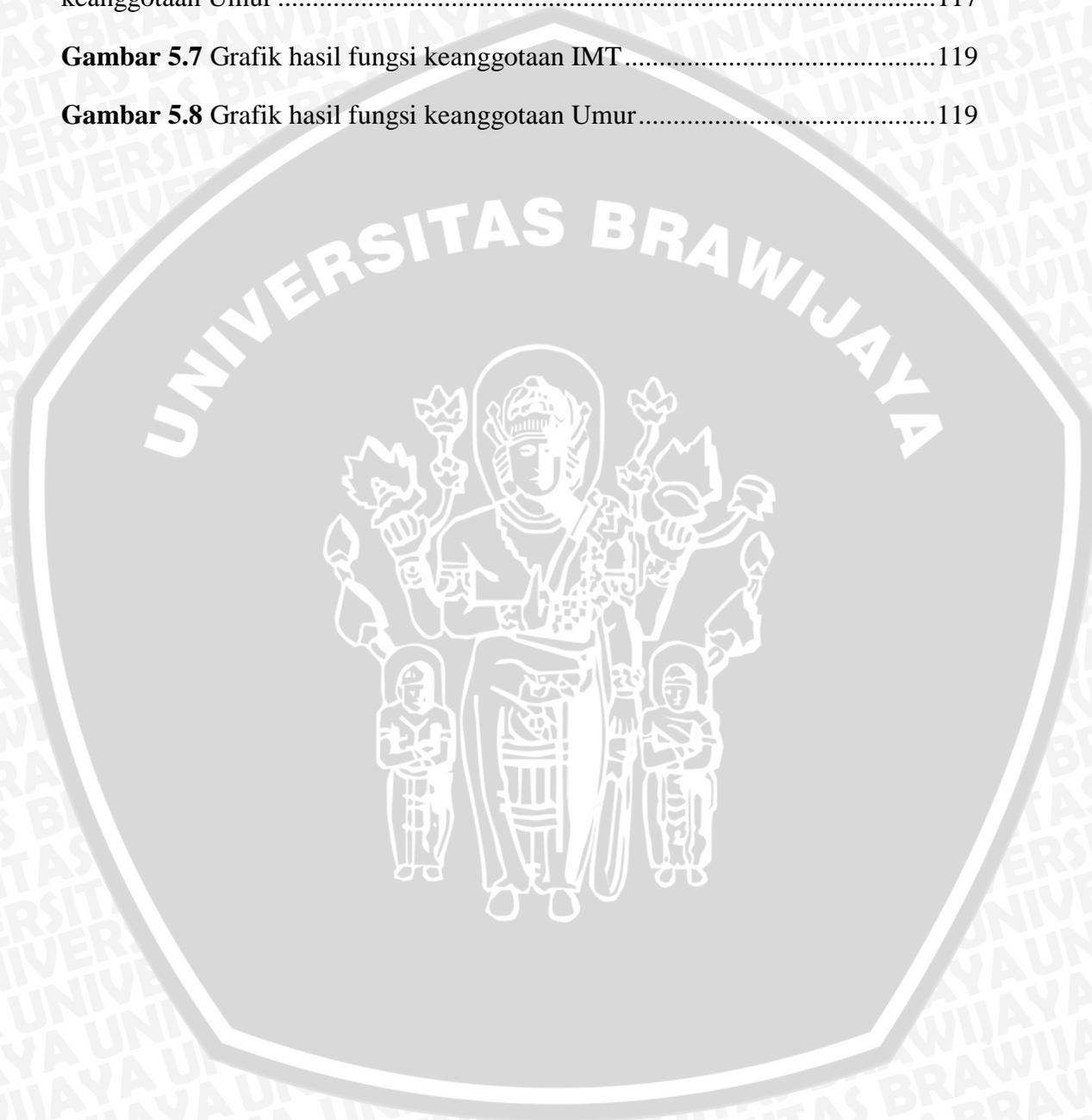
Gambar 3.22 Diagram alir proses Inferensi.....	59
Gambar 3.23 Diagram alir proses defuzzifikasi	60
Gambar 3.24 Diagram fungsi keanggotaan Umur.....	75
Gambar 3.25 Diagram fungsi keanggotaan IMT	75
Gambar 3.26 Sitemap antarmuka sistem	80
Gambar 3.27 Rancangan <i>interface</i> halaman Login	81
Gambar 3.28 Rancangan <i>interface</i> proses fungsi keanggotaan.....	81
Gambar 3.29 Rancangan <i>interface</i> akurasi sistem.....	82
Gambar 3.30 Rancangan <i>interface</i> prediksi AMB Pasien.....	83
Gambar 3.31 Rancangan <i>interface</i> Help.....	84
Gambar 3.32 Rancangan <i>interface</i> About	84
Gambar 4.1 Halaman antarmuka Login pengguna.....	101
Gambar 4.2 Halaman antarmuka pembentukan fungsi keanggotaan	102
Gambar 4.3 Halaman antarmuka hasil akurasi sistem.....	103
Gambar 4.4 Halaman antarmuka prediksi AMB Pasien.....	104
Gambar 4.5 Halaman About sistem.....	105
Gambar 4.6 Halaman bantuan memasukkan data	105
Gambar 5.1 Grafik pengujian nilai Pc dan Pm terhadap nilai <i>fitness</i> pada fungsi keanggotaan IMT	111
Gambar 5.2 Grafik pengujian nilai Pc dan Pm terhadap nilai <i>fitness</i> pada fungsi keanggotaan Umur	111
Gambar 5.3 Grafik pengujian jumlah generasi terhadap nilai <i>fitness</i> pada fungsi keanggotaan IMT	114
Gambar 5.4 Grafik pengujian jumlah generasi terhadap nilai <i>fitness</i> pada fungsi keanggotaan Umur	114

Gambar 5.5 Grafik pengujian jumlah populasi terhadap nilai *fitness* pada fungsi keanggotaan IMT116

Gambar 5.6 Grafik pengujian jumlah populasi terhadap nilai *fitness* pada fungsi keanggotaan Umur117

Gambar 5.7 Grafik hasil fungsi keanggotaan IMT119

Gambar 5.8 Grafik hasil fungsi keanggotaan Umur119



DAFTAR TABEL

Tabel 2.1 Penelitian Sistem Perhitungan Angka Metabolisme Basal	8
Tabel 2.2 Perbandingan Kriteria Metode	8
Tabel 2.3 Rumus Perhitungan AMB	9
Tabel 2.4 Klasifikasi kategori IMT	10
Tabel 3.1 Daftar kebutuhan sistem	30
Tabel 3.2 Basis pengetahuan sistem	37
Tabel 3.3 Pembentukan bilangan random populasi awal	41
Tabel 3.4 Range Skor Umur	45
Tabel 3.5 Range Skor Nilai IMT	46
Tabel 3.6 Pembentukan bilangan random pada mutasi	50
Tabel 3.7 Pembentukan himpunan fuzzy	54
Tabel 3.8 Aturan Inferensi Model Sugeno	59
Tabel 3.9 Data uji perhitungan manual	61
Tabel 3.10 Hasil Perhitungan <i>Fitness</i> dari populasi Balita	71
Tabel 3.11 Hasil Perhitungan <i>Fitness</i> dari populasi Anak-anak	71
Tabel 3.12 Individu awal	72
Tabel 3.13 Individu setelah diurutkan	72
Tabel 3.14 Individu yang terpilih populasi balita	73
Tabel 3.15 Individu awal	73
Tabel 3.16 Individu setelah diurutkan	73
Tabel 3.17 Individu yang terpilih populasi anal-anak	74
Tabel 3.18 Hasil perhitungan manual inferensi	79
Tabel 4.1 Spesifikasi perangkat keras	88
Tabel 4.2 Spesifikasi perangkat lunak	89
Tabel 4.3 Kelas-kelas yang dibangun	89

Tabel 5.1 Hasil pengujian validasi	107
Tabel 5.2 Pengujian Pc 0.1-0.9 dan Pm 0.1-0.4 pada fungsi keanggotaan IMT .	109
Tabel 5.3 Pengujian Pc 0.1-0.9 dan Pm 0.5-0.9 pada fungsi keanggotaan IMT .	109
Tabel 5.4 Pengujian Pc 0.1-0.9 dan Pm 0.1-0.4 pada fungsi keanggotaan Umur	109
Tabel 5.5 Pengujian Pc 0.1-0.9 dan Pm 0.5-0.5 pada fungsi keanggotaan Umur	110
Tabel 5.6 Hasil uji coba 1-5 pengaruh jumlah generasi terhadap nilai <i>fitness</i> pada fungsi keanggotaan IMT	113
Tabel 5.7 Hasil uji coba 1-5 pengaruh jumlah generasi terhadap nilai <i>fitness</i> pada fungsi keanggotaan Umur	113
Tabel 5.8 Hasil uji coba 1-5 pengaruh jumlah populasi terhadap nilai <i>fitness</i> pada fungsi keanggotaan IMT	115
Tabel 5.9 Hasil uji coba 1-5 pengaruh jumlah populasi terhadap nilai <i>fitness</i> pada fungsi keanggotaan Umur	113
Tabel 5.10 Hasil uji coba akurasi data latih terhadap parameter Algoritma Genetika dengan nilai rata-rata fitness tertinggi	118
Tabel 5.11 Hasil pengujian akurasi data uji	120

DAFTAR SOURCE CODE

Source Code 4.1 Inisialisasi Populasi Awal90

Source Code 4.2 Perhitungan *Fitness*91

Source Code 4.3 Pemilihan Induk Crossover92

Source Code 4.4 Proses *Simple Arithmetic Crossover*94

Source Code 4.5 Pemilihan Induk Mutasi95

Source Code 4.6 Proses Mutasi Gen.....96

Source Code 4.7 Seleksi *Rank-based Fitness*97

Source Code 4.8 Proses Fuzzyfikasi.....98

Source Code 4.9 Proses Inferensi Model Sugeno.....100

Source Code 4.10 Proses Defuzzyfikasi.....100



BAB I PENDAHULUAN

1.1 Latar Belakang

Diabetes merupakan jenis penyakit yang perlu diperhatikan oleh seluruh kalangan masyarakat pada zaman sekarang. Diabetes adalah suatu penyakit, dimana tubuh penderitanya tidak bisa secara otomatis mengendalikan tingkat gula (glukosa) dalam darahnya [VIT-05]. Setiap tahun angka kematian yang disebabkan oleh diabetes terus mengalami peningkatan. Seperti yang dilangsir dalam situs resminya www.who.int, WHO memproyeksikan bahwa kematian akibat diabetes akan berlipat ganda antara tahun 2005-2030 [WHO-13].

Penyebab utama dari penyakit diabetes adalah perilaku tidak sehat, yaitu diet tidak sehat dan tidak seimbang, kurang aktivitas fisik, merokok, hipertensi dan obesitas. Selain itu adanya faktor keturunan (gen) dan juga pola makan yang kurang terkontrol [DKS-08]. Pengaturan pola makan yang diberikan harus sesuai dengan nilai angka metabolisme basal (AMB) penderita diabetes. Perhitungan tersebut merupakan kombinasi dari parameter berat badan, tinggi badan, umur, jenis kelamin, dan faktor aktivitas [ALM-09].

Penetapan parameter berat badan, tinggi badan, umur, jenis kelamin, dan faktor aktivitas memiliki sifat ketidakpastian (*fuzzy*). Dengan adanya parameter yang memiliki ketidakpastian, maka konsep logika *fuzzy* tepat digunakan dalam memecahkan masalah tersebut [ROS-04]. Parameter yang digunakan telah mendasari beberapa rumus perhitungan nilai AMB total yang digunakan oleh ahli gizi. Adapun dalam penelitian ini akan digunakan lima parameter tersebut menggunakan inferensi *fuzzy* Model Sugeno yang diperkenalkan pada tahun 1985 oleh Takagi-Sugeno Kang [KUS-04].

Fuzzy metode sugeno merupakan metode inferensi *fuzzy* untuk aturan yang direpresentasikan dalam bentuk *IF – THEN*, dimana output (konsekuen) sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linear [KUS-04]. Inferensi *fuzzy* Model Sugeno hampir sama dengan penalaran Mamdani, hanya saja output (konsekuen) sistem tidak

berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linier. Metode ini cocok digunakan dalam menghitung nilai AMB total, karena hasil perhitungannya merupakan sebuah konstanta.

Dari penelitian yang telah dilakukan sebelumnya, dimana ketepatan hasil perhitungan Angka Metabolisme Basal (AMB) secara manual dengan hasil perhitungan yang dilakukan oleh sistem menggunakan Metode Inferensi Sugeno mendapatkan akurasi sebesar 75% [FEB-12]. Penyebab kurang maksimal tingkat akurasi ketepatan hasil perhitungan yang dilakukan oleh sistem tersebut adalah pada penentuan fungsi keanggotaannya, dimana penentuan nilai fungsi keanggotaannya belum tepat sesuai dengan nilai yang seharusnya. Oleh karena itu dalam penelitian ini akan dilakukan optimasi terhadap nilai pada fungsi keanggotaan (*membership function*). Ada beberapa metode yang digunakan untuk mendapatkan fungsi keanggotaan yang optimal diantaranya yaitu Metode *Neural Network*, Algoritma Genetika, *Inductive Reasoning*, *Batch Least Square Algorithm* (BLSA), *Recursive Least Square Algorithm* (RLSA) dan sebagainya [ROS-04]. Dari berbagai metode optimasi fungsi keanggotaan yang ada, pada penelitian ini menerapkan Algoritma Genetika sebagai metode untuk melakukan optimasi pada fungsi keanggotaan, dikarenakan Algoritma Genetika mempunyai tingkat kesuksesan yang tinggi dalam memecahkan masalah optimasi di dalam permasalahan *computer science*. Selain itu, Algoritma Genetika menyediakan kemampuan menyelesaikan berbagai masalah optimasi dengan efisien dan efektif [MRE-06]. Algoritma ini didasari pada proses genetik yang ada dalam makhluk hidup, yaitu perkembangan generasi dalam sebuah populasi yang alami, secara lambat laun mengikuti proses seleksi alam. Dengan meniru teori evolusi ini, Algoritma Genetika dapat digunakan mencari solusi permasalahan-permasalahan dalam dunia nyata [ITS-2013], sehingga diharapkan fungsi keanggotaan (*membership function*) yang dihasilkan dari Algoritma Genetika dapat membuat tingkat akurasi dari sistem lebih baik.

Berdasarkan latar belakang tersebut dapat diketahui bahwa sangat penting menentukan fungsi keanggotaan (*membership function*) dengan tepat yang akan digunakan dalam fase fuzzyfikasi di dalam metode *Fuzzy*, sehingga

hasil akhir dari proses *Fuzzy Model Sugeno* yang merupakan angka metabolisme basal (AMB) bagi penderita diabetes akan sesuai dengan hasil yang sebenarnya. Dari paparan tersebut skripsi ini diberi judul “**Penerapan Algoritma Genetika Untuk Optimasi Fungsi Keanggotaan Fuzzy Inference System Model Sugeno Pada Perhitungan Angka Metabolisme Basal (AMB)**”.

1.2 Rumusan Masalah

Dalam skripsi ini permasalahan yang akan dibahas :

1. Bagaimana implementasi *Algoritma Genetika* pada optimasi fungsi keanggotaan Inferensi Model Sugeno pada perhitungan Angka Metabolisme Basal (AMB).
2. Berapa nilai probabilitas *crossover*, probabilitas mutasi, jumlah generasi dan jumlah populasi sehingga mendapatkan nilai *fitness* yang optimal pada Algoritma Genetika.
3. Bagaimana akurasi Sistem pada Aplikasi Optimasi Fungsi Keanggotaan dengan Algoritma Genetika pada Perhitungan Angka Metabolisme Basal (AMB) terhadap perhitungan manual menggunakan rumus WHO/FAO/UNU.

1.3 Tujuan Penelitian

Melakukan optimasi terhadap fungsi keanggotaan (*membership function*) *Fuzzy Inference System Model Sugeno* dengan menggunakan Algoritma Genetika pada permasalahan perhitungan Angka Metabolisme Basal (AMB) serta menentukan parameter Algoritma Genetika yaitu probabilitas *crossover*, probabilitas mutasi, jumlah generasi dan jumlah populasi yang menghasilkan batas-batas fungsi keanggotaan dengan nilai *fitness* yang optimal.

1.4 Batasan Masalah

Batasan masalah dalam skripsi ini adalah :

1. Perhitungan nilai Angka Metabolisme Basal (AMB) menggunakan rumus FAO/WHO/UNU.

2. Data uji dan data latih yang digunakan adalah data pasien penderita diabetes yang diambil dari penelitian sebelumnya [FEB-12].
3. Metode-metode yang digunakan di dalam Algoritma Genetika yaitu Metode seleksi yang digunakan adalah Metode *Rank Based Fitness Selection*, metode *Crossover* yang digunakan adalah Metode *Simple Arithmetic Crossover* dan metode Mutasi menggunakan Metode Mutasi Random Gen.

1.5 Manfaat

Hasil penelitian ini diharapkan dapat membantu ahli gizi dalam menghitung nilai Angka Metabolisme Basal (AMB) pada pasien penderita diabetes. Serta dapat membantu pasien penderita diabetes mengetahui nilai Angka Metabolisme Basal (AMB).

Penelitian ini juga sebagai pengembangan terhadap ilmu pengetahuan yang telah didapat penulis selama perkuliahan, khususnya di dalam bidang Sistem Pakar dan *Artificial Intelligent* yang telah digunakan pada penelitian ini yaitu metode Logika Fuzzy dan Algoritma Genetika.

1.6 Metodologi Penelitian

Metode penyelesaian masalah yang dilakukan pada skripsi ini, yaitu :

1. Studi Literatur.

Mempelajari dan mengkaji beberapa literatur (jurnal, buku dan artikel dari website) mengenai Angka Metabolisme Basal (AMB), diet makanan Diabetes Militus, Logika Fuzzy, *Fuzzy Inference System* (FIS) Model Sugeno dan Algoritma Genetika.

2. Perumusan masalah dan analisis kebutuhan.

Mengkaji permasalahan sebagai hasil dari studi pustaka dan menganalisis yang dibutuhkan.

3. Perancangan dan implementasi sistem.

Mengimplementasikan Algoritma Genetika sebagai metode untuk optimasi fungsi keanggotaan dan *Fuzzy Inference System* Model Sugeno dengan merancang dan membangun sebuah perangkat lunak untuk membuat sistem pendukung keputusan.

4. Uji coba dan analisis hasil implementasi.

Menganalisa kombinasi probabilitas *crossover*, probabilitas mutasi, jumlah generasi dan jumlah populasi yang menghasilkan *fitness* terbaik, serta menganalisa akurasi ketepatan perhitungan sistem Optimasi Algoritma Genetika pada fungsi keanggotaan *Fuzzy Inference System Model Sugeno* dengan perhitungan manual menggunakan rumus WHO/FAO/UNU.

1.7 Sistematika Penulisan

Tugas akhir ini secara keseluruhan terdiri dari atas enam bab. Penentuan sistematika penulisan ini bertujuan untuk mempermudah dalam menyusun dan mempelajari bagian-bagian dari seluruh rangkaian penelitian tugas akhir. Secara garis besar masing-masing bab akan membahas hal-hal sebagai berikut:

BAB I : PENDAHULUAN

Pada bab ini berisi tentang gambaran umum dari penulisan Tugas Akhir yang meliputi latar belakang masalah, perumusan masalah, batasan masalah, tujuan dan manfaat penelitian, dan sistematika penulisan.

BAB II : KAJIAN PUSTAKA DAN DASAR TEORI

Bab ini membahas tentang teori-teori dasar yang relevan dengan penelitian sehingga mampu memecahkan permasalahan yang ada dan mencapai tujuan yang diinginkan. Teori-teori yang dipergunakan dalam penelitian ini meliputi teori logika *fuzzy*, himpunan *crisp* dan himpunan *fuzzy*, operator dasar himpunan *fuzzy*, *Fuzzy Inference System Model Sugeno*, optimasi fungsi keanggotaan, Algoritma Genetika dan teori-teori dasar dari beberapa penyakit diabetes dan Angka Metabolisme Basal (AMB). Selain itu juga berisi kajian pustaka penelitian sebelumnya.

BAB III : METODE DAN PERANCANGAN SISTEM

Pada bab ini akan berisi langkah-langkah sistematis yang dipergunakan dalam penelitian dengan tujuan memecahkan masalah. Di dalamnya akan dijelaskan secara singkat mengenai metode yang menjadi pedoman pengerjaan yang perlu dilakukan sehingga dihasilkan output yang diinginkan. Pada subbab perancangan membahas mengenai analisis kebutuhan perangkat lunak

dan perancangan sistem. Di dalamnya akan dijelaskan secara bertahap mengenai kerangka perencanaan pembuatan sistem mulai dari *use case*, *sequence diagram*, *class diagram*, perhitungan manual dan antarmuka pengguna.

BAB IV : IMPLEMENTASI SISTEM

Pada bab ini akan dibahas hasil dari metode dan perancangan yang telah diterapkan. Yaitu berisi hasil dari implementasi perangkat lunak digunakan untuk membuat batas-batas fungsi keanggotan menggunakan Algoritma Genetika yang selanjutnya kan digunakan sebagai proses fuzzyfikasi untuk menghitung nilai Angka Metabolisme Basal (AMB) bagi pasien penderita diabetes menggunakan metode *Fuzzy Inference System Model Sugeno*.

BAB V : PENGUJIAN DAN ANALISIS

Pada bab ini akan dibahas tentang analisa hasil dari implementasi yang telah diterapkan. Pada tahap ini melakukan validasi program, analisis pengaruh parameter Algoritma Genetika terhadap nilai fitness dan perhitungan akurasi ketepatan sistem dalam menentukan angka metabolisme basal (AMB) terhadap hasil yang didapat dari pakar serta membandingkan akurasi sistem dari perhitungan secara manual.

BAB VI : PENUTUP

Bab terakhir akan memuat kesimpulan isi dari keseluruhan uraian bab-bab sebelumnya dan menentukan saran dari hasil yang diperoleh dan diharapkan dapat bermanfaat dalam pengembangan selanjutnya.

BAB II

KAJIAN PUSTAKA DAN DASAR TEORI

2.1 Kajian Pustaka

Dalam menentukan tingkat Angka Metabolisme Basal (AMB) pada pasien penderita diabetes sangat erat hubungannya dengan *Clinical Decision Support System* (CDSS). CDSS merupakan system yang dirancang untuk membantu para pekerja klinis dalam mengambil keputusan [ETA-09]. Karakteristik yang diperoleh dari pasien akan dicocokkan dengan pengetahuan-pengetahuan yang ada pada basis pengetahuan. Pada penelitian ini dilakukan mekanisme penarikan kesimpulan dengan konsep fuzzy yaitu mekanisme penalaran *fuzzy/fuzzy reasoning*. *Fuzzy reasoning* merupakan bagaimana suatu kesimpulan berbasis *fuzzy* dapat dibuat.

Dalam penelitian ini diimplementasikan menggunakan Algoritma Genetika untuk melakukan optimasi pada fungsi keanggotaan pada permasalahan perhitungan Angka Metabolisme Basal (AMB) dengan *Fuzzy Inference System* Sugeno. Algoritma Genetika dipilih sebagai metode untuk melakukan optimasi pada fungsi keanggotaan, dikarenakan Algoritma Genetika mempunyai tingkat kesuksesan yang tinggi dalam memecahkan masalah optimasi di dalam permasalahan *computer science*. Selain itu, Algoritma Genetika menyediakan kemampuan menyelesaikan berbagai masalah optimasi dengan efisien dan efektif [MRE-06]. Penelitian yang sedang diusulkan kali ini akan berfokus pada optimasi fungsi keanggotaan yang dibuat menggunakan metode Algoritma Genetika dalam menentukan Angka Metabolisme Basal (AMB). Kriteria dan referensi yang digunakan bersumber dari penelitian sebelumnya yaitu Implementasi Inferensi Fuzzy Model Sugeno pada Perhitungan Angka Metabolisme Basal (AMB) pada Penderita Diabetes [FEB-12]. Serta referensi dari penelitian terhadap optimasi fungsi keanggotaan menggunakan Algoritma Genetika [MRE-06]. Berikut ini adalah tabel perbandingan data penelitian yang akan dilakukan dengan penelitian sebelumnya :

Tabel 2.1 Penelitian Sistem Perhitungan Angka Metabolisme Basal

No.	Penelitian	Objek	Metode
1.	Sebelumnya	Perhitungan Angka Metabolisme Basal (AMB)	Inferensi Fuzzy Model Sugeno
2.	Dilakukan	Pembentukan Batas-batas Fungsi Keanggotaan dan Perhitungan Angka Metabolisme Basal (AMB)	Optimasi Fungsi Keanggotaan Inferensi Fuzzy Model Sugeno dengan Algoritma Genetika

Berikut adalah tabel perbandingan kriteria yang digunakan dalam penelitian :

Tabel 2.2 Perbandingan Kriteria Metode

No.	Tahapan	Penelitian Sebelumnya	Penelitian yang Dilakukan
1.	Input Kriteria	1. Berat Badan 2. Tinggi Badan 3. Umur 4. Jenis Kelamin 5. Faktor Aktivitas	1. Berat Badan dan Tinggi badan menjadi Nilai IMT 2. Umur 3. Jenis Kelamin 4. Faktor Aktivitas 5. Probabilitas Crossover 6. Probabilitas Mutasi 7. Jumlah Generasi 8. Jumlah Populasi
2.	Metode	Inferensi Fuzzy Model Sugeno	Optimasi Fungsi Keanggotaan Inferensi Fuzzy Model Sugeno dengan Algoritma Genetika
3.	Output	Nilai Angka Metabolisme Basal (AMB)	Batas-batas fungsi keanggotaan dan Nilai Angka Metabolisme Basal (AMB)

2.2 Angka Metabolisme Basal (AMB)

Angka Metabolisme Basal (AMB) adalah kebutuhan energi minimal yang dibutuhkan tubuh untuk menjalankan proses tubuh yang vital. Kebutuhan energi metabolisme basal termasuk jumlah energi yang diperlukan untuk pernapasan, peredaran darah, pekerjaan ginjal, pankreas, dan alat tubuh lainnya, serta untuk proses metabolisme di dalam sel-sel

dan untuk mempertahankan suhu tubuh. Kurang lebih dua pertiga energi yang dikeluarkan seseorang sehari digunakan untuk kebutuhan aktivitas metabolisme basal tubuh.

Angka metabolisme basal dinyatakan dalam kilokalori berat badan perjam. Angka ini berbeda antar orang dan mungkin pada orang yang sama bila terjadi perubahan dalam keadaan fisik dan lingkungan [ALM-03]. Angka Metabolisme Basal (AMB) perlu diketahui oleh ahli gizi karena nilai ini dijadikan dasar dalam menentukan nilai AMB pada pasien penderita diabetes. Pembagian kategori umur berdasarkan *Basal Metabolic Rate* (BMR) [ALM-09] seperti rumus yang tertera pada tabel 2.3.

Tabel 2.3 Rumus Perhitungan BMR

Kelompok Umur (th)	BMR (kkal/hari)	
	Laki-laki	Perempuan
0 – 3	$(60,9 \times BB) - 54$	$(61,0 \times BB) - 51$
3 – 10	$(22,7 \times BB) + 495$	$(22,5 \times BB) + 499$
10 – 18	$(17,5 \times BB) + 651$	$(12,2 \times BB) + 746$
18 – 30	$(15,3 \times BB) + 679$	$(14,7 \times BB) + 496$
30 – 60	$(11,6 \times BB) + 879$	$(8,7 \times BB) + 829$
≥ 60	$(13,5 \times BB) + 487$	$(10,5 \times BB) + 596$

Sumber : [ALM-09]

Perhitungan nilai AMB mengacu pada standart yang digunakan oleh FAO/WHO/UNU [ALM-09] dengan persamaan sebagai berikut.

$$AMB = ((BMR) + IMT) \times FA \dots \dots \dots (2-1)$$

Keterangan :

AMB = Angka Metabolisme Basal atau AMB

BMR = Basal Metabolic Rate yang didapat dari tabel 2.3

IMT = Skor Indeks Massa Tubuh

FA = Faktor Aktivitas

2.3 Indeks Massa Tubuh (IMT)

IMT dihitung sebagai berat badan dalam kilogram (kg) dibagi tinggi badan dalam meter dikuadratkan (m²) dan tidak terikat pada jenis kelamin. IMT secara signifikan berhubungan dengan kadar lemak tubuh



toal sehingga dapat dengan mudah mewakili kadar lemak tubuh [SUP-01]. Saat ini, IMT secara internasional diterima sebagai alat untuk mengidentifikasi kelebihan berat badan dan obesitas. Tabel 2.4 menampilkan klasifikasi Indeks Massa Tubuh berdasarkan Departemen Kesehatan RI tahun 1994.

Tabel 2.4 Klasifikasi kategori IMT

IMT (kg/m ²)	Keterangan klasifikasi	Kategori	Skor
< 17.0	Kekurangan berat badan tingkat berat	Sangat Kurus	+500
17.0 – 18.4	Kekurangan berat badan tingkat ringan	Kurus	0
18.5 – 25.0	Normal	Normal	0
25.1 – 27.0	Kelebihan berat badan tingkat ringan	Gemuk	-500
> 27.0	Kelebihan berat badan tingkat berat	Obesitas	-500

Sumber : [SUP-01]

2.4 Logika Fuzzy

Pengertian paling dasar dari logika *fuzzy* adalah metode yang digunakan untuk menangani segala sesuatu yang bersifat tidak pasti, ambigu dari suatu sistem. Logika *fuzzy* pertama kali diperkenalkan oleh Zadeh pada tahun 1965. Dalam tulisannya yang menyatakan bahwa banyak sekali pengambil keputusan dalam dunia nyata tidak mengetahui dengan pasti hasil dan konsekuensi yang diperoleh dari keputusan yang dibuatnya, maka dibuatlah suatu metode untuk mengurangi pengaruh dari ketidakpastian tersebut. Ketidakpastian inilah yang menjadi dasar yang paling utama dalam logika *fuzzy* [KUS-04]. Logika *fuzzy* sengaja dibuat sebagai generalisasi dari semua teori yang konvensional, sehingga akibatnya logika *fuzzy* tetap menjadi bidang yang membutuhkan keahlian analisis dan matematis yang sangat baik. Logika *fuzzy* mencoba untuk menyelesaikan hal-hal yang tidak pasti dalam bidang teknik dengan mengikutsertakan semua parameter yang nantinya digunakan untuk membuat pernyataan menjadi jelas [TMW-99].

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output. Sehingga dengan menggunakan logika *fuzzy* ini, output yang diinginkan merupakan hasil proses dari input yang ada dengan menggunakan aturan-aturan terkait. Terdapat beberapa alasan orang menggunakan logika *fuzzy*, antara lain adalah [KUS-04]:

1. Konsep logika *fuzzy* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. Logika *fuzzy* sangat fleksibel.
3. Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
4. Logika *fuzzy* mampu memodelkan fungsi-fungsi *nonlinear* yang sangat kompleks.
5. Logika *fuzzy* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui proses pelatihan.
6. Logika *fuzzy* dapat bekerja sama dengan teknik-teknik kendali secara konvensional.
7. Logika *fuzzy* didasarkan pada bahasa alami.

Berikut komponen-komponen di dalam Logika Fuzzy [ROS-04] :

a. Himpunan Fuzzy

Himpunan *fuzzy* didasarkan pada gagasan untuk memperluas jangkauan fungsi karakteristik sedemikian hingga fungsi tersebut akan mencakup bilangan real pada interval $[0,1]$. Nilai keanggotaannya menunjukkan bahwa suatu *item* dalam semesta pembicaraan tidak hanya berada pada 0 atau 1, namun juga nilai yang terletak diantaranya. Dengan kata lain, nilai kebenaran suatu *item* tidak hanya bernilai benar atau salah. Nilai 0 menunjukkan salah, nilai 1 menunjukkan masih ada nilai-nilai yang terletak antara benar dan salah.

b. Fungsi Keanggotaan

Fungsi Keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat

keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Ada beberapa fungsi yang bisa digunakan.

- Representasi Linear
- Representasi Kurva Segitiga
- Representasi Kurva Trapesium
- Representasi Kurva Bentuk Bahu
- Representasi Kurva-S
- Representasi Kurva Bentuk Lonceng (*Bell Curve*)

c. Operator Dasar untuk Himpunan Fuzzy

i. Operator AND

Operator ini berhubungan dengan operasi interseksi pada himpunan. α -predikat sebagai hasil operasi dengan operator AND diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu A \cap B = \min (\mu A[x], \mu B[y]) \dots \dots \dots (2-2)$$

ii. Operator OR

Operator ini berhubungan dengan operasi *union* pada himpunan. α -predikat sebagai hasil operasi dengan operator OR diperoleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan- himpunan yang bersangkutan.

$$\mu A \cup B = \max (\mu A[x], \mu B[y]) \dots \dots \dots (2-3)$$

iii. Operator NOT

Operator ini berhubungan dengan operasi komplemen pada himpunan. α -predikat sebagai hasil operasi dengan operator not diperoleh dengan mengurangi nilai keanggotaan pada himpunan yang bersangkutan dari 1.

$$\mu A' = 1 - \mu A[x] \dots \dots \dots (2-4)$$



d. Fungsi Implikasi

Tiap-tiap aturan pada basis pengetahuan *fuzzy* akan berhubungan dengan suatu relasi *fuzzy*. Bentuk umum dari aturan yang digunakan dalam fungsi implikasi adalah:

IF x is A THEN y is B(2-5)

Dengan x dan y adalah skalar, dan A dan B adalah himpunan *fuzzy*. Aturan yang mengikuti *IF* disebut sebagai anteseden, sedangkan aturan yang mengikuti *THEN* disebut sebagai konsekuen. Proposisi ini dapat diperluas dengan menggunakan operator *fuzzy*, seperti:

IF (x_1 is A_1) \bullet (x_2 is A_2) \bullet (x_3 is A_3) \bullet ... \bullet (x_N is A_N) THEN y is B(2-6)

Dengan \bullet adalah operator (misal: *OR* atau *AND*).

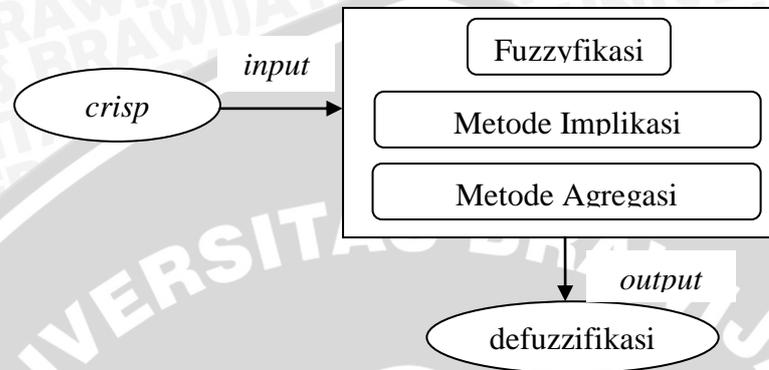
2.5 Fuzzy Inference System

Sistem inferensi *fuzzy* merupakan kerangka komputasi yang didasarkan pada teori himpunan *fuzzy*, aturan *fuzzy* berbentuk *IF-THEN*, dan penalaran *fuzzy*. Hal-hal yang terdapat dalam sistem *fuzzy* yaitu sebagai berikut [KUS-04].

1. Variabel *fuzzy* merupakan variabel yang dibahas dalam suatu sistem *fuzzy* seperti umur, temperatur, permintaan dsb.
2. Himpunan *fuzzy*, merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*. Contoh: variabel umur, terbagi atas 3 himpunan *fuzzy*, yaitu: MUDA, PAROBAYA, TUA
3. Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Adakalanya nilai semesta pembicaraan tidak dibatasi batas atasnya. Contoh: Semesta pembicaraan untuk variabel umur: [0 ... 40]

4. Domain adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam himpunan *fuzzy*.

Proses *fuzzy inference system* secara umum dapat dilihat dalam bentuk diagram seperti yang tertera pada Gambar 2.1.



Gambar 2.1 Diagram *Fuzzy Inference System*
Sumber : [KUS-04]

Dari gambar di atas, dapat dijelaskan bahwa *input* data yang diterima berupa nilai *fuzzy* kemudian ditentukan derajat keanggotaannya. Apabila kondisi mempunyai aturan lebih dari satu maka diterapkan operasi *fuzzy*. Operasi *fuzzy* diperlukan apabila anteseden untuk suatu aturan lebih dari satu dan digunakan untuk menentukan fungsi keanggotaan hasil inferensi setiap aturan tersebut. *Aggregation* digunakan untuk menggabungkan seluruh *output* gugus *fuzzy* menjadi sebuah *output* gugus *fuzzy* kemudian dilakukan proses *defuzzifikasi* sebagai proses akhir (*output*) yaitu proses perubahan hasil *fuzzy* menjadi hasil yang memiliki nilai tunggal (*crisp*). Dalam sistem inferensi *fuzzy* ada beberapa komponen utama yang dibutuhkan. Komponen tersebut meliputi data variable *input*, data variabel *output* dan data aturan. Untuk mengolah data masukan dibutuhkan beberapa fungsi meliputi fungsi *fuzzyfikasi* yang terbagi dua, yaitu fungsi untuk menentukan nilai jenis keanggotaan suatu himpunan dan fungsi penggunaan operator. Fungsi *fuzzifikasi* akan mengubah nilai *crisp* (nilai aktual) menjadi dinilai *fuzzy* (nilai kabur). Selain itu, dibutuhkan pula fungsi *defuzzifikasi*, yaitu fungsi untuk

memetakan kembali nilai *fuzzy* menjadi nilai *crisp* yang menjadi *output* atau nilai solusi permasalahan [ROS-04].

Ada beberapa model yang digunakan dalam pembuatan sistem inferensi *fuzzy* diantaranya model Mamdani, Sugeno dan Tsukamoto. Model Mamdani sering juga dikenal dengan nama metode *Max-Min*. Model ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan output, diperlukan 4 tahapan yaitu Pembentukan himpunan *fuzzy*, Aplikasi fungsi implikasi (aturan), Komposisi aturan, dan Penegasan (*defuzzy*). Pada Model Tsukamoto, setiap konsekuen pada yang berbentuk *IF-Then* harus direpresentasikan dengan suatu himpunan *fuzzy* dengan fungsi keanggotaan yang monoton. Sebagai hasilnya, *output* hasil inferensi dari tiap-tiap aturan diberikan secara tegas (*crisp*) berdasarkan α -predikat (*fire strength*). Hasil akhirnya diperoleh dengan menggunakan rata-rata terbobot. Sedangkan Penalaran dengan model Sugeno hampir sama dengan penalaran Mamdani, hanya saja *output* (konsekuen) sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linear. Model ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985 [KUS-04].

2.6 Metode Sugeno

Penalaran dengan model sugeno hampir sama dengan penalaran mamdani, hanya saja *output* (konsekuen) sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linear. Model ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985 [KUS-04].

1. Model *Fuzzy Sugeno* Orde-Nol

Secara umum bentuk model *fuzzy* sugeno Orde-Nol adalah:

$$\text{IF } (x_1 \text{ is } A_1) \cdot (x_2 \text{ is } A_2) \cdot (x_3 \text{ is } A_3) \cdot \dots \cdot (x_N \text{ is } A_N) \text{ THEN } z=k \dots\dots\dots(2-7)$$

Dengan AN adalah himpunan *fuzzy* ke-N sebagai anteseden, dan k adalah suatu konstanta (tegas) sebagai konsekuen.

2. Model *Fuzzy Sugeno* Orde-Satu

Secara umum bentuk model *fuzzy* sugeno Orde-Satu adalah:

$$\text{IF } (x_1 \text{ is } A_1) \cdot \dots \cdot (x_N \text{ is } A_N) \text{ THEN } z = p_1 \cdot x_1 + \dots + p_N \cdot x_N + q \dots\dots\dots(2-8)$$

dengan AN adalah himpunan *fuzzy* ke-N sebagai anteseden, dan pN adalah suatu konstanta (tegas) ke-N dan q juga merupakan konstanta dalam konsekuen. Apabila komposisi aturan menggunakan model Sugeno, maka defuzzifikasi dilakukan dengan cara mencari nilai rata-ratanya.

Tahapan-tahapan dalam model Sugeno yaitu sebagai berikut:

a) Pembentukan himpunan *fuzzy*

Pada tahapan ini variabel input (*crisp*) dari sistem *fuzzy* ditransfer kedalam himpunan *fuzzy* untuk dapat digunakan dalam perhitungan nilai kebenaran dari premis pada setiap aturan dalam basis pengetahuan. Dengan demikian tahap ini mengambil nilai-nilai *crisp* dan menentukan derajat di mana nilai-nilai tersebut menjadi anggota dari setiap himpunan *fuzzy* yang sesuai.

b) Aplikasi fungsi implikasi

Tiap-tiap aturan pada basis pengetahuan *fuzzy* akan berhubungan dengan suatu relasi *fuzzy*. Bentuk umum dari aturan yang digunakan dalam fungsi implikasi.

Dengan x dan y adalah skalar, dan A dan B adalah himpunan *fuzzy*. Aturan yang mengikuti *IF* disebut sebagai anteseden sedangkan aturan yang mengikuti *THEN* disebut konsekuen. Aturan ini dapat diperluas dengan menggunakan operator *fuzzy*.

Secara umum fungsi implikasi yang dapat digunakan yaitu sebagai berikut:

1. Min (minimum) adalah fungsi ini dipergunakan untuk memotong output himpunan *fuzzy*.
2. Dot (*product*) adalah fungsi ini dipergunakan untuk menskala output himpunan *fuzzy*. Pada model Sugeno ini, fungsi implikasi yang digunakan adalah fungsi min.

c) Defuzzifikasi (*Defuzzification*) adalah proses pembentukan output berupa nilai *crisp* kembali. Metode defuzzifikasi pada model Sugeno dilakukan dengan perhitungan *Weight Average (WA)*

$$Z = \frac{a_1z_1 + a_2z_2 + a_3z_3 + \dots + a_nz_n}{a_1 + a_2 + a_3 + \dots + a_n} \dots\dots\dots(2-9)$$

Keterangan :

Z = Hasil Defuzzifikasi/Nilai akhir

a_i = Nilai keanggotaan anteseden

z_i = Hasil inferensi masing-masing aturan

Dalam penelitian ini, penggunaan fungsi keanggotaan untuk proses fuzzyfikasi di dalam metode *Logika Fuzzy* membutuhkan suatu penyesuaian yang dinamis, dikarenakan faktor utama yang menentukan hasil akhir adalah pemetaan input ke dalam tahap fuzzyfikasi. Oleh karena itu, dalam penelitian ini dikembangkan perumusan fuzzyfikasi agar dapat disesuaikan secara dinamis, sehingga fungsi keanggotaan yang dibuat diharapkan mendapatkan hasil akhir yang optimal. Salah satu metode pembentukan fungsi keanggotaan adalah dengan metode Algoritma Genetika yang merupakan salah satu metode untuk mencapai optimasi dari permasalahan yang ada. Kajian pustaka terhadap Algoritma Genetika akan dijelaskan pada bagian di bawah ini.

2.7 Algoritma Genetika

Sejak algoritma genetika (AG) pertama kali dirintis oleh John Holland dari Universitas Michigan pada tahun 1960-an, AG telah diaplikasikan secara luas pada berbagai bidang. Algoritma Genetika banyak digunakan untuk memecahkan masalah optimasi, walaupun pada kenyataannya juga memiliki kemampuan yang baik untuk masalah-masalah selain optimasi. John Holland menyatakan bahwa setiap masalah yang berbentuk adaptasi (alami maupun buatan) dapat diformulasikan dalam terminologi genetika [ITS-2013].

Algoritma genetika adalah simulasi dari proses evolusi Darwin dan operasi genetika atas kromosom. Pada algoritma genetika, teknik pencarian dilakukan sekaligus atas sejumlah kandidat solusi yang dikenal dengan istilah populasi. Suatu solusi yang terdapat dalam satu populasi disebut dengan istilah kromosom. Kromosom ini merupakan suatu solusi yang masih berbentuk simbol. Populasi awal dibangun secara acak,

sedangkan populasi berikutnya merupakan hasil evolusi kromosom-kromosom melalui iterasi yang disebut dengan generasi. Pada setiap generasi, kromosom akan melalui proses evaluasi dengan menggunakan alat ukur yang disebut dengan fungsi *fitness*. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas dari kromosom dalam populasi tersebut [KUS-04].

Generasi berikutnya dikenal dengan istilah anak (*offspring*) terbentuk dari gabungan dua kromosom generasi sekarang yang bertindak sebagai induk (*parent*) dengan menggunakan operator penyilangan (*crossover*). Selain operator penyilangan, suatu kromosom dapat juga dimodifikasi dengan menggunakan operator mutasi. Populasi generasi yang baru dibentuk dengan cara menyeleksi nilai *fitness* dari kromosom induk (*parent*) dan nilai *fitness* dari kromosom anak (*offspring*), serta menolak kromosom-kromosom yang lainnya sehingga ukuran populasi (jumlah kromosom dalam suatu populasi) konstan [ITS-03].

Struktur umum algoritma genetika adalah sebagai berikut:

- a. Populasi, istilah pada teknik pencarian yang dilakukan sekaligus atas sejumlah solusi yang mungkin.
- b. Kromosom, individu yang terdapat dalam satu populasi dan merupakan suatu solusi yang masih berbentuk simbol.
- c. Generasi, populasi awal dibangun secara acak sedangkan populasi selanjutnya merupakan hasil evolusi kromosom-kromosom melalui iterasi.
- d. Fungsi *Fitness*, alat ukur yang digunakan untuk proses evaluasi kromosom. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas kromosom dalam populasi tersebut.
- e. Generasi berikutnya dikenal dengan anak (*offspring*) terbentuk dari gabungan 2 kromosom generasi sekarang yang bertindak sebagai induk (*parent*) dengan menggunakan operator penyilangan (*crossover*).
- f. Mutasi, operator untuk memodifikasi kromosom.

Ada 6 komponen utama algoritma genetika:

1. Teknik Penyandian

Teknik penyandian meliputi penyandian gen dari kromosom. Gen merupakan bagian dari kromosom, satu gen biasanya akan mewakili satu variable. Gen dapat direpresentasikan dalam bentuk: string bit, pohon, array bilangan *real*, daftar aturan, elemen permutasi, elemen program dan lain-lain. Kromosom dapat direpresentasikan dengan menggunakan:

- String bit: 10011, 11101
- Bilangan Real: 65.65, 562.88
- Elemen Permutasi: E2, E10
- Daftar Aturan: R1, R2, R3
- Elemen Program: pemrograman genetika
- Struktur lainnya

2. Prosedur Inisialisasi

Ukuran populasi tergantung pada masalah yang akan dipecahkan dan jenis operator genetika yang akan diimplementasikan. Setelah ukuran populasi ditentukan, kemudian harus dilakukan inisialisasi terhadap kromosom yang terdapat pada populasi tersebut. Inisialisasi kromosom dilakukan secara acak, namun demikian harus tetap memperhatikan domain solusi dan kendala permasalahan yang ada.

3. Fungsi *fitness*

Fungsi yang digunakan untuk mengevaluasi setiap individu yang terdapat dalam sebuah populasi. Menurut Suyanto, fungsi *fitness* bisa berupa fungsi yang memaksimalkan dan meminimalkan fungsi *Pi*, dimana keduanya sama dan bersifat kebalikan. Dalam permasalahan yang memaksimalkan fungsi *fitness* maka akan meminimalkan fungsi *Pi*. Fungsi *fitness* memiliki arti bahwa individu yang tinggi adalah individu yang mampu bertahan hidup dalam sebuah populasi [SUY-05]. Rumus fungsi *fitness* dengan model maksimasi yang digunakan adalah sebagai berikut :

$$\text{Fitness} = \frac{1}{1+P_i} \dots \dots \dots (2 - 10)$$

Keterangan :

P_i = Fungsi objektif atau fungsi penalti.

4. Seleksi

Proses seleksi akan melakukan pemilihan terhadap individu yang akan diikuti dalam proses reproduksi. Pada proses seleksi, keanekaragaman populasi memegang peranan penting dalam proses seleksi yaitu daerah sampling dan mekanisme seleksi [SET-03].

Sebelum dilakukan seleksi, jumlah anggota populasi ditambah dengan hasil *offspring* dari proses operasi genetik *crossover* dan mutasi. Hasil operasi genetik dan populasi selanjutnya di seleksi dengan metode tertentu untuk diambil sejumlah n anggota populasi yang terbaik sesuai dengan jumlah populasi awal [WID-07]. Proses seleksi juga bertujuan untuk memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang memiliki *fitness* lebih tinggi daripada individu lain. Ada beberapa metode seleksi, antara lain :

- *Rank-based Fitness* yaitu populasi diurutkan menurut nilai *fitnessnya*. Nilai *fitness* dari tiap-tiap individu akan diurutkan dari terbesar hingga terkecil yang kemudian akan diambil individu dengan *fitness* yang tinggi sejumlah populasi awal yang terbentuk yang akan digunakan untuk generasi selanjutnya.
- *Roulette Wheel Selection* Istilah lainnya adalah *stochastic sampling with replacement*. Individu-individu dipetakan dalam suatu segmen garis secara berurutan sedemikian hingga tiap-tiap segmen individu memiliki ukuran yang sama dengan ukuran *fitnessnya*. Sebuah bilangan random dibangkitkan dan individu yang memiliki segmen dalam kawasan segmen dalam kawasan bilangan random tersebut akan terseleksi. Proses ini berulang hingga didapatkan sejumlah individu yang diharapkan.
- *Stochastic universal sampling*
- *Local selection*

- *Truncation selection*
- *Tournament selection* akan menentukan individu-individu mana saja yang akan dipilih untuk dilakukan rekombinasi dan bagaimana *offspring* terbentuk dari individu-individu terpilih tersebut. Langkah pertama: Pencarian nilai *fitness*. Langkah kedua: Nilai *fitness* yang diperoleh digunakan pada tahap-tahap seleksi selanjutnya.

Pada penelitian ini digunakan metode seleksi *Rank-based Fitness* dikarenakan metode seleksi ini berfungsi untuk menyimpan individu yang terbaik pada tiap generasi yang akan digunakan untuk generasi selanjutnya.

5. Operator Genetika

Pada Metode Algoritma Genetika, terdapat 2 operator genetika yaitu operator perkawinan silang/*crossover* dan operator mutasi:

a. Perkawinan Silang/*Crossover*

Proses perkawinan silang (*crossover*) bertujuan menghasilkan anak/keturunan. Dalam *crossover*, diperlukan dua kromosom terpilih sebagai induk atau *parent*. Dalam proses *crossover* ini, akan dihasilkan dua kromosom anak. Kromosom anak yang dihasilkan merupakan kombinasi gen dari dua kromosom induk. Dalam *crossover* terdapat nilai Probabilitas *crossover* (P_c) yang akan menentukan frekuensi dari *crossover* yang terjadi dalam sebuah populasi. Semakin tinggi nilai P_c maka akan semakin besar kemungkinan dilakukan kawin silang [SET-03].

Dalam proses *crossover*, jumlah populasi sangat mempengaruhi hasil dari proses *crossover*. Jumlah populasi yang sangat kecil akan berakibat buruk, hal itu akan menyebabkan suatu kromosom dengan gen-gen yang mengarah pada solusi, akan sangat cepat menyebar ke kromosom-kromosom yang lain. Untuk mengatasi masalah tersebut digunakan suatu aturan bahwa proses *crossover* hanya dapat dilakukan dengan suatu probabilitas tertentu, artinya *crossover* hanya dapat

dilakukan dengan membangkitkan bilangan random [0..1] dan nilai bangkitan random adalah kurang dari nilai P_c yang telah ditentukan.

Secara umum, mekanisme *crossover* adalah sebagai berikut [SET-03] :

1. Memilih dua buah kromosom sebagai induk. Kromosom yang terpilih merupakan kromosom yang mempunyai nilai random dibawah nilai P_c .
2. Memilih secara acak posisi dalam kromosom, biasa disebut *crossover point*, sehingga masing-masing kromosom induk terbagi menjadi dua segmen.
3. Lakukan pertukaran antar bagian kromosom induk untuk menghasilkan kromosom anak.

Berikut beberapa teknik *crossover* yang dapat digunakan sesuai dengan teknik pengkodean kromosom yang dipilih :

a) Rekombinasi bernilai *real*

- Rekombinasi diskrit yaitu menukar nilai variabel antar kromosom induk.
- Rekombinasi garis : Hampir sama dengan rekombinasi menengah, hanya saja nilai *alpha* untuk semua variabel sama.
- Rekombinasi garis yang diperluas.
- *Simple Arithmetic Crossover*

Penelitian ini menggunakan teknik *crossover* ini, dimana teknik *Simple Arithmetic Crossover* dapat melakukan persilangan untuk teknik representasi berupa bilangan real. Teknik ini melakukan persilangan dua buah *parent* yang dilakukan dengan menukarkan sebesar k-bagian satu sama lain untuk menghasilkan anak (*child*), dimana sebelumnya akan ditentukan titik potong ke-k.

$$x'_1 = (x_1, \dots, x_k, y_{k+1} \times a + x_{k+1} \times (1 - a), \dots, y_q \times a + x_q \times (1 - a)) \dots \dots (2-11)$$

$$x'_2 = (y_1, \dots, y_k, x_{k+1} \times a + y_{k+1} \times (1 - a), \dots, x_q \times a + y_q \times (1 - a)) \dots \dots (2-12)$$

Dimana :

X = parent 1 ; Y = parent 2 ; x'_1 = child 1 ; x'_2 = child 2 ; a = nilai random antara 0 .. 1

Contoh Simple Arithmetic Crossover :

Parent 1 :

0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9
-----	-----	-----	-----	-----	-----	-----	-----	-----

Parent 2 :

0,3	0,2	0,3	0,2	0,3	0,2	0,3	0,2	0,3
-----	-----	-----	-----	-----	-----	-----	-----	-----

Child 1 :

0,1	0,2	0,3	0,4	0,5	0,6	0,5	0,5	0,6
-----	-----	-----	-----	-----	-----	-----	-----	-----

Child 2 :

0,3	0,2	0,3	0,2	0,3	0,2	0,5	0,5	0,6
-----	-----	-----	-----	-----	-----	-----	-----	-----

Gambar 2.2 Simple Arithmetic Crossover dengan $\alpha = 0,5$
Sumber : [SET-03]

b) Rekombinasi bernilai biner (*Crossover*)

- *Crossover* satu titik

Posisi penyilangan k ($k=1,2,\dots,N-1$) dengan $N =$ panjang kromosom diseleksi secara random. Variabel-variabel ditukar antar kromosom pada titik tersebut untuk menghasilkan anak. Misalkan terdapat 2 kromosom dengan panjang 12 :

induk1 : 0 1 1 1 0 | 0 1 0 1 1 1 0

induk2 : 1 1 0 1 0 | 0 0 0 1 1 0 1

Posisimenyilang yang terpilih: Misalkan 5. Setelah penyilangan, diperoleh kromosom-kromosom baru:

anak1 : 0 1 1 1 0 | 0 0 0 1 1 0 1

anak2 : 1 1 0 1 0 | 0 1 0 1 1 1 0

- *Crossover* banyak titik

Pada penyilangan ini, m posisi penyilangan k_i ($k=1,2,\dots,N-1, i=1,2,\dots,m$) dengan $N =$ panjang kromosom diseleksi secara random dan tidak diperbolehkan ada posisi yang sama, serta diurutkan naik. Variabel-variabel ditukar antar kromosom pada titik tersebut untuk menghasilkan anak.

- *Crossover* seragam



Setiap lokasi memiliki potensi sebagai tempat penyilangan. Sebuah mask penyilangan dibuat sepanjang panjang kromosom secara random yang menunjukkan bit-bit dalam mask yang mana induk akan mensupply anak dengan bit-bit yang ada. Induk yang mana yang akan menyumbangkan bit keanak yang dipilih secara random dengan probabilitas yang sama.

- *Crossover* dengan permutasi

Dengan permutasi, kromosom-kromosom anak diperoleh dengan cara memilih sub-barisan suatu *tour* dari satu induk dengan tetap menjaga urutan dan posisi sejumlah kota yang mungkin terhadap induk yang lainnya.

b. Mutasi

Mutasi merupakan bentuk operator genetika yang menukar nilai gen dengan nilai gen lain, dalam mutasi biner misalnya, gen yang bernilai 0 menjadi bernilai 1. Proses ini dilakukan secara acak pada posisi gen tertentu pada individu-individu yang terpilih untuk dimutasikan. Selain mutasi yang bernilai biner, terdapat juga mutasi yang bernilai *real*. Banyaknya individu yang mengalami mutasi ditentukan oleh probabilitas mutasi (P_m) [BAS-03].

Dalam representasi integer, ada beberapa teknik mutasi, diantaranya adalah *random resetting*. Dalam *random resetting*, akan dibangkitkan nilai P_m secara random dalam tiap gen sepanjang kromosom, dimana sebelumnya sudah ditentukan P_m , yang menjadi batasan. Kemudian nilai random pada tiap individu yang kurang dari P_m akan mengalami mutasi. Dalam representasi real atau floating point terdapat metode mutasi *uniform mutation*, teknik ini dianalogikan sama dengan *random resetting* seperti pada representasi integer [GEN-00].

Proses mutasi pada penelitian ini menggunakan metode Mutasi Random Gen dengan langkah-langkah dari metode mutasi adalah sebagai berikut [PRA-11] :

1. Membangkitkan bilangan random R $[0..1]$ sebanyak jumlah individu.
2. Apabila bilangan random $r < p_m$, maka individu tersebut terpilih untuk dikenai mutasi.
3. Memilih gen yang akan dimutasi (k) secara random.
4. Membangkitkan bilangan random antara 1 hingga jumlah kromosom sebanyak 1 bilangan dengan ketentuan bilangan (m) merupakan bilangan yang belum pernah terpakai pada populasi awal.
5. Mengganti nilai pada gen k dengan nilai m .

2.8 Diabetes *Melitus*

Diabetes *Melitus* adalah suatu penyakit, dimana tubuh penderitanya tidak bisa secara otomatis mengendalikan tingkat gula (glukosa) dalam darahnya atau glukosa berada pada kondisi yang melebihi nilai normal. Tingginya kadar gula karena kurang maksimalnya pemanfaatan hormon insulin yang diproduksi oleh pankreas atau tidak berfungsinya hormon insulin dalam menyerap gula secara maksimal, oleh sebab itu penyakit ini juga biasa disebut atau didefinisikan sebagai penyakit gula darah [DKS-08]. Klasifikasi Diabetes *Melitus* berdasarkan etiologis DM yaitu :

- a. Diabetes *Melitus* tipe 1 adalah penyakit gangguan metabolik yang ditandai oleh kenaikan kadar gula darah akibat destruksi (kerusakan) sel beta pankreas (kelenjar ludah perut) karena suatu sebab tertentu yang menyebabkan produksi insulin tidak ada sama sekali sehingga penderita memerlukan tambahan insulin dari luar.
- b. Diabetes *Melitus* tipe 2 adalah penyakit gangguan metabolik yang ditandai oleh kenaikan kadar gula darah akibat penurunan sekresi insulin oleh sel beta pankreas atau fungsi insulin (resistensi insulin).
- c. Diabetes *Melitus* tipe lain adalah penyakit gangguan metabolik yang ditandai oleh kenaikan kadar gula darah akibat defek genetik fungsi sel beta, defek genetik kerja insulin, penyakit eksokrin pankreas,

endokrinopati, karena atau zat kimia, infeksi, sebab imunologi yang jarang, sindrom genetik lain yang berkaitan dengan DM.

- d. Diabetes Melitus tipe Gestasional adalah penyakit gangguan metabolik yang ditandai oleh kenaikan kadar gula darah yang terjadi pada wanita hamil, biasanya terjadi pada usia 24 minggu masa kehamilan, dan setelah melahirkan kadar gula darah kembali normal.

Menurut konsensus PERKENI tahun 2006, diagnosis DM dapat dipastikan jika terdapat salah satu hasil pemeriksaan sebagai berikut :

- a. Gejala klasik DM dengan kadar glukosa darah sewaktu > 200 mg/dL. Gejala klasik DM yaitu sering kencing, sering lapar, sering haus, berat badan menurun cepat tanpa penyebab yang jelas.
- b. Gejala klasik DM dengan kadar glukosa darah puasa > 126 mg/dL.
- c. Pada tes toleransi glukosa oral (TTGO) didapatkan hasil pemeriksaan kadar glukosa darah 2 jam > 200 mg/dL sesudah pemberian beban glukosa 75 gr.

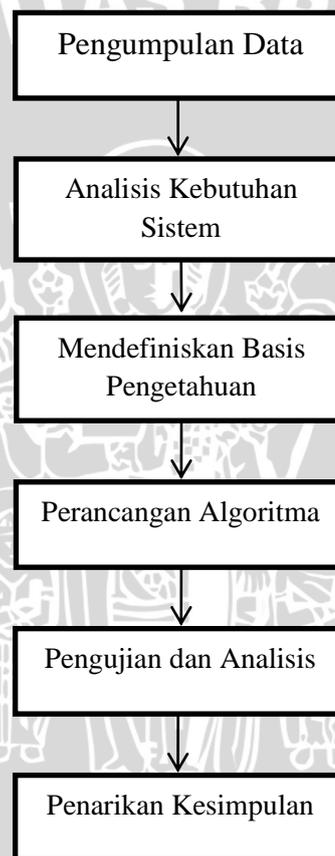
Diabetes Melitus mempunyai faktor resiko bersama dengan 5 (lima) penyakit tidak menular utama lainnya yaitu penyakit *kardiovaskuler*, *stroke*, penyakit paru *obstruktif kronis*/menahun, dan kanker. Faktor resiko bersama tersebut *preventable* (dapat dicegah) dan mempunyai kontribusi satu sama lainnya dalam menyebabkan terjadinya Penyakit Utama.

BAB III

METODE PENELITIAN DAN PERANCANGAN

3.1. Metode Penelitian

Metode penelitian yang digunakan dalam pembuatan Sistem Penerapan Algoritma Genetika pada Optimasi Fungsi Keanggotaan Inferensi Model Sugeno untuk menentukan Angka Metabolisme Basal (AMB) pada penderita Diabetes Melitus adalah sebagai berikut :



Gambar 3.1 Metode Penelitian Optimasi Fungsi Keanggotaan pada Perhitungan AMB dengan Algoritma Genetika

Penyusunan perhitungan yang sistematis menggunakan metode Algoritma Genetika dalam kasus Optimasi Fungsi Keanggotaan Inferensi Model Sugeno pada Perhitungan Angka Metabolisme Basal (AMB) dilakukan melalui beberapa tahapan yaitu diawali pada tahap

pengumpulan data pasien yang didapatkan dari penelitian sebelumnya, analisis kebutuhan sistem yang meliputi analisa penelitian sebelumnya dan penelitian yang dibangun, selanjutnya adalah mendefinisikan basis pengetahuan dari semua parameter yang digunakan. Pada tahap perancangan algoritma akan dijelaskan proses alur kerja dan perhitungan dari metode Algoritma Genetika dan *Fuzzy*. Setelah sistem diimplementasikan maka masuk ke dalam tahap pengujian dan analisis, kemudian tahap terakhir adalah penarikan kesimpulan yang berasal dari analisis terhadap kinerja sistem.

3.2. Teknik Pengumpulan Data

Dalam teknik pengumpulan data nantinya akan dikumpulkan data dan informasi yang akan dibutuhkan berkaitan dengan variabel yang dibutuhkan oleh sistem. Pengumpulan data dilakukan dengan menggunakan data dari penelitian yang telah dilakukan sebelumnya [FEB-12]. Untuk menentukan variabel numeris terhadap gejala-gejala yang ada maka diambil data dari pasien yang terkena penyakit diabetes melitus. Di bawah ini dijelaskan variabel yang digunakan oleh sistem dan sampel yang digunakan untuk proses pengujian.

3.2.1. Menentukan Variabel

Dalam penentuan variabel pada sistem Optimasi Fungsi Keanggotaan pada Perhitungan AMB dengan Algoritma Genetika yaitu:

1. Berat Badan dan Tinggi Badan menjadi Nilai IMT
2. Umur
3. Jenis Kelamin
4. Faktor Aktivitas
5. Probabilitas *Crossover*
6. Probabilitas Mutasi
7. Jumlah Generasi
8. Jumlah Populasi

Maka yang menjadi objek penelitian adalah bagaimana masukan dari probabilitas *crossover*, probabilitas mutasi, jumlah generasi dan

jumlah populasi akan menghasilkan batas-batas fungsi keanggotaan dengan menggunakan Algoritma Genetika. Serta masukan data dari berat badan, tinggi badan, umur, jenis kelamin dan faktor aktivitas diolah menjadi keluaran yang berupa nilai dari angka metabolisme basal (AMB) dari pasien penderita diabetes melitus menggunakan metode *Fuzzy Inference System* Model Sugeno dengan menggunakan fungsi keanggotaan yang dihasilkan oleh Algoritma Genetika.

3.2.2. Menentukan Sampel

Mengumpulkan data terhadap beberapa pasien dari penelitian yang telah dilakukan sebelumnya [FEB-12]. Jumlah data pasien yang menderita diabetes yaitu 40 orang sebagai *sample* untuk data latih dan 100 orang sebagai data uji dengan informasi data yang didapat berupa berat badan, tinggi badan, jenis kelamin, umur dan jenis aktivitas.

3.3. Analisis Kebutuhan Sistem

Dalam penelitian sebelumnya yang dijadikan acuan dalam pengembangan penelitian ini, telah diterapkan perhitungan Angka Metabolisme Basal (AMB) dengan metode Fuzzy Inference System model Sugeno dimana fungsi keanggotaannya masih terbentuk secara statis. Sedangkan akurasi pada penelitian tersebut mencatat bahwa hasil perhitungan yang dilakukan oleh sistem mencapai 75% terhadap perhitungan manual AMB menggunakan rumus WHO/FAO/UNU. Dari analisa yang dilakukan terhadap penelitian tersebut didapat bahwa penyebab salahnya hasil perhitungan adalah terletak pada pemetaan fuzzyfikasi yang dilakukan terhadap fungsi keanggotaan yang masih bersifat statis.

Oleh karena itu dalam penelitian ini digunakan sebuah metode untuk membuat secara dinamis batas-batas fungsi keanggotaan dengan menggunakan metode Algoritma Genetika. Dalam melakukan perhitungan Angka Metabolisme Basal (AMB) diperlukan beberapa nilai dari 5 kriteria, kriteria tersebut yakni berat badan, tinggi badan, umur, faktor aktivitas dan jenis kelamin, seperti pada penelitian sebelumnya.

Sedangkan untuk membuat batas-batas fungsi keanggotaan membutuhkan parameter masukan yaitu probabilitas *crossover* dan mutasi, jumlah generasi dan jumlah populasi yang diinginkan. Pada hasil akhir sistem didapatkan batas-batas fungsi keanggotaan yang digunakan untuk proses menghasilkan nilai Angka Metabolisme Basal (AMB) dari seorang pasien.

Maka dengan adanya sistem ini, diharapkan dapat untuk membantu para pekerja klinis dalam menentukan Angka Metabolisme Basal (AMB) dari pasien penderita diabetes. Pada sistem perhitungan Angka Metabolisme Basal, karakteristik dari data pasien akan dicocokkan dengan pengetahuan-pengetahuan yang ada pada basis pengetahuan. Sistem ini diimplementasikan berbasis *desktop* dengan bahasa pemrograman Java.

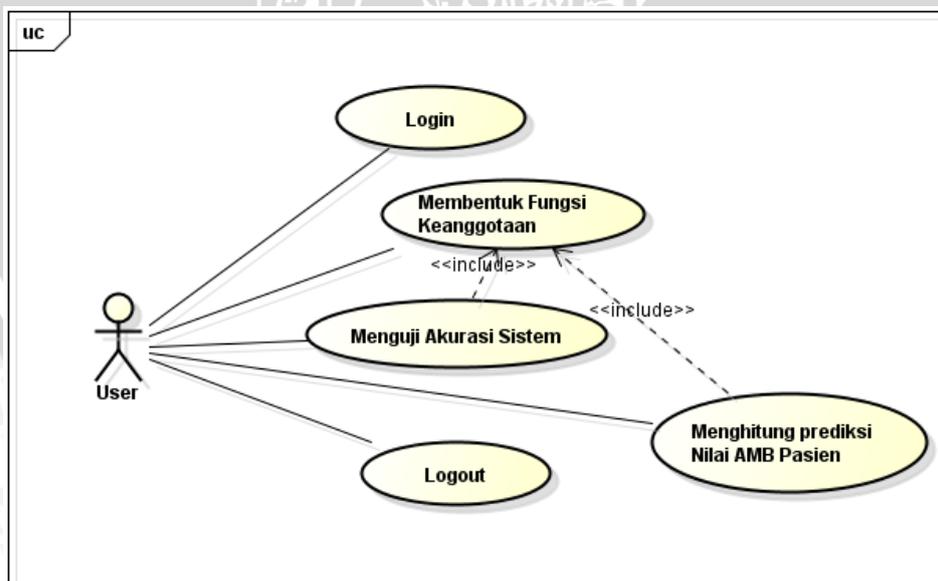
Analisis kebutuhan perangkat lunak meliputi analisis spesifikasi perangkat lunak menggunakan bahasa pemodelan UML (*Unified Modeling Language*). Permodelan dimaksudkan untuk mempermudah dalam memahami sistem yang akan dibangun. Analisis kebutuhan dilakukan dengan mengidentifikasi semua kebutuhan (*requirements*) system yang akan dimodelkan dalam diagram *use case* diagram, *sequence* diagram dan *class* diagram. Daftar kebutuhan sistem secara keseluruhan ditunjukkan pada Tabel 3.1.

Tabel 3.1 Daftar kebutuhan sistem

Kebutuhan	Aktor	Nama <i>Use case</i>
Sistem menyediakan <i>interface</i> untuk login bagi <i>user</i>	<i>user</i>	Login
Sistem menyediakan <i>interface</i> untuk form nilai probabilitas <i>crossover</i> , probabilitas mutasi jumlah generasi dan jumlah populasi	<i>user</i>	Membentuk Fungsi Keanggotaan
Sistem menyediakan <i>interface</i> untuk melihat hasil akurasi dari sistem.	<i>user</i>	Menguji Akurasi Sistem
Sistem menyediakan <i>interface</i> untuk form data pasien	<i>user</i>	Menghitung prediksi AMB Pasien
Sistem menyediakan <i>interface</i> untuk logout bagi <i>user</i>	<i>user</i>	Logout

3.3.1. Use Case Diagram

Use case diagram menggambarkan fungsionalitas dari suatu sistem, sehingga pengguna sistem paham dan mengerti mengenai kegunaan sistem yang akan dibangun. Semua perilaku dimodelkan sebagai *use case* yang mungkin dispesifikasikan mandiri dari realisasinya. *Use case* mendeskripsikan kumpulan urutan (*sequence*) di mana tiap urutan menjelaskan interaksi sistem dengan sesuatu di luar sistem (sering dinamakan dengan aktor). Penjabaran setiap *use case* diagram yaitu menjelaskan secara detail mengenai fungsionalitas keseluruhan dari sistem Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno. Fungsionalitas dari beberapa aktor yang terdapat pada *use case* dapat digambarkan secara detail mengenai aktivitas-aktivitas yang dilakukan oleh aktor. Perancangan *Use case* untuk Sistem Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno akan dijelaskan pada Gambar 3.2.



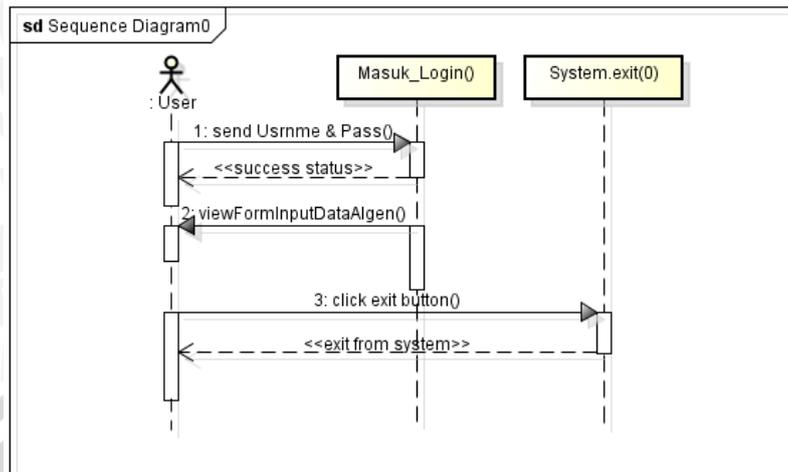
Gambar 3.2 Diagram *use case* sistem

3.3.2. Skenario Use Case

Masing-masing *use case* yang terdapat pada diagram *use case*, dijabarkan dalam skenario *use case* secara lebih detail. Pada skenario *use case*, akan diberikan uraian nama *use case*, aktor yang berhubungan dengan *use case* tersebut, tujuan dari *use case*, deskripsi tentang *use case*, kondisi awal yang harus dipenuhi dan kondisi akhir yang diharapkan setelah berjalannya fungsional *use case*. Pada skenario *use case* juga akan diberikan ulasan yang berkaitan dengan tanggapan dari sistem terhadap aksi yang diberikan oleh aktor. Skenario *use case* juga terdapat kejadian alternatif yang merupakan jalannya sistem jika terdapat kondisi tertentu. Penjelasan skenario dari masing-masing *use case* akan dijelaskan seluruhnya dalam bentuk tabel yang tertera pada Lampiran 1.

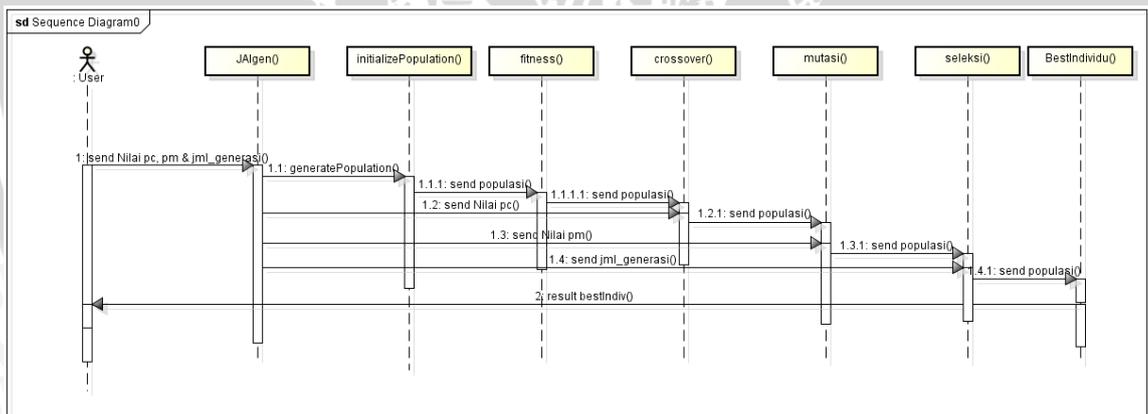
3.3.3. Sequence Diagram

Sequence diagram adalah diagram yang digunakan untuk melihat *behavior* beberapa objek di dalam *use case* tunggal. *Sequence diagram* baik untuk memperlihatkan kolaborasi antar objek. Masing-masing *sequence diagram* menggambarkan aliran-aliran pada suatu *use case*. *Sequence diagram* dapat dibaca dengan melihat pada objek-objek dan pesan-pesan (*message*). Objek-objek yang berperan dalam aliran diperlihatkan pada kotak persegi panjang yang melintas pada bagian atas diagram. Setiap objek memiliki garis hidup (*lifetime*), yang digambarkan sebagai garis vertikal di bawah nama suatu objek. Garis hidup dimulai pada saat objek terbentuk. Pesan-pesan digambarkan di antara garis hidup yang dimiliki objek untuk memperlihatkan bagaimana objek-objek itu saling berkomunikasi. Pada perancangan *sequence diagram* ini, dijelaskan secara rinci dari tiap-tiap *use case* tunggal yang telah dirancanag sebelumnya. Gambar 3.3 menunjukkan *Sequence Diagram* pada *use case* Login Pengguna.



Gambar 3.3 Sequence diagram Login

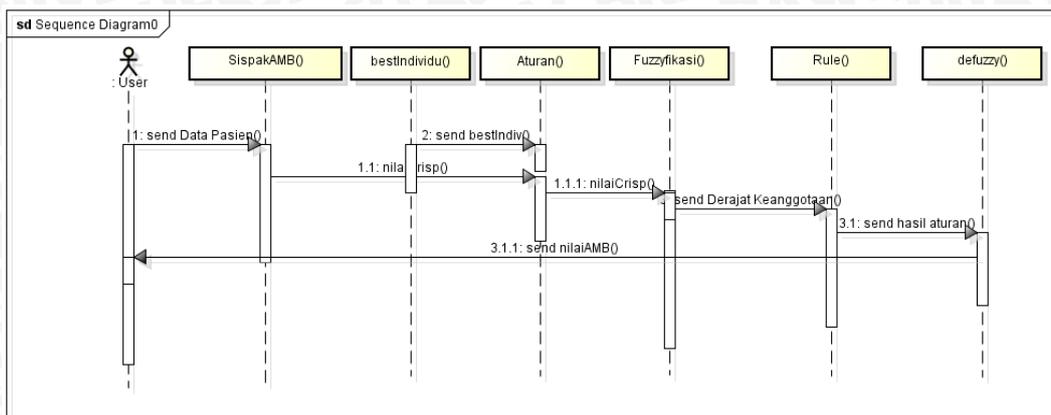
Selanjutnya pada gambar 3.4 adalah *sequence diagram* dari *use case* Membentuk Fungsi Keanggotaan. *Sequence diagram* dibawah ini akan menjelaskan setiap objek dan pesan yang digunakan untuk menggambarkan sistem di dalam proses pembentukan batas-batas fungsi keanggotaan dengan Algoritma Genetika.



Gambar 3.4 Sequence diagram pembentukan fungsi keanggotaan

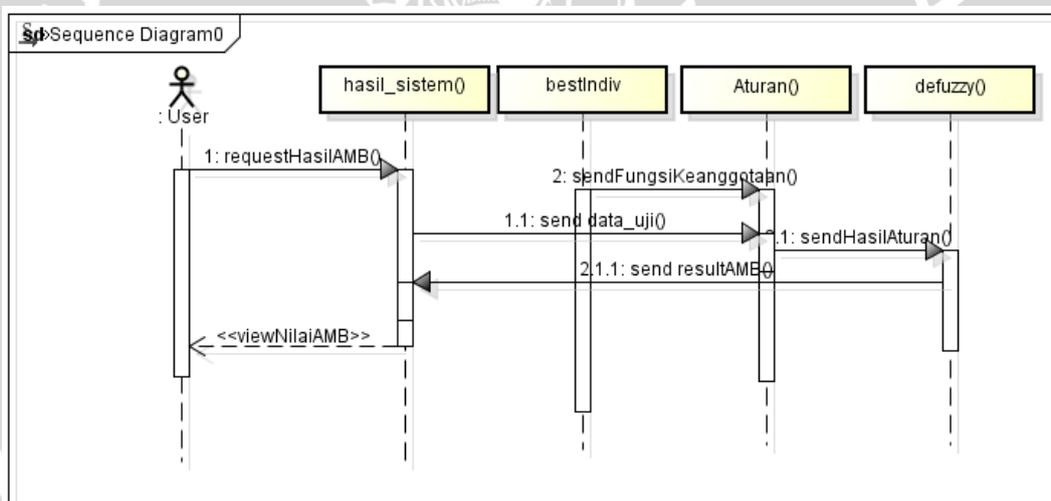
Pada gambar 3.5 adalah *sequence diagram* dari *use case* Memproses Perhitungan AMB. *Sequence diagram* dibawah ini akan menjelaskan setiap objek dan pesan yang digunakan untuk menggambarkan sistem di dalam proses untuk menghasilkan Angka Metabolisme Basal (AMB) dengan metode *Fuzzy* model Sugeno.





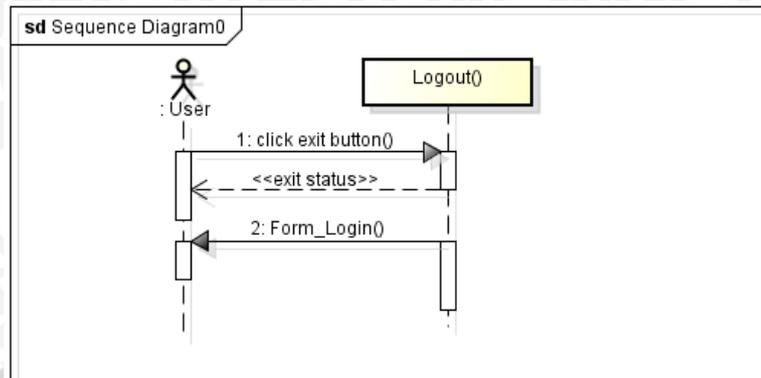
Gambar 3.5 Sequence diagram perhitungan nilai AMB

Gambar 3.6 adalah sequence diagram dari use case pengujian akurasi data. Sequence diagram dibawah ini akan menjelaskan setiap objek dan pesan yang digunakan untuk menggambarkan sistem di dalam menguji akurasi data sistem.



Gambar 3.6 Sequence diagram pengujian akurasi data

Gambar 3.7 adalah sequence diagram dari use case Logout. Sequence diagram dibawah ini akan menjelaskan setiap objek dan pesan yang digunakan untuk menggambarkan sistem di dalam proses untuk keluar dari aplikasi.

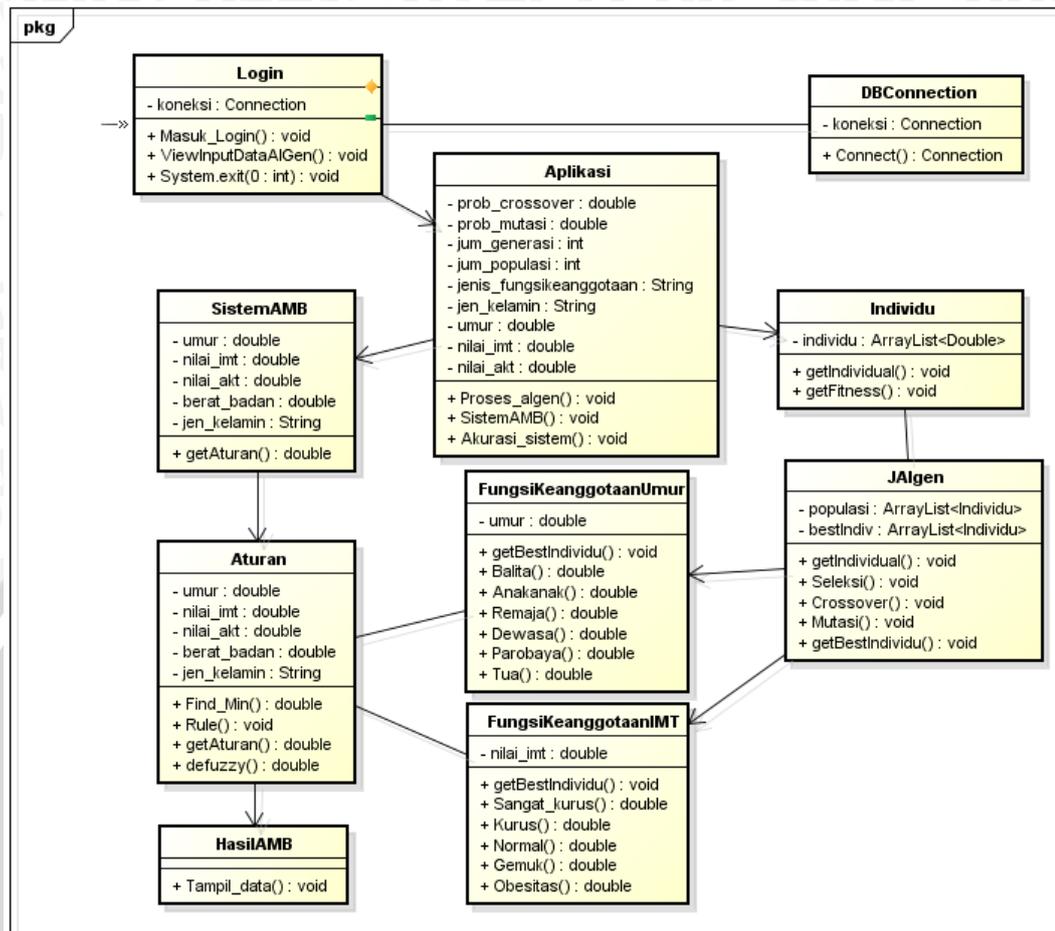


Gambar 3.7 Sequence diagram logout

3.3.4. Class Diagram

Class adalah sebuah spesifikasi yang akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class Diagram* menggambarkan keadaan (atribut / properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode / fungsi). *Class diagram* menggambarkan struktur dan deskripsi *class*, *package* dan *object* beserta hubungan satu sama lain seperti pewarisan, asosiasi, dan lain – lain. *Class* memiliki tiga cara area pokok yaitu : Nama (dan *stereotype*), Atribut dan Metode.

Gambar 3.8 menunjukkan *Class Diagram* Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno.



Gambar 3.8 Class diagram sistem

3.4. Basis Pengetahuan

Basis pengetahuan berisi tentang pengetahuan yang relevan yang diperlukan untuk memahami, merumuskan, dan memecahkan persoalan. Basis pengetahuan tersebut mencakup dua elemen dasar, yaitu fakta dan aturan khusus yang mengarahkan pengguna pengetahuan untuk memecahkan persoalan khusus dalam domain tertentu.

Pada tabel 3.2 merupakan tabel penjelasan dari kriteria-kriteria masukan pada Sistem Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System Model Sugeno*.

Tabel 3.2 Basis pengetahuan sistem

No	Kriteria	Keterangan
1	Berat Badan	Kriteria yang digunakan adalah berat badan pasien dalam satuan kg.
2	Tinggi Badan	Kriteria yang digunakan adalah tinggi badan pasien dalam satuan cm. Nilai dari berat badan dan tinggi badan akan menghasilkan nilai IMT
3	Nilai IMT	IMT (Indeks Massa Tubuh) merupakan salah satu metode yang digunakan untuk mengukur status gizi individu. Cara menghitung IMT adalah dengan membagi berat badan dalam kg dengan kuadrat dari tinggi badan dalam meter. Nilai IMT dibagi menjadi 5 kategori yaitu sangat kurus, kurus, normal, gemuk dan obesitas.
3	Umur	Kriteria dari umur pasien dibagi menjadi 6, yaitu balita, anak-anak, remaja, dewasa, parobaya dan tua.
4	Jenis Kelamin	Kriteria dari jenis kelamin dibagi dua yaitu laki-laki dan perempuan.
5	Jenis Aktivitas	Kriteria dari jenis aktivitas dikategorikan dalam 4 jenis yaitu aktivitas sangat ringan, aktivitas ringan, aktivitas sedang dan aktivitas berat.
6	Probabilitas Crossover	Nilai yang dimasukkan ke dalam sistem yang berfungsi sebagai nilai untuk menentukan probabilitas jumlah <i>parent</i> pada <i>crossover</i> .
7	Probabilitas Mutasi	Nilai yang dimasukkan ke dalam sistem yang berfungsi sebagai nilai untuk menentukan probabilitas jumlah induk yang mengalami mutasi.
8	Jumlah Generasi	Menyatakan satu-satuan siklus proses evolusi.
9	Jumlah Populasi	Kumpulan individu yang membentuk suatu populasi pada tiap-tiap generasi.
10	Nilai AMB	Angka Metabolisme Basal (AMB) adalah kebutuhan energi minimal yang dibutuhkan tubuh untuk menjalankan proses tubuh yang vital yang dinyatakan dalam satuan kkal.

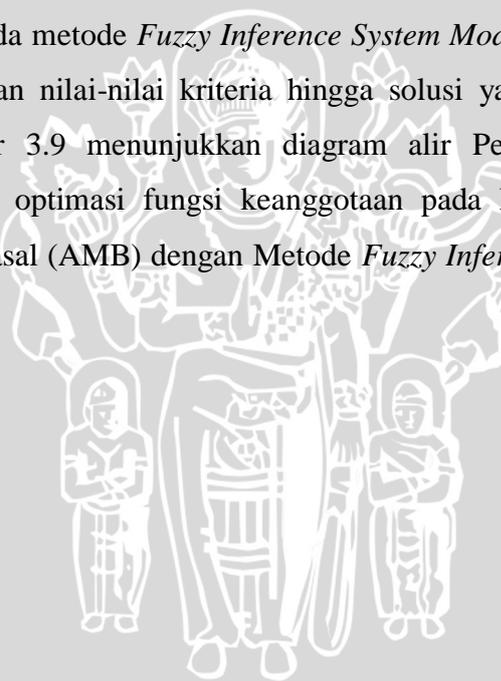
3.5. Perancangan Sistem

Perancangan aplikasi dilakukan setelah semua kebutuhan sistem didapatkan melalui tahap analisis kebutuhan. Tahap perancangan sistem ini memiliki tujuan untuk mengubah model informasi yang telah dibuat

selama tahapan analisa menjadi model yang sesuai dengan teknologi yang akan digunakan dalam pengimplementasian sistem.

Pada tahap perancangan ini, dijelaskan perancangan yang dilakukan pada setiap proses untuk membentuk batas-batas fungsi keanggotaan menggunakan Algoritma Genetika dan proses untuk melakukan perhitungan Angka Metabolisme Basal (AMB) menggunakan metode *Fuzzy Inference System Model Sugeno*.

Perancangan algoritma sistem menggunakan notasi diagram alir. Diagram alir menggunakan notasi-notasi untuk menggambarkan arus data yang membantu dalam proses memahami pemodelan menggunakan Algoritma Genetika yang diterapkan untuk melakukan optimasi fungsi keanggotaan pada metode *Fuzzy Inference System Model Sugeno*. Dimulai dari memasukkan nilai-nilai kriteria hingga solusi yang dihasilkan oleh sistem. Gambar 3.9 menunjukkan diagram alir Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System Model Sugeno*.





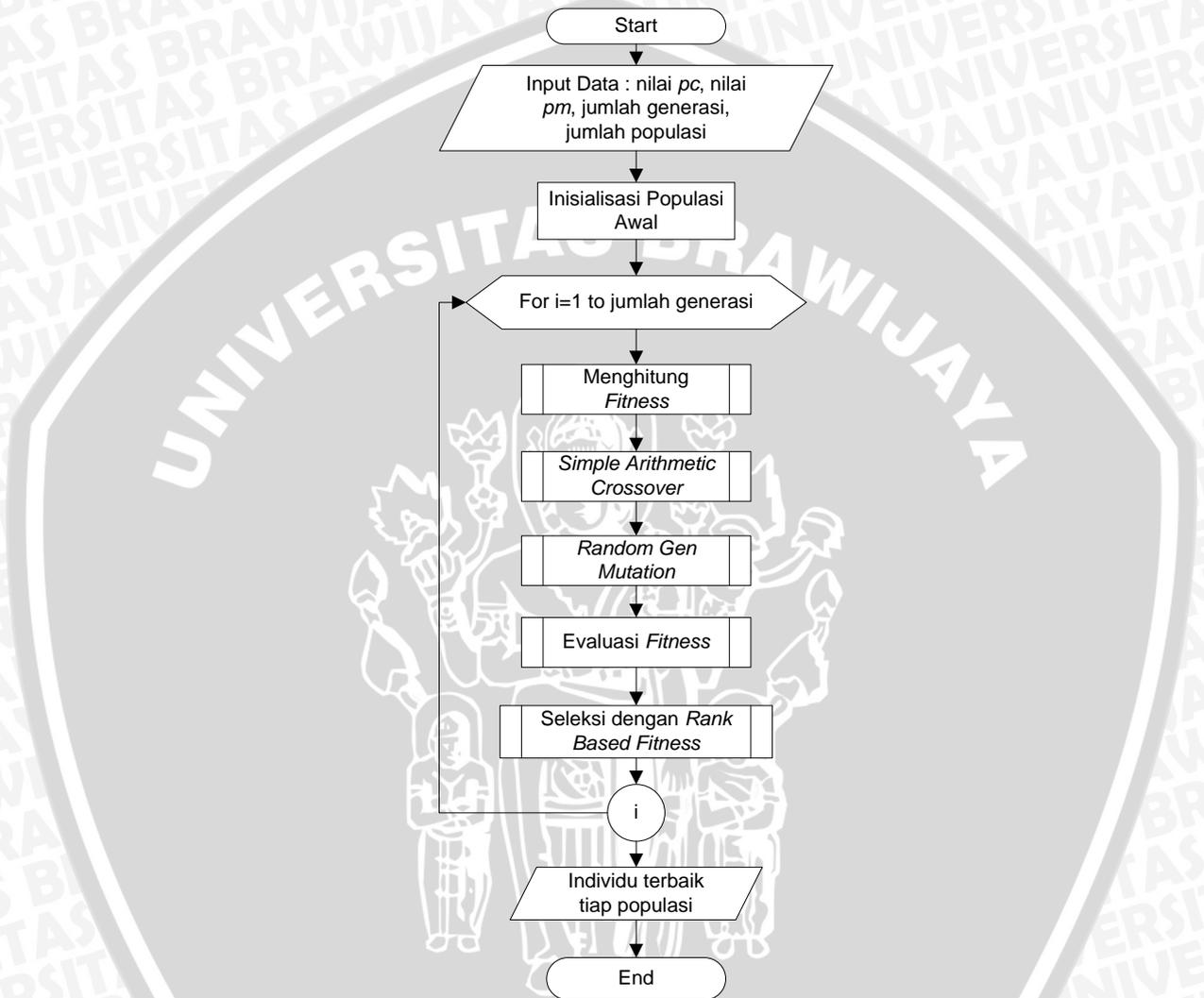
Gambar 3.9 Diagram alir perancangan sistem

Pada bagian ini dibahas perancangan dari Sistem Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System Model Sugeno*. Sistem ini memiliki beberapa proses algoritma yang harus dibuat agar sistem dapat berjalan sesuai kebutuhan. Proses algoritma yang diperlukan untuk penyusunan perhitungan nilai Angka Metabolisme Basal (AMB) antara lain Algoritma Genetika dan Algoritma Fuzzy Model Inferensi Sugeno.

3.5.1. Perancangan Fungsi Keanggotaan dengan Algoritma Genetika

Metode Algoritma Genetika digunakan sebagai optimasi di dalam pembentukan fungsi keanggotaan (*membership function*) pada permasalahan Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System Model Sugeno*. Tahapan yang

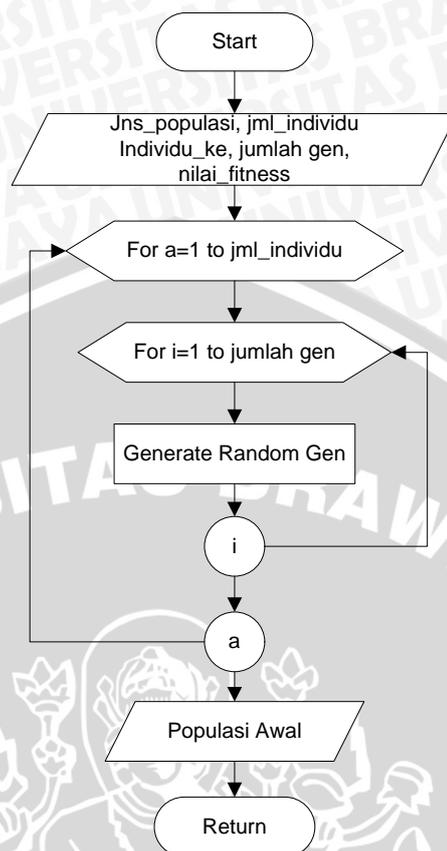
dilakukan di dalam Metode Algoritma Genetika meliputi Inisialisasi Kromosom, Seleksi, *Crossover*, dan Mutasi. Hasil akhir dari Algoritma Genetika ini adalah *range* nilai fungsi keanggotaan. Diagram alir proses Algoritma Genetika dapat dilihat pada Gambar 3.10.



Gambar 3.10 Diagram alir proses Algoritma Genetika

3.5.1.1. Inisialisasi Populasi Awal

Individu terdiri dari gen-gen yang merupakan suatu pengkodean yang digunakan untuk merepresentasikan masalah kombinasi batas-batas fungsi keanggotaan pada dua fungsi keanggotaan yakni fungsi keanggotaan Umur dan fungsi keanggotaan IMT. Diagram alir untuk proses inisialisasi populasi awal akan ditunjukkan dalam Gambar 3.11.



Gambar 3.11 Diagram alir proses Inisialisasi Populasi Awal

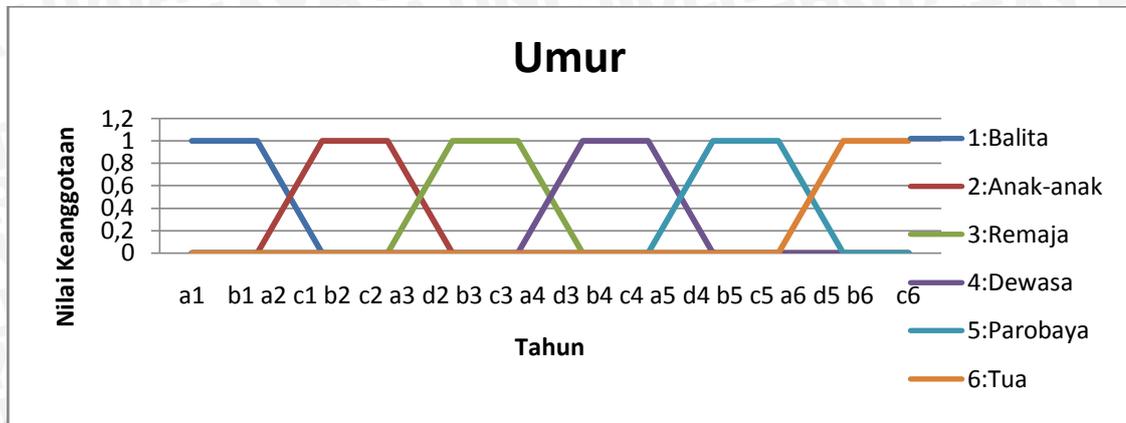
Masing masing individu akan mempunyai gen yang berisi angka acak yang merupakan bilangan *real* bertipe *floating point* yang dibangkitkan secara random oleh sistem. Random yang dibangkitkan, dibatasi rantang nilainya berdasarkan jenis variabel. Berikut pada tabel 3.3 menjelaskan nilai random yang harus dibangkitkan pada tiap populasi dengan digit 1 dibelakang koma.

Tabel 3.3 Pembentukan bilangan random populasi awal

Keanggotaan IMT	Range Random	Keanggotaan Umur	Range Random
Sangat Kurus	0.0 – 19.0	Balita	0.0 – 4.0
Kurus	15.0 – 20.0	Anak-anak	2.0 – 11.0
Normal	17.0 – 27.0	Remaja	9.0 – 19.0
Gemuk	23.0 – 29.0	Dewasa	17.0 – 31.0
Obesitas	25.0 – 50.0	Parobaya	29.0 – 62.0
		Tua	54.0 – 110.0

Sehingga tabel kromosom dibawah ini merupakan representasi individu dari masing-masing derajat keanggotaan pada fungsi keanggotaan.

a) Fungsi Keanggotaan Umur



Gambar 3.12 Diagram fungsi keanggotaan Umur

1. Kromosom Balita:

a1	b1	c1
----	----	----

2. Kromosom Anak-anak:

a2	b2	c2	d2
----	----	----	----

3. Kromosom Remaja:

a3	b3	c3	d3
----	----	----	----

4. Kromosom Dewasa:

a4	b4	c4	d4
----	----	----	----

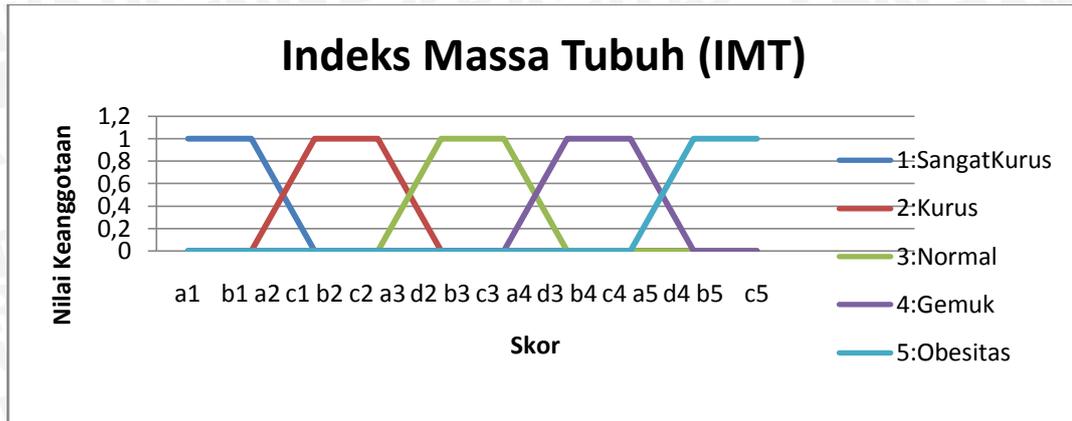
5. Kromosom Parobaya:

a5	b5	c5	d5
----	----	----	----

6. Kromosom Tua:

a6	b6	c6
----	----	----

b) Fungsi Keanggotaan IMT



Gambar 3.13 Diagram fungsi keanggotaan IMT

1. Kromosom Sangat Kurus:

a1	b1	c1
----	----	----

2. Kromosom Kurus:

a2	b2	c2	d2
----	----	----	----

3. Kromosom Normal:

a3	b3	c3	d3
----	----	----	----

4. Kromosom Gemuk:

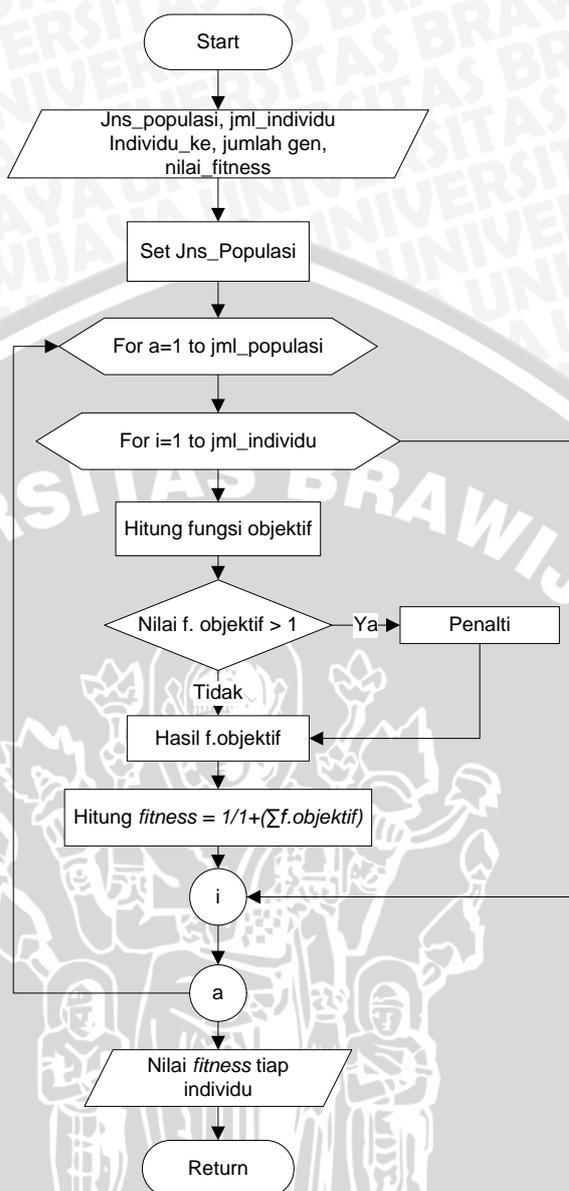
a4	b4	c4	d4
----	----	----	----

5. Kromosom Obesitas :

a6	b6	c6
----	----	----

3.5.1.2. Menghitung Nilai Fitness

Fungsi *fitness* merupakan fungsi yang digunakan untuk mengevaluasi setiap individu yang terdapat dalam sebuah populasi. Fungsi *fitness* dalam penelitian ini merupakan suatu fungsi optimasi, sehingga semakin besar nilai *fitness*, maka individu semakin baik. Diagram alir pada Gambar 3.14 menjelaskan dari proses pada perhitungan nilai *fitness*.



Gambar 3.14 Diagram alir proses perhitungan nilai *fitness*

Rumus fungsi *fitness* yang digunakan seperti pada rumus (2-9). Sedangkan fungsi objektif yang dibuat disesuaikan dengan rentang domain nilai dari variabel IMT dan variabel umur. Fungsi objektif digunakan sebagai perhitungan *fitness* suatu individu. Sehingga apabila nilai-nilai yang dihasilkan individu mendekati fungsi objektif yang telah dibuat maka individu tersebut semakin baik hasil *fitness*nya. Berikut tabel 3.4 menjelaskan range dan domain *fuzzy* dari pakar.

Tabel 3.4 Range Skor Umur

No.	Derajat Keanggotaan	Range Tahun	Domain Himpunan Fuzzy
1.	Balita	0 – 3	[0 3,5]
2.	Anak-anak	3 – 10	[2,5 10,5]
3.	Remaja	10 – 18	[9,5 18,5]
4.	Dewasa	18 – 30	[17,5 30,5]
5.	Parobaya	30 – 60	[29,5 60,5]
6.	Tua	≥60	[55,5 110,0]

Untuk persamaan fungsi objektif diambil dari range domain *fuzzy* yang diambil dari pakar. Sehingga persamaan fungsi objektif dibuat dengan membandingkan apakah nilai-nilai pada individu mendekati range domain *fuzzy*. Untuk jumlah gen pada individu sebanyak 3 maka ada 3 fungsi objektif, sedangkan untuk 4 gen maka ada 6 fungsi objektif. Fungsi objektif akan membandingkan seluruh kombinasi 2 gen terhadap domain himpunan fuzzy dari pakar. Berikut salah satu contoh fungsi objektif dari fungsi keanggotaan Umur pada variabel Anak-anak.

$$\text{Fungsi objektif anak-anak } (f1) = \frac{b2-a2}{3,5-2,5}$$

$$\text{Fungsi objektif anak-anak } (f2) = \frac{c2-b2}{9,5-3,5}$$

$$\text{Fungsi objektif anak-anak } (f3) = \frac{c2-a2}{9,5-2,5}$$

$$\text{Fungsi objektif anak-anak } (f4) = \frac{d2-c2}{10,5-9,5}$$

$$\text{Fungsi objektif anak-anak } (f5) = \frac{d2-b2}{10,5-3,5}$$

$$\text{Fungsi objektif anak-anak } (f6) = \frac{d2-a2}{10,5-2,5}$$

Sedangkan untuk tabel *range* skor dan domain himpunan *fuzzy* dari Indeks Massa Tubuh (IMT) dijelaskan pada Tabel 3.5.

Tabel 3.5 Range Skor Nilai IMT

No.	Derajat Keanggotaan	Range Skor	Domain Himpunan Fuzzy
1.	Sangat Kurus	0 – 17	[0 17,5]
2.	Kurus	17 – 18,5	[16,5 19]
3.	Normal	18,5 – 25	[18 25,5]
4.	Gemuk	25 – 27	[24,5 27,5]
5.	Obesitas	< 27	[26,5 50]

Untuk persamaan fungsi objektif pada fungsi keanggotaan IMT salah satu contohnya adalah variabel Kurus. Berikut di bawah ini merupakan persamaan fungsi objektif variabel kurus.

$$\text{Fungsi objektif kurus } (f1) = \frac{b2-a2}{17,5-16,5}$$

$$\text{Fungsi objektif kurus } (f2) = \frac{c2-b2}{18,0-17,5}$$

$$\text{Fungsi objektif kurus } (f3) = \frac{c2-a2}{18,0-16,5}$$

$$\text{Fungsi objektif kurus } (f4) = \frac{d2-c2}{19,0-18,0}$$

$$\text{Fungsi objektif kurus } (f5) = \frac{d2-b2}{19,0-17,5}$$

$$\text{Fungsi objektif kurus } (f6) = \frac{d2-a2}{19,0-16,5}$$

Tahap selanjutnya adalah memasukkan hasil perhitungan dari fungsi objektif kedalam fungsi *fitness* pada masing-masing individu pada tiap populasi. Nilai *fitness* inilah yang akan digunakan sebagai penentu baik tidaknya suatu individu. Di bawah ini merupakan fungsi *fitness* yang digunakan.

$$\text{Nilai Fitness Jumlah gen 3} = \frac{1}{1+(3-(f1+f2+f3))} \quad (3-1)$$

$$\text{Nilai Fitness Jumlah gen 4} = \frac{1}{1+(6-(f1+f2+f3+f4+f5+f6))} \quad (3-2)$$

Keterangan : F(n) merupakan hasil dari perhitungan fungsi objektif pada masing-masing fungsi keanggotaan berdasarkan tabel 3.4 dan tabel 3.5.

3.5.1.3. Pendefinisian Penalti

Nilai maksimum tiap fungsi objektif telah ditentukan dimana harus mempunyai nilai maksimum 1.0. Penggunaan penalti ini dimaksudkan untuk menghindari nilai fungsi objektif yang berlebih akibat dari gen yang mempunyai *range* terlalu jauh akibat random. Sehingga apabila nilai fungsi objektif melebihi batas maksimum yang ditentukan akan dikenai sebuah penalti atau hukuman. Rumus penalti yang digunakan adalah sebagai berikut :

$$\text{Penalti} = f.obj\ max - (\text{nilai } f.obj - f.obj\ max) \quad (3-3)$$

Keterangan : *f.obj max* merupakan nilai maksimum fungsi objektif yakni selalu 1 sedangkan *nilai f.obj* adalah masukan yang berupa hasil perhitungan *f.obj* untuk dikenai penalti.

3.5.1.4. Crossover

Pada proses *crossover*, metode yang digunakan salah satunya adalah *Simple Arithmetic Crossover*, yaitu melakukan persilangan dua buah *parent* yang dilakukan dengan menukarkan sebesar *k*-bagian satu sama lain untuk menghasilkan anak (*offspring*), dimana sebelumnya akan ditentukan titik potong ke-*k* berdasarkan nilai *alpha*. Kromosom yang dijadikan induk dipilih secara acak dan jumlah kromosom yang mengalami *crossover* dipengaruhi oleh parameter *crossover rate* atau probabilitas *crossover* (*pc*).

Tiap-tiap individu dibangkitkan nilai Random $R[0..1]$, nilai random ini akan melekat pada tiap individu. Misal pada perhitungan ditentukan *crossover probability* adalah sebesar 0.5, maka diharapkan kromosom yang memiliki nilai *R* dibawah nilai *Pc* dari tiap satu populasi mengalami proses *crossover*. Maka kromosom ke *k* akan dipilih sebagai induk jika $R[k] < pc$. Selanjutnya posisi *single cut-point crossover* dibangkitkan secara random dari 0 hingga jumlah gen kromosom, seperti contoh pada *crossover* dibawah ini :

Contoh Simple Arithmetic Crossover dengan $\alpha = 0,5$ dan *cut point* = 2 :

Parent 1 :

0	0,5	3
---	-----	---

Parent 2 :

0	1,5	2,5
---	-----	-----

Selanjutnya melakukan proses perhitungan sesuai dengan persamaan *Simple Arithmetic Crossover* yang telah dijelaskan pada persamaan (2-11) dan (2-12) maka perhitungan manual untuk mendapat hasil child 1 dan child 2 seperti dibawah ini :

Child 1 gen ke 1 (setelah *cut point*)

$$C_1 = (P_2 \times \alpha) + (P_1 \times (1 - \alpha)) = (2,5 \times 0,5) + (3 \times 0,5) = 1,25 + 1,5 \\ = 2,75$$

Child 2 gen ke 1 (setelah *cut point*)

$$C_2 = (P_1 \times \alpha) + (P_2 \times (1 - \alpha)) = (3 \times 0,5) + (2,5 \times 0,5) = 1,5 + 1,25 \\ = 2,75$$

Sehingga hasil individu setelah mengalami *crossover* dengan teknik *Simple Arithmetic Crossover* adalah sebagai berikut.

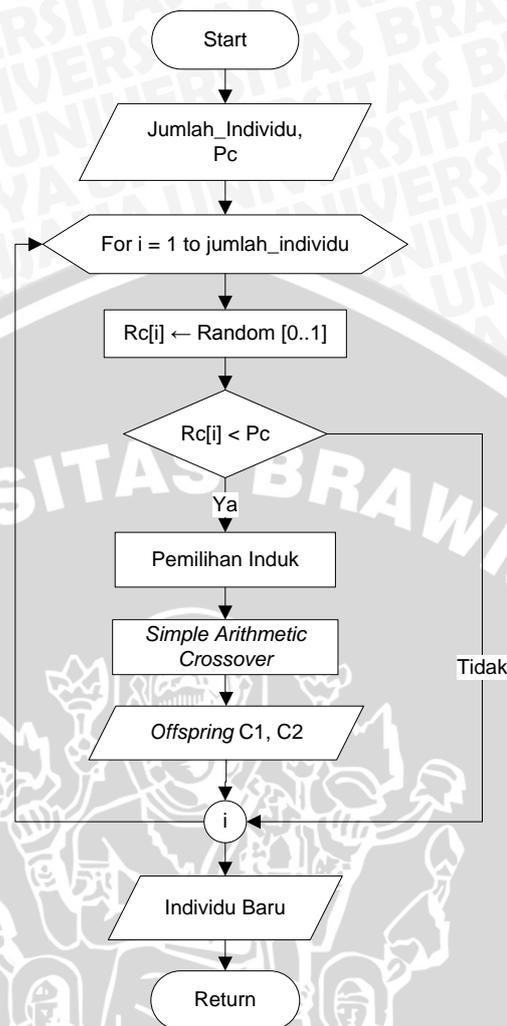
Child 1 :

0	0,5	2,75
---	-----	------

Child 2 :

0	1,5	2,75
---	-----	------

Flowchart atau diagram alir untuk proses *Simple Arithmetic Crossover* akan ditunjukkan dalam Gambar 3.15.



Gambar 3.15 Diagram alir proses *Crossover*

3.5.1.5. Mutasi

Jumlah kromosom yang mengalami mutasi dalam satu populasi ditentukan oleh parameter *mutation rate* atau probabilitas mutasi (pm). Proses mutasi dilakukan dengan metode Random Gen yaitu dengan cara mengganti satu gen yang terpilih secara acak dengan suatu nilai baru yang didapat secara acak.

Langkah mutasi hampir sama dengan *crossover*, yakni tiap-tiap individu dibangkitkan nilai Random $R[0..1]$, nilai random ini akan melekat pada tiap individu. Misal pada perhitungan ditentukan probabilitas mutasi (Pm) adalah sebesar 0.5, maka diharapkan chromosome yang memiliki nilai

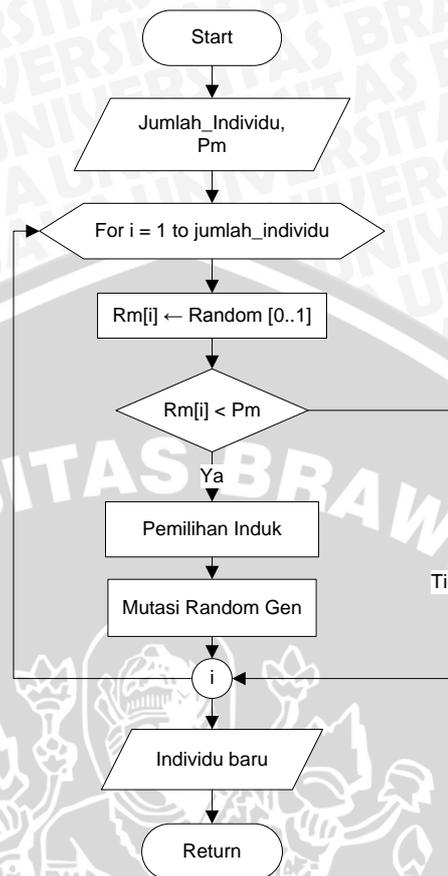
R dibawah nilai P_m dari tiap satu populasi mengalami proses mutasi. Maka kromosom ke k akan dipilih untuk dikenai mutasi jika $R[k] < p_m$.

Setelah individu yang mengalami mutasi terpilih, selanjutnya menentukan posisi gen pada tiap-tiap individu yang akan mengalami mutasi. Penentuan posisi gen dilakukan secara random agar didapat hasil individu baru yang bervariasi. Maka nilai gen pada posisi tersebut diganti dengan bilangan acak yang telah ditentukan. Berikut tabel bilangan acak yang dibangkitkan yang telah diatur panjang rangenya dengan digit 1 dibelakang koma, agar nilai random yang dihasilkan tidak terlalu jauh hasilnya.

Tabel 3.6 Pembentukan bilangan random pada mutasi

Keanggotaan IMT	Range Random	Keanggotaan Umur	Range Random
Sangat Kurus	0.0 – 19.0	Balita	0.0 – 4.0
Kurus	15.0 – 20.0	Anak-anak	2.0 – 11.0
Normal	17.0 – 27.0	Remaja	9.0 – 19.0
Gemuk	23.0 – 29.0	Dewasa	17.0 – 31.0
Obesitas	25.0 – 50.0	Parobaya	29.0 – 62.0
		Tua	54.0 – 110.0

Flowchart untuk proses mutasi akan ditunjukkan dalam Gambar 3.16.



Gambar 3.16 Diagram alir proses Mutasi Gen

3.5.1.6. Evaluasi *Fitness*

Setelah proses mutasi selesai dilakukan, maka tahapan selanjutnya adalah proses evaluasi *fitness*. Proses ini bertujuan untuk mengevaluasi *fitness* pada individu yang merupakan hasil dari proses mutasi. Hasil dari perhitungan *fitness* pada individu yang telah mengalami mutasi akan digunakan pada tahap selanjutnya yaitu pada proses seleksi, dimana proses seleksi menggunakan metode perankingan dari keseluruhan hasil *fitness*, mulai dari nilai *fitness* individu awal dan nilai *fitness* individu hasil mutasi. Proses yang dilakukan sama seperti pada subbab perancangan perhitungan *fitness* diatas yakni menggunakan rumus *fitness* seperti pada persamaan (3-1).

3.5.1.7. Seleksi

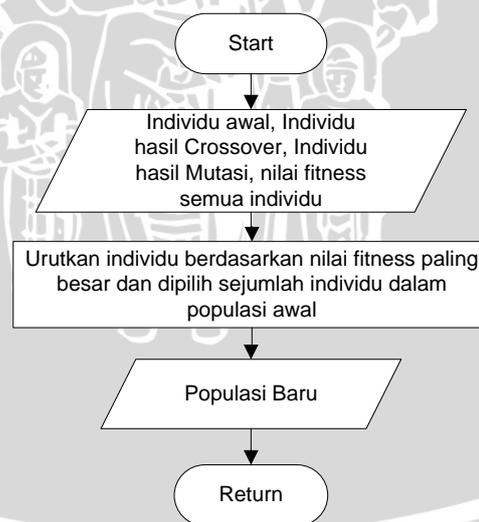
Proses seleksi digunakan untuk memilih individu-individu yang akan digunakan dalam iterasi atau generasi berikutnya. Proses *crossover* dan mutasi yang telah dilakukan sebelumnya menghasilkan keturunan (*offspring*)

sebagai individu baru. Ukuran satu populasi tiap generasi selalu sama, oleh karena itu hanya individu yang terbaiklah yang mampu bertahan dalam populasi atau keturunan terbaiklah yang akan masuk ke dalam populasi untuk membentuk generasi berikutnya yang lebih baik. Dengan demikian harus dilakukan proses seleksi.

Metode seleksi yang digunakan adalah metode *Rank-Based Fitness Selection*. Adapun langkah-langkah dalam metode seleksi ini adalah sebagai berikut :

1. Masukkan semua individu yaitu individu awal, *offspring* dan individu hasil mutasi.
2. Urutkan berdasarkan nilai *fitness* yang tertinggi.
3. Ambil individu berdasarkan urutan *fitness* maksimum sejumlah individu dalam populasi awal untuk digunakan sebagai populasi baru.
4. Gunakan populasi baru untuk perhitungan pada generasi berikutnya sampai ditemukan generasi yang optimum.

Flowchart atau diagram alir untuk proses seleksi akan ditunjukkan dalam Gambar 3.17.



Gambar 3.17 Diagram alir proses Seleksi *Rank-based Fitness*

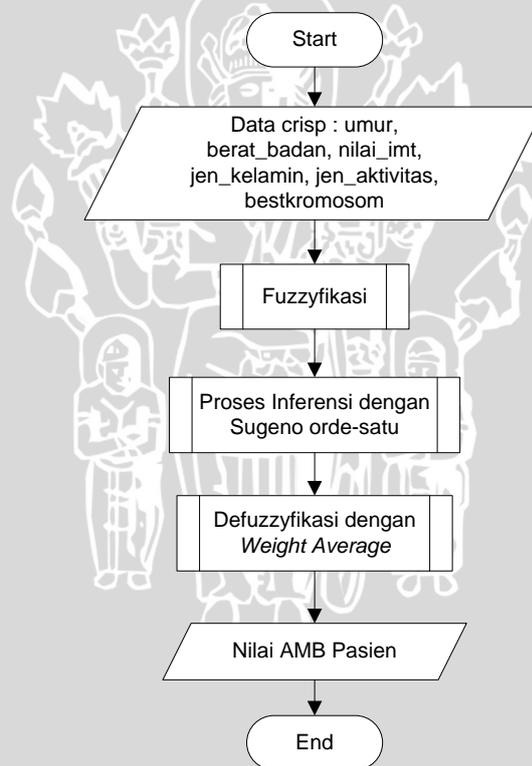
3.5.1.8. Hasil Fungsi Keanggotaan

Apabila jumlah generasi yang dilakukan oleh sistem mencapai batas generasi yang telah ditentukan oleh pengguna, maka individu yang terpilih dengan nilai *fitness* tertinggi akan ditampilkan sebagai hasil akhir dari proses

Algoritma Genetika yang digunakan sebagai batas-batas fungsi keanggotaan dalam proses fuzzyfikasi.

3.5.2. Perancangan Perhitungan AMB dengan Fuzzy Model Sugeno

Apabila proses penentuan batas-batas fungsi keanggotaan dengan menggunakan metode Algoritma Genetika telah selesai dilakukan, maka proses selanjutnya adalah menghitung nilai Angka Metabolisme Basal dengan menggunakan *Fuzzy Inference System Model Sugeno*. Tahapan proses dari metode *Fuzzy Inference System Model Sugeno* adalah Fuzzyfikasi, Inferensi dan yang terakhir Defuzzyfikasi. Diagram alir dari proses perhitungan Angka Metabolisme Basal (AMB) dengan Fuzzy Model Sugeno seperti yang ditunjukkan pada Gambar 3.18.



Gambar 3.18 Diagram alir proses perhitungan AMB dengan Fuzzy Model Sugeno

Hasil dari proses defuzzyfikasi dengan menggunakan perhitungan *Weight Average* (WA) menjadi nilai Angka Metabolisme Basal (AMB) dari

pasien. Berikut tahapan dari perancangan perhitungan AMB dengan *Fuzzy Model Sugeno*.

3.5.2.1. Pembentukan Himpunan Fuzzy

Perhitungan sistem fuzzy pada metode *Fuzzy Model Sugeno* diawali dengan menentukan variabel fungsi keanggotaan *fuzzy*. Parameter fungsi keanggotaan yang digunakan adalah nilai IMT, umur, jenis kelamin dan jenis aktivitas sesuai dengan penelitian yang dilakukan sebelumnya. Rentang nilai domain atau batas-batas fungsi keanggotaan pada parameter nilai IMT dan Umur berasal dari hasil perhitungan Algoritma Genetika yang digunakan sebagai pembentuk domain fungsi keanggotaan. Himpunan fuzzy yang telah dibuat seperti yang ditampilkan pada tabel 3.7.

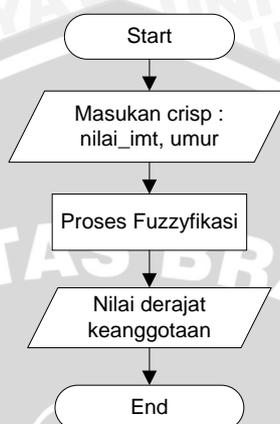
Tabel 3.7 Pembentukan himpunan fuzzy

Variabel	Himpunan Fuzzy		Satuan Domain
	Nama	Notasi	
Nilai IMT	Sangat Kurus	s	Skor
	Kurus	k	
	Normal	n	
	Gemuk	g	
	Obesitas	o	
Umur	Balita	b	Tahun
	Anak-anak	a	
	Remaja	r	
	Dewasa	d	
	Parobaya	p	
	Tua	t	
Jenis Kelamin	Laki-laki	l	Skor
	Perempuan	p	
Jenis Aktivitas	Sangat Ringan	s	Skor
	Ringan	r	
	Sedang	sd	
	Berat	t	

3.5.2.2. Fuzzyfikasi

Setelah pendefinisian himpunan *fuzzy* telah dilakukan, maka selanjutnya memulai proses pengolahan data dengan metode *Fuzzy Inference System Model Sugeno*. Proses perhitungan diawali dengan fuzzyfikasi. Proses dari fuzzyfikasi adalah mengambil masukan nilai crisp dan menentukan

derajat keanggotaan berdasarkan range fungsi keanggotaan. *Range* nilai fungsi keanggotaan (*membership function*) telah terbentuk sebelumnya yang merupakan hasil dari proses Algoritma Genetika. Diagram alir proses *Fuzzyfikasi* dapat dilihat pada Gambar 3.19.



Gambar 3.19 Diagram alir proses *Fuzzyfikasi*

Fungsi Keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (sering juga disebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi. Berikut persamaan fuzzyfikasi pada masing-masing variabel sesuai dengan model kurva fungsi keanggotaanya.

a. Fungsi derajat keanggotaan variabel IMT

- Derajat keanggotaan Sangat Kurus :

$$\mu_s(x) = \begin{cases} 0 & ; x \geq c1 \\ \frac{c1 - x}{c1 - b1} & ; b1 < x < c1 \\ 1 & ; x \leq b1 \end{cases}$$

- Derajat keanggotaan Kurus :

$$\mu_k(x) = \begin{cases} 0; & x \leq a2 \\ \frac{x - a2}{b2 - a2} & ; a2 < x < b2 \\ 1; & b2 \leq x \leq c2 \\ \frac{d2 - x}{d2 - c2} & ; c2 < x < d2 \\ 0; & x \geq d2 \end{cases}$$

- Derajat keanggotaan Normal :

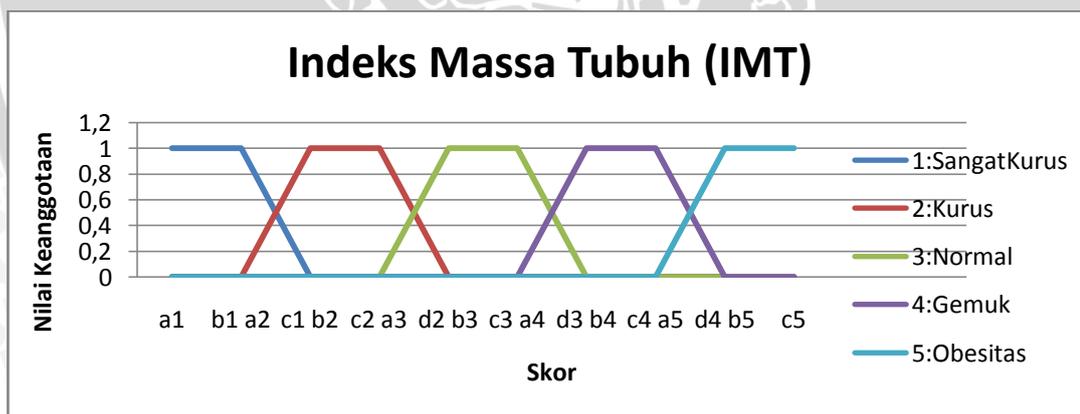
$$\mu_n(x) = \begin{cases} 0; & x \leq a3 \\ \frac{x - a3}{b3 - a3}; & a3 < x < b3 \\ 1; & b3 \leq x \leq c3 \\ \frac{d3 - x}{d3 - c3}; & c3 < x < d3 \\ 0; & x \geq d3 \end{cases}$$

- Derajat keanggotaan Gemuk :

$$\mu_g(x) = \begin{cases} 0; & x \leq a4 \\ \frac{x - a4}{b4 - a4}; & a4 < x < b4 \\ 1; & b4 \leq x \leq c4 \\ \frac{d4 - x}{d4 - c4}; & c4 < x < d4 \\ 0; & x \geq d4 \end{cases}$$

- Derajat keanggotaan Obesitas :

$$\mu_o(x) = \begin{cases} 0 & ; & x \leq a5 \\ \frac{x - a5}{b5 - a5} & ; & a5 < x < b5 \\ 1 & ; & x \geq b5 \end{cases}$$



Gambar 3.20 Grafik fungsi keanggotaan Variabel IMT

- b. Fungsi derajat keanggotaan variabel Umur

- Derajat keanggotaan Balita :

$$\mu_{bl}(x) = \begin{cases} 0 & ; & x \geq c1 \\ \frac{c1 - x}{c1 - b1} & ; & b1 < x < c1 \\ 1 & ; & x \leq b1 \end{cases}$$



- Derajat keanggotaan Anak-anak :

$$\mu_{ak}(x) = \begin{cases} 0; & x \leq a2 \\ \frac{x - a2}{b2 - a2} & ; a2 < x < b2 \\ 1; & b2 \leq x \leq c2 \\ \frac{d2 - x}{d2 - c2} & ; c2 < x < d2 \\ 0; & x \geq d2 \end{cases}$$

- Derajat keanggotaan Remaja :

$$\mu_r(x) = \begin{cases} 0; & x \leq a3 \\ \frac{x - a3}{b3 - a3} & ; a3 < x < b3 \\ 1; & b3 \leq x \leq c3 \\ \frac{d3 - x}{d3 - c3} & ; c3 < x < d3 \\ 0; & x \geq d3 \end{cases}$$

- Derajat keanggotaan Dewasa :

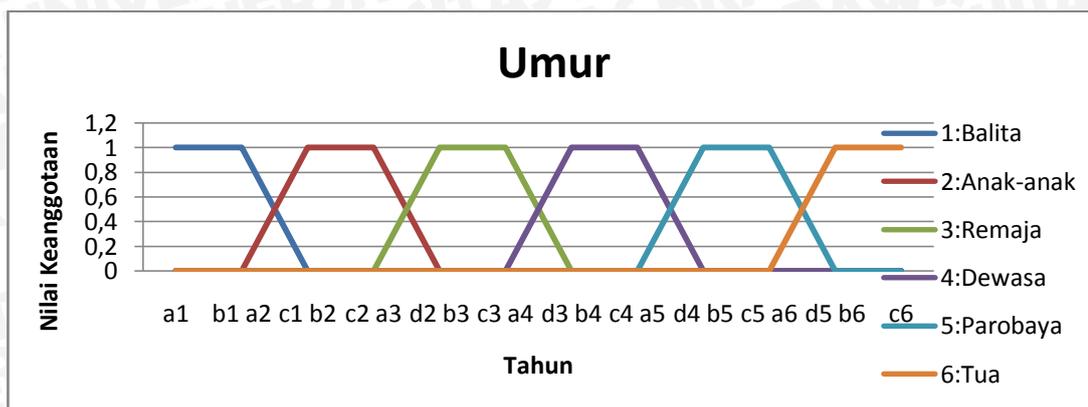
$$\mu_d(x) = \begin{cases} 0; & x \leq a4 \\ \frac{x - a4}{b4 - a4} & ; a4 < x < b4 \\ 1; & b4 \leq x \leq c4 \\ \frac{d4 - x}{d4 - c4} & ; c4 < x < d4 \\ 0; & x \geq d4 \end{cases}$$

- Derajat keanggotaan Parobaya :

$$\mu_p(x) = \begin{cases} 0; & x \leq a5 \\ \frac{x - a5}{b5 - a5} & ; a5 < x < b5 \\ 1; & b5 \leq x \leq c5 \\ \frac{d5 - x}{d5 - c5} & ; c5 < x < d5 \\ 0; & x \geq d5 \end{cases}$$

- Derajat keanggotaan Tua :

$$\mu_o(x) = \begin{cases} 0 & ; x \leq a6 \\ \frac{x - a6}{b6 - a6} & ; a6 < x < b6 \\ 1 & ; x \geq b6 \end{cases}$$



Gambar 3.21 Grafik fungsi keanggotaan Variabel Umur

3.5.2.3. Inferensi

Aturan atau *Rules* adalah pernyataan-pernyataan yang ditulis di dalam penelitian ini menggunakan bentuk aturan *if-then*, sedangkan fungsi implikasi yang digunakan adalah fungsi *min*. Fungsi implikasi *min* bertujuan untuk menentukan nilai α -predikat dari hasil nilai crisp yang dihasilkan oleh tahap fuzzyfikasi dengan mencari nilai yang paling minimum. Pada proses penalarannya menggunakan orde-satu dimana hasil konstanta yang dihasilkan dari suatu persamaan yang telah ditetapkan dari penelitian sebelumnya. Contoh aturan yang akan dibuat seperti berikut:

[R1] IF IMT SANGATKURUS And Umur BALITA And Jen_Kelamin PEREMPUAN And Jen_Aktivitas SANGATRINGAN THEN $((61*bb)-51+500)*1.3$;

Nilai keanggotaan anteseden untuk aturan *fuzzy* [R1] yang dinotasikan dengan α_1 diperoleh dengan rumus sebagai berikut:

$$\alpha_1 = \mu_{\text{IMTSANGATKURUS}} \cap \mu_{\text{UmurBALITA}} \cap \mu_{\text{Jen_Kelamin PEREMPUAN}} \cap \mu_{\text{Jen_AktivitasSANGATRINGAN}}$$

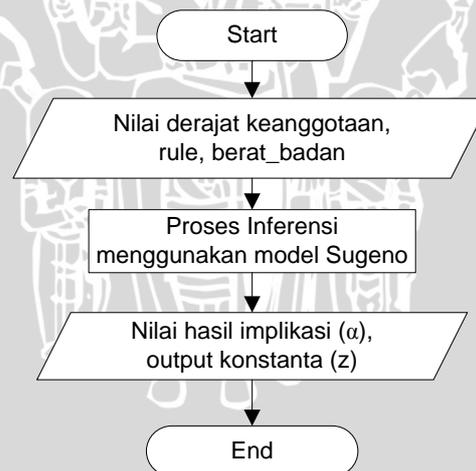
$$\alpha_1 = \min (\mu_{\text{IMTSANGATKURUS}} , \mu_{\text{UmurBALITA}} , \mu_{\text{Jen_Kelamin PEREMPUAN}} , \mu_{\text{Jen_AktivitasSANGATRINGAN}})$$

Tabel 3.8 menunjukkan beberapa aturan yang telah dibuat, untuk aturan selengkapnya akan dimasukkan ke dalam Lampiran 2. Jumlah aturan yang terbentuk sebanyak 240 aturan yang didapat dari penelitian yang telah dilakukan sebelumnya.

Tabel 3.8 Aturan Inferensi Model Sugeno

Rule	IMT	Umur	Jen_Kelamin	Jen_Aktivitas	Perhitungan AMB
1	Sangat Kurus	Balita	Perempuan	Sangat Ringan	$((61*bb)-51+500)*1.3$
2	Sangat Kurus	Balita	Perempuan	Ringan	$((61*bb)-51+500)*1.55$
3	Sangat Kurus	Balita	Perempuan	Sedang	$((61*bb)-51+500)*1.7$
4	Sangat Kurus	Balita	Perempuan	Berat	$((61*bb)-51+500)*2$
5	Sangat Kurus	Anak-anak	Perempuan	Sangat Ringan	$((22.5*bb)+499+500)*1.3$
6	Sangat Kurus	Anak-anak	Perempuan	Ringan	$((22.5*bb)+499+500)*1.55$
7	Sangat Kurus	Anak-anak	Perempuan	Sedang	$((22.5*bb)+499+500)*1.7$
8	Sangat Kurus	Anak-anak	Perempuan	Berat	$((22.5*bb)+499+500)*2$
9	Sangat Kurus	Remaja	Perempuan	Sangat Ringan	$((12.2*bb)+746+500)*1.3$
10	Sangat Kurus	Remaja	Perempuan	Ringan	$((12.2*bb)+746+500)*1.55$
11	Sangat Kurus	Remaja	Perempuan	Sedang	$((12.2*bb)+746+500)*1.7$
12	Sangat Kurus	Remaja	Perempuan	Berat	$((12.2*bb)+746+500)*2$
13	Sangat Kurus	Dewasa	Perempuan	Sangat Ringan	$((14.7*bb)+496+500)*1.3$
14	Sangat Kurus	Dewasa	Perempuan	Ringan	$((14.7*bb)+496+500)*1.55$
15	Sangat Kurus	Dewasa	Perempuan	Sedang	$((14.7*bb)+496+500)*1.7$
16	Sangat Kurus	Dewasa	Perempuan	Berat	$((14.7*bb)+496+500)*2$
17	Sangat Kurus	Parobaya	Perempuan	Sangat Ringan	$((8.7*bb)+829+500)*1.3$
18	Sangat Kurus	Parobaya	Perempuan	Ringan	$((8.7*bb)+829+500)*1.55$
19	Sangat Kurus	Parobaya	Perempuan	Sedang	$((8.7*bb)+829+500)*1.7$
20	Sangat Kurus	Parobaya	Perempuan	Berat	$((8.7*bb)+829+500)*2$

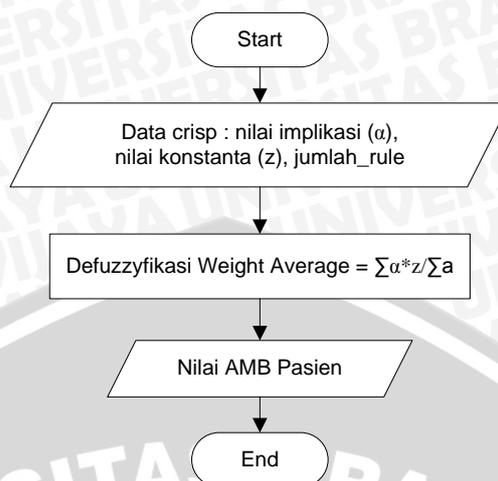
Selanjutnya untuk diagram alir proses Inferensi Model Sugeno dapat dilihat pada Gambar 3.22.



Gambar 3.22 Diagram alir proses Inferensi

3.5.2.4. Defuzzyfikasi

Pada model *Fuzzy Inference System Model Sugeno* ada beberapa cara yang dapat digunakan dalam proses defuzzyfikasi diantaranya adalah dengan menggunakan metode perhitungan *Weight Average (WA)*. Diagram alir proses Defuzzyfikasi dapat dilihat pada Gambar 3.23.



Gambar 3.23 Diagram alir proses defuzzyfikasi

Pada penelitian ini menggunakan perhitungan *weight average*, dimana metode ini melakukan perhitungan dengan membagi jumlah hasil dari α -predikat*Z dengan nilai α -predikat. Dengan demikian maka hasil dari defuzzifikasi akan digunakan sebagai nilai perhitungan Angka Metabolisme Basal (AMB) dari seorang pasien dengan skor dengan satuan domain kkal. Rumus perhitungan defuzzyfikasi seperti yang telah dijelaskan pada persamaan (2-8).

3.6. Perhitungan Manual

Pada subbab ini, menjelaskan secara manual dan alur dari perhitungan dari Sistem Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno. Pada tabel 3.9 tertera contoh permasalahan untuk pembuatan fungsi keanggotaan dan melakukan perhitungan Angka Metabolisme Basal (AMB) dari data pasien dengan parameter masukan sesuai kriteria yang telah ditetapkan.

Tabel 3.9 Data uji perhitungan manual

Kriteria Gejala	Input Penilaian
	Pasien 1
Nama Pasien	Adi
Jenis Kelamin	Laki-laki
Usia	19 Tahun
Berat Badan	78 kg
Tinggi Badan	170 cm
Aktivitas	Memasak
Probabilitas Crossover	0,25
Probabilitas Mutasi	0,1
Jumlah Generasi	1
Jumlah Populasi	4

3.6.1. Pembentukan Fungsi Keanggotaan dengan Algoritma Genetika

Tahap pertama di dalam sistem ini adalah membentuk batas-batas keanggotaan dengan menggunakan metode Algoritma Genetika. Di bawah ini merupakan tahap-tahap di dalam pembentukan fungsi keanggotaan diantaranya adalah inisialisasi populasi awal, menghitung nilai *fitness*, *crossover*, mutasi, evaluasi *fitness*, seleksi dan hasil kromosom akhir.

3.6.1.1. Inisialisasi Populasi Awal

Pada tahap inisialisasi populasi awal, gen pada tiap individu atau kromosom akan dibangkitkan secara acak dengan representasi *floating point* sesuai dengan pada subbab perancangan inisialisasi kromosom. Panjang gen pada kromosom yang mempunyai jenis fungsi keanggotaan linier naik maupun turun mempunyai panjang gen 3, sedangkan untuk kromosom yang mempunyai jenis fungsi keanggotaan trapesium mempunyai panjang gen sebanyak 4. Pada perhitungan manual ini akan dicontohkan pada derajat keanggotaan untuk Balita dan Anak-anak.

- Dibangkitkan 4 Kromosom dari Populasi Balita secara random:

Individu 1 :

0	0,5	3
---	-----	---

Individu 2 :

0	1,5	2,5
---	-----	-----

Individu 3 :

0	2	3,5
---	---	-----

Individu 4 :

0	2,5	3
---	-----	---

- Dibangkitkan 4 Kromosom dari Populasi Anak-anak secara random:

Individu 1 :

3	4	8,5	10
---	---	-----	----

Individu 2 :

2	3,5	9	10,5
---	-----	---	------

Individu 3 :

2,5	3	9,5	10
-----	---	-----	----

Individu 4 :

2	4,5	8	11
---	-----	---	----

3.6.1.2. Menghitung Nilai Fitness

Tahap selanjutnya adalah menghitung nilai *fitness* dari masing-masing kromosom pada tiap populasi. Persamaan yang digunakan untuk melakukan perhitungan *fitness* sesuai dengan yang telah tertera pada subbab perancangan Algoritma Genetika. Berikut merupakan perhitungan fitness dari masing-masing kromosom atau individu.

- Perhitungan Fitness dari populasi **Balita**:

Fitness Individu 1 :

$$\text{Fungsi objektif balita (f1)} = \frac{b1-a1}{2,5-0} = \frac{0,5-0}{2,5-0} = 0,2$$

$$\text{Fungsi objektif balita (f2)} = \frac{c1-b1}{3,5-2,5} = \frac{3-0,5}{3,5-2,5} = 2,5$$

$$\text{Penalti (f2)} = 1 - (f2 - 1) = 1 - (2,5 - 1) = -0,5$$

$$\text{Fungsi objektif balita (f3)} = \frac{c1-a1}{3,5-0} = \frac{3-0}{3,5-0} = 0,85$$

$$\text{Fungsi Fitness Individu 1} = \frac{1}{1+(3-(f1+f2+f3))} = \frac{1}{1+(3-(0,55))} = 0,28$$

Fitness Individu 2 :

$$\text{Fungsi objektif balita (f1)} = \frac{b1-a1}{2,5-0} = \frac{1,5-0}{2,5-0} = 0,6$$

$$\text{Fungsi objektif balita (f2)} = \frac{c1-b1}{3,5-2,5} = \frac{2,5-1,5}{3,5-2,5} = 1$$

$$\text{Fungsi objektif balita (f3)} = \frac{c1-b1}{3,5-2,5} = \frac{2,5-0}{3,5-0} = 0,71$$

$$\text{Fungsi Fitness Individu 2} = \frac{1}{1+(3-(f1+f2+f3))} = \frac{1}{1+(3-(2,31))} = 0,59$$

Fitness Individu 3 :

$$\text{Fungsi objektif balita (f1)} = \frac{b1-a1}{2,5-0} = \frac{2-0}{2,5-0} = 0,8$$

$$\text{Fungsi objektif balita (f2)} = \frac{c1-b1}{3,5-2,5} = \frac{3,5-2}{3,5-2,5} = 1,5$$

$$\text{Penalti (f2)} = 1 - (f2 - 1) = 1 - (1,5 - 1) = 0,5$$

$$\text{Fungsi objektif balita (f3)} = \frac{c1-b1}{3,5-2,5} = \frac{3,5-0}{3,5-0} = 1$$

$$\text{Fungsi Fitness Individu 3} = \frac{1}{1+(3-(f1+f2+f3))} = \frac{1}{1+(3-(2,3))} = 0,58$$

Fitness Individu 4 :

$$\text{Fungsi objektif balita (f1)} = \frac{b1-a1}{2,5-0} = \frac{2,5-0}{2,5-0} = 1$$

$$\text{Fungsi objektif balita (f2)} = \frac{c1-b1}{3,5-2,5} = \frac{3-2,5}{3,5-2,5} = 0,5$$

$$\text{Fungsi objektif balita (f3)} = \frac{c1-b1}{3,5-2,5} = \frac{3-0}{3,5-0} = 0,86$$

$$\text{Fungsi Fitness Individu 4} = \frac{1}{1+(3-(f1+f2+f3))} = \frac{1}{1+(3-(2,31))} = 0,61$$

- Perhitungan Fitness dari populasi **Anak-anak:**

Fitness Individu 1 :

$$\text{Fungsi objektif anak-anak (f1)} = \frac{b2-a2}{3,5-2,5} = \frac{4-3}{3,5-2,5} = 1$$

$$\text{Fungsi objektif anak-anak (f2)} = \frac{c2-b2}{9,5-3,5} = \frac{8,5-4}{9,5-3,5} = 0,75$$

$$\text{Fungsi objektif anak-anak (f3)} = \frac{c2-a2}{9,5-2,5} = \frac{8,5-3}{9,5-2,5} = 0,78$$

$$\text{Fungsi objektif anak-anak (f4)} = \frac{d2-c2}{10,5-9,5} = \frac{10-8,5}{10,5-9,5} = 1,5$$

$$\text{Penalti (f4)} = 1 - (f4 - 1) = 1 - (1,5 - 1) = 0,5$$

$$\text{Fungsi objektif anak-anak (f5)} = \frac{d2-b2}{10,5-3,5} = \frac{10-4}{10,5-3,5} = 0,85$$

$$\text{Fungsi objektif anak-anak (f6)} = \frac{d2-a2}{10,5-2,5} = \frac{10-3}{10,5-2,5} = 0,87$$

$$\text{Fungsi Fitness Individu 1} = \frac{1}{1+(6-(f1+f2+f3+f4+f5+f6))} = \frac{1}{1+(6-(4,75))} = 0,44$$

Fitness Individu 2 :

$$\text{Fungsi objektif anak-anak (f1)} = \frac{b2-a2}{3,5-2,5} = \frac{3,5-2}{3,5-2,5} = 1,5$$

$$\text{Penalti (f1)} = 1 - (f1 - 1) = 1 - (1,5 - 1) = 0,5$$

$$\text{Fungsi objektif anak-anak (f2)} = \frac{c2-b2}{9,5-3,5} = \frac{9-3,5}{9,5-3,5} = 0,91$$

$$\text{Fungsi objektif anak-anak (f3)} = \frac{c2-a2}{9,5-2,5} = \frac{9-2}{9,5-2,5} = 1$$

$$\text{Fungsi objektif anak-anak (f4)} = \frac{d2-c2}{10,5-9,5} = \frac{10,5-9}{10,5-9,5} = 1,5$$

$$\text{Penalti (f4)} = 1 - (f4 - 1) = 1 - (1,5 - 1) = 0,5$$

$$\text{Fungsi objektif anak-anak (f5)} = \frac{d2-b2}{10,5-3,5} = \frac{10,5-3,5}{10,5-3,5} = 1$$

$$\text{Fungsi objektif anak-anak (f6)} = \frac{d2-a2}{10,5-2,5} = \frac{10,5-2}{10,5-2,5} = 1,06$$

$$\text{Penalti (f6)} = 1 - (f6 - 1) = 1 - (1,06 - 1) = 0,94$$

$$\text{Fungsi Fitness Individu 2} = \frac{1}{1+(6-(f1+f2+f3+f4+f5+f6))} = \frac{1}{1+(6-(4,85))} = 0,47$$

Fitness Individu 3 :

$$\text{Fungsi objektif anak-anak (f1)} = \frac{b2-a2}{3,5-2,5} = \frac{3-2,5}{3,5-2,5} = 0,5$$

$$\text{Fungsi objektif anak-anak (f2)} = \frac{c2-b2}{9,5-3,5} = \frac{9,5-3}{9,5-3,5} = 1,08$$

$$\text{Penalti (f2)} = 1 - (f2 - 1) = 1 - (1,08 - 1) = 0,92$$

$$\text{Fungsi objektif anak-anak (f3)} = \frac{c2-a2}{9,5-2,5} = \frac{9,5-2,5}{9,5-2,5} = 1$$

$$\text{Fungsi objektif anak-anak (f4)} = \frac{d2-c2}{10,5-9,5} = \frac{10-9,5}{10,5-9,5} = 0,5$$

$$\text{Fungsi objektif anak-anak (f5)} = \frac{d2-b2}{10,5-3,5} = \frac{10-3}{10,5-3,5} = 1$$

$$\text{Fungsi objektif anak-anak (f6)} = \frac{d2-a2}{10,5-2,5} = \frac{10-2,5}{10,5-2,5} = 0,93$$

$$\text{Fungsi Fitness Individu 3} = \frac{1}{1+(6-(f1+f2+f3+f4+f5+f6))} = \frac{1}{1+(6-(4,85))} = 0,47$$

Fitness Individu 4 :

$$\text{Fungsi objektif anak-anak (f1)} = \frac{b2-a2}{3,5-2,5} = \frac{4,5-2}{3,5-2,5} = 2,5$$

$$\text{Penalti (f2)} = 1 - (f2 - 1) = 1 - (2,5 - 1) = -0,5$$

$$\text{Fungsi objektif anak-anak (f2)} = \frac{c2-b2}{9,5-3,5} = \frac{8-4,5}{9,5-3,5} = 0,58$$

$$\text{Fungsi objektif anak-anak (f3)} = \frac{c2-a2}{9,5-2,5} = \frac{8-2}{9,5-2,5} = 0,85$$

$$\text{Fungsi objektif anak-anak (f4)} = \frac{d2-c2}{10,5-9,5} = \frac{11-9}{10,5-9,5} = 2$$

$$\text{Penalti (f4)} = 1 - (f4 - 1) = 1 - (2 - 1) = 0$$

$$\text{Fungsi objektif anak-anak (f5)} = \frac{d2-b2}{10,5-3,5} = \frac{11-3,5}{10,5-3,5} = 1,07$$

$$\text{Penalti (f5)} = 1 - (f4 - 1) = 1 - (1,07 - 1) = 0,93$$

$$\text{Fungsi objektif anak-anak (f6)} = \frac{d2-a2}{10,5-2,5} = \frac{11-2}{10,5-2,5} = 1,12$$

$$\text{Penalti (f6)} = 1 - (f6 - 1) = 1 - (1,12 - 1) = 0,88$$

$$\text{Fungsi Fitness Individu 4} = \frac{1}{1+(6-(f1+f2+f3+f4+f5+f6))} = \frac{1}{1+(6-(2,74))} = 0,23$$

3.6.1.3. Crossover

Setelah proses perhitungan *fitness* selesai dilakukan, maka proses selanjutnya adalah proses *crossover*. Di bawah ini merupakan proses perhitungan manual dari *crossover*.

Membangkitkan bilangan R secara random dari 0 sampai 1 untuk tiap-tiap kromosom:

- Bilangan R populasi Balita :

$$\mathbf{R[1]} = 0.191$$

$$\mathbf{R[2]} = 0.760$$

$$\mathbf{R[3]} = 0.006$$

$$\mathbf{R[4]} = 0.159$$

- Bilangan R populasi Anak-anak :

$$\mathbf{R[1]} = 0.230$$

$$\mathbf{R[2]} = 0.113$$

$$\mathbf{R[3]} = 0.980$$

$$\mathbf{R[4]} = 0.159$$

Selanjutnya dimisalkan probabilitas *crossover* yang dimasukkan sebesar 0,5, maka kromosom ke k akan dipilih sebagai induk jika $R[k] < p_c$, dari bilangan acak R diatas maka yang dijadikan induk adalah sebagai berikut:

- Populasi Balita :
Kromosom[1], Kromosom[3], Kromosom[4]
- Populasi Anak-anak :
Kromosom[1], Kromosom[2], Kromosom[4]

Setelah kromosom terpilih, selanjutnya dilakukan proses crossover sejumlah persamaan kombinasi yang telah dijelaskan pada subbab perancangan Algoritma Genetika. Dalam contoh ini kromosom yang terpilih sebanyak 3, sehingga persilangan yang dilakukan sebanyak 3 pasang.

- Pasangan crossover populasi Balita :
Kromosom[1] \times Kromosom[3]
Kromosom[1] \times Kromosom[4]
Kromosom[3] \times Kromosom[4]
- Pasangan crossover populasi Anak-anak :
Kromosom[1] \times Kromosom[2]
Kromosom[1] \times Kromosom[4]
Kromosom[2] \times Kromosom[4]

Posisi *cut-point* crossover ditentukan secara random dengan rentang 0 hingga jumlah gen pada satu kromosom, seperti pada *crossover* dengan menggunakan metode *Simple Arithmetic Crossover* yang telah dijelaskan perhitungannya pada subbab perancangan *crossover* maka hasil *crossover* terhadap populasi balita dibawah ini dengan $\alpha = 0,5$ dan *cut-point* = 2:

✓ **Populasi Balita :**

- **Crossover 1** = chromosome[1] \times chromosome[3]
= [0 ; 0,5 ; 3] \times [0 ; 2 ; 3,5]
offspring[1] = [0 ; 0,5 ; 3,25]
offspring[2] = [0 ; 2 ; 3,25]
- **Crossover 2** = chromosome[1] \times chromosome[4]
= [0 ; 0,5 ; 3] \times [0 ; 2,5 ; 3]
offspring[3] = [0 ; 0,5 ; 3]
offspring[4] = [0 ; 2,5 ; 3]

- **Crossover 3** = chromosome[3] \times chromosome[4]
 = [0 ; 2 ; **3,5**] \times [0 ; 2,5 ; **3**]
 offspring[5] = [0 ; 2 ; 3,25]
 offspring[6] = [0 ; 2,5 ; 3,25]

Dengan demikian populasi kromosom dari variabel keanggotaan Balita setelah mengalami proses *crossover* menjadi:

- offspring**[1] = [0 ; 0,5 ; 3,25]
- offspring** [2] = [0 ; 2 ; 3,25]
- offspring** [3] = [0 ; 0,5 ; 3]
- offspring** [4] = [0 ; 2,5 ; 3]
- offspring** [5] = [0 ; 2 ; 3,25]
- offspring** [6] = [0 ; 2,5 ; 3,25]

Pada *crossover* terhadap populasi Anak-anak dijelaskan sebagai berikut :

✓ **Populasi Anak-anak :**

- **Crossover 1** = chromosome[1] \times chromosome[2]
 = [3 ; 4 ; **8,5 ; 10**] \times [2 ; 3,5 ; **9 ; 10,5**]
 offspring[1] = [3 ; 4 ; 8,75 ; 10,25]
 offspring[2] = [2 ; 3,5 ; 8,75 ; 10,25]
- **Crossover 2** = chromosome[1] \times chromosome[4]
 = [3 ; 4 ; **8,5 ; 10**] \times [2 ; 4,5 ; **8 ; 11**]
 offspring[3] = [3 ; 4 ; 8,25 ; 10,5]
 offspring[4] = [2 ; 4,5 ; 8,25 ; 10,5]
- **Crossover 3** = chromosome[2] \times chromosome[4]
 = [2 ; 3,5 ; **9 ; 10,5**] \times [2 ; 4,5 ; **8 ; 11**]
 offspring[5] = [2 ; 3,5 ; 8,5 ; 10,75]
 offspring[6] = [2 ; 4,5 ; 8,5 ; 10,75]

Dengan demikian populasi kromosom dari variabel keanggotaan anak-anak setelah mengalami proses *crossover* menjadi:

- offspring** [1] = [3 ; 4 ; 8,75 ; 10,25]
- offspring** [2] = [2 ; 3,5 ; 8,75 ; 10,25]

offspring [3] = [3 ; 4 ; 8,25 ; 10,5]

offspring [4] = [2 ; 4,5 ; 8,25 ; 10,5]

offspring [5] = [2 ; 3,5 ; 8,5 ; 10,75]

offspring [6] = [2 ; 4,5 ; 8,5 ; 10,75]

3.6.1.4. Mutasi

Setelah *crossover* selesai dilakukan dan didapat *offspring* atau anak hasil dari proses *crossover*, tahap selanjutnya adalah proses mutasi. Di bawah ini merupakan proses perhitungan manual dari mutasi.

Membangkitkan bilangan R secara random dari 0 sampai 1 untuk tiap-tiap kromosom :

- Bilangan R populasi Balita :

$$\mathbf{R}[1] = 0.191$$

$$\mathbf{R}[2] = 0.760$$

$$\mathbf{R}[3] = 0.806$$

$$\mathbf{R}[4] = 0.159$$

$$\mathbf{R}[5] = 0.190$$

$$\mathbf{R}[6] = 0.623$$

- Bilangan R populasi Anak-anak :

$$\mathbf{R}[1] = 0.230$$

$$\mathbf{R}[2] = 0.613$$

$$\mathbf{R}[3] = 0.980$$

$$\mathbf{R}[4] = 0.159$$

$$\mathbf{R}[5] = 0.724$$

$$\mathbf{R}[6] = 0.263$$

Selanjutnya dimisalkan probabilitas mutasi yang dimasukkan sebesar 0,5, maka kromosom ke k akan dipilih sebagai induk jika $\mathbf{R}[k] < p_c$, dari bilangan acak R diatas maka yang dijadikan induk adalah sebagai berikut :

- Populasi Balita :

Kromosom[1], Kromosom[4], Kromosom[5]

- Populasi Anak-anak :

Kromosom[1], Kromosom[4], Kromosom[6]

Misalkan setelah terpilih individu yang dikenai mutasi, selanjutnya membangkitkan bilangan random untuk menentukan posisi gen didalam individu yang mengalami mutasi. Dengan demikian yang akan mengalami mutasi adalah:

- Populasi Balita :

Offspring[1] gen nomor 2

Offspring[4] gen nomor 2

Offspring[5] gen nomor 2

Sehingga gen yang akan mengalami mutasi seperti yang tercetak tebal di bawah ini :

offspring[1] = [0 ; **0,5** ; 3,25]

offspring [2] = [0 ; 2 ; 3,25]

offspring [3] = [0 ; 0,5 ; 3]

offspring [4] = [0 ; **2,5** ; 3]

offspring [5] = [0 ; 2 ; 3,25]

offspring [6] = [0 ; 2,5 ; 3,25]

- Populasi Anak-anak :

Offspring[1] gen nomor 4

Offspring[4] gen nomor 4

Offspring[6] gen nomor 4

Sehingga gen yang akan mengalami mutasi seperti yang tercetak tebal di bawah ini :

offspring [1] = [3 ; 4 ; 8,75 ; **10,25**]

offspring [2] = [2 ; 3,5 ; 8,75 ; 10,25]

offspring [3] = [3 ; 4 ; 8,25 ; 10,5]

offspring [4] = [2 ; 4,5 ; 8,25 ; **10,5**]

offspring [5] = [2 ; 3,5 ; 8,5 ; 10,75]

offspring [6] = [2 ; 4,5 ; 8,5 ; **10,75**]

Langkah terakhir dari proses mutasi adalah mengganti gen-gen yang terpilih dengan nilai yang dibangkitkan secara random dengan rentang nilai

random sesuai pada subbab perancangan Algoritma Genetika. Pada populasi balita bilangan pengganti yang dibangkitkan secara random di dalam rentang 0 – 4, sedangkan pada populasi anak-anak dibangkitkan secara random dalam rentang 2 – 11. Maka populasi kromosom setelah mengalami proses mutasi adalah sebagai berikut:

- Populasi Balita :

Bilangan random yang dibangkitkan adalah 1,0 , 2,5 dan 2,5.

Sehingga gen yang akan mengalami mutasi berganti dengan nilai random, seperti yang tercetak tebal di bawah ini :

offspring [1] = [0 ; **1,0** ; 3,25]

offspring [2] = [0 ; 2 ; 3,25]

offspring [3] = [0 ; 0,5 ; 3]

offspring [4] = [0 ; **2,5** ; 3]

offspring [5] = [0 ; **2,5** ; 3,25]

offspring [6] = [0 ; 2,5 ; 3,25]

- Populasi Anak-anak :

Bilangan random yang dibangkitkan adalah 11,5 , 10,5 dan 10.

Sehingga gen yang akan mengalami mutasi berganti dengan nilai random, seperti yang tercetak tebal di bawah ini :

offspring [1] = [3 ; 4 ; 8,75 ; **11,5**]

offspring [2] = [2 ; 3,5 ; 8,75 ; 10,25]

offspring [3] = [3 ; 4 ; 8,25 ; 10,5]

offspring [4] = [2 ; 4,5 ; 8,25 ; **10,5**]

offspring [5] = [2 ; 3,5 ; 8,5 ; 10,75]

offspring [6] = [2 ; 4,5 ; 8,5 ; **10**]

3.6.1.5. Evaluasi Fitness

Setelah proses mutasi dilanjutkan dengan menghitung fitness kembali dikarenakan nilai *fitness* hasil mutasi akan digunakan ke dalam proses seleksi. Proses perhitungan nilai fitness sama dengan perhitungan fitness awal. Tabel 3.10 merupakan hasil perhitungan fitness pada populasi Balita dan Anak-anak.

- Hasil Perhitungan *Fitness* dari populasi **Balita**:

Tabel 3.10 Hasil fitness populasi balita

No.	Nama Individu	Nilai <i>fitness</i>
1.	Offspring 1	0,342
2.	Offspring 2	0,657
3.	Offspring 3	0,290
4.	Offspring 4	0,608
5.	Offspring 5	0,756
6.	Offspring 6	0,756

- Hasil Perhitungan *Fitness* dari populasi **Anak-anak**:

Tabel 3.11 Hasil fitness populasi anak-anak

No.	Nama Individu	Nilai <i>fitness</i>
1.	Offspring 1	0,305
2.	Offspring 2	0,448
3.	Offspring 3	0,341
4.	Offspring 4	0,225
5.	Offspring 5	0,320
6.	Offspring 6	0,276

3.6.1.6. Seleksi

Setelah nilai fitness tiap individu yang mengalami mutasi selesai dilakukan, maka selanjutnya dilakukan proses seleksi dengan menggunakan metode *Rank-based fitness selection* dimana pemilihan individu untuk generasi selanjutnya maupun hasil akhir berdasarkan nilai *fitness*. Pada awalnya dilakukan pengumpulan seluruh individu, dari proses inisialisasi awal dan *offspring* hasil *crossover* dan mutasi. Di bawah ini merupakan contoh perhitungan seleksi menggunakan metode *Rank-based fitness selection* :

- Proses seleksi pada populasi **Balita**:

Individu Awal :

Tabel 3.12 Individu awal

No.	Nama Individu	Nilai <i>Fitness</i>
1	Individu 1	0,28
2	Individu 2	0,59
3	Individu 3	0,58
4	Individu 4	0,61
5	Offspring 1	0,342
6	Offspring 2	0,657
7	Offspring 3	0,290
8	Offspring 4	0,608
9	Offspring 5	0,756
10	Offspring 6	0,756

Tabel 3.13 merupakan hasil setelah diurutkan berdasarkan nilai *fitness* yang dimulai dari terbesar hingga terkecil :

Tabel 3.13 Individu setelah diurutkan

No.	Nama Individu	Nilai <i>Fitness</i>
1	Offspring 5	0,756
2	Offspring 6	0,756
3	Offspring 2	0,657
4	Individu 4	0,61
5	Offspring 4	0,608
6	Individu 2	0,59
7	Individu 3	0,58
8	Offspring 1	0,342
9	Offspring 3	0,290
10	Individu 1	0,28

Apabila jumlah generasi belum mencapai batas yang telah ditentukan oleh pengguna maka hasil individu dari generasi pertama akan diambil sebanyak jumlah individu awal yaitu sebanyak 4 individu. Individu yang terpilih berdasarkan nilai *fitness* yang terbesar pada populasi balita:

Tabel 3.14 Individu yang terpilih populasi balita

No.	Nama Individu	Nilai <i>Fitness</i>
1	Offspring 5	0,756
2	Offspring 6	0,756
3	Offspring 2	0,657
4	Individu 4	0,61

- Proses seleksi pada populasi **Anak-anak**:

Individu Awal :

Tabel 3.15 Individu awal

No.	Nama Individu	Nilai <i>Fitness</i>
1	Individu 1	0,44
2	Individu 2	0,47
3	Individu 3	0,47
4	Individu 4	0,23
5	Offspring 1	0,305
6	Offspring 2	0,448
7	Offspring 3	0,341
8	Offspring 4	0,225
9	Offspring 5	0,320
10	Offspring 6	0,276

Tabel 3.16 merupakan hasil setelah diurutkan berdasarkan nilai *fitness* yang dimulai dari terbesar hingga terkecil :

Tabel 3.16 Individu setelah diurutkan

No.	Nama Individu	Nilai <i>Fitness</i>
1	Individu 2	0,47
2	Individu 3	0,47
3	Offspring 2	0,448
4	Individu 1	0,44
5	Offspring 3	0,341
6	Offspring 5	0,320
7	Offspring 1	0,305
8	Offspring 6	0,276
9	Individu 4	0,23
10	Offspring 4	0,225

Apabila jumlah generasi belum mencapai batas yang telah ditentukan oleh pengguna maka hasil individu dari generasi pertama akan diambil

sebanyak jumlah individu awal yaitu sebanyak 4 individu. Individu yang terpilih berdasarkan nilai fitness yang terbesar pada populasi anak-anak:

Tabel 3.17 Individu yang terpilih populasi anak-anak

No.	Nama Individu	Nilai <i>Fitness</i>
1	Individu 2	0,47
2	Individu 3	0,47
3	Offspring 2	0,448
4	Individu 1	0,44

3.6.1.7. Hasil Fungsi Keanggotaan

Dalam perhitungan ini dimisalkan jumlah generasi sebanyak 1 kali iterasi, sehingga individu yang terpilih merupakan individu yang menempati posisi pertama dalam proses seleksi yang berarti mempunyai nilai fitness yang terbesar pada masing-masing populasi. Untuk fungsi keanggotaan umur yang lain seperti remaja, dewasa, parobaya dan tua serta fungsi keanggotaan indeks massa tubuh (IMT) perhitungan manualnya mengikuti seperti proses yang dilakukan ini. Sehingga hasil akhir dari keseluruhan individu pada masing-masing fungsi keanggotaan adalah sebagai berikut:

- Fungsi Keanggotaan **Umur** :

Balita

0	2,5	3,25
---	-----	------

Anak-anak

2	3,5	9	10,5
---	-----	---	------

Remaja

9	10,5	17,5	18,5
---	------	------	------

Dewasa

17,5	18,5	25,5	30,5
------	------	------	------

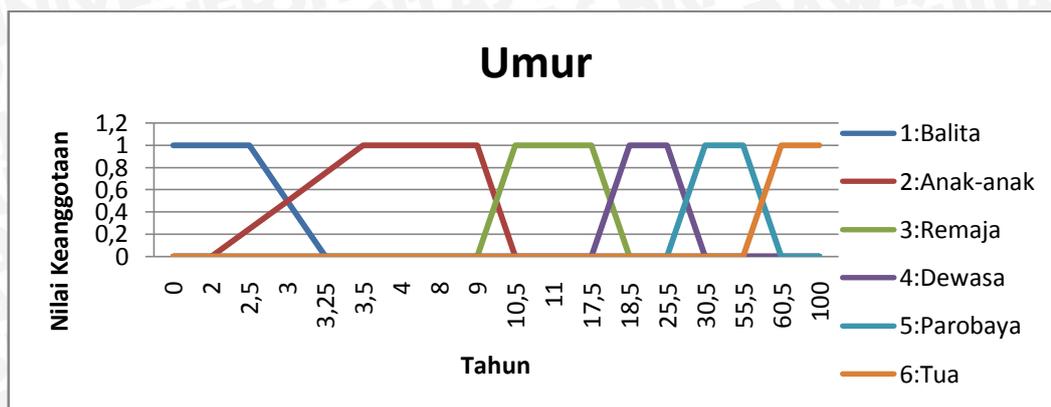
Parobaya

25,5	30,5	55,5	60,5
------	------	------	------

Tua

55,5	60,5	100
------	------	-----

Sehingga diagram fungsi keanggotaan Umur adalah sebagaimana yang ditunjukkan Gambar 3.24 berikut :



Gambar 3.24 Diagram fungsi keanggotaan Umur

- Fungsi Keanggotaan **Indeks Massa Tubuh (IMT)** :

Sangat Kurus

0	16,5	17,5
---	------	------

Kurus

16,5	17,5	18	19
------	------	----	----

Normal

18	19	24,5	25,5
----	----	------	------

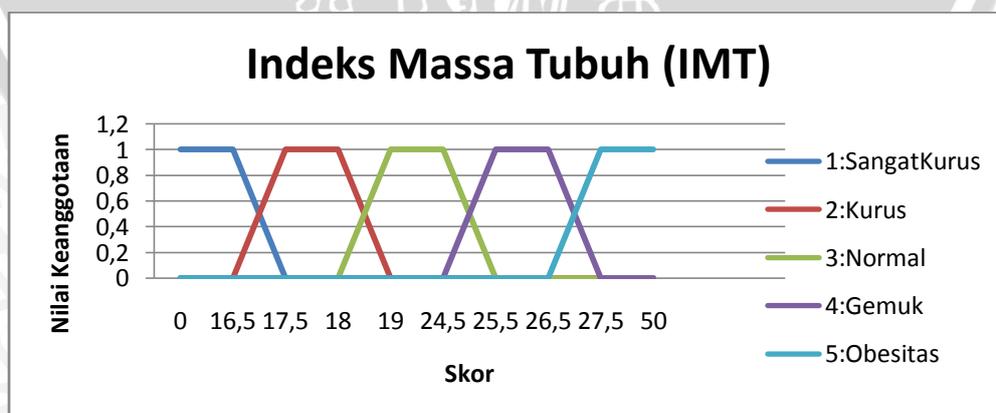
Gemuk

24,5	25,5	26,5	27,5
------	------	------	------

Obesitas

26,5	27,4	50
------	------	----

Sehingga diagram fungsi keanggotaan Umur adalah sebagaimana yang ditunjukkan Gambar 3.25 berikut :



Gambar 3.25 Diagram fungsi keanggotaan IMT

3.6.2. Perhitungan nilai AMB dengan FIS Model Sugeno

Proses penentuan batas-batas fungsi keanggotaan dengan menggunakan metode Algoritma Genetika telah selesai dilakukan, maka proses selanjutnya adalah menghitung nilai Angka Metabolisme Basal dengan menggunakan *Fuzzy Inference System* Model Sugeno. Tahapan proses dari metode *Fuzzy Inference System* Model Sugeno adalah Fuzzyfikasi, Inferensi dan yang terakhir Defuzzyfikasi. Hasil dari defuzzyfikasi inilah yang menjadi nilai Angka Metabolisme Basal (AMB) dari pasien tersebut. Berikut proses perhitungan manual dalam menghitung AMB dengan menggunakan metode *Fuzzy Inference System* Model Sugeno :

3.6.2.1. Fuzzyfikasi

Proses pertama pada metode FIS Model Sugeno adalah Fuzzyfikasi. Fuzzyfikasi digunakan untuk mencari nilai derajat keanggotaan masing-masing variabel dari masukan nilai *crisp* data pasien. Data pasien yang dibutuhkan pada proses fuzzyfikasi adalah umur, nilai imt, berat badan, jenis kelamin dan jenis aktivitas. Berikut perhitungan manual pada proses fuzzyfikasi dengan data masukan sesuai pada tabel 3.9:

1. Fuzzyfikasi Fungsi Keanggotaan Umur

Dari persamaan Fungsi derajat keanggotaan variabel umur, jika masukan nilai umur pasien = 19 maka derajat keanggotaan *fuzzy* pada setiap himpunan sesuai dengan persamaan fungsi keanggotaan pada subbab perancangan fuzzyfikasi adalah:

- a. Himpunan *fuzzy* Balita :

$$\mu_{bl}(x) = 0$$

- b. Himpunan *fuzzy* Anak-anak :

$$\mu_{ak}(x) = 0$$

- c. Himpunan *fuzzy* Remaja :

$$\mu_r(x) = 0$$

- d. Himpunan *fuzzy* Dewasa :

$$\mu_d(x) = 1$$

e. Himpunan *fuzzy* Parobaya :

$$\mu_p(x) = 0$$

f. Himpunan *fuzzy* Tua :

$$\mu_{pl}(x) = 0$$

2. Variabel Indeks Masa Tubuh (IMT)

Dari persamaan Fungsi derajat keanggotaan variabel IMT, jika masukan nilai berat badan = 78 dan tinggi badan = 170 sehingga didapatkan nilai IMT dengan persamaan = $(78/1.7^2) = 26,98$ maka derajat keanggotaan *fuzzy* pada setiap himpunan adalah :

a. Himpunan *fuzzy* Sangat Kurus :

$$\mu_s(x) = 0$$

b. Himpunan *fuzzy* Kurus :

$$\mu_k(x) = 0$$

c. Himpunan *fuzzy* Normal :

$$\mu_n(x) = 0$$

d. Himpunan *fuzzy* Gemuk :

$$\mu_g(x) = \frac{27,5 - x}{b - a}$$

$$\mu_g(x) = \frac{27,5 - 26,98}{27,5 - 26,5} = 0,52$$

e. Himpunan *fuzzy* Obesitas :

$$\mu_o(x) = \frac{x - 26,5}{27,5 - 26,5}$$

$$\mu_o(x) = \frac{26,98 - 26,5}{27,5 - 26,5} = 0,48$$

3. Variabel Jenis Kelamin

Jika pasien mempunyai jenis kelamin laki-laki, maka nilai keanggotaan *fuzzy* pada setiap himpunan adalah :

a. Himpunan *fuzzy* Laki-laki :

$$\mu_{lk}(x) = 1$$

b. Himpunan *fuzzy* Perempuan :

$$\mu_{pr}(x) = 0$$

4. Variabel Jenis Aktivitas

Jika pasien mempunyai aktivitas memasak maka digolongkan kedalam kategori aktivitas ringan, maka nilai keanggotaan *fuzzy* pada setiap himpunan adalah :

- a. Himpunan *fuzzy* Sangat Ringan :

$$\mu_{lk}(x) = 0$$

- b. Himpunan *fuzzy* Ringan :

$$\mu_{lk}(x) = 1$$

- c. Himpunan *fuzzy* Sedang :

$$\mu_{lk}(x) = 0$$

- d. Himpunan *fuzzy* Berat :

$$\mu_{lk}(x) = 0$$

3.6.2.2. Inferensi

Langkah selanjutnya adalah mencari nilai aturan atau yang disebut dengan Inferensi, dimana dalam penelitian ini metode inferensi yang digunakan adalah menggunakan metode Sugeno dengan konstanta orde satu. Dengan menggunakan fungsi min dalam menentukan α -predikat. Tahap inferensi dilakukan dengan mencocokkan hasil dari nilai derajat keanggotaan dengan tabel aturan yang telah dibuat. Selanjutnya pada setiap baris dalam aturan akan dicari nilai yang paling minimum diantara nilai derajat keanggotaan yang telah dimasukkan. Hasil dari nilai minimum akan disimpan dalam variabel Z yang akan digunakan dalam proses defuzzyfikasi. Selanjutnya adalah menghitung aturan α -predikat yang telah dibentuk pada subbab perancangan tabel aturan. Hasil α -predikat juga akan digunakan dalam proses defuzzyfikasi. Berikut perhitungan inferensinya:

Tabel 3.18 Hasil perhitungan manual inferensi

Rule	IMT	Umur	Jen_Kelamin	Jen_Aktivitas	α -predikat (Min)	Nilai Z	α -predikat * Z
180	0.52	0	1	0	0	(((17.5*78)+651-500)*2.1)	0
181	0.52	1	1	0	0	(((15.3*78)+679-500)*1.3)	0
182	0.52	1	1	1	0.52	(((15.3*78)+679-500)*1.65)	1177.52
183	0.52	1	1	0	0	(((15.3*78)+679-500)*1.76)	0
184	0.52	1	1	0	0	(((15.3*78)+679-500)*2.1)	0
185	0.52	0	1	0	0	(((11.6*78)+879-500)*1.3)	0
186	0.52	0	1	1	0	(((11.6*78)+879-500)*1.65)	0
187	0.52	0	1	0	0	(((11.6*78)+879-500)*1.76)	0
188	0.52	0	1	0	0	(((11.6*78)+879-500)*2.1)	0
189	0.52	0	1	0	0	(((13.5*78)+487-500)*1.3)	0
190	0.52	0	1	1	0	(((13.5*78)+487-500)*1.65)	0
220	0.48	0	1	0	0	(((60.9*78)-54-500)*2.1)	0
221	0.48	0	1	0	0	(((22.7*78)+495-500)*1.3)	0
222	0.48	0	1	1	0	(((22.7*78)+495-500)*1.65)	0
223	0.48	0	1	0	0	(((22.7*78)+495-500)*1.76)	0
224	0.48	0	1	0	0	(((22.7*78)+495-500)*2.1)	0
225	0.48	0	1	0	0	(((17.5*78)+651-500)*1.3)	0
226	0.48	0	1	1	0	(((17.5*78)+651-500)*1.65)	0
227	0.48	0	1	0	0	(((17.5*78)+651-500)*1.76)	0
228	0.48	0	1	0	0	(((17.5*78)+651-500)*2.1)	0
229	0.48	1	1	0	0	(((15.3*78)+679-500)*1.3)	0
230	0.48	1	1	1	0.48	(((15.3*78)+679-500)*1.65)	1086.94
$\Sigma \alpha$ -predikat					1	$\Sigma \alpha$ -predikat * Z	2264.46

3.6.2.3. Defuzzyfikasi

Pada model *Fuzzy Inference System* Sugeno ada beberapa cara yang dapat digunakan dalam proses defuzzyfikasi diantaranya adalah dengan menggunakan perhitungan *Weight Average* (WA).

Pada penelitian ini peneliti menggunakan perhitungan *Weight Average*, dengan membagi jumlah hasil dari α -predikat*Z dengan jumlah nilai α -predikat yang merupakan hasil proses inferensi, dimana untuk $\Sigma \alpha$ -predikat = 1 dan $\Sigma \alpha$ -predikat*Z = 2264,46.

Dengan demikian maka hasil dari defuzzyfikasi akan digunakan sebagai nilai akhir dari Angka Metabolisme Basal (AMB) pada pasien. Perhitungan defuzzyfikasi sebagaimana dijelaskan dibawah ini.

$$\text{Weighted Average (WA)} = \frac{a_1 * Z_1 + a_2 * Z_2 + a_3 * Z_3 + \dots + a_n * Z_n}{a_1 + a_2 + a_3 + \dots + a_n}$$

$$WA = \frac{\Sigma a * Z}{\Sigma a}$$

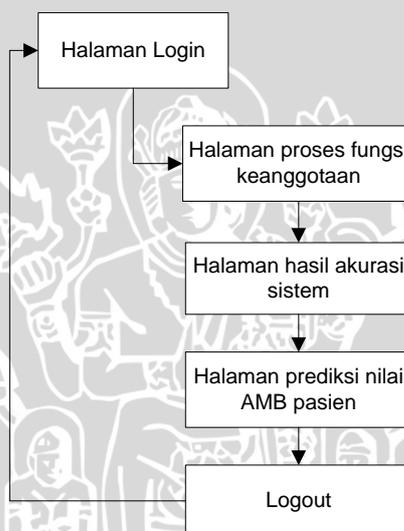
$$WA = \frac{2264,46}{1}$$

$$WA = 2264,46$$

Sesuai dengan nilai defuzzyfikasi yang dihasilkan, maka pasien yang bersangkutan mendapat nilai Angka Metabolisme Basal (AMB) sebesar **2264,46 kkal**.

3.7. Perancangan Antarmuka Pengguna

Subsistem antarmuka pengguna digunakan user untuk berkomunikasi dan memberi perintah kepada sistem yang sedang di operasikan. Perancangan antarmuka pengguna sistem ini dijelaskan pada alur *sitemap* dan desain antarmuka tiap-tiap halaman aplikasi java. *Sitemap* Sistem Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno ditunjukkan pada gambar 3.26 yang terdiri dari halaman Login, halaman proses fungsi keanggotaan, halaman hasil akurasi sistem dan halaman prediksi nilai AMB pasien.



Gambar 3.26 *Sitemap* antarmuka sistem

Rancangan halaman yang dibuat digunakan sebagai acuan dalam membuat halaman antarmuka aplikasi yang di bangun dengan menggunakan bahasa pemrograman *Java*. Berikut ini adalah rancangan antarmuka aplikasi Sistem Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno.

3.7.1. Interface Login

Gambar 3.27 Rancangan *interface* halaman *Login*

Gambar 3.27 merupakan gambar perancangan halaman login. Terdapat field user id dan password yang harus diisi oleh user. Setelah mengisi field maka klik tombol login untuk mengautentifikasi apakah isi dari field yang telah diisi sudah sesuai dengan data yang ada dalam database. Terdapat menu bar *Help* yang berfungsi menampilkan halaman *help* yang menunjukkan panduan dari halaman antarmuka yang sedang di jalankan.

3.7.2. Interface Proses Fungsi Keanggotaan

Gambar 3.28 Rancangan *interface* proses fungsi keanggotaan

Setelah *user* melalui tahap login maka selanjutnya seperti yang ditampilkan pada gambar 3.28, merupakan gambar perancangan interface halaman Input Data Algoritma Genetika. *User* harus memasukkan nilai probabilitas *crossover*, nilai probabilitas mutasi, jumlah generasi dan jumlah populasi yang akan digunakan oleh sistem untuk melakukan proses pembuatan fungsi keanggotaan dengan Metode Algoritma Genetika. Terdapat dua tombol dalam halaman ini yaitu tombol proses digunakan untuk menghasilkan batas-batas fungsi keanggotaan yang akan tampil pada kolom bawah. Tombol clear digunakan untuk menghapus nilai pada form. Terdapat menu bar Help yang berfungsi menampilkan halaman help yang menunjukkan panduan dari halaman antarmuka yang sedang di jalankan.

3.7.3. Interface Hasil Akurasi Sistem

HASIL AKURASI SISTEM				
Help				X
Akurasi Sistem				
Tabel Hasil Akurasi Sistem				
Kategori	<input type="text" value="Search"/>	<input type="button" value="Hitung Manual"/>	<input type="button" value="Hitung Sistem"/>	
Cek Akurasi	<input type="text"/>	Hasil Akurasi	<input type="text"/>	%

Gambar 3.29 Rancangan *interface* akurasi sistem

Gambar 3.29 merupakan gambar perancangan interface halaman hasil akurasi sistem, yang merupakan hasil perhitungan yang dihasilkan oleh Algoritma Genetika dan *Fuzzy Inference System Model Sugeno*. Dalam halaman ini ditampilkan tabel data uji pasien dengan perhitungan manual dan perhitungan yang dilakukan oleh sistem,

kemudian diproses seberapa besar prosesntase akurasi nilai AMB yang dihasilkan oleh sistem terhadap perhitungan manual. Terdapat menu bar *Help* yang berfungsi menampilkan halaman *help* yang menunjukkan panduan dari halaman antarmuka yang sedang di jalankan.

3.7.4. Interface Prediksi AMB Pasien

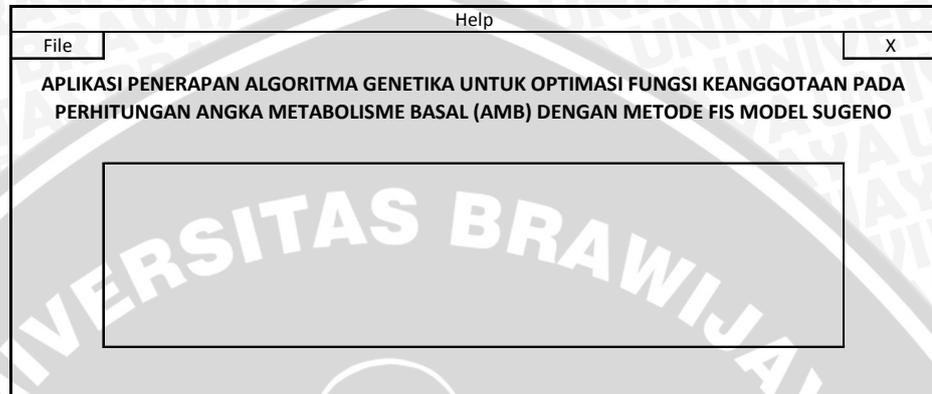
Input Data			
Help			X
INPUT DATA PERHITUNGAN AMB			
Jenis Aktivitas			
Sangat Ringan	Ringan	Sedang	Berat
Akt 1	Akt 1	Akt 1	Akt 1
Akt 2	Akt 2	Akt 2	Akt 2
Akt 3	Akt 3	Akt 3	Akt 3
Akt 4	Akt 4	Akt 4	Akt 4
Input Data Pasien			
Nama	<input type="text"/>	Berat Badan	<input type="text"/>
Umur	<input type="text"/>	Tinggi Badan	<input type="text"/>
Jenis Kelamin	<input type="text"/>	Jenis Aktivitas	<input type="text"/>
Hasil Manual	<input type="text"/>		
Hasil Sistem	<input type="text"/>		
<input type="button" value="Clear"/> <input type="button" value="Proses"/> <input type="button" value="Log Out"/>			

Gambar 3.30 Rancangan *interface* prediksi AMB Pasien

Setelah pengguna selesai pada proses pembuatan batas-batas fungsi keanggotaan dengan Algoritma Genetika, maka selanjutnya seperti yang ditampilkan pada gambar 3.30, merupakan gambar perancangan *interface* halaman prediksi AMB pasien. Halaman ini terdapat *field* data pasien yang harus diisi oleh *user* yang meliputi nama, berat badan, tinggi badan, jenis kelamin, usia, dan aktivitas yang dilakukan oleh pasien. Terdapat tombol dalam halaman ini yaitu tombol proses sistem dimana hasil yang ditampilkan adalah nilai AMB dari perhitungan sistem dengan metode *Fuzzy Model Sugeno* dengan optimasi fungsi keanggotaan yang dilakukan oleh Algoritma Genetika.

Terdapat menu bar Help yang berfungsi menampilkan halaman help yang menunjukkan panduan dari halaman antarmuka yang sedang di jalankan.

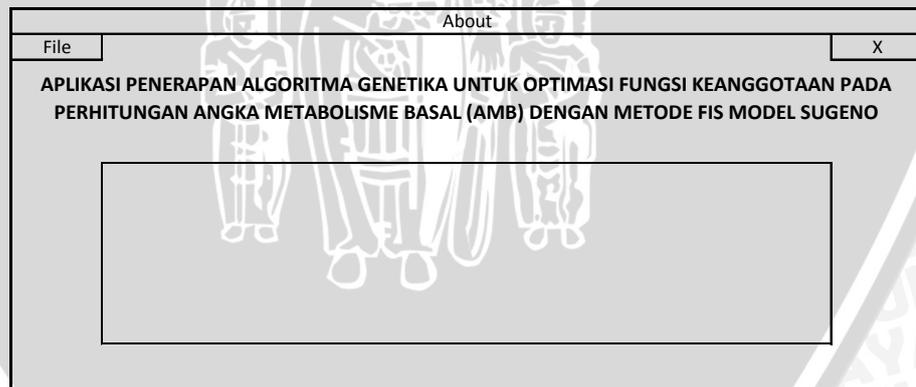
3.7.5. Interface Help



Gambar 3.31 Rancangan *interface help*

Pada gambar 3.31 merupakan gambar perancangan antarmuka halaman *help* yang berisi panduan cara menggunakan atau mengoperasikan halaman.

3.7.6. Interface About



Gambar 3.32 Rancangan *interface about*

Pada gambar 3.32 merupakan gambar perancangan *interface* halaman *About* yang berisi informasi tentang aplikasi Sistem Penerapan Algoritma Genetika untuk Optimasi Fungsi Keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno.

3.8. Mekanisme Pengujian

Pengujian perangkat lunak pada penelitian ini dilakukan agar dapat menunjukkan bahwa perangkat lunak telah mampu bekerja sesuai dengan spesifikasi dari kebutuhan yang melandasinya. Serta melakukan analisis terhadap hasil perhitungan sistem yang menggunakan metode Algoritma Genetika untuk optimasi fungsi keanggotaan dan menemukan nilai probabilitas *crossover* probabilitas mutasi, jumlah generasi dan jumlah populasi yang menghasilkan *fitness* terbesar. Pengujian yang dilakukan meliputi:

- Pengujian parameter Algoritma Genetika meliputi pengujian nilai probabilitas *crossover* dan mutasi, pengujian jumlah generasi dan pengujian ukuran populasi, pengujian yang dilakukan adalah membandingkan perubahan nilai parameter terhadap nilai *fitness* yang dihasilkan. Berikut merupakan hasil pengujian parameter Algoritma Genetika.

1. Pengujian yang pertama adalah mengetahui pengaruh perubahan probabilitas *crossover* dan probabilitas mutasi terhadap nilai *fitness* yang dihasilkan serta mengetahui berapa besar nilai peluang *crossover* dan peluang mutasi terbaik. Dalam pengujian ini digunakan parameter jumlah populasi 25, jumlah generasi 25, dan nilai *alpha* 0.5. Selanjutnya, dalam proses pengujian akan dilakukan terhadap kombinasi dari 9 nilai probabilitas *crossover* dan 9 nilai probabilitas mutasi dengan nilai antara 0.1 – 0.9. Dalam setiap kali kombinasi, akan dilakukan 5 kali percobaan lalu diambil nilai rata-rata *fitness*nya.
2. Pengujian yang kedua adalah mengetahui pengaruh perubahan jumlah generasi terhadap besar nilai *fitness* maksimum yang dihasilkan. Dalam pengujian ini, digunakan parameter nilai P_c dan P_m dengan nilai *fitness* terbesar sesuai pengujian pertama, *alpha* 0.5 dan jumlah populasi 25. Proses pengujian dilakukan dengan mengubah nilai generasi 10, 20, 30, 40, 50, 75, 100 dan 150 generasi. Pada tiap pengujian satu jumlah generasi, dilakukan 5

kali percobaan lalu diambil nilai rata-rata *fitness* yang kemudian akan dibandingkan dengan setiap nilai generasi yang lain untuk dianalisa.

3. Pengujian yang ketiga adalah mengetahui pengaruh perubahan ukuran populasi terhadap besar nilai *fitness* maksimum yang dihasilkan. Dalam pengujian ini, digunakan parameter nilai P_c dan P_m dengan *fitness* terbesar sesuai pada pengujian pertama, α 0.5, dan jumlah generasi yang menghasilkan *fitness* terbesar sesuai dengan pengujian kedua. Proses pengujian dilakukan dengan mengubah ukuran populasi 10, 20, 30, 40 dan 50 populasi. Pada tiap pengujian populasi, dilakukan 5 kali percobaan lalu diambil nilai rata-rata *fitness* yang kemudian akan dibandingkan dengan setiap nilai *fitness* populasi yang lain untuk dianalisa.
- Pengujian data latih dilakukan dengan menggunakan parameter fungsi keanggotaan yang mendapatkan nilai *fitness* yang terbaik dan diujikan terhadap 40 orang yang merupakan data latih yang didapatkan dari data pada penelitian sebelumnya. Pengujian ini dilakukan dengan membangkitkan batas-batas keanggotaan sebanyak 5 kali percobaan dan diambil batas-batas fungsi keanggotaan yang mempunyai hasil akurasi paling tinggi untuk dibuat sebagai fungsi keanggotaan yang dikenakan pada pengujian akurasi data uji.
 - Pengujian akurasi data dilakukan dengan menggunakan fungsi keanggotaan yang mempunyai nilai akurasi yang paling tinggi pada pengujian data latih dan diujikan terhadap 100 orang atau pasien objek uji yang didapatkan dari data pada penelitian sebelumnya. Pengujian ini berfungsi untuk memastikan sistem dapat melakukan perhitungan dengan benar maka akan dibandingkan hasil perhitungan manual AMB menggunakan rumus FAO/WHO/UNU dengan hasil perhitungan dari sistem menggunakan Algoritma Genetika dan *Fuzzy Inference System* (FIS) Model Sugeno. Rumus perhitungan akurasi perhitungan yang dilakukan oleh sistem terhadap perhitungan manual adalah sebagai berikut :

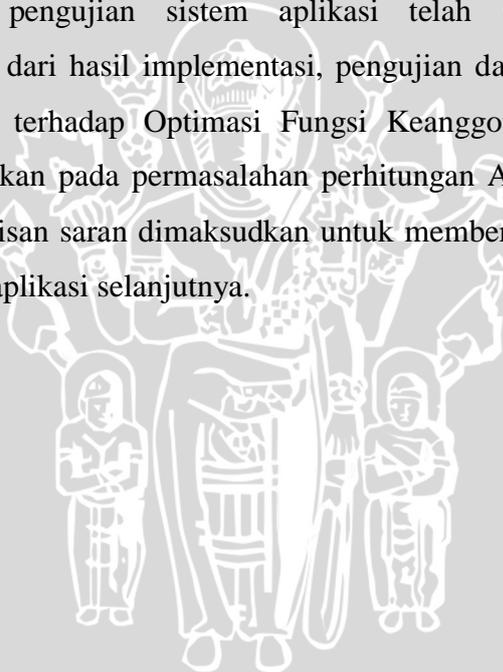
$$\text{Nilai akurasi} = \frac{\text{Jumlah data akurat}}{\text{jumlah h seluruh data}} \times 100\% \quad (3-3)$$

- Pengujian dengan metode *Functional Testing*, dilakukan dengan menguji segala fitur maupun tombol pada antarmuka aplikasi apakah telah berjalan sesuai yang telah direncanakan. Rumus perhitungan akurasi kesesuaian aplikasi terhadap skenario yang dijalankan adalah sebagai berikut :

$$\text{Fungsionalitas} = \frac{\text{jumlah h tindakan yang dilakukan}}{\text{jumlah h tindakan dalam daftar kebutuhan}} \times 100\% \quad (3-4)$$

3.9. Penarikan Kesimpulan

Pembuatan kesimpulan dilakukan setelah semua perancangan, implementasi dan pengujian sistem aplikasi telah selesai dilakukan. Kesimpulan diambil dari hasil implementasi, pengujian dan analisis terhadap Algoritma Genetika terhadap Optimasi Fungsi Keanggotaan *Fuzzy Model Sugeno* yang digunakan pada permasalahan perhitungan Angka Metabolisme Basal (AMB). Penulisan saran dimaksudkan untuk memberikan pertimbangan atas pengembangan aplikasi selanjutnya.



BAB IV

IMPLEMENTASI SISTEM

Bab ini membahas mengenai implementasi perangkat lunak berdasarkan hasil yang telah diperoleh dari analisis kebutuhan dan proses perancangan perangkat lunak yang telah dibuat. Pembahasan terdiri dari penjelasan tentang implementasi spesifikasi sistem dan implementasi program Sistem Penerapan Algoritma Genetika untuk Optimasi Fungsi Keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno.

4.1. Spesifikasi Sistem

Hasil analisis kebutuhan dan perancangan perangkat lunak yang telah diuraikan pada Bab 3 menjadi acuan untuk melakukan implementasi menjadi sistem yang dapat berfungsi sesuai dengan kebutuhan. Spesifikasi sistem diimplementasikan pada spesifikasi perangkat keras dan perangkat lunak.

4.1.1. Spesifikasi Perangkat Keras

Pengembangan Aplikasi Sistem Penerapan Algoritma Genetika untuk Optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno menggunakan sebuah komputer dengan minimal spesifikasi perangkat keras yang dijelaskan pada tabel 4.1.

Tabel 4.1 Spesifikasi perangkat keras

Komponen	Spesifikasi
Processor	Intel(R) Core i5
Memori (RAM)	2 GB RAM
Kartu Grafis	Nvidia 1 GB
Harddisk	500 GB HDD
Layar LCD Monitor	14" WXGA LCD

4.1.2. Spesifikasi Perangkat Lunak

Pengembangan Aplikasi Sistem Penerapan Algoritma Genetika untuk Optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Metode *Fuzzy Inference System* Model Sugeno, menggunakan

sebuah laptop dengan minimal spesifikasi perangkat lunak yang dijelaskan pada tabel 4.2.

Tabel 4.2 Spesifikasi perangkat lunak

Sistem Operasi	Microsoft Windows 8 (64 bit)
Bahasa Pemrograman	Java
Tools Pemrograman	Netbeans IDE 7.0.1

4.2. Implementasi Program

Berdasarkan perancangan perangkat lunak yang telah diuraikan pada bab 3, maka akan dibahas mengenai implementasi program sesuai dengan perancangan yang telah dibangun. Sistem diimplementasikan kedalam pemrograman *Java* berbasis *Desktop Application*. Selanjutnya tiap-tiap proses dilakukan oleh kelas-kelas sesuai dengan fungsi dan pembagiannya. Kelas-kelas yang dibangun akan dijelaskan dalam tabel 4.3.

Tabel 4.3 Kelas-kelas yang dibangun

No	Nama Kelas	Keterangan
1.	Individu.java	Kelas ini berisi method-method yang digunakan sebagai inisialisasi pembentuk individu dan melakukan penghitungan fitness.
2.	JAlgen.java	Kelas ini berisi method-method yang digunakan untuk membentuk populasi, melakukan proses <i>crossover</i> , mutasi dan seleksi.
3.	Aturan.java	Kelas ini berisi method-method yang digunakan sebagai inferensi model Sugeno, dan proses defuzzyfikasi.
4.	FuzzyfikasiUmur.java	Kelas ini berfungsi melakukan proses penghitungan fuzzyfikasi untuk variabel umur.
5.	FuzzyfikasiIMT.java	Kelas ini berfungsi melakukan proses penghitungan fuzzyfikasi untuk variabel IMT.
6.	Aplikasi.java	Kelas ini bertipe GUI dimana pada bagian pertama berfungsi untuk menampilkan form yang berupa masukan paramater Algoritma Genetika dan menampilkan hasil fungsi keanggotaan yang terbentuk. Pada bagian kedua berfungsi untuk menampilkan form yang berupa masukan parameter untuk perhitungan AMB. Pada bagian terakhir berfungsi untuk menampilkan hasil akurasi data perhitungan sistem terhadap perhitungan manual.

4.2.1. Inisialisasi Populasi Awal

Pada proses inisialisasi populasi awal dimulai dengan membangkitkan kromosom tiap individu secara random yang dilakukan oleh sistem seperti pada baris kode ke 8-10, 16-19, 25-27 dan seterusnya. Setelah beberapa individu dibangkitkan maka selanjutnya individu disimpan kedalam populasi seperti pada baris ke 12, 20, 29 dan 37. Proses Inisialisasi individu ini dapat dilihat pada *Source code 4.1* berikut:

```

public ArrayList<Individu> initPopulasi(int jumlah, String jen_fk,
String jen_variabel)
1. int sizeLinear = 3;
2. int sizeTrapeسيوم = 4;
3.
4. for(int i = 0 ; i < jumlah; i++){
5. ArrayList<Double> kromosom = new ArrayList<Double>();
6. if(jen_fk == "Umur"){
7. if(jen_variabel == "Balita"){
8. for (int j=0; j<sizeLinear; j++){
9. kromosom.add(Math.round((Math.random()*4)*Math.pow(10,1))/Math.
10. pow(10,1));
11. }
12. populasi.add(new Individu(kromosom, jen_fk, jen_variabel));
13. }
14.
15. else if(jen_variabel == "Anak-anak"){
16. for (int j=0; j<sizeTrapeسيوم; j++){
17. kromosom.add(Math.round((Math.random()*(11-
18. 2))+2)*Math.pow(10,1))/Math. Pow(10,1));
19. }
20. populasi.add(new Individu(kromosom, jen_fk, jen_variabel));
21. }
22.
23. if(jen_fk == "IMT"){
24. if(jen_variabel == "Sangat Kurus"){
25. for (int j=0; j<sizeLinear; j++){
26. kromosom.add(Math.round((Math.random()*19)* Math.pow (10,1))/
27. Math.pow (10,1));
28. }
29. populasi.add(new Individu(kromosom, jen_fk, jen_variabel));
30. }
31.
32. else if(jen_variabel == "Kurus"){
33. for (int j=0; j<sizeTrapeسيوم; j++){
34. kromosom.add(Math.round(((Math.random()*(20-15))+15) *
35. Math.pow(10,1))/Math.pow(10,1));
36. }
37. populasi.add(new Individu(kromosom, jen_fk, jen_variabel));
38. }
39.
40. return populasi;

```

Source Code 4.1 Inisialisasi Populasi Awal

4.2.2. Perhitungan *Fitness*

Proses perhitungan *fitness* dilakukan dengan cara memasukkan persamaan *fitness* yang telah ditentukan pada bab perancangan kepada setiap individu yang telah dibangkitkan. Kemudian nilai *fitness* akan digunakan untuk menentukan baik atau tidaknya individu tersebut. Perhitungan *fitness* dilakukan dengan memasukkan persamaan yang telah dirancang pada bab 3 terhadap gen individu yang terbentuk seperti yang tertera pada potongan kode pada baris 3-21, 22-52 dan selanjutnya sama pada tiap jenis variabel. Proses perhitungan *fitness* dapat dilihat dalam *Source code* 4.2:

```

public double fitness()
1. double obj1, obj2, obj3, obj4, obj5, obj6;
2. if(jen_fk == "Umur"){
3.   if(jen_variabel == "Balita"){
4.     obj1 = ((individu.get(2) - individu.get(0)) / (3.5-0.0));
5.     if(obj1 >= 1){
6.       obj1 = 1 - (obj1 - 1);
7.     }
8.     obj2 = ((individu.get(1) - individu.get(0)) / (2.5-0.0));
9.     if(obj2 >= 1){
10.      obj2 = 1 - (obj2 - 1);
11.    }
12.    obj3 = ((individu.get(2) - individu.get(1)) / (3.5-2.5));
13.    if(obj3 >= 1){
14.      obj3= 1 - (obj3 - 1);
15.    }
16.    fitness = obj1 + obj2 + obj3;
17.    if (fitness >= 3){
18.      fitness = 3-(fitness-3);
19.    }
20.    fitness = 1/(1+(3-fitness));
21.  }
22. else if(jen_variabel == "Anak-anak") {
23.   obj1 = ((individu.get(1) - individu.get(0)) / (3.5-2.5));
24.   if(obj1 >= 1){
25.     obj1 = 1 - (obj1 - 1);
26.   }
27.   obj2 = ((individu.get(2) - individu.get(1)) / (9.5-3.5));
28.   if(obj2 >= 1){
29.     obj2 = 1 - (obj2 - 1);
30.   }
31.   obj3 = ((individu.get(2) - individu.get(0)) / (9.5-2.5));
32.   if(obj3 >= 1){
33.     obj3= 1 - (obj3 - 1);
34.   }
35.   obj4 = ((individu.get(3) - individu.get(2)) / (10.5-9.5));
36.   if(obj4 >= 1){
37.     obj4= 1 - (obj4 - 1);
38.   }
39.   obj5 = ((individu.get(3) - individu.get(1)) / (10.5-3.5));
40.   if(obj5 >= 1){
41.     obj5= 1 - (obj5 - 1);

```

```

42.     }
43.     obj6 = ((individu.get(3) - individu.get(0)) / (10.5-2.5));
44.     if(obj6 >= 1){
45.         obj6= 1 - (obj6 - 1);
46.     }
47.     fitness = obj1 + obj2 + obj3 + obj4 + obj5 + obj6;
48.     if (fitness >=6){
49.         fitness=6-(fitness-6);
50.     }
51.     fitness = 1/(1+(6-fitness));
52. }

```

Source Code 4.2 Perhitungan Fitness

4.2.3. Proses Crossover

Dalam proses *crossover* terdapat proses pemilihan induk, untuk proses *crossover* ini akan diambil dengan membandingkan nilai Pc yang dimasukkan dengan nilai R yang dirandom untuk tiap individu. Individu dengan nilai $R < P_c$ input akan dipilih sebagai *parent*. Setelah didapatkan *parent*, maka akan dilakukan *crossover* pada tiap individu terpilih dengan metode *Simple Arithmetic Crossover*. Dalam proses pemilihan induk atau *parent*, diinisialisasikan variabel random, *parent*, *child* (1-5). Lalu akan dibangkitkan nilai R secara acak (baris 6-7) dan dibandingkan dengan nilai Pc, jika nilainya kurang dari Pc, maka akan disimpan sebagai *parent* dalam *Arraylist Parent* (baris 10-15). Proses pemilihan *parent* ini dapat dilihat pada *Source code 4.3*.

```

public void Crossover(double crossoverProb)
1. Random random = new Random();
2. double alpha = 0.5;
3.
4. ArrayList<Individu> parent = new ArrayList<Individu>();
5. ArrayList<Individu> child = new ArrayList<Individu>();
6. ArrayList<Double> randomCrossoverProb =
7. JAlgen.generateRandomDouble(populasi.size(), 0, 1);
8.
9. int i = 0;
10. for (double crossoverProbIndividu : randomCrossoverProb){
11.     if(crossoverProbIndividu < crossoverProb){
12.         parent.add(populasi.get(i));
13.     }
14.     i++;
15. }
16.
17. System.out.println(" ");
18. System.out.println("PROSES CROSSOVER : ");
19. System.out.println("Induk yang terpilih : ");
20. for (Individu c : parent){
21.     System.out.println(c);
22. }

```

Source Code 4.3. Pemilihan Induk Crossover

Setelah didapatkan *parent* dari proses sebelumnya, maka akan dilakukan proses *crossover*. Proses *crossover* dilakukan dengan menyilangkan semua kemungkinan dari *parent* yang terpilih seperti pada (baris 1-2). Dari semua *parent* akan dikombinasikan 1 *crossover* untuk setiap pasangan dengan menggunakan *cut point* yang dibangkitkan secara random, sebagai titik persilangan atau tukar gen seperti pada kode (baris 10). Proses *Simple Arithmetic Crossover* dilakukan pada kode (baris 34-39 dan baris 43-49). Untuk hasil *offspring*, pada masing-masing *parent* akan diambil nilai dari setiap gen setelah titik potong yang dilakukan kemudian dimasukkan ke dalam *ArrayList* Populasi seperti pada kode (baris 51 - 52). *Source code* dari proses tersebut dapat dilihat pada *Source code* 4.4.

```

public void Crossover(double crossoverProb)
1.  for(int x = 0 ; x < parent.size()-1 ; x++){
2.  for(int y = x+1 ; y <parent.size() ; y++){
3.
4.  //int x = 0, y =1;
5.  Individu child1 = parent.get(x);
6.  Individu child2 = parent.get(y);
7.
8.  //random cut point 0 sampai panjang kromosom
9.
10. int cutpoint = random.nextInt (parent.get(0). getIndividu().
11. size());
12. System.out.println("Cut Point terpilih : " +cutpoint);
13. System.out.println("Parent 1 : " );
14. String print = "";
15. for(double Gen : parent.get(x).getIndividu()){
16.     print += ((String.format("%1$, .2f", Gen)) + " | ");
17. }
18. System.out.println(print);
19. System.out.println("Parent 2 : " );
20. print = "";
21. for(double Gen : parent.get(y).getIndividu()){
22.     print += ((String.format("%1$, .2f", Gen)) + " | ");
23. }
24. System.out.println(print);
25. System.out.println("");
26.
27. ArrayList<Double> child1Hasil = new ArrayList<Double>();
28. child1Hasil.addAll (child1.getIndividu());
29. ArrayList<Double> child2Hasil = new ArrayList<Double>();
30. child2Hasil.addAll (child2.getIndividu());
31.
32. //child1
33. //substitute from index cut point to end
34. for(i = cutpoint ; i < child1Hasil.size() ; i++){
35.     double newGen = Math.round(((alpha * parent.get(y).
36. getIndividu().get(i)) + ((1 - alpha) * parent.get(x).
37. individu.get(i)) * Math.pow(10,1))/Math.pow(10,1));

```

```

38. child1Hasil.set(i, newGen);
39. //System.out.println("Childhasil1 "+child1Hasil);
40. }
41.
42. //child2
43. for(i = cutpoint ; i < child2Hasil.size() ; i++){
44. double newGen = Math.round(((alpha * parent.get(x).
45. getIndividu().get(i)) + ((1 - alpha) * parent.get(y).
46. getIndividu().get(i)) * Math.pow(10,1))/Math.pow(10,1));
47. child2Hasil.set(i, newGen);
48. //System.out.println("Childhasil2 "+child2Hasil);
49. }
50.
51. child.add(new Individu(child1Hasil, jen_fk, jen_variabel));
52. child.add(new Individu(child2Hasil, jen_fk, jen_variabel));
53. }
54. }
55. if(child.size() > 0){
56.     populasi.addAll(child);
57. }
58. System.out.println("HASIL CROSSOVER : " );
59. for (Individu c : populasi){
60.     System.out.println(c);
61. }

```

Source Code 4.4 Proses Simple Arithmetic Crossover

4.2.4. Mutasi

Dalam proses mutasi, terdiri dari dua prosedur. Prosesur yang pertama adalah memilih individu mana yang akan mengalami mutasi. Prosedur yang kedua adalah prosedur mutasi itu sendiri.

Dalam pemilihan individu yang mengalami mutasi ini akan dilakukan secara random berdasarkan nilai probabilitas mutasi (pm), jadi akan dibangkitkan nilai R tiap individu secara acak (baris 1) dan nilai R masing-masing individu dibandingkan dengan nilai Pm yang sudah ditentukan (baris 7-12), jika nilai R kurang dari yang ditentukan, maka individu akan mengalami mutasi (baris 9). Proses tersebut dapat dilihat pada *Source code 4.5*.

```

public void Mutasi(double mutationProb)
1. ArrayList<Double> randomMutationProb =
2. JAlgen.generateRandomDouble(populasi.size(), 0, 1);
3.
4. ArrayList<Integer> indexToMutate = new ArrayList<Integer>();
5.
6. int i = 0;
7. for (double individuMutationProb : randomMutationProb ){
8.     if(individuMutationProb < mutationProb){
9.         indexToMutate.add(i);
10.    }
11.    i++;
12. }

```

```

13. System.out.println(" ");
14. System.out.println("PROSES MUTASI : ");
15. System.out.println("Indeks yang terpilih : ");
16. for (Integer c : indexToMutate){
17.     System.out.println(c);
18. }

```

Source Code 4.5 Pemilihan Induk Mutasi

Proses berikutnya setelah didapatkan individu mana yang akan mengalami mutasi, maka akan dilakukan mutasi terhadap individu yang terpilih. Jadi dalam proses ini juga akan dilakukan pembangkitan nilai random untuk menentukan gen yang mengalami mutasi (baris 1). Gen yang mengalami mutasi akan dirubah nilainya dengan nilai yang dibangkitkan secara random (21-45). Kemudian hasil mutasi akan ditambahkan ke dalam ArrayList Populasi (baris 48). Proses mutasi tersebut dapat dilihat pada *Source code 4.7*.

```

public void Mutasi(double mutationProb)
1. double genMutationProb = random.nextDouble();
2.
3. ArrayList<Double> randomGenMutationProb =
4. JAlgen.generateRandomDouble(populasi.size(), 0, 1);
5.
6. for (int indexMutate : indexToMutate){
7. Individu mutated = new Individu(populasi.get(indexMutate).
8. getIndividu(), jen_fk, jen_variabel);
9.
10. ArrayList<Double> mutatedGen = new ArrayList<Double>();
11. mutatedGen.addAll(populasi.get(indexMutate).getIndividu());
12. mutated.setIndividu(mutatedGen);
13.
14. i = 0 ;
15. for (double individuGenMutationProb : randomGenMutationProb){
16. if(individuGenMutationProb < genMutationProb && i <
17. populasi.get(indexMutate).getIndividu().size() ){
18. System.out.println("Melakukan perubahan pada individu "
19. +(indexMutate+1) + " Gen ke " + (i+1));
20.
21. //mutasi gen
22. if(jen_fk == "IMT"){
23. if(jen_variabel == "Sangat Kurus"){
24. double newRandomGen = Math.round((Math.random()*19) *
25. Math.pow(10,1))/Math.pow(10,1);
26. //jika hasil masih sama
27. while (newRandomGen ==
28. populasi.get(indexMutate).getIndividu().get(i)){
29. newRandomGen = Math.round((Math.random()*19) *
30. Math.pow(10,1))/Math.pow(10,1);
31. }
32. mutatedGen.set( i , newRandomGen);
33. }
34.
35. if(jen_fk == "Umur"){
36. if(jen_variabel == "Balita"){

```

```

37. double newRandomGen = Math.round((Math.random()*4) *
38. Math.pow(10,1))/Math.pow(10,1);
39. //jika hasil masih sama
40. while (newRandomGen ==
41. populasi.get(indexMutate).getIndividu().get(i)){
42. newRandomGen = Math.round((Math.random()*4) *
43. Math.pow(10,1))/Math.pow(10,1);
44. }
45. mutatedGen.set( i , newRandomGen);
46. }
47. i++;
48. }
49. populasi.add(new Individu(mutatedGen, jen_fk,jen_variabel));
50. }

```

Source Code 4.6 Proses Mutasi Gen

4.2.5. Seleksi

Proses seleksi dilakukan dengan metode *Rank-Based Fitness*, dimana semakin besar nilai *fitness* suatu individu, maka semakin besar kesempatan terpilih menjadi individu dalam populasi generasi berikutnya. Seleksi ini dimulai dengan mengurutkan nilai *fitness* dari tiap individu yang sudah dihitung dari posisi tertinggi ke posisi terendah (baris 4-11). Kemudian menghapus individu yang berada diluar jumlah populasi awal (baris 13-16). Dalam proses ini akan diambil individu sejumlah populasi awal, sehingga individu dengan nilai *fitness* terendah tidak digunakan kembali pada generasi berikutnya. Proses seleksi ini dapat dilihat pada *Source code 4.7* berikut:

```

public void Seleksi(int jumlahPopAwal)
1. public void Seleksi(int jumlahPopAwal){
2. //sorting by fitness
3. ArrayList<Individu> populasiSorted = populasi;
4. for(int i=0 ; i < populasiSorted.size(); i++){
5. double maxFitness = Double.MIN_VALUE;
6. int toSwap = i;
7. for(int j=i ; j<populasiSorted.size(); j++){
8. if(populasi.get(j).fitness() > maxFitness){
9. maxFitness = populasi.get(j).fitness();
10. toSwap = j;
11. }
12. }
13. if(toSwap != i){
14. Individu oldindividu = populasi.get(i);
15. populasi.set(i, populasi.get(toSwap));
16. populasi.set(toSwap, oldindividu);
17. }
18. }
19. for(int i=0; i<jumlahPopAwal; i++){
20. populasiSelected.add(populasi.get(i));
21. }
22. populasi.removeAll (populasi);

```

```

23. for(int i=0; i<jumlahPopAwal; i++){
24.     populasi.add(populasiSelected.get(i));
25. }
26. System.out.println(" " );
27. System.out.println("HASIL SELEKSI : " );
28. for (Individu c : populasi){
29.     System.out.println(c);
30. }
31. }

```

Source Code 4.7 Seleksi Rank-Based Fitness

4.2.6. Fuzzyfikasi

Fuzzyfikasi merupakan tahap awal dari *Fuzzy Model Sugeno*, dimana data dari kriteria yang telah dimasukkan oleh user akan dibagi ke dalam derajat keanggotaan yang telah ditentukan. Batas-batas rentang nilai pada fungsi keanggotaan sesuai dengan rentang nilai yang telah dihasilkan melalui metode Algoritma Genetika. Sehingga output pada tahap ini akan menghasilkan angka pada masing-masing derajat keanggotaan untuk selanjutnya diproses ke dalam tahap inferensi. Pada baris 8-74 merupakan proses perhitungan fuzzyfikasi sesuai fungsi keanggotaan yang ditentukan. Proses perhitungan fuzzyfikasi menggunakan persamaan sesuai model fungsi keanggotaannya. Dalam penelitian ini model fungsi keanggotaan yang dipakai adalah trapesium. Pada Proses fuzzyfikasi ini dapat dilihat pada *Source code 4.8* berikut:

AGFuzzyKeanggotaanAMB.java

```

1. public class AGFuzzyKeanggotaanAMB {
2.     public double umur;
3.
4.     AGFuzzyKeanggotaanAMB(double umur){
5.         this.umur=umur;
6.     }
7.
8.     public double Balita() {
9.         if ((umur >=gb1) && (umur <=gb2)) {
10.            return 1;
11.        } else if ((umur > gb2) && (umur <= gb3)) {
12.            return (gb3-umur)/1;
13.        } else {
14.            return 0;
15.        }
16.    }
17.
18.    public double Anakanak() {
19.        if ((umur < ga1) || (umur > ga4)) {
20.            return 0;
21.        } else if ((umur >= ga1) && (umur <= ga2)) {
22.            return (umur-ga1)/1;
23.        } else if ((umur >= ga3) && (umur <= ga4)) {
24.            return (ga4-umur)/1;

```

```

25.         } else {
26.             return 1;
27.         }
28.     }
29.
30.     public double Remaja() {
31.         if ((umur < gr1) || (umur > gr4)) {
32.             return 0;
33.         } else if ((umur >= gr1) && (umur <= gr2)) {
34.             return (umur-gr1)/1;
35.         } else if ((umur >= gr3) && (umur <= gr4)) {
36.             return (gr4-umur)/1;
37.         } else {
38.             return 1;
39.         }
40.     }
41.
42.     public double Dewasa() {
43.         if ((umur < gd1) || (umur > gd4)) {
44.             return 0;
45.         } else if ((umur >= gd1) && (umur <= gd2)) {
46.             return (umur-gd1)/1;
47.         } else if ((umur >= gd3) && (umur <= gd4)) {
48.             return (gd4-umur)/(gd4-gd3);
49.         } else {
50.             return 1;
51.         }
52.     }
53.
54.     public double Parobaya() {
55.         if ((umur < gp1) || (umur > gp4)) {
56.             return 0;
57.         } else if ((umur >= gp1) && (umur <= gp2)) {
58.             return (umur-gp1)/(gp2-gp1);
59.         } else if ((umur >= gp3) && (umur <= gp4)) {
60.             return (gp4-umur)/(gp4-gp3);
61.         } else {
62.             return 1;
63.         }
64.     }
65.     public double Tua() {
66.         if ((umur <= gt1) && (umur > gt2)) {
67.             return (umur-gt1)/(gt2-gt1);
68.         } else if (umur < gt1) {
69.             return 0;
70.         } else {
71.             return 1;
72.         }
73.     }
74. }

```

Source Code 4.8 Proses Fuzzyfikasi

4.2.7. Inferensi

Aturan atau *Rules* berupa pernyataan-pernyataan yang ditulis dalam bentuk aturan *if-then*, aturan-aturan penentuan tersebut dapat dibentuk berdasarkan referensi dari penelitian sebelumnya (baris 34-62). Pada kesimpulan setiap rule

menggunakan metode inferensi *Sugeno* dimana hasil dari fungsi *min* pada aturan akan digunakan pada fungsi inferensi *Sugeno* untuk mendapatkan nilai *z*. Hasil inferensi pada model *Sugeno* ini, menggunakan persamaan orde satu yaitu menggunakan inferensi hasil yang dihasilkan oleh suatu rumus. Rumus yang digunakan berasal dari penelitian sebelumnya. Proses inferensi ini dapat dilihat pada *Source code* 4.9 berikut:

```

Public void Rule( )
1.   hasil = new double[240];
2.   hasilR = new double[240];
3.   Hasil uji = new Hasil();
4.
5.       FuzzyKeanggotaanIMT      imt      =      new
6. FuzzyKeanggotaanIMT(nilai_imt);
7.       FuzzyKeanggotaanAMB      amb      =      new
8. FuzzyKeanggotaanAMB(nilai_amb);
9.       JenisKelamin jk = new JenisKelamin(jns_kelamin);
10.      JenisAktivitas ak= new JenisAktivitas(n_akt);
11.
12.      double imt_sangatkurus= imt.Sangat_kurus();
13.      double imt_kurus= imt.Kurus();
14.      double imt_normal= imt.Normal();
15.      double imt_gemuk= imt.Gemuk();
16.      double imt_obesitas= imt.Obesitas();
17.
18.      double umur_balita= amb.Balita();
19.      double umur_anakanak= amb.Anakanak();
20.      double umur_remaja= amb.Remaja();
21.      double umur_dewasa= amb.Dewasa();
22.      double umur_parobaya= amb.Parobaya();
23.      double umur_tua= amb.Tua();
24.
25.      double jk_p=jk.Wanita();
26.      double jk_l=jk.Lakilaki();
27.
28.      double aktivitas_sangatringan= ak.SangatRingan();
29.      double aktivitas_ringan= ak.Ringan();
30.      double aktivitas_sedang= ak.Sedang();
31.      double aktivitas_berat= ak.Berat();
32.      //== RULE =====
33.
34.      hasil[0]=Find_Min(imt_sangatkurus,umur_balita,jk_p,aktivitas_s
35.      angatringan);
36.      uji.setHasil(hasil[0]);
37.      hasilR[0]=(((61*bb)-51+500)*1.3);
38.
39.      hasil[1]=Find_Min(imt_sangatkurus,umur_balita,jk_p,aktivitas_r
40.      ingan);
41.      uji.setHasil(hasil[1]);
42.      hasilR[1]=(((61*bb)-51+500)*1.55);
43.
44.      hasil[2]=Find_Min(imt_sangatkurus,umur_balita,jk_p,aktivitas_s
45.      edang);
46.      uji.setHasil(hasil[2]);
47.      hasilR[2]=(((61*bb)-51+500)*1.7);
48.

```

```

49.  hasil[3]=Find_Min(imt_sangatkurus,umur_balita,jk_p,aktivitas_b
50.  erat);
51.      uji.setHasil(hasil[3]);
52.      hasilR[3]=(((61*bb)-51+500)*2);
53.
54.  hasil[4]=Find_Min(imt_sangatkurus,umur_anakanak,jk_p,aktivitas
55.  _sangatringan);
56.      uji.setHasil(hasil[4]);
57.      hasilR[4]=(((22.5*bb)+499+500)*1.3);
58.
59.  hasil[5]=Find_Min(imt_sangatkurus,umur_anakanak,jk_p,aktivitas
60.  _ringan);
61.      uji.setHasil(hasil[5]);
62.      hasilR[5]=(((22.5*bb)+499+500)*1.55);

```

Source Code 4.9 Proses Inferensi Model Sugeno

4.2.8. Defuzzyfikasi

Setelah semua nilai inferensi keanggotaan diketahui pada proses fuzzyfikasi, dan nilai z telah diketahui yang merupakan hasil dari tahap inferensi, selanjutnya pada proses penentuan output crisp disini dilakukan dengan cara menggunakan rata-rata berbobot (*Weight Average*) (baris 5-11). Hasil keluaran dari defuzzifikasi ini adalah nilai Angka Metabolisme Basal (AMB) dari seorang pasien dalam satuan *kcal*. Proses defuzzyfikasi ini dapat dilihat pada *Source code* 4.10 berikut:

```

Public double defuzzy( )
1.  double temp1 = 0, temp2 = 0;
2.      DecimalFormat bulat = new DecimalFormat("#");
3.      double temp1 = 0, temp2 = 0;
4.
5.      double hasil1, hasil2;
6.      int i = 0;
7.
8.      for (i = 0; i < 240; i++) {
9.          temp1 = temp1+ hasil[i] * hasilR[i];
10.         temp2 = temp2 + hasil[i];
11.     }
12.     System.out.println("temp1 :" +temp1);
13.     System.out.println("temp2 :" +temp2);
14.     hasil1 = temp1 / temp2;
15.     hasil2 = Double.valueOf(bulat.format(hasil1));
16.     System.out.println("hasil defuzzifikasi :" +hasil2);
17.     return hasil2;

```

Source Code 4.10 Proses Defuzzyfikasi

4.3. Implementasi Antarmuka Aplikasi

Antarmuka Aplikasi Sistem Penerapan Algoritma Genetika untuk optimasi fungsi keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan

Metode *Fuzzy Inference System* Model Sugeno ini digunakan oleh pengguna untuk berinteraksi dengan sistem perangkat lunak. Antarmuka perangkat lunak ini dibagi menjadi 4 halaman otoritas, yaitu halaman *login*, halaman proses pembentukan fungsi keanggotaan, halaman akurasi sistem dan halaman prediksi nilai AMB pasien. Berikut hasil implementasi antarmuka pada masing-masing halaman pada aplikasi.

4.3.1. Tampilan Halaman *Login*

Halaman *login* merupakan halaman yang tampil pertama kali ketika mengoperasikan aplikasi. Berdasarkan perancangan antarmuka pada bab 3, halaman *login* ini terdapat dua kolom yang harus diisi oleh *user*. Kolom tersebut yaitu kolom *username* dan kolom *password*. *User* harus mengisi kolom tersebut sesuai dengan data yang benar agar dapat mengakses halaman utama aplikasi. Berikut pada gambar 4.1 menampilkan halaman *login* yang telah diimplementasikan pada sistem.



Gambar 4.1 Halaman antarmuka *login* pengguna

4.3.2. Tampilan Halaman Pembentukan Fungsi Keanggotaan

Setelah *user* melalui tahap *login* maka selanjutnya masuk ke dalam interface halaman pembentukan fungsi keanggotaan. *User* harus memasukkan nilai probabilitas *crossover*, nilai probabilitas mutasi, jumlah generasi dan jumlah populasi yang akan digunakan oleh sistem untuk melakukan proses pembuatan fungsi keanggotaan dengan Metode Algoritma Genetika. Terdapat

dua tombol dalam halaman ini yaitu tombol proses digunakan untuk menghasilkan batas-batas fungsi keanggotaan yang akan tampil pada kolom bawah. Tombol clear digunakan untuk menghapus nilai pada *form*. Terdapat menu bar Help yang berfungsi menampilkan halaman help yang menunjukkan panduan dari halaman antarmuka yang sedang di jalankan.

**APLIKASI OPTIMASI MEMBERSHIP FUNCTION DENGAN ALGORITMA GENETIKA PADA
PENENTUAN ANGKA METABOLISME BASAL (AMB) DENGAN METODE FUZZY MODEL SUGENO
2014**

Proses Fungsi Keanggotaan Uji Akurasi Data Prediksi AMB Pasien

INPUT DATA PERHITUNGAN ALGORITMA GENETIKA

Jenis Fungsi Keanggotaan:

Nilai Probabilitas Crossover: Skala (0.1 - 0.9)

Nilai Probabilitas Mutasi: Skala (0.1 - 0.9)

Jumlah Generasi:

Jumlah Populasi:

HASIL BATAS-BATAS FUNGSI KEANGGOTAAN

Fungsi Keanggotaan IMT
Batas-batas fungsi keanggotaan:

Variabel	Batas Fungsi Keanggotaan	Fitness
Sangat Kurus	[[0.0, 11.9, 13.0]]	0.61127282349828
Kurus	[[16.4, 17.1, 17.6, 18.5]]	0.5474452554744534
Normal	[[18.0, 19.6, 24.7, 25.4]]	0.4706940817625237
Gemuk	[[24.5, 25.9, 26.5, 27.6]]	0.4800000000000005
Obesitas	[[38.1, 39.2, 48.9]]	0.45262910094805964

Nilai Fitness IMT:

Fungsi Keanggotaan Umur
Batas-batas fungsi keanggotaan:

Variabel	Batas Fungsi Keanggotaan	Fitness
Balita	[[0.0, 2.5, 3.6]]	0.8860759493670882
Anak-anak	[[2.5, 2.9, 8.9, 9.9]]	0.5679513184584177
Remaja	[[10.2, 11.5, 18.2, 18.6]]	0.4712482468443202
Dewasa	[[17.5, 18.4, 25.6, 28.8]]	0.5665163911038248
Parobaya	[[36.3, 41.4, 53.3, 57.3]]	0.3281557646029315
Tua	[[55.5, 59.7, 106.3]]	0.7773174013640335

Nilai Fitness Umur:

@PTIHK-2014

HASIL FUNGSI KEANGGOTAAN FITNESS TERBAIK

F.K IMT Terbaik

SK1	SK2	SK3	K1	K2	K3	K4	N1	N2	N3	N4	G1	G2	G3	G4	O1	O2	O3	Fitness
0.0	16.5	17.5	15.0	16.0	16.5	17.5	17.1	18.1	23.6	24.6	23.0	24.0	25.0	26.0	25.0	26.0	48.5	5.0

F.K Umur Terbaik

B1	B2	B3	A1	A2	A3	A4	R1	R2	R3	R4	D1	D2	D3	D4	P1	P2	P3	P4	T1	T2	T3	Fitness
0.2	2.7	3.7	2.5	3.5	9.3	10.3	9.5	10.5	17.5	18.5	17.5	18.5	25.5	30.5	25.5	30.5	54.4	25.5	55.5	60.5	110.0	5.75...

Gambar 4.2 Halaman antarmuka pembentukan fungsi keanggotaan

4.3.3. Halaman Hasil Akurasi Sistem

Gambar 4.3 merupakan hasil implementasi halaman hasil akurasi sistem, yang merupakan hasil perhitungan yang dihasilkan oleh Algoritma Genetika dan *Fuzzy Inference System Model Sugeno*. Dalam halaman ini ditampilkan tabel data uji pasien dengan perhitungan manual dan perhitungan yang dilakukan oleh sistem, kemudian diproses seberapa besar prosesntase akurasi nilai AMB yang

dihasilkan oleh sistem terhadap perhitungan manual. Terdapat menu bar *Help* yang berfungsi menampilkan halaman *help* yang menunjukkan panduan dari halaman antarmuka yang sedang di jalankan.

TABEL HASIL AKURASI SISTEM

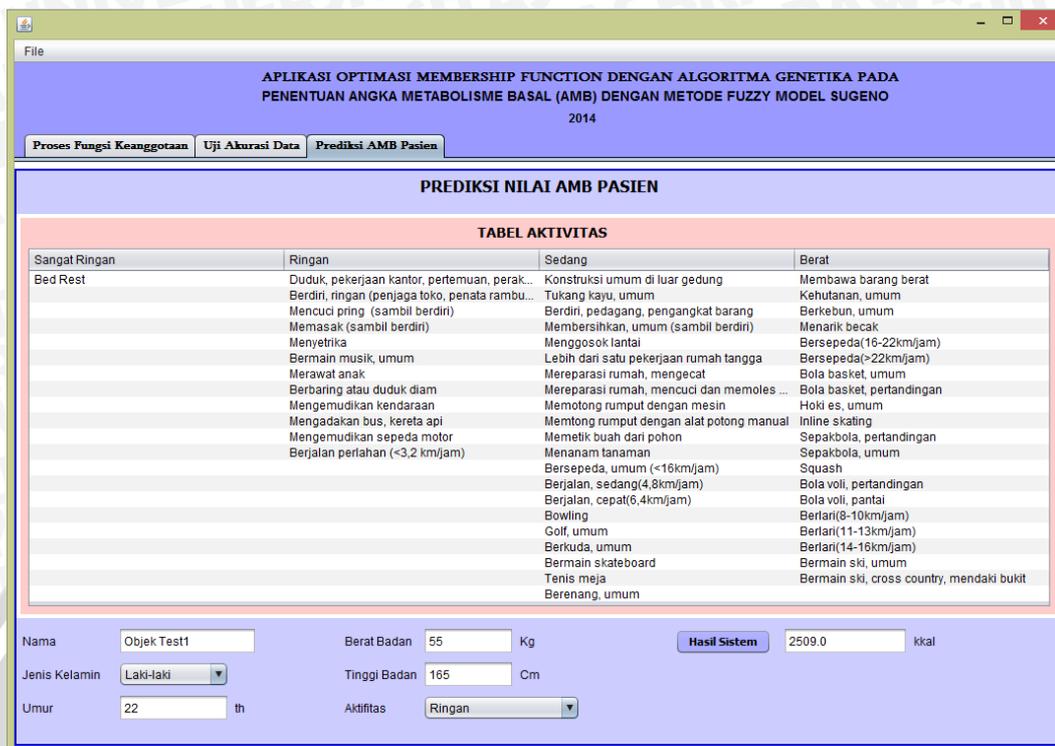
ID Pasien	Nama	Jen. Kelamin	Umur	Tinggi	Berat	Jen. Aktivitas	Hasil Manual	Hasil Sistem
43	pasien 43	2	19	155	86	1	1953	1953
44	pasien 44	1	43	155	80	2	2300	2300
45	pasien 45	2	46	160	56	2	2238	2238
46	pasien 46	1	50	160	54	3	3161	3161
47	pasien 47	1	84	175	55	0	2248	2248
48	pasien 48	1	84	175	55	1	2854	2881
49	pasien 49	1	84	175	55	2	3044	3044
50	pasien 50	1	53	157	42	0	2426	2426
51	pasien 51	1	75	175	55	1	2854	2881
52	pasien 52	2	80	140	42	3	2074	2074
53	pasien 53	2	62	145	48	1	1705	1705
54	pasien 54	2	77	145	52	2	1941	1941
55	pasien 55	2	39	155	86	0	1400	1400
56	pasien 56	1	43	155	80	2	2300	2300
57	pasien 57	2	51	155	63	3	2754	2754
58	pasien 58	1	65	180	63	3	2809	2809
59	pasien 59	1	55	180	84	3	3892	3892
60	pasien 60	2	52	180	70	1	2229	2229
61	pasien 61	2	52	180	70	0	1960	1960

Prosentase Akurasi Sistem **94.0** %

Gambar 4.3 Halaman antarmuka hasil akurasi sistem

4.3.4. Tampilan Halaman Prediksi AMB Pasien

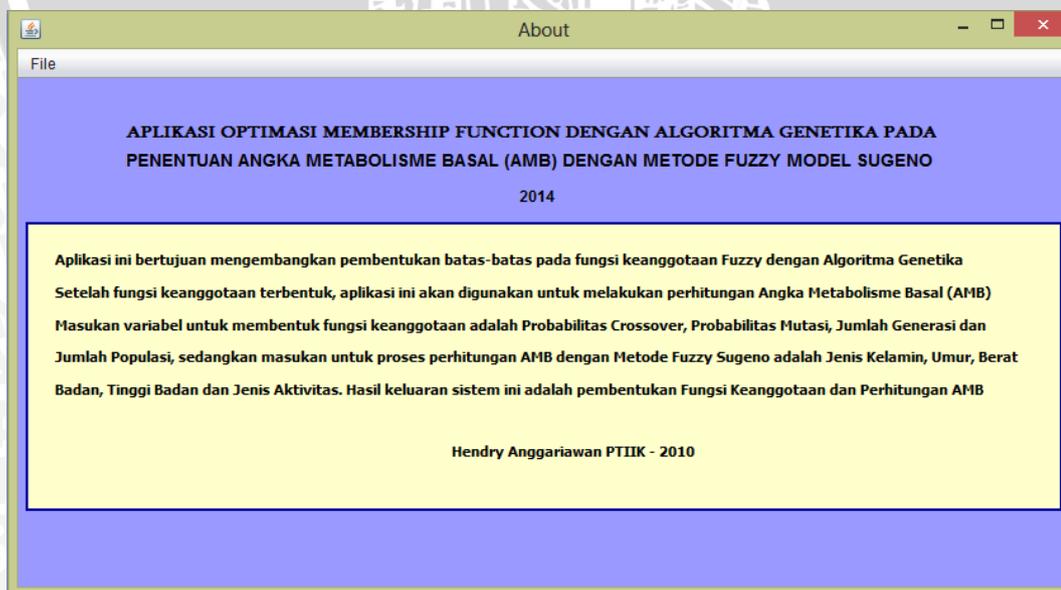
Setelah pengguna selesai pada proses pembuatan batas-batas fungsi keanggotaan dengan Algoritma Genetika, maka selanjutnya seperti yang ditampilkan pada gambar 4.4, merupakan hasil implementasi halaman prediksi AMB pasien. Halaman ini terdapat *field* data pasien yang harus diisi oleh *user* yang meliputi nama, berat badan, tinggi badan, jenis kelamin, usia, dan aktivitas yang dilakukan oleh pasien. Terdapat tombol dalam halaman ini yaitu tombol proses sistem dimana hasil yang ditampilkan adalah nilai AMB dari perhitungan sistem dengan metode *Fuzzy Model Sugeno*. Selanjutnya, terdapat *menu bar* Help yang berfungsi menampilkan halaman help yang menunjukkan panduan dari halaman antarmuka yang sedang di jalankan.



Gambar 4.4 Halaman antarmuka prediksi AMB Pasien

4.3.5. Halaman About

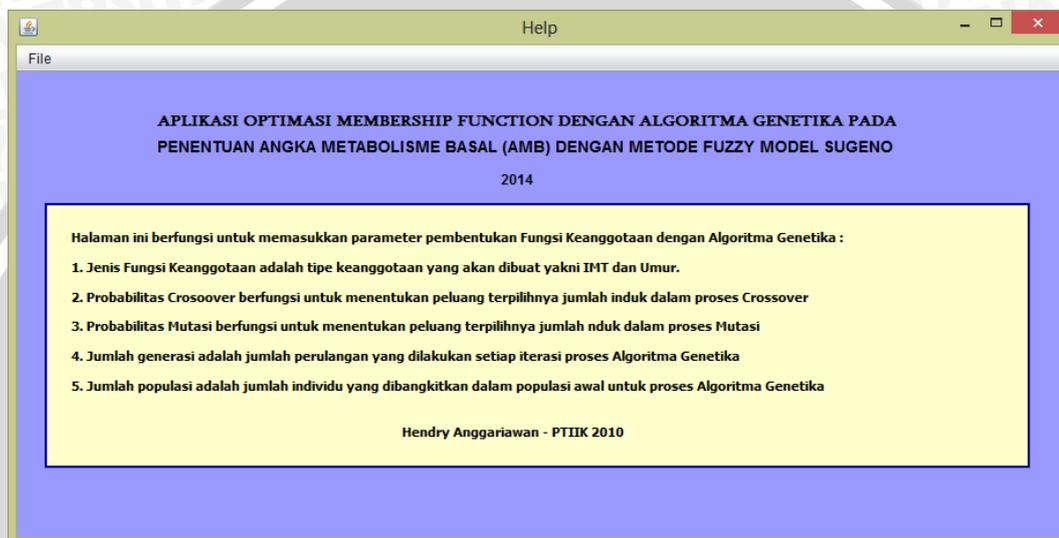
Halaman ini akan menampilkan informasi tentang deskripsi atau penjelasan dari aplikasi sistem yang dibangun. Cara mengakses halaman ini adalah menekan tombol Menu kemudian tekan About.



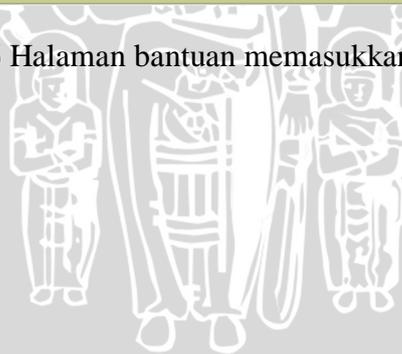
Gambar 4.5 Halaman about sistem

4.3.6. Halaman *Help*

Halaman ini akan muncul berbeda informasi berdasarkan tiap menu bar halaman aplikasi yang di jalankan. Cara mengakses halaman ini adalah menekan tombol Help kemudian tekan Go To Help. Halaman ini bertujuan memberikan bantuan kepada pengguna yang berguna untuk cara menjalankan program secara benar dan memberi panduan terhadap input data secara benar.



Gambar 4.6 Halaman bantuan memasukkan data



BAB V

PENGUJIAN DAN ANALISIS

Pada bab ini dilakukan proses pengujian terhadap Sistem yang telah dibangun. Proses pengujian dilakukan melalui empat tahap yaitu pengujian validasi sistem, pengujian parameter Algoritma Genetika, pengujian data latih dan pengujian akurasi. Pada pengujian validasi akan digunakan teknik pengujian *Black Box (Black Box Testing)*. Pengujian parameter Algoritma Genetika meliputi pengujian nilai probabilitas *crossover* dan mutasi, pengujian jumlah generasi dan pengujian ukuran populasi, pengujian yang dilakukan adalah membandingkan perubahan nilai parameter terhadap nilai *fitness* yang dihasilkan. Selanjutnya, setelah ditemukan kombinasi parameter Algoritma Genetika dengan nilai rata-rata *fitness* yang tertinggi kemudian akan dilakukan percobaan terhadap data latih yang diuji nilai prosentase akurasinya, hasil fungsi keanggotaan yang mempunyai nilai prosentase yang tinggi akan dipilih untuk digunakan dalam pengujian akurasi. Pengujian akurasi digunakan untuk menguji tingkat akurasi antara perhitungan secara manual yaitu menggunakan rumus FAO/WHO/UNU dengan perhitungan sistem yang telah diimplementasikan.

5.1. Pengujian Validasi

Pengujian validasi digunakan untuk mengetahui apakah 107system yang dibangun sudah benar sesuai dengan yang dibutuhkan. Item-item yang telah dirumuskan dalam daftar kebutuhan akan menjadi acuan untuk melakukan pengujian validasi. Pengujian validasi menggunakan metode pengujian *Black Box*, tidak difokuskan terhadap alur jalannya algoritma program namun lebih ditekankan untuk menemukan kesesuaian antara kinerja 107system dengan daftar kebutuhan. Pada setiap kebutuhan dilakukan proses pengujian dengan kasus uji masing-masing untuk mengetahui kesesuaian antara kebutuhan dengan kinerja sistem. Pada Tabel 5.1 menunjukkan hasil pengujian validasi aplikasi Sistem.

Tabel 5.1 Hasil pengujian validasi

No	Kasus	Hasil Yang diharapkan	Hasil yang didapat	Status
1	Login	Sistem mampu menerima masukan dan memproses <i>Login</i>	Sistem dapat menerima masukan form <i>login</i> sehingga <i>user</i> dapat mengakses halaman utama.	Valid
2	Memproses Fungsi Keanggotaan	Sistem mampu menerima masukan nilai probabilitas <i>crossover</i> , mutasi, jumlah generasi, jumlah populasi dan menghasilkan batas-batas fungsi keanggotaan.	Sistem dapat menghasilkan fungsi keanggotaan secara dinamis sesuai masukan parameter Algoritma Genetika.	Valid
3	Uji Akurasi Sistem	Sistem mampu menerima masukan data uji dan melakukan perhitungan AMB dengan metode <i>Fuzzy Sugeno</i> serta metode manual	Sistem mampu menghasilkan perhitungan AMB hasil metode <i>Fuzzy Sugeno</i> dan Perhitungan manual, serta memproses akurasi kesamaan hasil.	Valid
4	Proses Perhitungan AMB	Sistem mampu menampilkan hasil perhitungan AMB pada satu pasien baik secara metode manual atau metode <i>Fuzzy Sugeno</i>	Sistem dapat menampilkan hasil perhitungan AMB pada seorang pasien baik secara metode manual atau metode <i>Fuzzy Sugeno</i>	valid
5	Logout	Sistem mampu merespon perintah untuk keluar dari halaman yang dioperasikan menuju halaman <i>login</i>	Sistem merespon ketika tombol logout di klik maka langsung kembali ke halaman <i>login</i>	valid

Berdasarkan pengujian fungsionalitas terhadap 5 tindakan dalam daftar kebutuhan dengan metode *black box testing* menunjukkan bahwa Sistem Penerapan Algoritma Genetika untuk Optimasi Fungsi Keanggotaan *Fuzzy Inference System* Model Sugeno pada Perhitungan AMB memiliki fungsionalitas sebagai berikut:

$$\begin{aligned}\text{Fungsionalitas} &= \frac{\text{jumlah tindakan yang dilakukan}}{\text{jumlah tindakan dalam daftar kebutuhan}} \times 100\% \\ &= \frac{5}{5} \times 100\% \\ &= 100\%\end{aligned}$$

Dari 5 kasus uji yang telah dilakukan pengujian *black box* menunjukkan nilai valid sebesar 100% yang menandakan bahwa fungsionalitas 109system dapat berjalan dengan baik sesuai dengan daftar kebutuhan.

5.2. Pengujian parameter Algoritma Genetika

Pengujian parameter Algoritma Genetika meliputi pengujian nilai probabilitas *crossover* dan mutasi, pengujian jumlah generasi dan pengujian ukuran populasi, pengujian yang dilakukan adalah membandingkan perubahan nilai parameter terhadap nilai *fitness* yang dihasilkan. Berikut merupakan hasil pengujian parameter Algoritma Genetika.

5.2.1. Pengujian probabilitas *crossover* dan probabilitas mutasi terhadap *fitness*

Pengujian yang pertama adalah mengetahui apakah perubahan nilai probabilitas *crossover* (P_c) dan probabilitas mutasi (P_m) berpengaruh terhadap perubahan nilai *fitness* serta mengetahui berapa besar nilai peluang *crossover* dan peluang mutasi terbaik. Dalam pengujian ini digunakan parameter jumlah populasi 25, jumlah generasi 25, dan nilai *alpha* 0.5. Selanjutnya, dalam proses pengujian akan dilakukan terhadap kombinasi dari 9 nilai probabilitas *crossover* dan 9 nilai probabilitas mutasi dengan nilai antara 0.1 – 0.9. Dalam setiap kali kombinasi, akan dilakukan 5 kali percobaan lalu diambil nilai rata-rata *fitness*nya. Nilai *fitness* inilah yang akan dibandingkan pada tiap kombinasi probabilitas *crossover* dan probabilitas mutasi. Hasil uji coba pengujian probabilitas *crossover* dan probabilitas mutasi terhadap nilai *fitness* dapat dilihat pada tabel 5.2 dan 5.3 untuk fungsi keanggotaan IMT dan tabel 5.4 dan 5.5 untuk fungsi keanggotaan Umur.

Tabel 5.2 Pengujian Pc 0.1-0.9 dan Pm 0.1-0.4 pada Fungsi Keanggotaan IMT

Nilai Pc	Nilai Pm			
	0.1	0.2	0.3	0.4
0.1	2,8200	3,0704	3,1986	3,4046
0.2	2,944	3,1222	3,2908	3,3332
0.3	3,1696	3,1134	3,3338	3,5217
0.4	3,3284	3,3725	3,5211	4,0326
0.5	3,3753	3,6486	3,6584	3,805
0.6	3,586	3,9357	3,9512	4,1561
0.7	3,7445	3,8001	4,3387	4,3413
0.8	4,1236	4,1625	4,4269	4,5707
0.9	4,3984	4,4152	4,4340	4,6279

Tabel 5.3 Pengujian Pc 0.1-0.9 dan Pm 0.5-0.9 pada Fungsi Keanggotaan IMT

Nilai Pc	Nilai Pm				
	0.5	0.6	0.7	0.8	0.9
0.1	3,493	3,6779	3,7134	3,7463	3,8991
0.2	3,2056	3,3840	3,451	3,6751	4,1118
0.3	3,5013	3,6238	3,6112	3,7143	4,3478
0.4	3,8948	3,7931	3,7945	3,868	4,2806
0.5	3,9328	4,0918	4,0413	3,8108	4,4761
0.6	4,0098	3,9439	4,1938	4,1654	4,4340
0.7	4,5826	4,2279	4,2255	4,3523	4,6683
0.8	4,4761	4,3998	4,4761	4,5576	4,7443
0.9	4,5571	4,6572	4,7018	4,7379	4,8991

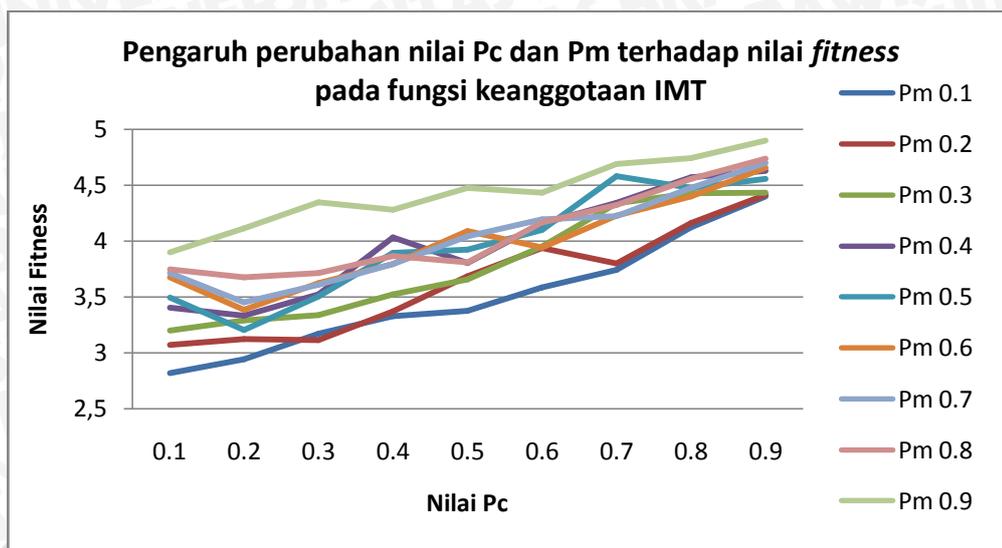
Tabel 5.4 Pengujian Pc 0.1-0.9 dan Pm 0.1-0.4 pada Fungsi Keanggotaan Umur

Nilai Pc	Nilai Pm			
	0.1	0.2	0.3	0.4
0.1	3,6337	3,7649	3,8954	4,0180
0.2	3,7233	3,9386	4,0909	4,1776
0.3	4,0722	4,2400	4,3075	4,2940
0.4	4,1857	4,3725	4,4439	4,6326
0.5	4,3635	4,5670	4,4202	4,6332
0.6	4,5099	4,7299	4,9012	4,9701
0.7	4,4799	4,9182	5,2002	5,1356
0.8	4,6011	5,2670	4,9040	5,4400
0.9	5,2009	5,3422	5,3137	5,5109

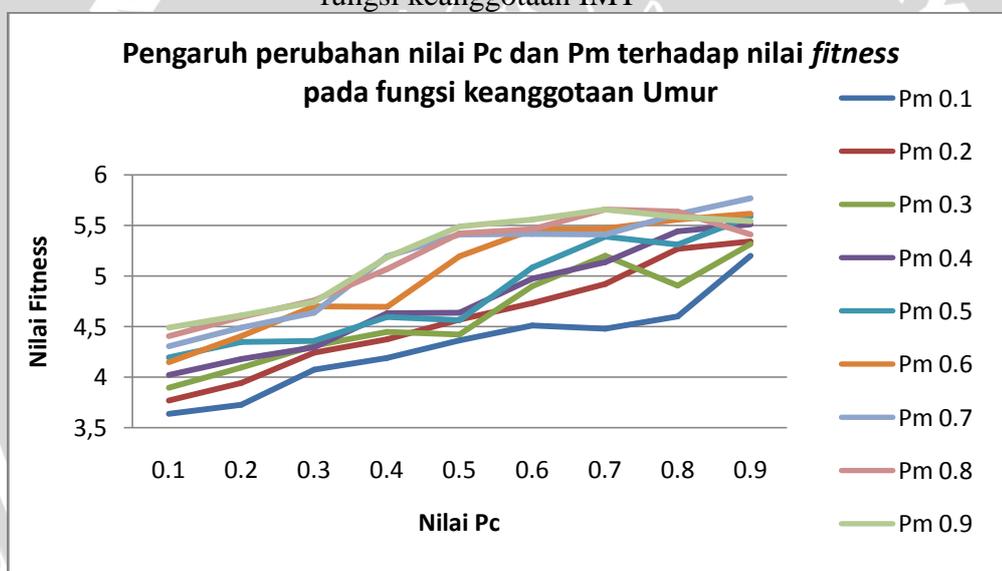
Tabel 5.5 Pengujian Pc 0.1-0.9 dan Pm 0.5-0.9 pada Fungsi Keanggotaan Umur

Nilai Pc	Nilai Pm				
	0.5	0.6	0.7	0.8	0.9
0.1	4,1960	4,1460	4,3014	4,4025	4,4894
0.2	4,3472	4,4019	4,4875	4,5951	4,6068
0.3	4,3588	4,6992	4,6372	4,7543	4,7412
0.4	4,5498	4,6931	5,1945	5,0680	5,1806
0.5	4,5628	5,1918	5,4075	5,4187	5,4884
0.6	5,0823	5,4577	5,4165	5,4625	5,5556
0.7	5,3857	5,4656	5,4087	5,6581	5,6566
0.8	5,3109	5,5552	5,6113	5,6367	5,5844
0.9	5,5874	5,6145	5,7656	5,4072	5,5416

Dari tabel 5.2 dan 5.3 yang merupakan hasil pengujian terhadap fungsi keanggotaan IMT dapat dilihat bahwa nilai *fitness* rata-rata terbesar adalah 4,8991 dari skala 1 sampai 5. Sedangkan pada tabel 5.4 dan 5.5 yang merupakan hasil pengujian terhadap fungsi keanggotaan Umur dapat dilihat bahwa nilai *fitness* rata-rata terbesar adalah 5,7656 dari skala 1 sampai 5. Nilai ini terdapat pada saat nilai Pc 0.9 dan nilai Pm 0.8 untuk fungsi keanggotaan IMT dan Pc 0.9 dan nilai Pm 0.8 untuk fungsi keanggotaan Umur. Adapun grafik pengaruh probabilitas *crossover* dan probabilitas mutasi pada fungsi keanggotaan IMT terhadap nilai *fitness* rata-rata dapat dilihat pada gambar 5.1 dan grafik pengaruh probabilitas *crossover* dan probabilitas mutasi pada fungsi keanggotaan Umur terhadap nilai *fitness* rata-rata dapat dilihat pada gambar 5.2.



Gambar 5.1 Grafik pengujian nilai Pc dan Pm terhadap nilai *fitness* pada fungsi keanggotaan IMT



Gambar 5.2 Grafik pengujian nilai Pc dan Pm terhadap nilai *fitness* pada fungsi keanggotaan Umur

Berdasarkan grafik perbandingan nilai Pc dan nilai Pm dengan nilai *fitness* rata-rata dari gambar 5.1 dan gambar 5.2, dapat dilihat seiring perubahan nilai probabilitas *crossover* dan probabilitas mutasi maka nilai *fitness* rata-rata yang didapat fluktuatif. Nilai *fitness* rata-rata seperti ini, dimungkinkan terjadi karena proses random yang dilakukan pada saat proses *crossover* dan mutasi. Untuk hasil *fitness* yang tertinggi terdapat apabila nilai

Pc dan Pm semakin tinggi angkanya, dan didapat Pc 0,9 dengan Pm 0,9 pada fungsi keanggotaan IMT dan Pc 0,9 dengan Pm 0,7 pada fungsi keanggotaan Umur.

Nilai *fitness* rata-rata yang relatif tinggi tersebut dapat disebabkan karena semakin seringnya proses *crossover* terjadi sehingga dimungkinkan diperoleh variasi individu yang semakin banyak. Semakin besar nilai probabilitas *crossover* maka semakin banyak kemungkinan individu yang disilangkan (*crossover*), tentunya akan semakin banyak individu baru yang diciptakan. Dengan demikian peluang untuk didapatkan individu yang memiliki nilai *fitness* yang tinggi akan semakin besar. Sedangkan untuk probabilitas mutasi, terjadi peningkatan nilai *fitness* rata-rata saat nilai Pm tinggi, karena semakin banyak individu yang ikut terkena mutasi sehingga variasi *fitness* dari individu semakin beragam karena ada beberapa individu yang kemungkinan setelah di mutasi akan menjadi solusi yang lebih baik dari keadaan sebelum di mutasi.

Untuk nilai Pm dan Pc yang rendah, dihasilkan nilai *fitness* rata-rata yang buruk. Hal ini disebabkan sedikitnya variasi individu yang terbentuk, sehingga nilai *fitness* yang dihasilkan cenderung seragam dan semakin sedikit individu yang mengalami perbaikan menuju solusi yang lebih baik dengan *crossover* maupun mutasi.

5.2.2. Pengujian jumlah generasi terhadap nilai *fitness*

Pengujian yang kedua adalah mengetahui pengaruh perubahan jumlah generasi terhadap besar nilai *fitness* maksimum yang dihasilkan. Dalam pengujian ini, digunakan parameter nilai Pc 0.9, Pm 0.9 untuk fungsi keanggotaan IMT dan nilai Pc 0.9, Pm 0.7 untuk fungsi keanggotaan Umur dikarenakan kombinasi ini mendapat nilai *fitness* terbaik, α 0.5 dan jumlah individu 25. Proses pengujian dilakukan dengan mengubah nilai generasi 10, 20, 30, 40,50, 75, 100 dan 150 generasi. Pada tiap pengujian satu jumlah generasi, dilakukan 5 kali percobaan lalu diambil nilai rata-rata *fitness* yang kemudian akan dibandingkan dengan setiap nilai generasi yang lain untuk

dianalisa. Hasil uji coba dapat dilihat pada tabel 5.6 untuk fungsi keanggotaan IMT dan tabel 5.7 untuk fungsi keanggotaan Umur.

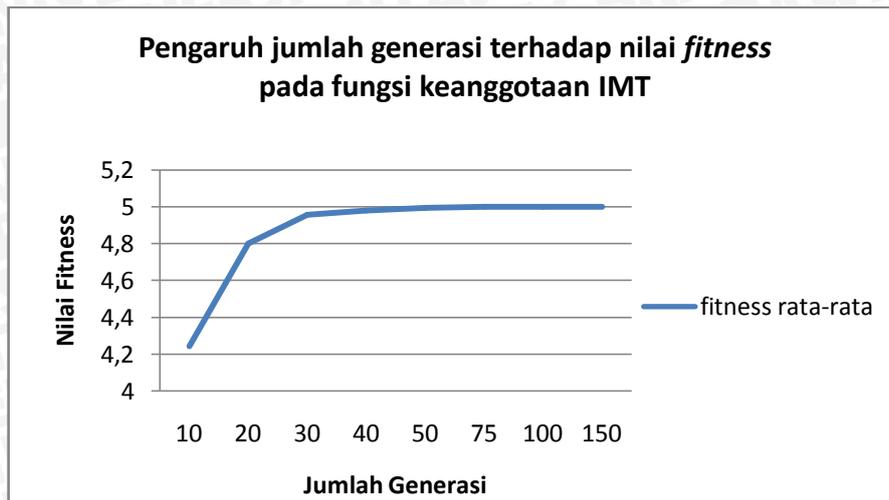
Tabel 5.6 Hasil Uji coba 1-5 pengaruh jumlah generasi terhadap nilai *fitness* pada fungsi keanggotaan IMT

Jumlah Generasi	Percobaan ke-					Fitnes rata-rata
	1	2	3	4	5	
10	4,294	4,332	4,217	4,102	4,278	4,2446
20	4,825	4,783	4,690	4,876	4,831	4,801
30	5,0	4,935	5,0	4,890	4,958	4,956
40	5,0	5,0	4,927	4,976	5,0	4,9806
50	5,0	4,965	5,0	5,0	5,0	4,993
75	5,0	5,0	5,0	5,0	5,0	5,0
100	5,0	5,0	5,0	5,0	5,0	5,0
150	5,0	5,0	5,0	5,0	5,0	5,0

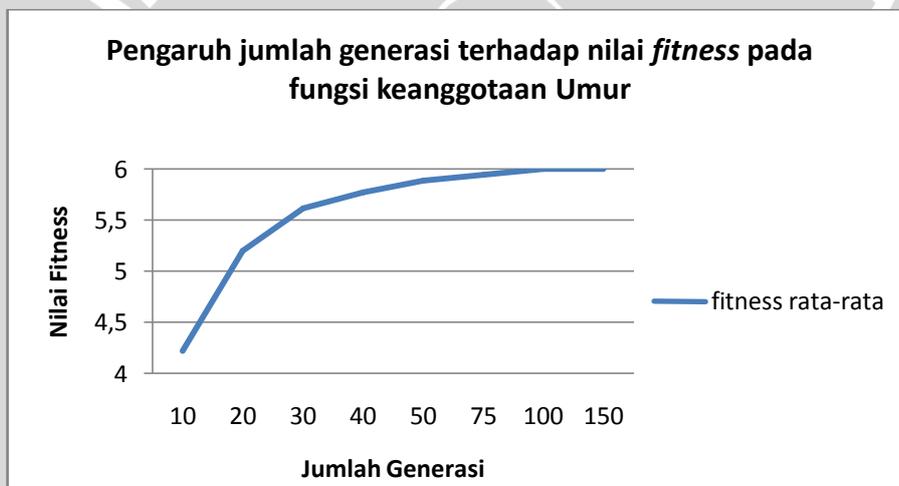
Tabel 5.7 Hasil Uji coba 1-5 pengaruh jumlah generasi terhadap nilai *fitness* pada fungsi keanggotaan Umur

Jumlah Generasi	Percobaan ke-					Fitnes rata-rata
	1	2	3	4	5	
10	4,009	4,533	4,187	4,154	4,209	4,218
20	5,016	5,172	5,388	5,104	5,299	5,198
30	5,676	5,589	5,610	5,687	5,504	5,6132
40	5,720	5,692	5,713	5,824	5,711	5,732
50	5,885	5,841	5,798	5,857	5,952	5,866
75	6,0	5,986	5,992	6,0	5,989	5,993
100	6,0	6,0	6,0	6,0	6,0	6,0
150	6,0	6,0	6,0	6,0	6,0	6,0

Dari tabel 5.6 yang merupakan hasil uji generasi pada fungsi keanggotaan IMT ditunjukkan bahwa nilai fitness rata-rata paling besar yang dihasilkan adalah 5,0 pada saat jumlah generasi sebesar 75 dan generasi selanjutnya memiliki nilai yang telah konvergen, sedangkan pada tabel 5.7 yakni pada fungsi keanggotaan Umur mempunyai nilai fitness paling besar yang dihasilkan adalah 6,0 pada saat jumlah generasi sebesar 100 dan generasi selanjutnya memiliki nilai yang konvergen. Pergerakan nilai fitness terhadap jumlah generasi digambarkan pada gambar grafik 5.3 untuk fungsi keanggotaan IMT dan gambar 5.4 untuk fungsi keanggotaan umur.



Gambar 5.3 Grafik pengujian jumlah generasi terhadap nilai *fitness* pada fungsi keanggotaan IMT



Gambar 5.4 Grafik pengujian jumlah generasi terhadap nilai *fitness* pada fungsi keanggotaan Umur

Pada gambar 5.3 dan gambar 5.4 dapat dilihat bahwa perbandingan jumlah generasi dan nilai *fitness* rata-rata yang terus meningkat. Dalam grafik terlihat nilai *fitness* rata-rata naik terus pada saat generasi ke 10 hingga ke generasi 75, lambat laun nilai *fitness* menjadi konvergen dengan semakin banyaknya jumlah generasi. Pada setiap satu generasi, akan dilakukan beberapa kali proses *crossover* dan mutasi, dimana pada proses ini individu baru akan dihasilkan. Semakin sering proses *crossover* dilakukan maka individu baru

yang terbentuk akan semakin banyak dan variasi *fitness* akan semakin beragam.

Jumlah generasi yang tinggi akan mengakibatkan proses evolusi semakin sering dilakukan sehingga kesempatan untuk terjadinya proses pembentukan individu-individu baru semakin besar pula. Semakin banyak individu yang dihasilkan, memungkinkan semakin banyak variasi nilai *fitness*nya, sehingga peluang untuk mendapatkan nilai *fitness* yang tinggi akan semakin besar.

5.2.3. Pengujian jumlah populasi terhadap nilai *fitness*

Pengujian yang ketiga adalah mengetahui pengaruh perubahan ukuran populasi terhadap besar nilai *fitness* maksimum yang dihasilkan. Dalam pengujian ini, digunakan parameter nilai P_c 0.7, nilai P_m 0.9, α 0.5, jumlah generasi 75 pada fungsi keanggotaan IMT dan P_c 0.7, nilai P_m 0.9, α 0.5, jumlah generasi 100 pada fungsi keanggotaan Umur yang merupakan parameter yang mendapat nilai *fitness* terbaik pada pengujian sebelumnya. Proses pengujian dilakukan dengan mengubah ukuran populasi 10, 20, 30, 40 dan 50 populasi. Pada tiap pengujian populasi, dilakukan 10 kali percobaan lalu diambil nilai rata-rata *fitness* yang kemudian akan dibandingkan dengan setiap nilai *fitness* populasi yang lain untuk dianalisa. Hasil pengujian ditunjukkan pada tabel 5.8 untuk fungsi keanggotaan IMT dan tabel 5.9 untuk fungsi keanggotaan Umur.

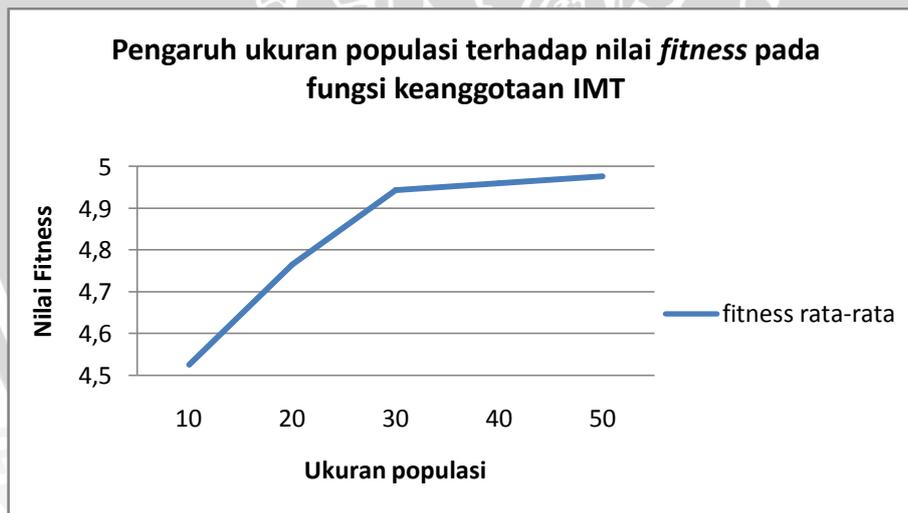
Tabel 5.8 Hasil Uji coba 1-5 pengaruh jumlah populasi terhadap nilai *fitness* pada fungsi keanggotaan IMT

Jumlah Populasi	Percobaan ke-					Fitnes rata-rata
	1	2	3	4	5	
10	4,537	4,368	4,679	4,552	4,491	4,525
20	4,805	4,739	4,866	4,615	4,798	4,764
30	5,0	4,946	4,893	4,878	5,0	4,943
40	4,884	5,0	5,0	4,986	4,927	4,959
50	5,0	4,987	5,0	5,0	4,893	4,976

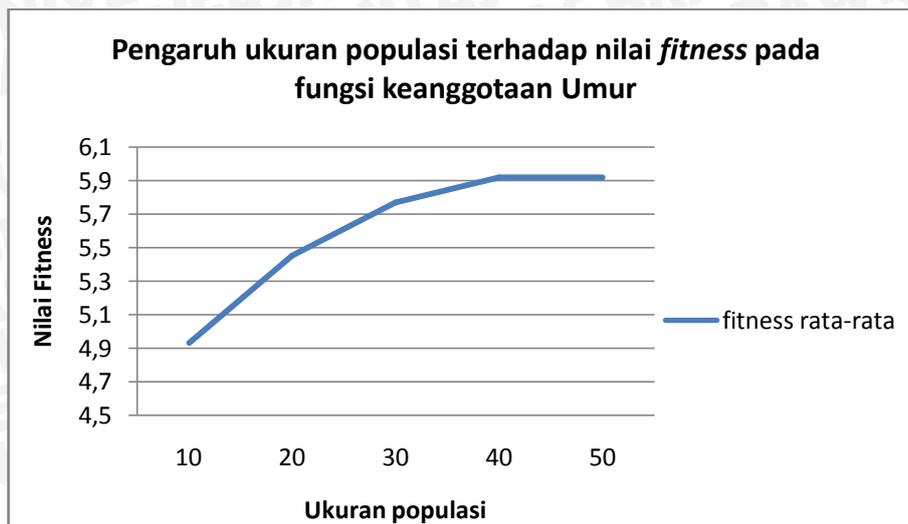
Tabel 5.9 Hasil Uji coba 1-5 pengaruh jumlah populasi terhadap nilai *fitness* pada fungsi keanggotaan Umur

Jumlah Populasi	Percobaan ke-					Fitnes rata-rata
	1	2	3	4	5	
10	4,971	4,863	5,018	4,882	4,927	4,932
20	5,468	5,547	5,492	5,336	5,425	5,453
30	5,884	5,736	5,839	5,761	5,628	5,769
40	5,913	5,927	5,898	5,935	5,917	5,918
50	5,937	5,945	5,889	5,907	5,924	5,920

Dari tabel 5.8 yang merupakan hasil uji generasi pada fungsi keanggotaan IMT ditunjukkan bahwa nilai *fitness* rata-rata paling besar yang dihasilkan adalah 0,400 pada saat jumlah populasi sebesar 50 sedangkan pada tabel 5.9 yakni pada fungsi keanggotaan Umur mempunyai nilai *fitness* paling besar yang dihasilkan adalah 0,500 pada saat jumlah populasi sebesar 30. Pergerakan nilai *fitness* terhadap jumlah generasi digambarkan pada gambar grafik 5.5 untuk fungsi keanggotaan IMT dan gambar 5.4 untuk fungsi keanggotaan Umur.



Gambar 5.5 Grafik pengujian jumlah populasi terhadap nilai *fitness* pada fungsi keanggotaan IMT



Gambar 5.6 Grafik pengujian jumlah populasi terhadap nilai *fitness* pada fungsi keanggotaan Umur

Dari gambar 5.5 dan gambar 5.6, dapat dilihat bahwa perbandingan jumlah populasi terhadap nilai *fitness* rata-rata cenderung naik baik pada fungsi keanggotaan IMT maupun pada fungsi keanggotaan Umur, yaitu setiap penambahan jumlah populasi akan diikuti pula bertambahnya nilai *fitness* rata-rata. Banyaknya ukuran populasi mempengaruhi jumlah individu baru yang dihasilkan. Dengan adanya proses *crossover* dan proses mutasi dalam ukuran populasi yang besar, memungkinkan jumlah individu baru yang dihasilkan juga semakin besar dan beragam. Tentunya hal ini akan berpengaruh pula terhadap variasi nilai *fitness* yang dihasilkan oleh individu-individu baru. Sehingga peluang untuk mendapatkan nilai *fitness* yang tinggi semakin besar.

Dari hasil pengujian parameter pengaruh nilai PC dan Pm, pengujian jumlah generasi dan pengujian ukuran populasi, didapatkan nilai maksimum *fitness* rata-rata $> 4,5$ dengan nilai $P_c = 0,9$, $P_m = 0,9$, jumlah generasi 75 untuk fungsi keanggotaan IMT dan nilai $P_c = 0,9$, $P_m = 0,7$, jumlah generasi 100 untuk fungsi keanggotaan Umur, serta ukuran populasi 50 untuk kedua fungsi keanggotaan.

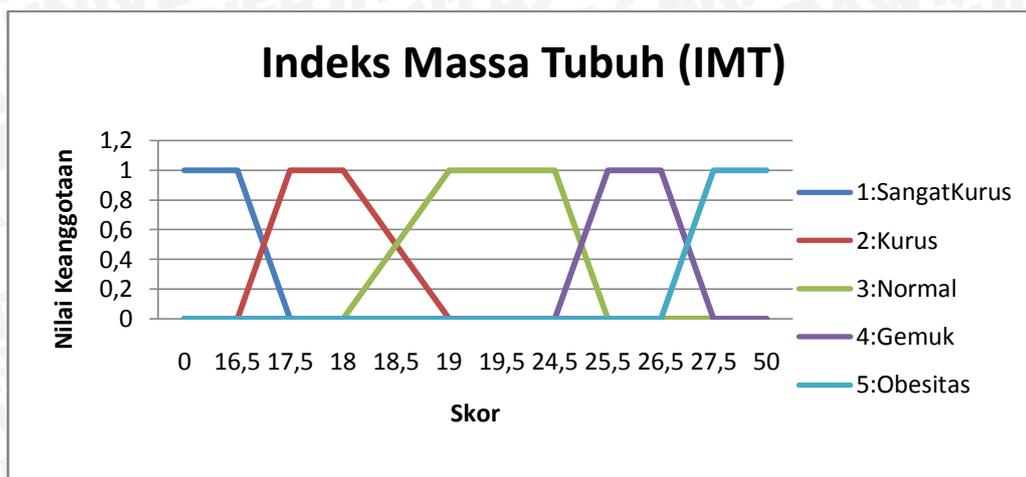
5.3. Pengujian Data Latih

Pengujian yang keempat adalah mengetahui pengaruh perubahan prosentase akurasi ketepatan perhitungan yang dilakukan sistem terhadap perhitungan manual pada permasalahan perhitungan Angka Metabolisme Basal (AMB) dengan menggunakan 40 data latih yang berupa data pasien yang didapat dari penelitian sebelumnya. Dalam pengujian ini, digunakan parameter nilai P_c 0.7, nilai P_m 0.9, α 0.5, jumlah generasi 50 dan jumlah populasi 50 pada fungsi keanggotaan IMT dan P_c 0.7, nilai P_m 0.9, α 0.5, jumlah generasi 50 dan jumlah populasi 50 pada fungsi keanggotaan Umur yang merupakan parameter yang mendapat nilai rata-rata *fitness* terbaik pada pengujian sebelumnya.. Pada pengujian data latih, dilakukan 5 kali percobaan dengan menggunakan parameter yang telah ditentukan di atas, kemudian diambil batas-batas fungsi keanggotaan yang memiliki nilai akurasi pengujian data latih yang paling tinggi. Hasil pengujian ditunjukkan pada tabel 5.10.

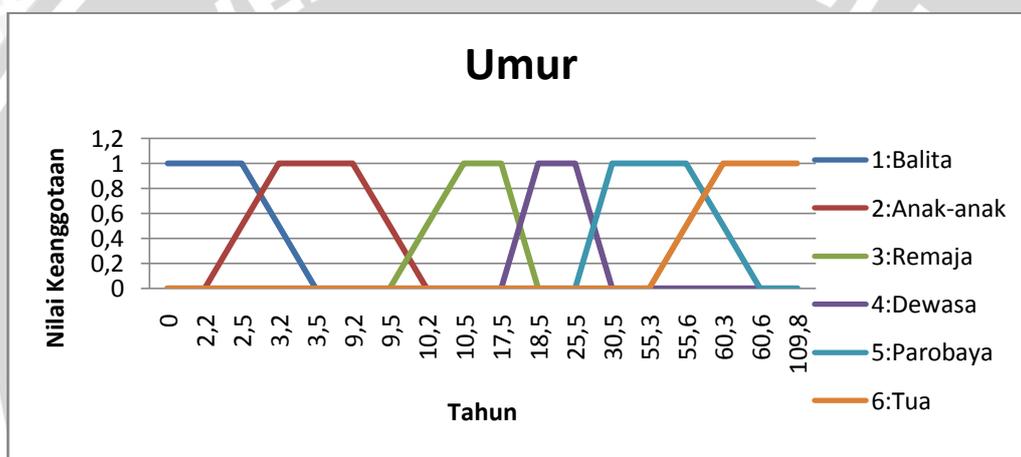
Tabel 5.10 Hasil Uji coba akurasi data latih terhadap parameter Algoritma Genetika dengan nilai rata-rata *fitness* tertinggi

Percobaan ke-	Hasil prosentase akurasi
1	92.5%
2	95%
3	92.5%
4	92.5%
5	95%

Dari tabel 5.10 dapat dilihat bahwa prosentase akurasi pengujian data latih mendapatkan nilai yang baik pada semua percobaan. Hasil akurasi yang baik membuktikan bahwa nilai dari parameter algoritma genetika yang digunakan sudah tepat. Sehingga hasil fungsi keanggotaan yang dipilih adalah pada percobaan ke 2 atau ke 5 dengan hasil fungsi keanggotaan keduanya adalah sama dan hasil uji data latih tertera pada Lampiran 3. Fungsi keanggotaan yang terpilih untuk dilakukan pengujian terhadap data uji yang sesungguhnya ditunjukkan pada Gambar 5.8 dan Gambar 5.9.



Gambar 5.7 Grafik hasil fungsi keanggotaan IMT



Gambar 5.8 Grafik hasil fungsi keanggotaan Umur

5.4. Pengujian Akurasi Data

Pengujian akurasi dilakukan untuk mengetahui performa dari sistem untuk memberikan hasil perhitungan nilai AMB pada seorang pasien. Penggunaan fungsi keanggotaan yang digunakan pada uji akurasi data menggunakan hasil batas-batas fungsi keanggotaan IMT dan Umur yang mempunyai nilai prosentase tertinggi pada pengujian data latih.

Data yang diuji sebanyak 100 sampel data yang selanjutnya akan di bandingkan antara hasil yang didapat dari perhitungan sistem menggunakan metode *Fuzzy Model Sugeno* dengan fungsi keanggotaan yang dihasilkan oleh metode *Algoritma Genetika* dengan perhitungan manual dengan rumus WHO/FAO/UNU. Hasil pengujian akurasi sistem ini ditampilkan 15 data pada

tabel 5.11 dan untuk 100 data keseluruhan terlampir pada halaman Lampiran 4 dalam bentuk tabel.

Tabel 5.11 Hasil pengujian akurasi data uji

No.	Nama	ID Jen Kelamin	Umur	Tinggi	Berat	Jen Aktivitas	Hasil Manual	Hasil Sistem
1	pasien 1	1	84	165	55	3	2582	2582
2	pasien 2	1	84	170	55	3	2582	2582
3	pasien 3	1	84	180	55	3	3632	3632
4	pasien 4	1	84	185	55	3	3632	3632
5	pasien 5	1	53	157	42	3	3919	3919
6	pasien 6	1	53	152	42	3	3919	3919
7	pasien 7	1	53	162	42	3	3919	3919
8	pasien 8	1	53	167	42	3	3919	3919
9	pasien 9	2	39	145	86	1	1670	1670
10	pasien 10	2	39	150	86	1	1670	1670
11	pasien 11	2	39	160	86	1	1670	1670
12	pasien 12	2	39	165	86	1	1670	1670
13	pasien 13	1	85	165	55	1	2029	2029
14	pasien 14	1	85	170	55	1	2029	2029
15	pasien 15	1	85	180	55	1	2854	2767

Di dalam tabel hasil pengujian akurasi sistem, terdapat baris yang berwarna jingga yang merupakan hasil perhitungan AMB yang berbeda dari perhitungan manual dengan rumus WHO/FAO/UNU dengan perhitungan yang dilakukan oleh sistem. Setelah dilakukan pengujian akurasi dengan 100 sampel data pasien dan menghasilkan nilai akurasi sesuai perhitungan berikut :

$$\text{Nilai akurasi} = \frac{\text{Jumlah data akurat}}{\text{jumlah seluruh data}} \times 100\%$$

$$\text{Nilai akurasi} = \frac{94}{100} \times 100\% = 94\%$$

5.5. Analisis Hasil Pengujian

Dari keseluruhan pengujian yang telah dilakukan terhadap sistem yang dilakukan untuk mengoptimasi fungsi keanggotaan *Fuzzy Inference System* Model Sugeno didapat hasil bahwa parameter Algoritma Genetika untuk membentuk fungsi keanggotaan dengan nilai *fitness* yang terbaik yaitu menggunakan probabilitas crossover 0.9, probabilitas mutasi 0.9, jumlah generasi 75 dan jumlah populasi 50 untuk fungsi keanggotaan IMT sedangkan untuk fungsi keanggotaan Umur didapat hasil probabilitas crossover 0.9, probabilitas mutasi 0.7, jumlah generasi 100 dan jumlah populasi 50. Dari

nilai parameter yang didapat membuktikan bahwa semakin besar nilai probabilitas *crossover*, probabilitas mutasi, jumlah generasi dan jumlah populasi, maka akan semakin banyak individu baru yang diciptakan. Dengan demikian peluang untuk didapatkan individu yang memiliki nilai *fitness* yang tinggi akan semakin besar.

Kemudian hasil pengujian akurasi data uji setelah dilakukan optimasi pada fungsi keanggotaan menggunakan Algoritma Genetika yang diuji terhadap 100 data pasien adalah sebesar 94% yang menunjukkan bahwa sistem ini dapat berfungsi dengan baik dan mempunyai peningkatan nilai akurasi sistem sebesar 19% dari penelitian sebelumnya yang mempunyai nilai akurasi 75%. Penyebab naiknya tingkat akurasi sistem ini adalah pada sudah tepatnya penentuan batas-batas fungsi keanggotaan yang dihasilkan setelah dilakukan optimasi menggunakan Algoritma Genetika, karena hasil yang didapat menunjukkan batas-batas fungsi keanggotaan telah sesuai dengan yang ditentukan oleh pakar sedangkan pada penelitian sebelumnya penentuan batas-batas fungsi keanggotaan ditentukan secara manual sehingga hasil yang didapat kurang maksimal karena tidak menggunakan optimasi. Ketidakakurasian sistem sebesar 6% yang disebabkan karena beberapa kemungkinan antara lain kesalahan data dan kesalahan pada saat fuzzyfikasi, dimana batas-batas keanggotaan yang dihasilkan ada yang masih belum sesuai dengan data uji yang dimasukkan.

BAB VI PENUTUP

6.1. Kesimpulan

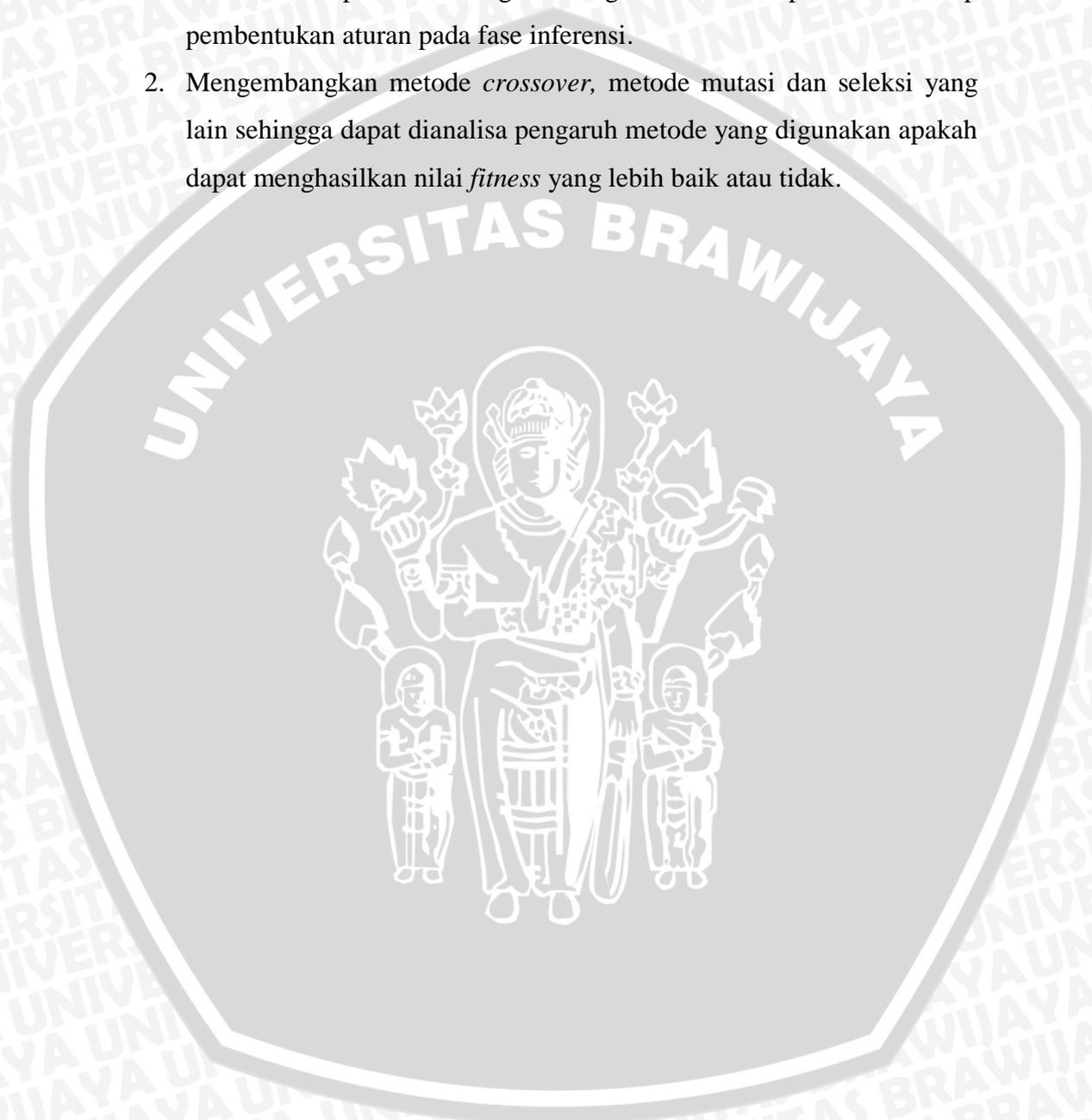
Berdasarkan hasil pengujian dan analisis yang telah dilakukan pada Sistem Penerapan Algoritma Genetika untuk Optimasi Fungsi Keanggotaan pada Perhitungan Angka Metabolisme Basal (AMB) dengan Fuzzy Model Sugeno, maka dapat diambil kesimpulan sebagai berikut :

1. Dari hasil analisa pengujian yang dilakukan, disimpulkan bahwa Algoritma Genetika dapat digunakan untuk melakukan optimasi fungsi keanggotaan pada metode Fuzzy dengan menerapkan *Simple Arithmetic Crossover*, *Random Mutation* dan *Rank Based Fitness Selection* sehingga menghasilkan batas-batas nilai pembentuk fungsi keanggotaan dengan nilai *fitness* yang optimal.
2. Hasil parameter Algoritma Genetika yang terbaik untuk menentukan fungsi keanggotaan IMT diperoleh dari probabilitas *crossover* 0.9 dan probabilitas mutasi 0.9, jumlah generasi 75 dan jumlah populasi 50 sedangkan untuk fungsi keanggotaan Umur diperoleh dari probabilitas *crossover* 0.9 dan probabilitas mutasi 0.7, jumlah generasi 100 dan jumlah populasi 50. Pengaruh parameter probabilitas *crossover* (P_c) menunjukkan bahwa semakin besar nilai P_c dan diikuti dengan nilai probabilitas mutasi (P_m) yang seimbang, maka nilai *fitness* cenderung semakin tinggi (baik). Hal tersebut juga terjadi pada jumlah populasi dan jumlah generasi, seiring dengan bertambahnya jumlah populasi dan/atau jumlah generasi akan cenderung menaikkan nilai *fitness*.
3. Hasil akurasi perhitungan sistem di dalam melakukan perhitungan Angka Metabolisme Basal (AMB) terhadap 100 data uji menunjukkan nilai akurasi sebesar 94% terhadap rumus manual menggunakan FAO/WHO/UNU. Hasil akurasi ini mengalami peningkatan daripada penelitian sebelumnya yang tidak menggunakan optimasi pada fungsi keanggotaannya, yakni sebesar 75% [FEB-12].

6.2. Saran

Saran yang dapat diberikan untuk pengembangan lebih lanjut adalah sebagai berikut :

1. Sistem ini dapat dikembangkan dengan membuat optimasi terhadap pembentukan aturan pada fase inferensi.
2. Mengembangkan metode *crossover*, metode mutasi dan seleksi yang lain sehingga dapat dianalisa pengaruh metode yang digunakan apakah dapat menghasilkan nilai *fitness* yang lebih baik atau tidak.



DAFTAR PUSTAKA

- [ALM-09] Almatsier, Sunita.2009. *Prinsip Dasar Ilmu Gizi*. Jakarta : PT Gramedia PustakaUtama. Jakarta.
- [DKS-08] Departemen Kesehatan Republik Indonesia. 2008. Pedoman Pengendalian Diabetes Melitus dan Penyakit Metabolik. Bakti Husada. Jakarta.
- [ETA-09] Eta S. Berner, Ed.D . 2009. *Clinical Decission Support System : State of the Art*. Department of Health Service Administration. University of Alabama Birmingham
- [FEB-12] Permata Hidayat, Febriyani. 2012. Implementasi Inferensi Fuzzy Model Sugeno pada Perhitungan Angka Metabolisme Basal (AMB) pada Penderita Diabetes. Program Studi Teknik Informatika Universitas Brawijaya. Malang.
- [ITS-13] Lecture Informatika. 2013. Algoritma Genetika . Program Studi Teknik Informatika. Universitas Sepuluh Nopember. Surabaya
- [KUS-04] Kusumadewi. Sri dan Hari Purnomo. 2004. *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu.
- [MRE-06] Kaya, Mehmet And Alhajj, Reda. 2006. Utilizing Genetic Algorithm to Optimize Membership Function for Fuzzy Weighted Association Rules Mining. Firat University Turkey And Calgary University Canada, Springer Science an Business Media inc. Netherlands.
- [ROS-04] Ross, Timothy J. 2004. Fuzzy Logic With Engineering Application. University of New Mexico. USA.
- [SET-03] Setiawan, Kuswara. 2003. *Paradigma Sistem Cerdas*. Malang : Banyumedia Publishing.
- [SUP-01] Supriasa, N, Bakri, B, Fajar, I. 2001. "Penilaian Status Gizi". Penerbit EGC. Jakarta.
- [SUY-05] Suyanto. 2001. *Algoritma Genetika Dalam Matlab*. Yogyakarta : Andi.
- [TMW-99] The MathWorks, Inc. 1999. "Fuzzy Logic Toolbo User's Huide".

[VIT-05] VitaHealth. 2005. Informasi Lengkap Diabetes untuk Penderita dan Keluarga. Penerbit GramediaPustaka Utama. Jakarta

[WHO-13] World Health Organization (WHO). 2013. Diabetes. <http://www.who.int/mediacentre/factsheets/fs312/en/> (diakses pada 28 Februari 2014)



LAMPIRAN 1

TABEL SKENARIO USE CASE

1. *Use case* Login

Pada *use case* login, akan dijelaskan secara mendetail tentang proses login yang dilakukan oleh *User*. Skenario *use case* login dijelaskan pada Tabel L.1.

Tabel L.1 *Use Case* Login

<i>Use case</i>	Login
Aktor	<i>User</i>
Tujuan	Menyeleksi akun yang sah untuk masuk ke dalam sistem.
Deskripsi	<i>Use case</i> ini menjelaskan tentang proses login atau masuk oleh admin ke dalam sistem.
Kondisi Awal	<i>User</i> sudah memiliki akun dalam sistem.
Kondisi Akhir	<i>User</i> bisa masuk ke dalam sistem sehingga dapat mengakses halaman admin.
Skenario : Login	
Aksi dari Aktor	Tanggapan dari Sistem
Aktor memilih menu login	Sistem menampilkan halaman login
Aktor memasukkan username dan password pada menu login	Sistem melakukan verifikasi terhadap username dan password, jika sesuai dengan database maka akan masuk ke halaman admin.

2. *Use case* Membentuk Fungsi Keanggotaan

Pada *Use case* ini akan dijelaskan mengenai bagaimana *user* memasukkan nilai probabilitas *crossover*, probabilitas mutasi, jumlah populasi dan jumlah generasi ke dalam sistem. Nilai masukan tersebut bertujuan untuk membuat batas-batas fungsi keanggotaan menggunakan Algoritma Genetika. Skenario *use case* membentuk fungsi keanggotaan dijelaskan pada Tabel L.2.

Tabel L.2 *Use Case* Membentuk Fungsi Keanggotaan

<i>Use case</i>	Membentuk Fungsi Keanggotaan
Aktor	<i>User</i>
Tujuan	Untuk memasukkan nilai probabilitas, nilai probabilitas <i>crossover</i> , jumlah populasi dan jumlah generasi ke dalam sistem.
Deskripsi	<i>Use case</i> ini menjelaskan tentang bagaimana <i>user</i> memasukkan nilai probabilitas mutasi, nilai probabilitas <i>crossover</i> , jumlah populasi dan jumlah generasi ke dalam sistem. Nilai probabilitas dan jumlah generasi digunakan oleh sistem untuk membentuk batas-batas fungsi keanggotaan dengan Algoritma Genetika.
Kondisi Awal	<i>User</i> harus login ke dalam sistem terlebih dahulu.
Kondisi Akhir	Menghasilkan batas-batas fungsi keanggotaan.
Skenario : Membentuk Fungsi Keanggotaan	
Aksi dari Aktor	Tanggapan dari Sistem
Aktor memasukkan nilai probabilitas <i>crossover</i> , nilai probabilitas mutasi, jumlah populasi dan jumlah generasi	Sistem akan melakukan pengolahan data untuk menghasilkan batas-batas fungsi keanggotaan dengan metode Algoritma Genetika.

3. *Use case* Menguji Akurasi Sistem

Pada *Use case* ini akan dijelaskan mengenai hubungan antara *user* dengan sistem dalam kaitannya pengujian terhadap hasil prosentase akurasi hasil perhitungan AMB dengan Metode Optimasi Fungsi Keanggotaan Fuzzy Model Sugeno dengan Algoritma Genetika terhadap hasil perhitungan manual dengan rumus WHO/FAO/UNU. Skenario *use case* menguji akurasi sistem dijelaskan pada Tabel L.4.

Tabel L.3 *Use Case* Menguji Akurasi Sistem

<i>Use case</i>	Menguji Akurasi Sistem
Aktor	<i>User</i>
Tujuan	Untuk mengetahui hasil akurasi data hasil perhitungan AMB metode manual dengan metode sistem.
Deskripsi	<i>Use case</i> ini menjelaskan tentang bagaimana <i>user</i> mengetahui hasil pengujian terhadap prosentase akurasi hasil perhitungan AMB dengan Metode Optimasi Fungsi Keanggotaan Fuzzy Model Sugeno dengan Algoritma Genetika terhadap hasil perhitungan manual dengan rumus WHO/FAO/UNU.
Kondisi Awal	<i>User</i> telah selesai dalam melakukan proses pemebentukan fungsi keanggotaan.
Kondisi Akhir	Sistem akan menampilkan hasil nilai AMB perhitungan manual dan perhitungan yang dilakukan dengan sistem serta menampilkan prosentase akurasinya.
Skenario : Menguji Akurasi Sistem	
Aksi dari Aktor	Tanggapan dari Sistem
Setelah aktor memasukkan semua nilai yang dibutuhkan, selanjutnya aktor menekan tombol proses.	Sistem akan menampilkan hasil dari perhitungan nilai AMB.

4. *Use case* Menghitung Prediksi AMB Pasien

Pada *Use case* ini akan dijelaskan mengenai peran *user* terhadap sistem dalam kaitannya dengan melakukan prediksi perhitungan Angka Metabolisme Basal (AMB) terhadap seorang pasien dengan menggunakan metode *Fuzzy Inference System* model *Sugeno*. Skenario *use case* menghitung prediksi AMB pasien dijelaskan pada Tabel L.3.

Tabel L.4 Use case Menghitung Prediksi AMB Pasien

<i>Use case</i>	Menghitung Prediksi AMB Pasien
Aktor	<i>User</i>
Tujuan	Melakukan prediksi nilai AMB seorang pasien sesuai dengan parameter yang dimasukkan <i>user</i> .
Deskripsi	<i>Use case</i> ini menjelaskan tentang proses memasukkan data pasien oleh <i>user</i> ke dalam sistem yang selanjutnya dilakukan prediksi perhitungan nilai AMB pasien tersebut menggunakan metode Fuzzy Model Sugeno untuk menghasilkan nilai AMB.
Kondisi Awal	<i>user</i> harus sudah melakukan proses pembentukan fungsi keanggotaan dengan Algoritma Genetika
Kondisi Akhir	Sistem menampilkan nilai AMB dari pasien.
Skenario : Menghitung Prediksi AMB Pasien	
Aksi dari Aktor	Tanggapan dari Sistem
Aktor memilih <i>form</i> input data pasien	Sistem menampilkan halaman input data pasien
Aktor memasukkan data pasien.	Sistem melakukan pemrosesan data yang dimasukkan <i>user</i> kemudian sistem akan melakukan pemrosesan data dengan metode Fuzzy model Sugeno untuk menghasilkan nilai AMB pasien.

5. *Use case* Logout

Pada *use case* logout, akan dijelaskan secara mendetail tentang proses logout yang dilakukan oleh *User*. Skenario *use case* logout dijelaskan pada Tabel L.5.

Tabel L.5 Use Case Logout

<i>Use case</i>	Logout
Aktor	<i>User</i>
Tujuan	<i>User keluar dari aplikasi.</i>
Deskripsi	<i>Use case ini menjelaskan tentang proses logout atau keluar oleh user dan kembali pada form Login.</i>
Kondisi Awal	User telah masuk pada <i>interface</i> halaman utama.
Kondisi Akhir	User berada pada halaman Login.
	Skenario : Logout
Aksi dari Aktor	Tanggapan dari Sistem
Aktor menekan tombol Logout	Sistem menampilkan halaman login.



LAMPIRAN 2

TABEL ATURAN INFERENSI MODEL SUGENO

Rule	IMT	Umur	Jen_Kelamin	Jen_Aktivitas	Perhitungan AMB
1	Sangat Kurus	Balita	Perempuan	Sangat Ringan	$((61*bb)-51+500)*1.3$
2	Sangat Kurus	Balita	Perempuan	Ringan	$((61*bb)-51+500)*1.55$
3	Sangat Kurus	Balita	Perempuan	Sedang	$((61*bb)-51+500)*1.7$
4	Sangat Kurus	Balita	Perempuan	Berat	$((61*bb)-51+500)*2$
5	Sangat Kurus	Anak-anak	Perempuan	Sangat Ringan	$((22.5*bb)+499+500)*1.3$
6	Sangat Kurus	Anak-anak	Perempuan	Ringan	$((22.5*bb)+499+500)*1.55$
7	Sangat Kurus	Anak-anak	Perempuan	Sedang	$((22.5*bb)+499+500)*1.7$
8	Sangat Kurus	Anak-anak	Perempuan	Berat	$((22.5*bb)+499+500)*2$
9	Sangat Kurus	Remaja	Perempuan	Sangat Ringan	$((12.2*bb)+746+500)*1.3$
10	Sangat Kurus	Remaja	Perempuan	Ringan	$((12.2*bb)+746+500)*1.55$
11	Sangat Kurus	Remaja	Perempuan	Sedang	$((12.2*bb)+746+500)*1.7$
12	Sangat Kurus	Remaja	Perempuan	Berat	$((12.2*bb)+746+500)*2$
13	Sangat Kurus	Dewasa	Perempuan	Sangat Ringan	$((14.7*bb)+496+500)*1.3$
14	Sangat Kurus	Dewasa	Perempuan	Ringan	$((14.7*bb)+496+500)*1.55$
15	Sangat Kurus	Dewasa	Perempuan	Sedang	$((14.7*bb)+496+500)*1.7$
16	Sangat Kurus	Dewasa	Perempuan	Berat	$((14.7*bb)+496+500)*2$
17	Sangat Kurus	Parobaya	Perempuan	Sangat Ringan	$((8.7*bb)+829+500)*1.3$
18	Sangat Kurus	Parobaya	Perempuan	Ringan	$((8.7*bb)+829+500)*1.55$
19	Sangat Kurus	Parobaya	Perempuan	Sedang	$((8.7*bb)+829+500)*1.7$
20	Sangat Kurus	Parobaya	Perempuan	Berat	$((8.7*bb)+829+500)*2$
21	Sangat Kurus	Tua	Perempuan	Sangat Ringan	$((10.5*bb)+596+500)*1.3$
22	Sangat Kurus	Tua	Perempuan	Ringan	$((10.5*bb)+596+500)*1.55$
23	Sangat Kurus	Tua	Perempuan	Sedang	$((10.5*bb)+596+500)*1.7$
24	Sangat Kurus	Tua	Perempuan	Berat	$((10.5*bb)+596+500)*2$
25	Sangat Kurus	Balita	Laki-laki	Sangat Ringan	$((60.9*bb)-54+500)*1.3$
26	Sangat Kurus	Balita	Laki-laki	Ringan	$((60.9*bb)-54+500)*1.65$
27	Sangat Kurus	Balita	Laki-laki	Sedang	$((60.9*bb)-54+500)*1.76$
28	Sangat Kurus	Balita	Laki-laki	Berat	$((60.9*bb)-54+500)*2.1$
29	Sangat Kurus	Anak-anak	Laki-laki	Sangat Ringan	$((22.7*bb)+495+500)*1.3$
30	Sangat Kurus	Anak-anak	Laki-laki	Ringan	$((22.7*bb)+495+500)*1.65$
31	Sangat Kurus	Anak-anak	Laki-laki	Sedang	$((22.7*bb)+495+500)*1.76$
32	Sangat Kurus	Anak-anak	Laki-laki	Berat	$((22.7*bb)+495+500)*2.1$
33	Sangat Kurus	Remaja	Laki-laki	Sangat Ringan	$((17.5*bb)+651+500)*1.3$
34	Sangat Kurus	Remaja	Laki-laki	Ringan	$((17.5*bb)+651+500)*1.65$
35	Sangat Kurus	Remaja	Laki-laki	Sedang	$((17.5*bb)+651+500)*1.76$
36	Sangat Kurus	Remaja	Laki-laki	Berat	$((17.5*bb)+651+500)*2.1$
37	Sangat Kurus	Dewasa	Laki-laki	Sangat Ringan	$((15.3*bb)+679+500)*1.3$
38	Sangat Kurus	Dewasa	Laki-laki	Ringan	$((15.3*bb)+679+500)*1.65$
39	Sangat Kurus	Dewasa	Laki-laki	Sedang	$((15.3*bb)+679+500)*1.76$
40	Sangat Kurus	Dewasa	Laki-laki	Berat	$((15.3*bb)+679+500)*2.1$

Rule	IMT	Umur	Jen_Kelamin	Jen_Aktivitas	Perhitungan AMB
41	Sangat Kurus	Parobaya	Laki-laki	Sangat Ringan	$((11.6*bb)+879+500)*1.3$
42	Sangat Kurus	Parobaya	Laki-laki	Ringan	$((11.6*bb)+879+500)*1.65$
43	Sangat Kurus	Parobaya	Laki-laki	Sedang	$((11.6*bb)+879+500)*1.76$
44	Sangat Kurus	Parobaya	Laki-laki	Berat	$((11.6*bb)+879+500)*2.1$
45	Sangat Kurus	Tua	Laki-laki	Sangat Ringan	$((13.5*bb)+487+500)*1.3$
46	Sangat Kurus	Tua	Laki-laki	Ringan	$((13.5*bb)+487+500)*1.65$
47	Sangat Kurus	Tua	Laki-laki	Sedang	$((13.5*bb)+487+500)*1.76$
48	Sangat Kurus	Tua	Laki-laki	Berat	$((13.5*bb)+487+500)*2.1$
49	Kurus	Balita	Perempuan	Sangat Ringan	$((61*bb)-51+500)*1.3$
50	Kurus	Balita	Perempuan	Ringan	$((61*bb)-51+500)*1.55$
51	Kurus	Balita	Perempuan	Sedang	$((61*bb)-51+500)*1.7$
52	Kurus	Balita	Perempuan	Berat	$((61*bb)-51+500)*2$
53	Kurus	Anak-anak	Perempuan	Sangat Ringan	$((22.5*bb)+499+500)*1.3$
54	Kurus	Anak-anak	Perempuan	Ringan	$((22.5*bb)+499+500)*1.55$
55	Kurus	Anak-anak	Perempuan	Sedang	$((22.5*bb)+499+500)*1.7$
56	Kurus	Anak-anak	Perempuan	Berat	$((22.5*bb)+499+500)*2$
57	Kurus	Remaja	Perempuan	Sangat Ringan	$((12.2*bb)+746+500)*1.3$
58	Kurus	Remaja	Perempuan	Ringan	$((12.2*bb)+746+500)*1.55$
59	Kurus	Remaja	Perempuan	Sedang	$((12.2*bb)+746+500)*1.7$
60	Kurus	Remaja	Perempuan	Berat	$((12.2*bb)+746+500)*2$
61	Kurus	Dewasa	Perempuan	Sangat Ringan	$((14.7*bb)+496+500)*1.3$
62	Kurus	Dewasa	Perempuan	Ringan	$((14.7*bb)+496+500)*1.55$
63	Kurus	Dewasa	Perempuan	Sedang	$((14.7*bb)+496+500)*1.7$
64	Kurus	Dewasa	Perempuan	Berat	$((14.7*bb)+496+500)*2$
65	Kurus	Parobaya	Perempuan	Sangat Ringan	$((8.7*bb)+829+500)*1.3$
66	Kurus	Parobaya	Perempuan	Ringan	$((8.7*bb)+829+500)*1.55$
67	Kurus	Parobaya	Perempuan	Sedang	$((8.7*bb)+829+500)*1.7$
68	Kurus	Parobaya	Perempuan	Berat	$((8.7*bb)+829+500)*2$
69	Kurus	Tua	Perempuan	Sangat Ringan	$((10.5*bb)+596+500)*1.3$
70	Kurus	Tua	Perempuan	Ringan	$((10.5*bb)+596+500)*1.55$
71	Kurus	Tua	Perempuan	Sedang	$((10.5*bb)+596+500)*1.7$
72	Kurus	Tua	Perempuan	Berat	$((10.5*bb)+596+500)*2$
73	Kurus	Balita	Laki-laki	Sangat Ringan	$((60.9*bb)-54+500)*1.3$
74	Kurus	Balita	Laki-laki	Ringan	$((60.9*bb)-54+500)*1.65$
75	Kurus	Balita	Laki-laki	Sedang	$((60.9*bb)-54+500)*1.76$
76	Kurus	Balita	Laki-laki	Berat	$((60.9*bb)-54+500)*2.1$
77	Kurus	Anak-anak	Laki-laki	Sangat Ringan	$((22.7*bb)+495+500)*1.3$
78	Kurus	Anak-anak	Laki-laki	Ringan	$((22.7*bb)+495+500)*1.65$
79	Kurus	Anak-anak	Laki-laki	Sedang	$((22.7*bb)+495+500)*1.76$
80	Kurus	Anak-anak	Laki-laki	Berat	$((22.7*bb)+495+500)*2.1$

Rule	IMT	Umur	Jen_Kelamin	Jen_Aktivitas	Perhitungan AMB
81	Kurus	Remaja	Laki-laki	Sangat Ringan	$((17.5*bb)+651+500)*1.3$
82	Kurus	Remaja	Laki-laki	Ringan	$((17.5*bb)+651+500)*1.65$
83	Kurus	Remaja	Laki-laki	Sedang	$((17.5*bb)+651+500)*1.76$
84	Kurus	Remaja	Laki-laki	Berat	$((17.5*bb)+651+500)*2.1$
85	Kurus	Dewasa	Laki-laki	Sangat Ringan	$((15.3*bb)+679+500)*1.3$
86	Kurus	Dewasa	Laki-laki	Ringan	$((15.3*bb)+679+500)*1.65$
87	Kurus	Dewasa	Laki-laki	Sedang	$((15.3*bb)+679+500)*1.76$
88	Kurus	Dewasa	Laki-laki	Berat	$((15.3*bb)+679+500)*2.1$
89	Kurus	Parobaya	Laki-laki	Sangat Ringan	$((11.6*bb)+879+500)*1.3$
90	Kurus	Parobaya	Laki-laki	Ringan	$((11.6*bb)+879+500)*1.65$
91	Kurus	Parobaya	Laki-laki	Sedang	$((11.6*bb)+879+500)*1.76$
92	Kurus	Parobaya	Laki-laki	Berat	$((11.6*bb)+879+500)*2.1$
93	Kurus	Tua	Laki-laki	Sangat Ringan	$((13.5*bb)+487+500)*1.3$
94	Kurus	Tua	Laki-laki	Ringan	$((13.5*bb)+487+500)*1.65$
95	Kurus	Tua	Laki-laki	Sedang	$((13.5*bb)+487+500)*1.76$
96	Kurus	Tua	Laki-laki	Berat	$((13.5*bb)+487+500)*2.1$
97	Normal	Balita	Perempuan	Sangat Ringan	$((61*bb)-51)*1.3$
98	Normal	Balita	Perempuan	Ringan	$((61*bb)-51)*1.55$
99	Normal	Balita	Perempuan	Sedang	$((61*bb)-51)*1.7$
100	Normal	Balita	Perempuan	Berat	$((61*bb)-51)*2$
101	Normal	Anak-anak	Perempuan	Sangat Ringan	$((22.5*bb)+499)*1.3$
102	Normal	Anak-anak	Perempuan	Ringan	$((22.5*bb)+499)*1.55$
103	Normal	Anak-anak	Perempuan	Sedang	$((22.5*bb)+499)*1.7$
104	Normal	Anak-anak	Perempuan	Berat	$((22.5*bb)+499)*2$
105	Normal	Remaja	Perempuan	Sangat Ringan	$((12.2*bb)+746)*1.3$
106	Normal	Remaja	Perempuan	Ringan	$((12.2*bb)+746)*1.55$
107	Normal	Remaja	Perempuan	Sedang	$((12.2*bb)+746)*1.7$
108	Normal	Remaja	Perempuan	Berat	$((12.2*bb)+746)*2$
109	Normal	Dewasa	Perempuan	Sangat Ringan	$((14.7*bb)+496)*1.3$
110	Normal	Dewasa	Perempuan	Ringan	$((14.7*bb)+496)*1.55$
111	Normal	Dewasa	Perempuan	Sedang	$((14.7*bb)+496)*1.7$
112	Normal	Dewasa	Perempuan	Berat	$((14.7*bb)+496)*2$
113	Normal	Parobaya	Perempuan	Sangat Ringan	$((8.7*bb)+829)*1.3$
114	Normal	Parobaya	Perempuan	Ringan	$((8.7*bb)+829)*1.55$
115	Normal	Parobaya	Perempuan	Sedang	$((8.7*bb)+829)*1.7$
116	Normal	Parobaya	Perempuan	Berat	$((8.7*bb)+829)*2$
117	Normal	Tua	Perempuan	Sangat Ringan	$((10.5*bb)+596)*1.3$
118	Normal	Tua	Perempuan	Ringan	$((10.5*bb)+596)*1.55$
119	Normal	Tua	Perempuan	Sedang	$((10.5*bb)+596)*1.7$
120	Normal	Tua	Perempuan	Berat	$((10.5*bb)+596)*2$

Rule	IMT	Umur	Jen_Kelamin	Jen_Aktivitas	Perhitungan AMB
121	Normal	Balita	Laki-laki	Sangat Ringan	$((60.9*bb)-54)*1.3$
122	Normal	Balita	Laki-laki	Ringan	$((60.9*bb)-54)*1.65$
123	Normal	Balita	Laki-laki	Sedang	$((60.9*bb)-54)*1.76$
124	Normal	Balita	Laki-laki	Berat	$((60.9*bb)-54)*2.1$
125	Normal	Anak-anak	Laki-laki	Sangat Ringan	$((22.7*bb)+495)*1.3$
126	Normal	Anak-anak	Laki-laki	Ringan	$((22.7*bb)+495)*1.65$
127	Normal	Anak-anak	Laki-laki	Sedang	$((22.7*bb)+495)*1.76$
128	Normal	Anak-anak	Laki-laki	Berat	$((22.7*bb)+495)*2.1$
129	Normal	Remaja	Laki-laki	Sangat Ringan	$((17.5*bb)+651)*1.3$
130	Normal	Remaja	Laki-laki	Ringan	$((17.5*bb)+651)*1.65$
131	Normal	Remaja	Laki-laki	Sedang	$((17.5*bb)+651)*1.76$
132	Normal	Remaja	Laki-laki	Berat	$((17.5*bb)+651)*2.1$
133	Normal	Dewasa	Laki-laki	Sangat Ringan	$((15.3*bb)+679)*1.3$
134	Normal	Dewasa	Laki-laki	Ringan	$((15.3*bb)+679)*1.65$
135	Normal	Dewasa	Laki-laki	Sedang	$((15.3*bb)+679)*1.76$
136	Normal	Dewasa	Laki-laki	Berat	$((15.3*bb)+679)*2.1$
137	Normal	Parobaya	Laki-laki	Sangat Ringan	$((11.6*bb)+879)*1.3$
138	Normal	Parobaya	Laki-laki	Ringan	$((11.6*bb)+879)*1.65$
139	Normal	Parobaya	Laki-laki	Sedang	$((11.6*bb)+879)*1.76$
140	Normal	Parobaya	Laki-laki	Berat	$((11.6*bb)+879)*2.1$
141	Normal	Tua	Laki-laki	Sangat Ringan	$((13.5*bb)+487)*1.3$
142	Normal	Tua	Laki-laki	Ringan	$((13.5*bb)+487)*1.65$
143	Normal	Tua	Laki-laki	Sedang	$((13.5*bb)+487)*1.76$
144	Normal	Tua	Laki-laki	Berat	$((13.5*bb)+487)*2.1$
145	Gemuk	Balita	Perempuan	Sangat Ringan	$((61*bb)-51-500)*1.3$
146	Gemuk	Balita	Perempuan	Ringan	$((61*bb)-51-500)*1.55$
147	Gemuk	Balita	Perempuan	Sedang	$((61*bb)-51-500)*1.7$
148	Gemuk	Balita	Perempuan	Berat	$((61*bb)-51-500)*2$
149	Gemuk	Anak-anak	Perempuan	Sangat Ringan	$((22.5*bb)+499-500)*1.3$
150	Gemuk	Anak-anak	Perempuan	Ringan	$((22.5*bb)+499-500)*1.55$
151	Gemuk	Anak-anak	Perempuan	Sedang	$((22.5*bb)+499-500)*1.7$
152	Gemuk	Anak-anak	Perempuan	Berat	$((22.5*bb)+499-500)*2$
153	Gemuk	Remaja	Perempuan	Sangat Ringan	$((12.2*bb)+746-500)*1.3$
154	Gemuk	Remaja	Perempuan	Ringan	$((12.2*bb)+746-500)*1.55$
155	Gemuk	Remaja	Perempuan	Sedang	$((12.2*bb)+746-500)*1.7$
156	Gemuk	Remaja	Perempuan	Berat	$((12.2*bb)+746-500)*2$
157	Gemuk	Dewasa	Perempuan	Sangat Ringan	$((14.7*bb)+496-500)*1.3$
158	Gemuk	Dewasa	Perempuan	Ringan	$((14.7*bb)+496-500)*1.55$
159	Gemuk	Dewasa	Perempuan	Sedang	$((14.7*bb)+496-500)*1.7$
160	Gemuk	Dewasa	Perempuan	Berat	$((14.7*bb)+496-500)*2$

Rule	IMT	Umur	Jen_Kelamin	Jen_Aktivitas	Perhitungan AMB
161	Gemuk	Parobaya	Perempuan	Sangat Ringan	$((8.7*bb)+829-500)*1.3$
162	Gemuk	Parobaya	Perempuan	Ringan	$((8.7*bb)+829-500)*1.55$
163	Gemuk	Parobaya	Perempuan	Sedang	$((8.7*bb)+829-500)*1.7$
164	Gemuk	Parobaya	Perempuan	Berat	$((8.7*bb)+829-500)*2$
165	Gemuk	Tua	Perempuan	Sangat Ringan	$((10.5*bb)+596-500)*1.3$
166	Gemuk	Tua	Perempuan	Ringan	$((10.5*bb)+596-500)*1.55$
167	Gemuk	Tua	Perempuan	Sedang	$((10.5*bb)+596-500)*1.7$
168	Gemuk	Tua	Perempuan	Berat	$((10.5*bb)+596-500)*2$
169	Gemuk	Balita	Laki-laki	Sangat Ringan	$((60.9*bb)-54-500)*1.3$
170	Gemuk	Balita	Laki-laki	Ringan	$((60.9*bb)-54-500)*1.65$
171	Gemuk	Balita	Laki-laki	Sedang	$((60.9*bb)-54-500)*1.76$
172	Gemuk	Balita	Laki-laki	Berat	$((60.9*bb)-54-500)*2.1$
173	Gemuk	Anak-anak	Laki-laki	Sangat Ringan	$((22.7*bb)+495-500)*1.3$
174	Gemuk	Anak-anak	Laki-laki	Ringan	$((22.7*bb)+495-500)*1.65$
175	Gemuk	Anak-anak	Laki-laki	Sedang	$((22.7*bb)+495-500)*1.76$
176	Gemuk	Anak-anak	Laki-laki	Berat	$((22.7*bb)+495-500)*2.1$
177	Gemuk	Remaja	Laki-laki	Sangat Ringan	$((17.5*bb)+651-500)*1.3$
178	Gemuk	Remaja	Laki-laki	Ringan	$((17.5*bb)+651-500)*1.65$
179	Gemuk	Remaja	Laki-laki	Sedang	$((17.5*bb)+651-500)*1.76$
180	Gemuk	Remaja	Laki-laki	Berat	$((17.5*bb)+651-500)*2.1$
181	Gemuk	Dewasa	Laki-laki	Sangat Ringan	$((15.3*bb)+679-500)*1.3$
182	Gemuk	Dewasa	Laki-laki	Ringan	$((15.3*bb)+679-500)*1.65$
183	Gemuk	Dewasa	Laki-laki	Sedang	$((15.3*bb)+679-500)*1.76$
184	Gemuk	Dewasa	Laki-laki	Berat	$((15.3*bb)+679-500)*2.1$
185	Gemuk	Parobaya	Laki-laki	Sangat Ringan	$((11.6*bb)+879-500)*1.3$
186	Gemuk	Parobaya	Laki-laki	Ringan	$((11.6*bb)+879-500)*1.65$
187	Gemuk	Parobaya	Laki-laki	Sedang	$((11.6*bb)+879-500)*1.76$
188	Gemuk	Parobaya	Laki-laki	Berat	$((11.6*bb)+879-500)*2.1$
189	Gemuk	Tua	Laki-laki	Sangat Ringan	$((13.5*bb)+487-500)*1.3$
190	Gemuk	Tua	Laki-laki	Ringan	$((13.5*bb)+487-500)*1.65$
191	Gemuk	Tua	Laki-laki	Sedang	$((13.5*bb)+487-500)*1.76$
192	Gemuk	Tua	Laki-laki	Berat	$((13.5*bb)+487-500)*2.1$
193	Obesitas	Balita	Perempuan	Sangat Ringan	$((61*bb)-51-500)*1.3$
194	Obesitas	Balita	Perempuan	Ringan	$((61*bb)-51-500)*1.55$
195	Obesitas	Balita	Perempuan	Sedang	$((61*bb)-51-500)*1.7$
196	Obesitas	Balita	Perempuan	Berat	$((61*bb)-51-500)*2$
197	Obesitas	Anak-anak	Perempuan	Sangat Ringan	$((22.5*bb)+499-500)*1.3$
198	Obesitas	Anak-anak	Perempuan	Ringan	$((22.5*bb)+499-500)*1.55$
199	Obesitas	Anak-anak	Perempuan	Sedang	$((22.5*bb)+499-500)*1.7$
200	Obesitas	Anak-anak	Perempuan	Berat	$((22.5*bb)+499-500)*2$

Rule	IMT	Umur	Jen_Kelamin	Jen_Aktivitas	Perhitungan AMB
201	Obesitas	Remaja	Perempuan	Sangat Ringan	$((12.2*bb)+746-500)*1.3$
202	Obesitas	Remaja	Perempuan	Ringan	$((12.2*bb)+746-500)*1.55$
203	Obesitas	Remaja	Perempuan	Sedang	$((12.2*bb)+746-500)*1.7$
204	Obesitas	Remaja	Perempuan	Berat	$((12.2*bb)+746-500)*2$
205	Obesitas	Dewasa	Perempuan	Sangat Ringan	$((14.7*bb)+496-500)*1.3$
206	Obesitas	Dewasa	Perempuan	Ringan	$((14.7*bb)+496-500)*1.55$
207	Obesitas	Dewasa	Perempuan	Sedang	$((14.7*bb)+496-500)*1.7$
208	Obesitas	Dewasa	Perempuan	Berat	$((14.7*bb)+496-500)*2$
209	Obesitas	Parobaya	Perempuan	Sangat Ringan	$((8.7*bb)+829-500)*1.3$
210	Obesitas	Parobaya	Perempuan	Ringan	$((8.7*bb)+829-500)*1.55$
211	Obesitas	Parobaya	Perempuan	Sedang	$((8.7*bb)+829-500)*1.7$
212	Obesitas	Parobaya	Perempuan	Berat	$((8.7*bb)+829-500)*2$
213	Obesitas	Tua	Perempuan	Sangat Ringan	$((10.5*bb)+596-500)*1.3$
214	Obesitas	Tua	Perempuan	Ringan	$((10.5*bb)+596-500)*1.55$
215	Obesitas	Tua	Perempuan	Sedang	$((10.5*bb)+596-500)*1.7$
216	Obesitas	Tua	Perempuan	Berat	$((10.5*bb)+596-500)*2$
217	Obesitas	Balita	Laki-laki	Sangat Ringan	$((60.9*bb)-54-500)*1.3$
218	Obesitas	Balita	Laki-laki	Ringan	$((60.9*bb)-54-500)*1.65$
219	Obesitas	Balita	Laki-laki	Sedang	$((60.9*bb)-54-500)*1.76$
220	Obesitas	Balita	Laki-laki	Berat	$((60.9*bb)-54-500)*2.1$
221	Obesitas	Anak-anak	Laki-laki	Sangat Ringan	$((22.7*bb)+495-500)*1.3$
222	Obesitas	Anak-anak	Laki-laki	Ringan	$((22.7*bb)+495-500)*1.65$
223	Obesitas	Anak-anak	Laki-laki	Sedang	$((22.7*bb)+495-500)*1.76$
224	Obesitas	Anak-anak	Laki-laki	Berat	$((22.7*bb)+495-500)*2.1$
225	Obesitas	Remaja	Laki-laki	Sangat Ringan	$((17.5*bb)+651-500)*1.3$
226	Obesitas	Remaja	Laki-laki	Ringan	$((17.5*bb)+651-500)*1.65$
227	Obesitas	Remaja	Laki-laki	Sedang	$((17.5*bb)+651-500)*1.76$
228	Obesitas	Remaja	Laki-laki	Berat	$((17.5*bb)+651-500)*2.1$
229	Obesitas	Dewasa	Laki-laki	Sangat Ringan	$((15.3*bb)+679-500)*1.3$
230	Obesitas	Dewasa	Laki-laki	Ringan	$((15.3*bb)+679-500)*1.65$
231	Obesitas	Dewasa	Laki-laki	Sedang	$((15.3*bb)+679-500)*1.76$
232	Obesitas	Dewasa	Laki-laki	Berat	$((15.3*bb)+679-500)*2.1$
233	Obesitas	Parobaya	Laki-laki	Sangat Ringan	$((11.6*bb)+879-500)*1.3$
234	Obesitas	Parobaya	Laki-laki	Ringan	$((11.6*bb)+879-500)*1.65$
235	Obesitas	Parobaya	Laki-laki	Sedang	$((11.6*bb)+879-500)*1.76$
236	Obesitas	Parobaya	Laki-laki	Berat	$((11.6*bb)+879-500)*2.1$
237	Obesitas	Tua	Laki-laki	Sangat Ringan	$((13.5*bb)+487-500)*1.3$
238	Obesitas	Tua	Laki-laki	Ringan	$((13.5*bb)+487-500)*1.65$
239	Obesitas	Tua	Laki-laki	Sedang	$((13.5*bb)+487-500)*1.76$
240	Obesitas	Tua	Laki-laki	Berat	$((13.5*bb)+487-500)*2.1$

LAMPIRAN 3

TABEL HASIL AKURASI DATA LATIH

No.	Nama	ID Jen Kelamin	Umur	Tinggi	Berat	Jen Aktivitas	Hasil Manual	Hasil Sistem
1	pasien 1	2	54	155	42	2	2880	2880
2	pasien 2	1	55	165	54	2	2650	2650
3	pasien 3	1	53	157	42	3	3919	3919
4	pasien 4	2	62	145	48	1	1705	1705
5	pasien 5	2	42	165	48	3	3493	3493
6	pasien 6	1	55	175	59	0	2032	2032
7	pasien 7	1	84	175	55	3	3632	3632
8	pasien 8	1	55	180	84	3	3892	3892
9	pasien 9	1	84	165	55	3	2582	2582
10	pasien 10	1	65	180	63	3	2809	2809
11	pasien 11	2	22	150	45	1	1794	1794
12	pasien 12	1	25	165	83	1	2391	2391
13	pasien 13	1	53	162	42	3	3919	3919
14	pasien 14	1	50	145	85	3	2867	2866
15	pasien 15	2	54	160	62	2	2326	2326
16	pasien 16	2	39	155	86	1	1670	1670
17	pasien 17	1	63	168	74	3	3121	3121
18	pasien 18	1	43	155	80	2	2300	2300
19	pasien 19	2	80	160	56	0	1539	1539
20	pasien 20	1	85	175	55	1	2854	2681
21	pasien 21	1	50	160	54	3	3161	3161
22	pasien 22	2	54	155	62	2	2326	2326
23	pasien 23	2	63	155	60	2	2084	2084
24	pasien 24	1	71	160	52	3	2497	2497
25	pasien 25	1	61	165	46	2	2830	2830
26	pasien 26	1	70	160	63	0	1739	1739
27	pasien 27	2	46	160	56	2	2238	2238
28	pasien 28	2	61	175	75	3	2767	2767
29	pasien 29	1	54	160	55	2	2670	2670
30	pasien 30	1	51	170	60	2	2772	2772
31	pasien 31	2	72	150	42	0	1348	1348
32	pasien 32	2	51	155	63	3	2754	2754
33	pasien 33	2	80	140	42	3	2074	2074
34	pasien 34	2	52	180	70	1	2229	2229
35	pasien 35	2	44	160	45	0	2237	2237
36	pasien 36	2	77	145	52	2	1941	1941
37	pasien 37	1	58	175	52	3	4163	3855
38	pasien 38	1	54	150	72	1	2003	2003
39	pasien 39	2	72	145	42	0	1348	1348
40	pasien 40	1	53	160	50	2	2568	2568

LAMPIRAN 4

TABEL HASIL AKURASI SISTEM

No.	Nama	ID Jen Kelamin	Umur	Tinggi	Berat	Jen Aktivitas	Hasil Manual	Hasil Sistem
1	pasien 1	1	84	165	55	3	2582	2582
2	pasien 2	1	84	170	55	3	2582	2582
3	pasien 3	1	84	180	55	3	3632	3632
4	pasien 4	1	84	185	55	3	3632	3632
5	pasien 5	1	53	157	42	3	3919	3919
6	pasien 6	1	53	152	42	3	3919	3919
7	pasien 7	1	53	162	42	3	3919	3919
8	pasien 8	1	53	167	42	3	3919	3919
9	pasien 9	2	39	145	86	1	1670	1670
10	pasien 10	2	39	150	86	1	1670	1670
11	pasien 11	2	39	160	86	1	1670	1670
12	pasien 12	2	39	165	86	1	1670	1670
13	pasien 13	1	85	165	55	1	2029	2029
14	pasien 14	1	85	170	55	1	2029	2029
15	pasien 15	1	85	180	55	1	2854	2767
16	pasien 16	1	85	185	55	1	2854	2854
17	pasien 17	2	54	145	62	2	1476	1476
18	pasien 18	2	54	150	62	2	1476	1476
19	pasien 19	2	54	160	62	2	2326	2326
20	pasien 20	2	54	165	62	2	2326	2326
21	pasien 21	2	63	145	60	2	1234	1234
22	pasien 22	2	63	150	60	2	2084	1659
23	pasien 23	2	63	160	60	2	2084	2084
24	pasien 24	2	63	165	60	2	2084	2084
25	pasien 25	2	46	150	56	2	2238	2238
26	pasien 26	2	46	155	56	2	2238	2238
27	pasien 27	2	46	165	56	2	2238	2238
28	pasien 28	2	46	170	56	2	2238	2238
29	pasien 29	2	72	140	42	0	1348	1348
30	pasien 30	2	72	145	42	0	1348	1348
31	pasien 31	2	72	155	42	0	1998	1998
32	pasien 32	2	72	160	42	0	1998	1998
33	pasien 33	2	51	145	63	3	1754	1754
34	pasien 34	2	51	150	63	3	1754	1754
35	pasien 35	2	51	160	63	3	2754	2754
36	pasien 36	2	51	165	63	3	2754	2754
37	pasien 37	2	52	170	70	1	2229	2229
38	pasien 38	2	52	175	70	1	2229	2229
39	pasien 39	2	52	185	70	1	2229	2229
40	pasien 40	2	52	190	70	1	2229	2229
41	pasien 41	2	22	150	45	1	1794	1794
42	pasien 42	1	25	165	83	1	2391	2391
43	pasien 43	2	19	155	86	1	1953	1953
44	pasien 44	1	43	155	80	2	2300	2300
45	pasien 45	2	46	160	56	2	2238	2238
46	pasien 46	1	50	160	54	3	3161	3161
47	pasien 47	1	84	175	55	0	2248	2248
48	pasien 48	1	84	175	55	1	2854	2681
49	pasien 49	1	84	175	55	2	3044	3044
50	pasien 50	1	53	157	42	0	2426	2426

No.	Nama	ID Jen Kelamin	Umur	Tinggi	Berat	Jen Aktivitas	Hasil Manual	Hasil Sistem
51	pasien 51	1	75	175	55	1	2854	2681
52	pasien 52	2	80	140	42	3	2074	2074
53	pasien 53	2	62	145	48	1	1705	1705
54	pasien 54	2	77	145	52	2	1941	1941
55	pasien 55	2	39	155	86	0	1400	1400
56	pasien 56	1	43	155	80	2	2300	2300
57	pasien 57	2	51	155	63	3	2754	2754
58	pasien 58	1	65	180	63	3	2809	2809
59	pasien 59	1	55	180	84	3	3892	3892
60	pasien 60	2	52	180	70	1	2229	2229
61	pasien 61	2	52	180	70	0	1869	1869
62	pasien 62	2	52	180	70	1	2229	2229
63	pasien 63	2	52	180	70	2	2445	2445
64	pasien 64	2	52	180	70	3	2876	2876
65	pasien 65	2	51	155	63	0	1790	1790
66	pasien 66	2	51	155	63	1	2135	2135
67	pasien 67	2	51	155	63	2	2341	2341
68	pasien 68	2	54	155	62	0	1779	1779
69	pasien 69	2	54	155	62	1	2121	2121
70	pasien 70	2	54	155	62	3	2737	2737
71	pasien 71	2	46	160	56	0	1711	1711
72	pasien 72	2	46	160	56	1	2040	2040
73	pasien 73	2	46	160	56	3	2632	2632
74	pasien 74	2	39	155	86	0	1400	1400
75	pasien 75	2	39	155	86	1	1670	1670
76	pasien 76	2	39	155	86	3	2154	2154
77	pasien 77	2	39	155	66	1	1400	1787
78	pasien 78	2	39	155	76	1	1535	1535
79	pasien 79	2	39	155	96	1	1805	1805
80	pasien 80	2	39	155	96	1	1805	1805
81	pasien 81	1	85	175	35	1	2408	2408
82	pasien 82	1	85	175	45	1	2631	2631
83	pasien 83	1	85	175	65	1	2251	2251
84	pasien 84	1	85	175	75	1	2474	2474
85	pasien 85	2	52	180	50	1	2734	2734
86	pasien 86	2	52	180	60	1	2094	2094
87	pasien 87	2	52	180	80	1	2364	2364
88	pasien 88	2	52	180	90	1	1724	1724
89	pasien 89	2	42	180	70	1	2229	2229
90	pasien 90	2	47	180	70	1	2229	2229
91	pasien 91	2	57	180	70	1	2229	2132
92	pasien 92	2	62	180	70	1	2063	2063
93	pasien 93	2	41	155	63	3	2754	2754
94	pasien 94	2	46	155	63	3	2754	2754
95	pasien 95	2	54	155	63	3	2754	2754
96	pasien 96	2	61	155	63	3	2515	2515
97	pasien 97	1	84	175	35	3	3065	3065
98	pasien 98	1	84	175	45	3	3348	3348
99	pasien 99	1	84	175	65	3	2865	2865
100	pasien 100	1	84	175	75	3	3149	3149