

**PENGLASIFIKASIAN KELAS TA'LIM AFKAR MAHASANTRI
BARU MA'HAD SUNAN AMPEL AL-'ALI UIN MAULANA
MALIK IBRAHIM MALANG MENGGUNAKAN
ALGORITMA GENETIKA**

SKRIPSI

Oleh:

TRI HENDRY ANDHIKA

NIM. 09650211



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2014**

**PENGLASIFIKASIAN KELAS TA'LIM AFKAR MAHASANTRI
BARU MA'HAD SUNAN AMPEL AL-'ALI UIN MAULANA
MALIK IBRAHIM MALANG MENGGUNAKAN
ALGORITMA GENETIKA**

SKRIPSI

**Diajukan kepada:
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh:
TRI HENDRY ANDHIKA
NIM. 09650211**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2014**

**PENGLASIFIKASIAN KELAS TA'LIM AFKAR MAHASANTRI
BARU MA'HAD SUNAN AMPEL AL-'ALI UIN MAULANA
MALIK IBRAHIM MALANG MENGGUNAKAN
ALGORITMA GENETIKA**

SKRIPSI

Oleh :

Nama : Tri Hendry Andhika
NIM : 09650211
Jurusan : Teknik Informatika
Fakultas : Sains Dan Teknologi

Telah disetujui, 7 April 2014

Dosen Pembimbing I

Dosen Pembimbing II

Dr. M. Amin Hariyadi, M.T
NIP. 19670118 200501 1 001

M. Imamuddin, Lc., MA.
NIP. 19740602 200901 1 010

Mengetahui,
Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdiyan, MSC
NIP. 19740424 200901 1 008

**PENGLASIFIKASIAN KELAS TA'LIM AFKAR MAHASANTRI
BARU MA'HAD SUNAN AMPEL AL-'ALI UIN MAULANA
MALIK IBRAHIM MALANG MENGGUNAKAN
ALGORITMA GENETIKA**

SKRIPSI

Oleh :

TRI HENDRY ANDHIKA

NIM. 09650211

**Telah Dipertahankan Di Depan Dewan Penguji Skripsi
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)**

Tanggal 11 April 2014

Susunan Dewan Penguji:

	Tanda Tangan
1. Penguji Utama : <u>Linda Salma Angreani, M.T</u> NIP. 19770803 200912 2 005	()
2. Ketua Penguji : <u>Fatchurrochman, M.Kom</u> NIP. 19700731 200501 1 002	()
3. Sekretaris : <u>Dr. M. Amin Hariyadi, M.T</u> NIP. 19670118 200501 1 001	()
4. Anggota Penguji : <u>M. Imamuddin, Lc, M.A</u> NIP. 19740602 200901 1 010	()

Mengetahui,
Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdiyan, MSC
NIP. 19740424 200901 1 008

HALAMAN PERNYATAAN**KEASLIAN TULISAN**

Saya yang bertanda tangan di bawah ini:

Nama : Tri Hendry Andhika
NIM : 09650211
Jurusan : Teknik Informatika
Fakultas : Sains dan Teknologi
Judul Penelitian : Pengklasifikasian Kelas *Ta'lim Afkar* Mahasantri Baru
Ma'had Sunan Ampel Al-'Ali UIN Maulana Malik
Ibrahim Malang menggunakan Algoritma Genetika

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut

Malang, 4 April 2014

Yang Membuat Pernyataan,

Tri Hendry Andhika
NIM. 09650211

PERSEMBAHAN

Sembah sujud dan syukur kupersembahkan kepada Allah SWT yang telah memberikanku kekuatan, kemudahan di setiap langkahku, serta segala nikmat yang Engkau berikan akhirnya skripsi ini dapat terselesaikan. Sholawat serta salam selalu tercurah limpahkan kepada Rasulullah SAW yang telah menuntunku menjadi salah satu umatnya.

Kupersembahkan Karya Tulis sederhana ini kepada:

Bapakku terkasih Abdul Hamid dan Ibuku tercinta Salha, yang mengenalkanku akan arti hidup, kasih sayang, dan tanggungjawab yang mengajari aku untuk bertahan dan tegar cinta, kasih sayang, kesabaran, dan kepercayaan kalian tidak akan pudar semoga Allah selalu melimpahkan rahmat dan kasih sayang-Nya...

*Seluruh keluargaku,
yang telah memberi dukungan,
jasa-jasa kalian telah menjadi motivasi dalam hidupku.*

*Bapak dan Ibu guru, Bapak dan Ibu Dosen
yang telah memberikan ilmu dan nasehat,
Kalian adalah pelita dalam kegelapan
Semoga Allah membalas jasa-jasa kalian.*

Vera Mei

*senyum, kasih sayang, kesabaran, dan perhatianmu
telah menjadi motivasi dalam hidupku.*

*Sahabat-sahabatku,
Semua orang yang telah hadir dan mewarnai hidupku.*

MOTTO

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا ﴿٥﴾ إِنَّ مَعَ الْعُسْرِ يُسْرًا ﴿٦﴾

Artinya:

“Karena sesungguhnya sesudah kesulitan itu ada kemudahan, Sesungguhnya sesudah kesulitan itu ada kemudahan.” (QS. Alam Nasyrah:5-6)

**“Tidak ada yang mudah,
tapi tidak ada yang tidak mungkin”**

KATA PENGANTAR



Dengan mengucapkan syukur kehadiran Allah SWT. Alhamdulillah, berkat rahmat dan petunjuk-Nya, penulis bisa menyelesaikan skripsi ini dengan baik. Shalawat serta salam senantiasa tercurahkan kepada junjungan kita Nabi Agung Muhammad SAW. yang membawa kita dari zaman *jahiliyah* ke jalan yang benar.

Penulis menyadari keterbatasan pengetahuan yang penulis miliki. Karena tanpa keterlibatan berbagai pihak yang telah memberikan bantuan, dorongan semangat dan bimbingan, sulit bagi penulis untuk menyelesaikan skripsi ini. Oleh karena itu, penulis menyampaikan terimakasih yang sebesar-besarnya kepada :

1. Bapak dan Ibuku yang dengan ketulusan tidak pernah berhenti mendukung, yang menjadi sandaran, dan kebanggaan bagi penulis.
2. Bapak Prof. Dr. Mudjia Raharjo, selaku Rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Ibu Dr. Drh. Hj Bayyinatul Muchtaromah, M.Si selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Bapak Dr. Cahyo Crysdiyan, MSC selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
5. Bapak Dr. M. Amin Hariyadi, M.T dan Bapak M. Imamuddin, Lc., M.A selaku dosen pembimbing skripsi yang telah meluangkan waktu untuk membimbing, memotivasi, mengarahkan, memberi nasehat dan petunjuk dalam pengerjaan skripsi ini.

6. Bapak A'la Syauqi, M.Kom selaku dosen wali yang telah membimbing dan mengarahkan selama masa kuliah.
7. Segenap Dosen Teknik informatika yang telah memberikan bimbingan keilmuan kepada penulis selama masa studi.
8. Kakak-kakak Pelatih, Pembina dan segenap anggota keluarga besar Racana Maulana Malik Ibrahim-Dewi Chandra Wulan pangkalan UIN Maulana Malik Ibrahim Malang, khususnya angkatan 22 terimakasih telah berbagi suka, duka dan pengalaman berharga. *Salam Pramuka...!!*
9. Segenap staf *Ma'had* Sunan Ampel Al-'Ali, *murobbiyah* dan *musyrifah* yang telah membantu dan mendukung penulis dalam melaksanakan penelitian skripsi ini.
10. Semua teman-teman Jurusan Teknik Informatika angkatan 2009, Khususnya kelas G yang telah membantu dan menyemangati penulis selama masa kuliah.
11. Seluruh sahabat-sahabatku dan semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu demi terselesainya skripsi ini.

Penulisan serta penelitian skripsi ini masih jauh dari sempurna, banyak kekurangan dan kelemahan. Kritik dan saran yang membangun dari berbagai pihak, sangat penulis harapkan demi perbaikan selanjutnya. Penulis berharap semoga penelitian skripsi ini dapat memberikan manfaat bagi semua pihak.

Malang, 4 April 2014

Penulis

DAFTAR ISI

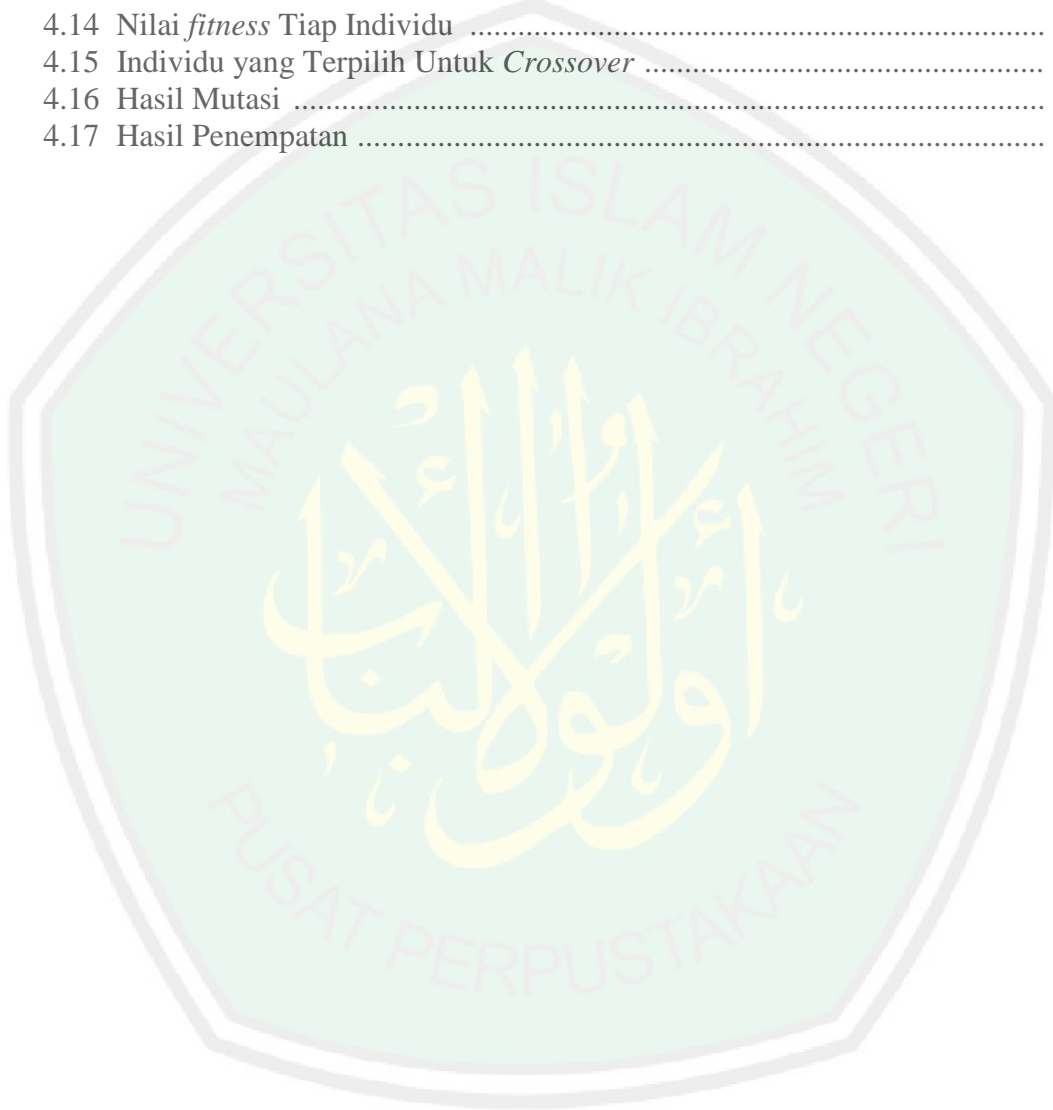
HALAMAN JUDUL	i
HALAMAN PENGANTAR	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
HALAMAN PERNYATAAN	v
HALAMAN PERSEMBAHAN	vi
HALAMAN MOTTO	vii
KATA PENGANTAR	viii
DAFTAR ISI	x
DAFTAR GAMBAR	xii
DAFTAR TABEL	xiv
ABSTRAK	xv
ABSTRACT	xvi
BAB I PENDAHULUAN	
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	6
1.3 Batasan Masalah	7
1.4 Tujuan Penelitian	7
1.5 Manfaat Penelitian	7
1.6 Metode Penelitian	7
1.7 Sistematika Penulisan	8
BAB II TINJAUAN PUSTAKA	
2.1 <i>Ma'had</i> Sunan Ampel Al-'Ali UIN Maliki Malang	10
2.2 <i>Ta'lim</i>	12
2.2.1 <i>Ta'lim al-Afkar al-Islamiyyah</i>	12
2.2.2 <i>Ta'lim</i> Al-Qur'an	13
2.2.3 <i>Khatm</i> Al-Qur'an	14
2.2.4 <i>Tashih Qiro'ah</i> Al-Qur'an	14
2.2.5 <i>Tahsin Tilawah</i> Al-Qur'an	14
2.3 Algoritma Genetika	15
2.3.1 Pengertian	15
2.3.2 Struktur Umum Algoritma Genetika	17
2.3.3 Komponen-komponen utama Algoritma Genetika	18
2.3.4 <i>Crossover</i>	22
2.3.5 Mutasi	24
2.3.6 Kriteria Penghentian	26
2.4 UML (<i>Unified Modelling Language</i>)	27
2.4.1 <i>Use Case Diagram</i>	28
2.4.2 <i>Activity Diagram</i>	28
2.4.3 <i>Sequence Diagram</i>	28
2.4.4 <i>Collaboration Diagram</i>	28
2.4.5 <i>Class Diagram</i>	29

2.4.6 <i>State Chart Diagram</i>	29
2.4.7 <i>Component Diagram</i>	29
2.4.8 <i>Deployment Diagram</i>	29
BAB III ANALISIS DAN PERANCANGAN	
3.1 Analisis Sistem	37
3.1.1 Analisa Proses	37
3.1.2 Analisa Keluaran	38
3.1.3 Analisa Masukan	39
3.1.4 Identifikasi Kebutuhan	39
3.1.5 Kebutuhan Sistem	39
3.2 Pemodelan UML	40
3.2.1 <i>Use Case Diagram</i>	40
3.2.2 <i>Activity Diagram</i>	42
3.2.3 <i>Sequence Diagram</i>	47
3.2.4 <i>Class Diagram</i>	48
3.3 Perancangan Sistem	49
3.3.1 Perancangan Algoritma Genetika	49
a. Teknik Pengkodean	49
b. Inisialisasi Kromosom	51
c. Fungsi Evaluasi (<i>Fitness</i>)	53
d. Seleksi	55
e. <i>Crossover</i>	56
f. Mutasi	57
3.3.2 <i>Output</i>	58
3.4 Perancangan <i>Database</i>	58
3.5 Perancangan <i>Interface</i>	61
BAB IV HASIL DAN PEMBAHASAN	
4.1 Implementasi	65
4.2 Uji Coba	82
4.3 Analisa Hasil	85
4.4 Pengklasifikasian menurut Islam	91
BAB V PENUTUP	
5.1 Kesimpulan	93
5.2 Saran	93
DAFTAR PUSTAKA	94
LAMPIRAN	95

DAFTAR GAMBAR

2.1	Blok diagram Algoritma Genetika	18
2.2	Contoh Populasi dengan Lima Kromosom	20
2.3	Probabilitas Kromosom dalam Roda <i>Roulette</i>	21
2.4	<i>Crossover</i> Satu Titik	23
2.5	<i>Crossover</i> Dua Titik	23
2.6	<i>Order Crossover</i>	24
2.7	Mutasi Tingkat Kromosom	26
2.8	Mutasi Tingkat Gen	26
2.9	Mutasi Tingkat Bit	26
2.10	Tampilan Awal <i>Rational Rose</i>	31
2.11	Komponen <i>Use Case Diagram</i>	33
2.12	Komponen <i>Activity Diagram</i>	35
3.1	Analisa Proses	38
3.2	<i>Use Case Diagram</i>	41
3.3	<i>Activity Diagram</i> Algoritma Genetika	43
3.4	<i>Activity Diagram</i> Tes Penempatan	44
3.5	<i>Activity Diagram</i> Mengumumkan Hasil Penempatan	43
3.6	<i>Activity Diagram</i> <i>Maintain</i> Data Mahasantri	45
3.7	<i>Activity Diagram</i> <i>Maintain</i> Nilai Tes	45
3.8	<i>Activity Diagram</i> <i>Maintain</i> Data Penempatan	46
3.9	<i>Activity Diagram</i> <i>Maintain</i> Data Kelas	46
3.10	<i>Sequence Diagram</i> Proses Genetika	47
3.11	<i>Sequence Diagram</i> <i>Maintain</i> Data Mahasantri	48
3.12	<i>Class Diagram</i>	49
3.13	Relasi Antar Tabel	59
3.14	Desain Halaman Utama	61
3.15	Desain Halaman Profil	61
3.16	Desain Login Admin	62
3.17	Desain <i>View</i> Data Mahasantri	62
3.18	Desain <i>View</i> Berita	62
3.19	Desain <i>View</i> Kelas	63
3.20	Desain <i>View</i> Nilai Tes	63
3.21	Desain Kriteria	63
3.22	Desain Parameter Genetika	64
3.23	Desain <i>View</i> Hasil Penempatan	64
4.1	<i>Sitemap</i> Program	65
4.2	Halaman Utama	66
4.3	Halaman Profil	67
4.4	Login Admin	67
4.5	Data Mahasantri	68
4.6	Berita	69
4.7	Data Kelas	69
4.8	Data Nilai	70

4.9 Kriteria	70
4.10 Parameter Genetika	71
4.11 Data <i>User Admin</i>	71
4.12 Hasil Penempatan	72
4.13 Inisialisasi Awal	83
4.14 Nilai <i>fitness</i> Tiap Individu	83
4.15 Individu yang Terpilih Untuk <i>Crossover</i>	84
4.16 Hasil Mutasi	84
4.17 Hasil Penempatan	85



DAFTAR TABEL

3.1 Tabel Pengkodean Asal Sekolah	50
3.2 Tabel Pengkodean <i>Range</i> Nilai Tes	50
3.3 Tabel Inisialisasi Kromosom	52
3.4 Tabel Ilustrasi Inisialisasi Kromosom	53
3.5 Tabel Nilai <i>Fitness</i>	54
3.6 Tabel Probabilitas Tiap Individu	55
3.7 Tabel Ilustrasi <i>crossover</i>	56
3.8 Tabel Ilustrasi Mutasi	57
3.9 Tabel Ilustrasi Hasil output	58
3.10 Tabel Mahasantri	59
3.11 Tabel <i>Users</i>	60
3.12 Tabel Kelas	60
3.13 Tabel Penempatan	60
3.14 Tabel Nilai	61
4.1 Tabel Sampel Data Mahasantri	82
4.2 Parameter Genetika	82
4.3 Nilai <i>Fitness</i> Inisialisasi Awal	86
4.4 Nilai <i>Fitness</i> Setelah Seleksi	87
4.5 Kromosom yang Mengalami <i>Crossover</i>	87
4.6 Kromosom yang Mengalami Mutasi	88
4.7 Populasi pada Generasi ke-2	88
4.8 Populasi pada Generasi ke-10	89
4.9 Perbandingan Hasil Uji Coba	90

ABSTRAK

Andhika, Tri Hendry. 2014. **Pengklasifikasian Kelas *Ta'lim Afkar* Mahasantri Baru *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang**. Skripsi Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing: (I) Dr. M. Amin Hariyadi, M.T (II) M. Imamuddin, Lc., M.A

Kata Kunci: *Pengklasifikasian, Ma'had, Algoritma Genetika*

Sistem Pengklasifikasian kelas *ta'lim afkar* mahasantri baru *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang masih dilakukan dengan cara konvensional, yaitu dengan metode penalaran seorang penguji yang bersifat subjektif. Sehingga memungkinkan dalam suatu kelas terdapat mahasantri yang berada dalam kelas yang tidak sesuai. Masalah pengklasifikasian kelas merupakan masalah pengelompokan, algoritma genetika bisa digunakan untuk menyelesaikan masalah ini. Algoritma genetika bekerja dengan mencari individu terbaik sebagai solusi dalam pengklasifikasian mahasantri baru dengan menggunakan perangkat lunak.

Algoritma genetika terdiri dari lima proses, inisialisasi kromosom, Seleksi menggunakan metode *roulette wheel selection*. *Crossover* menggunakan metode *single-point crossover*. Dalam proses ini bisa tidak terjadi mutasi, hal ini karena dipengaruhi peluang mutasi. Metode mutasi yang digunakan adalah *swapping mutation*. Hasil dari proses ini berupa data mahasantri yang ditempatkan dalam kelas sesuai dengan parameter yang ditentukan. Berdasarkan hasil tersebut didapatkan 91% mirip dengan hasil konvensional dan lebih tepat sasaran untuk pengklasifikasian kelas karena berdasarkan kemampuan mahasantri.

ABSTRACT

Andhika, Tri Hendry. 2014. **Classification of New University Student Ta'lim Afkar Class in Ma'had Sunan Ampel Al-'Ali Maulana Malik Ibrahim State Islamic University Malang**. Thesis. Informatics Engineering Department Faculty of Science and Technology. Maulana Malik Ibrahim State Islamic University Malang.

Adviser: (I) Dr. M. Amin Hariyadi, M.T (II) M. Imamuddin, Lc., M.A

Keywords: *Classification, Ma'had, Genetic Algorithm*

The classification system of ta'lim Afkar class for new university student in Ma'had Sunan Ampel Al-'Ali Maulana Malik Ibrahim State Islamic University Malang is still performed in a conventional manner. Its classified through a reasoning method of examiner which is very subjective. Thus allowing new university student contained in a class that is not appropriate for the class. Class classification problem is a grouping problem. Method that can be used for classification problems is genetic algorithms. Genetic algorithms work by finding the best individual as solution in classifying new university student using the software.

Genetic algorithm consists of five processes, initiated with chromosome. The selection uses roulette wheel selection method. Crossover method uses single point crossover. In this process, the mutation may not occur. It is due to its opportunity. Conversely, the swapping mutation method can be used whenever the mutation happens. The finding of this process is university student's data placed in the class by constructed parameter. Based on the result of this study, 91 % of the data found is similar with conventional result and considered more accurate for class classifying since it is adjusted with their ability.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Menuntut ilmu pengetahuan bagi seorang muslim wajib hukumnya, baik ilmu agama maupun ilmu umum. Ilmu agama hukumnya wajib untuk dipelajari karena ilmu agama mencakup seluruh aspek kehidupan baik sosial, politik dan budaya. Sedangkan mempelajari ilmu pengetahuan umum hukumnya sunnah selama digunakan untuk kebaikan dan tidak menyimpang dari hukum Islam. Dalam menuntut ilmu harus diniatkan untuk menghapus kebodohan, semakin tinggi pendidikan seorang muslim semakin tinggi pula derajatnya sesuai firman Allah dalam Al-Qur'an surat Al-Mujaadilah Ayat 11

يَتَأْتِيهَا الَّذِينَ ءَامَنُوا إِذَا قِيلَ لَكُمْ تَفَسَّحُوا فِي الْمَجَالِسِ فَأَفْسَحُوا يَفْسَحِ
 اللَّهُ لَكُمْ وَإِذَا قِيلَ أَنْشُرُوا فَأَنْشُرُوا يَرْفَعُ اللَّهُ الَّذِينَ ءَامَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا
 الْعِلْمَ دَرَجَاتٍ ۗ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ ﴿١١﴾

Artinya :

"Hai orang-orang beriman apabila kamu dikatakan kepadamu: "Berlapang-lapanglah dalam majlis", Maka lapangkanlah niscaya Allah akan memberi kelapangan untukmu. dan apabila dikatakan: "Berdirilah kamu", Maka berdirilah, niscaya Allah akan meninggikan orang-orang yang beriman di antaramu dan orang-orang yang diberi ilmu pengetahuan beberapa derajat. dan Allah Maha mengetahui apa yang kamu kerjakan. (Qs. Al-Mujaadilah:11).

Ayat tersebut tidak menyebut secara tegas bahwa Allah akan meninggikan derajat orang yang berilmu. Tetapi menegaskan bahwa mereka memiliki derajat-derajat yakni yang lebih tinggi dari dari sekadar beriman. Dalam ayat tersebut

membagi kaum yang beriman menjadi dua kelompok besar, pertama kelompok yang sekadar beriman dan beramal saleh, kedua kelompok yang beriman dan beramal saleh serta memiliki pengetahuan. Derajat kelompok yang kedua ini lebih tinggi, bukan karena nilai ilmu yang disandangnya tetapi amal juga pengajarannya kepada pihak lain baik secara lisan atau secara tulisan maupun dengan keteladanan. Ilmu yang dimaksud dalam ayat tersebut tentu saja bukan hanya ilmu agama, tetapi ilmu apapun yang bermanfaat. (Quraish Shihab 2003:79-80).

UIN Maulana Malik Ibrahim Malang memandang keberhasilan pendidikan mahasiswa apabila mereka memiliki identitas sebagai seseorang yang mempunyai ilmu pengetahuan yang luas, penglihatan yang tajam, otak yang cerdas, hati yang lembut, dan semangat tinggi karena Allah (Tarbiyatu Uli al-Albab: Dzikir, Fikir, dan Amal Saleh, 2000:5). Untuk mencapai keberhasilan tersebut, kegiatan kependidikan di UIN Maulana Malik Ibrahim Malang, baik kurikuler, ko-kurikuler, maupun ekstra kurikuler diarahkan pada pemberdayaan potensi dan kegemaran mahasiswa untuk mencapai target profil lulusan yang memiliki ciri-ciri: kemandirian, siap berkompetisi dengan lulusan perguruan tinggi lain, berwawasan akademik global, kemampuan memimpin atau sebagai penggerak umat, bertanggung jawab dalam mengembangkan agama Islam di tengah-tengah masyarakat, berjiwa besar, selalu peduli pada orang lain/gemar berkorban untuk kemajuan bersama, dan kemampuan menjadi tauladan bagi masyarakat sekelilingnya (UIN Malang, 2006:13).

Salah satu upaya pengembangan kelembagaan untuk mewujudkan strategi tersebut tercermin dalam kemampuan membangun *bi'ah* Islamiyah yang mampu menumbuhkan suburkan akhlakul karimah bagi setiap civitas akademika UIN Maulana Malik Ibrahim Malang. Karena itu, kehadiran *ma'had* di lingkungan kampus sangat dibutuhkan dan diharapkan secara intensif mampu memberikan resonansi dalam mewujudkan lembaga pendidikan tinggi Islam yang ilmiah-religius, sekaligus sebagai bentuk penguatan terhadap pembentukan lulusan yang intelek profesional yang ulama' atau ulama' yang intelek profesional.

Salah satu kegiatan di *Ma'had* Sunan Ampel Al-'Ali untuk menunjang peningkatan kompetensi akademik adalah *ta'lim*. Kegiatan ini dilaksanakan empat kali dalam sepekan setiap pagi. *Ta'lim* disini ada dua macam, yaitu *ta'lim* Qur'an dan *ta'lim afkar*. *Ta'lim* Qur'an membahas tata cara membaca Al-Qur'an beserta hukum bacaannya sedangkan *ta'lim afkar* mengkaji kitab yang berisi persoalan fiqh dan keimanan, salah satu kitab yang digunakan dalam *ta'lim afkar* adalah *Qomi' at-Thughyan*. *Qomi' at-Thughyan* merupakan salah satu kitab yang digunakan dalam pembelajaran Islam, kitab tersebut berisi tentang pokok-pokok keimanan ajaran Islam. Kitab yang dikarang oleh Syaikh Muhammad Nawawi bin Umar al Bantani itu menerangkan tentang 77 cabang iman dimana cabang-cabang tersebut merupakan rincian dari Rukun Iman, yaitu Iman Kepada Allah, Malaikat, Kitab-kitab Allah, Nabi, Hari Kiamat, serta *Qada'* dan *Qadar*. Kitab ini telah digunakan selama bertahun-tahun dalam pondok pesantren di Indonesia agar para santri bisa mendalami dan mendapatkan pengetahuan tentang keimanan. sehingga menjadi muslim yang mengamalkan prinsip "*amar ma'ruf nahi munkar*".

Firman Allah dalam Al-Qur'an surat Ali-Imran Ayat 104

وَلْتَكُنْ مِنْكُمْ أُمَّةٌ يَدْعُونَ إِلَى الْخَيْرِ وَيَأْمُرُونَ بِالْعُرْفِ وَيَنْهَوْنَ عَنِ الْمُنْكَرِ
 وَأُولَئِكَ هُمُ الْمُفْلِحُونَ ﴿١٠٤﴾

Artinya :

“dan hendaklah ada di antara kamu segolongan umat yang menyeru kepada kebajikan, menyuruh kepada yang ma'ruf dan mencegah dari yang munkar. merekalah orang-orang yang beruntung” (QS.Ali-Imran:104).

Ayat tersebut menyuruh kita untuk selalu mengajak kepada kebaikan dan mencegah kemungkaran agar menjadi orang yang beruntung. Berbuat *ma'ruf*, segala perbuatan yang mendekatkan kita kepada Allah, seperti menolong sesama, meninggalkan yang *munkar* yaitu segala perbuatan yang menjauhkan kita dari-Nya, seperti mencuri, membunuh dan sebagainya.

Firman Allah dalam Al-Qur'an surat Al-Jumuah Ayat 2.

هُوَ الَّذِي بَعَثَ فِي الْأُمِّيِّينَ رَسُولًا مِنْهُمْ يَتْلُوا عَلَيْهِمْ آيَاتِهِ وَيُزَكِّيهِمْ وَيُعَلِّمُهُمُ
 الْكِتَابَ وَالْحِكْمَةَ وَإِنْ كَانُوا مِنْ قَبْلُ لَفِي ضَلَالٍ مُبِينٍ ﴿٢﴾

Artinya :

“Dia-lah yang mengutus kepada kaum yang buta huruf seorang Rasul dari mereka, yang membacakan ayat-ayat-Nya kepada mereka, mensucikan mereka dan mengajarkan mereka kitab dan Hikmah (As Sunnah) dan sesungguhnya mereka sebelumnya benar-benar dalam kesesatan yang nyata” (QS. Al-Jumuah:2).

Ayat tersebut menjelaskan tentang pembelajaran dalam Islam, Allah mengutus Nabi Muhammad Saw. yang tidak pandai membaca dan menulis, membacakan ayat-ayat kepada mereka (bangsa arab) dan menyucikan mereka dari keburukan pikiran, hati dan tingkah laku serta menjelaskan dengan ucapan dan

perbuatannya sesuai Al-Qur'an dan hikmah berupa pemahaman agama, ilmu alamiah dan ilmu ilmiah. (Quraish Shihab 2003:219).

Sebelum mahasantri baru mengikuti kegiatan tersebut, terlebih dahulu diadakan *placement test*. Hal ini dimaksudkan untuk mengetahui kemampuan mahasantri baru tentang pengetahuan fiqh dan keimanan yang nantinya diklasifikasikan menurut kemampuannya. Tes tersebut terdiri dari tes tulis yang materinya diambil dari kitab *Al-Tadzhib* dan *Qomi' at-Thughyan*, berdasarkan hasil tes tersebut kemudian ditentukan di kelas mana mahasantri tersebut harus belajar.

Dalam proses penempatan kelas tersebut, dipilih orang-orang tertentu yang dianggap mampu untuk menguji mahasantri baru dan masih menggunakan penilaian langsung serta penalaran dari seorang penguji untuk menentukan pengklasifikasian kelas sehingga memungkinkan adanya penilaian subjektif dari seorang penguji. Tentunya hal ini sangat tidak efisien mengingat banyaknya mahasantri baru yang lebih dari 2000 orang dan sedikitnya jumlah penguji. Untuk itulah diperlukan cara yang lebih efektif dan efisien dalam menguji dan mengklasifikasikan mahasantri baru tersebut ke dalam kelas yang sesuai kemampuannya.

Firman Allah SWT dalam Surat Al-Baqarah ayat 286.

لَا يُكَلِّفُ اللَّهُ نَفْسًا إِلَّا وُسْعَهَا

Artinya :

“Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya.” (QS. Al-Baqarah:286).

Tugas-tugas yang dibebankan Allah kepada manusia adalah tugas yang mudah untuk dilaksanakan, bahkan setiap orang yang mengalami kesulitan dalam pelaksanaan tugas, mendapat kemudahan yang dibenarkan walaupun sebelumnya tidak dibenarkan. Shalat boleh duduk jika tidak mampu berdiri. (Quraish Shihab 2003:219).

Dari ayat tersebut, bisa dipahami bahwa Allah Maha Adil, Allah tidak akan memberikan beban lebih dari kemampuan seseorang. Begitu juga dalam hal belajar. Hendaknya mahasiswa ditempatkan dalam kelas sesuai tingkat pemahamannya.

Berdasarkan keadaan tersebut, perlu diadakan penelitian untuk mengklasifikasikan dalam penempatan kelas *ta'lim* mahasiswa baru secara efektif dan efisien. Berdasarkan masalah tersebut, Algoritma Genetika bisa digunakan untuk menyelesaikan masalah tersebut karena pengklasifikasian mahasiswa baru sesuai dengan algoritma yang memprioritaskan individu terbaik yang menempati posisi teratas. Hal inilah yang mengilhami peneliti untuk mengadakan penelitian dengan judul "*Pengklasifikasian Kelas Ta'lim Afkar Mahasiswa Baru Ma'had Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang Menggunakan Algoritma Genetika*".

1.2 Rumusan Masalah

Bagaimana sistem pengklasifikasian kelas *ta'lim afkar* bagi mahasiswa baru *Ma'had Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang* menggunakan Algoritma Genetika?

1.3 Tujuan Penelitian

Mengklasifikasikan kelas *ta'lim afkar* mahasantri baru *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang Menggunakan Algoritma Genetika.

1.4 Manfaat Penelitian

Memudahkan pengklasifikasian kelas *ta'lim afkar* mahasantri baru *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang.

1.5 Batasan Masalah

Batasan masalah dalam penelitian ini antara lain :

- a. Sistem digunakan untuk pengklasifikasian kelas *ta'lim afkar* mahasantri baru *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang.
- b. Sistem dibangun dengan bahasa pemrograman PHP.
- c. Metode yang digunakan adalah Algoritma Genetika.

1.6 Metode Penelitian

- a. Studi literatur
 1. Pengumpulan informasi yang berkaitan tentang *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang.
 2. Pengumpulan informasi tentang cara pengklasifikasian kelas.
 3. Pengumpulan informasi tentang metode Algoritma genetika.

b. Perancangan dan desain aplikasi

Perancangan aplikasi yang meliputi proses-proses dalam algoritma genetika, yaitu: inisialisasi, fungsi evaluasi, seleksi, *crossover* dan mutasi serta pengklasifikasian kelas. Desain aplikasi yang terdiri dari halaman utama dan desain aplikasi proses algoritma genetika.

c. Pembuatan aplikasi

Pembuatan aplikasi ini dengan mengimplementasikan algoritma genetika menggunakan bahasa pemrograman PHP.

d. Uji coba dan evaluasi

Uji coba dan evaluasi terhadap aplikasi yang telah di implementasikan.

e. Penyusunan laporan

Penyusunan laporan akhir merupakan dokumentasi pelaksanaan penelitian dan seluruh proses aplikasi sehingga diharapkan bisa bermanfaat untuk peneliti yang ingin mengembangkan aplikasi ini lebih lanjut.

1.7 Sistematika Penulisan

Laporan skripsi ini dibuat dengan sistem penulisan berikut:

Bab I Pendahuluan. Bab ini berisi tentang latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah serta sistematika penulisan.

Bab II Tinjauan Pustaka. Bab ini menjelaskan tentang teori yang terkait pengklasifikasian kelas *ta'lim afkar*, pembuatan analisa dan pemecahan masalah, serta membahas metode yang digunakan yaitu algoritma genetika.

Bab III Analisis dan Perancangan. Bab ini menjelaskan tahapan-tahapan yang dilalui, analisis dan perancangan sistem pengklasifikasian kelas *ta'lim afkar* mahasantri baru *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang menggunakan algoritma genetika.

Bab IV Hasil dan Pembahasan. Bab ini menjelaskan tentang implementasi dan uji coba sistem yang telah dibuat menggunakan algoritma genetika, serta analisis hasil program aplikasi.

Bab V Penutup. Bab ini berisi tentang kesimpulan dari pembahasan program aplikasi dan saran untuk pengembangan program aplikasi ini.



BAB II

TINJAUAN PUSTAKA

2.1 *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang

Pesantren adalah sebuah lembaga pendidikan keagamaan yang mempunyai kekhasan sendiri dan berbeda dengan lembaga pendidikan lainnya. Pendidikan di Pesantren meliputi pendidikan Islam, dakwah, pengembangan kemasyarakatan dan pendidikan lainnya yang sejenis. Para peserta didik pada pesantren disebut santri yang umumnya menetap di Pesantren. Tempat para santri menetap di lingkungan pesantren disebut dengan istilah Pondok. Dari sinilah timbul istilah Pondok Pesantren (Departemen Agama, 2003:1). Dari konsep Pondok Pesantren inilah *Ma'had* Sunan Ampel Al-'Ali didirikan.

Ma'had Sunan Ampel Al-'Ali didirikan pada masa kepemimpinan Prof. Dr. H. Imam Suprayogo, ketika itu masih menjabat sebagai Ketua STAIN Malang. Ide mendirikan *ma'had* sendiri di cetuskan oleh KH. Mansur. Peletakan batu pertama *ma'had* tersebut dimulai pada ahad wage, 4 April 1999 oleh 9 Kyai Jawa Timur dan disaksikan oleh sejumlah Kyai dari Kota dan Kabupaten Malang. Dalam jangka setahun UIN Maulana Malik Ibrahim Malang telah berhasil menyelesaikan 4 unit gedung yang terdiri dari 189 kamar (3 unit masing-masing 50 kamar dan 1 unit 39 kamar) yang nantinya dihuni oleh mahasantri baru UIN Maulana Malik Ibrahim Malang dari semua jurusan dan 5 rumah pengasuh serta 1 rumah *mudir* (direktur). Dengan selesainya pembangunan *ma'had* yang direncanakan sebanyak 10 unit, kini sudah terealisasi 5 unit. Sejak 26 Agustus

2000 *ma'had* tersebut mulai dihuni oleh 1041 mahasantri, 483 mahasantri putra dan 558 mahasantri putri. Mahasantri tersebut adalah mahasiswa baru yang terdaftar sebagai mahasiswa baru dari semua jurusan. Pada tanggal 17 April 2001, Presiden RI KH. Abdurrahman Wahid meresmikan penggunaan keempat unit hunian *ma'had*, yang masing-masing diberi nama *mabna* (unit hunian) Al-Ghazali, *mabna* Ibn Rusyd, *mabna* Ibn Sina dan *mabna* Ibn khaldun, beberapa bulan kemudian satu unit hunian yang diberi nama Al-Faraby diresmikan penggunaannya oleh wakil presiden RI, H. Hamzah Haz didampingi oleh Wakil Presiden I Sudan saat meresmikan alih status STAIN Malang menjadi Universitas Islam Indonesia Sudan (UIIS).

Semua unit hunian *ma'had* tersebut sekarang dihuni khusus untuk mahasantri putra, sementara untuk mahasantri putri sekarang menempati 4 unit hunian baru yang dibangun sejak tahun 2006 dan telah selesai pembangunannya. Dua unit di antaranya berkapasitas 64 kamar, masing-masing untuk 512 orang bernama *mabna* Ummu Salamah dan *mabna* Asma binti Abi Bakr, 1 unit berkapasitas 60 kamar untuk 480 orang bernama *mabna* Fatimah al-Zahra dan 1 unit berkapasitas 48 kamar untuk 348 orang. Masing-masing kamar dari 4 unit hunian tersebut untuk kapasitas 8 orang. Unit hunian untuk mahasantri putra dan untuk mahasantri putri berada di lokasi yang berjauhan dalam area kampus, semua unit hunian tersebut berkapasitas 425 kamar untuk 3022 orang mahasantri.

Melengkapi nuansa religius dan kultur religiusitas muslim Jawa Timur, maka dibangunlah monumen (prasasti) yang sekaligus menggambarkan visi dan

misi *ma'had* yang tertulis dalam bahasa Arab di depan pintu masuk area unit hunian untuk mahasantri putra. Prasasti tersebut berbunyi:

كونوا اولي الابصار (jadilah kamu orang-orang yang memiliki mata hati);

كونوا اولي النهي (jadilah kamu orang-orang yang memiliki kecerdasan);

كونوا اولي الباب (jadilah kamu orang-orang yang memiliki akal);

وجاهدوا في الله حق جهاده (dan berjuanglah untuk membela agama Allah dengan kesungguhan).

Selanjutnya, untuk mengenang jasa dan historisitas ulama' pejuang Islam di Pulau Jawa, maka ditanam tanah yang diambil dari Wali Songo (Wali Sembilan: simbol perjuangan para ulama' di Jawa) di sekeliling prasasti tersebut. Di samping itu dimaksudkan untuk menanamkan nilai historis perjuangan para ulama', sehingga para mahasantri selalu mengingat urgensi perjuangan atau jihad *li i'laai kalimatillah*. Prasasti yang sama kemudian juga dibangun di depan pintu masuk area unit hunian putri dan di depan kantor rektorat.

2.2 Ta'lim

2.2.1 Ta'lim al-Afkar al-Islamiyyah

Ta'lim Afkar sebagai media proses belajar-mengajar ini diselenggarakan dua kali dalam satu pekan selama dua semester, diikuti oleh semua mahasantri di masing-masing unit hunian dan diasuh oleh para pengasuh dengan menggunakan metode bandongan dan sorongan. Pada setiap akhir semester diselenggarakan tes/evaluasi. Kitab panduan yang dikaji adalah "*Al-Tadzhib*" karya Dr. Musthafa Dieb al Bigha. Kitab ini berisi persoalan fiqh dengan cantuman anotasi Al-Qur'an,

Al-Hadits sebagai dasar normatifnya dan pendapat para ulama sebagai elaborasi dan komparasinya. Capaian *ta'lim* ini adalah masing-masing mahasantri mampu menyebutkan hukum aktifitas/kewajiban tertentu dengan menyertakan dalil (dasar normatifnya), baik Al-Qur'an maupun Al-hadits beserta rawinya. Kitab lain yang dikaji adalah "*Qami' at-Thughyan*" karya Syaikh Muhammad Nawawi bin Umar al Bantani yang berisi tentang pokok-pokok keimanan dan interpretasinya dalam ranah implementatif. Capaian *ta'lim* ini adalah masing-masing mahasantri mampu menyebutkan pokok-pokok keimanan secara komprehensif dan mengaplikasikannya dalam kehidupan sehari-hari.

2.2.2 *Ta'lim* Al-Qur'an

Ta'lim Al-Qur'an ini diadakan dua kali dalam sepekan selama dua semester, diikuti oleh semua mahasantri di masing-masing *mabna* dengan materi yang meliputi *Tashwit*, *Qira'ah*, *Tartil*, *Tarjamah* dan Tafsir dan dibina oleh para *musyrif*, *murabbi*, komunitas HTQ (*Hai'ah Tahfidzul Qur'ani*). Capaian *ta'lim* ini adalah di akhir semester genap semua mahasantri telah mampu membaca Al-Qur'an dengan baik dan benar sesuai tajwid, hafal surat-surat tertentu, bagi mahasantri yang memiliki kemampuan lebih akan diikuti kelas *tarjamah* dan tafsir, sehingga memiliki kemampuan teknik-teknik menerjemahkan dan menafsirkan Al-Qur'an. *Ta'lim* ini diadakan dua kali dalam sepekan selama dua semester, diikuti oleh semua mahasantri di masing-masing *mabna* dengan materi yang meliputi *Tashwit*, *Qira'ah*, *Tartil*, *Tarjamah* dan Tafsir dan dibina oleh para *musyrif*, *murabbi*, komunitas HTQ (*Hai'ah Tahfidzul Qur'ani*). Capaian *ta'lim* ini adalah di akhir semester genap semua mahasantri telah mampu membaca Al-

Qur'an dengan baik dan benar sesuai tajwid, hafal surat-surat tertentu, bagi mahasantri yang memiliki kemampuan lebih akan diikutkan kelas *tarjamah* dan tafsir, sehingga memiliki kemampuan teknik-teknik menerjemahkan dan menafsirkan Al-Qur'an.

2.2.3 Khatm Al-Qur'an

Program *Khatm* Al-Qur'an ini diselenggarakan bersama setiap selesai shalat shubuh pada hari Jum'at. Melalui program ini diharapkan mahasantri mendapatkan kesempatan praktik membaca Al-Qur'an dengan baik dan benar dan diharapkan dapat memperhalus budi, memperkaya pengalaman religiusnya serta memperdalam spiritualitasnya. Program ini juga diselenggarakan tiap Kamis malam pada akhir bulan.

2.2.4 Tashih Qiro'ah Al-Qur'an

Program *tashih* dilaksanakan pada hari aktif belajar, tepatnya dilaksanakan selama 10 bulan dan 5 hari selama satu minggu mulai dari jam 08.00 sampai jam 14.00 WIB disela-sela mahasantri tidak memiliki jadwal kuliah dan dilaksanakan sampai mahasantri mengkhatamkan Al-Qur'an 30 Juz *Binnadhor*. Melalui program ini diharapkan mahasantri mampu mengamalkan teori yang didapatkan saat *ta'lim* Al-Qur'an. Mahasantri juga mengamalkan teori dengan membaca Al-Qur'an secara rutin didepan para *Mushahih* yang secara kapabilitas memiliki kemampuan hafalan Al-Qur'an 30 Juz.

2.2.5 Tahsin Tilawah Al-Qur'an

Program *tahsin* dilaksanakan setiap satu minggu sekali dengan tujuan memperdalam teori Al-Qur'an yang berhubungan dengan hal-hal yang langka

pada Al-Qur'an (ilmu *Gharaib* Al-Qur'an). Pada program ini, mahasantri juga praktik membaca Al-Qur'an dengan lagu yang dibawakan oleh *Muhassin* Al-Qur'an, sehingga santri mendapatkan ilmu tambahan terkait cara membaca Al-Qur'an dengan cara yang indah.

2.3 Algoritma Genetika

2.3.1 Pengertian

Algoritma genetika adalah algoritma pencarian heuristik yang didasarkan atas mekanisme evolusi biologis. Keberagaman pada evolusi biologis adalah variasi dari kromosom antar individu organisme. Variasi kromosom ini akan mempengaruhi laju reproduksi dan tingkat kemampuan organisme untuk tetap hidup. Pada dasarnya ada 4 kondisi yang sangat mempengaruhi proses evaluasi, yaitu:

- a. Kemampuan organisme untuk melakukan reproduksi.
- b. Keberadaan populasi organisme yang bisa melakukan reproduksi.
- c. Keberagaman organisme dalam suatu populasi.
- d. Perbedaan kemampuan untuk *survive*.

Algoritma genetika pertama kali dikembangkan oleh John Holland dari Universitas Michigan (1975). John Holland mengatakan bahwa setiap masalah yang berbentuk adaptasi (alami maupun buatan) dapat diformulasikan dalam terminologi genetika. Algoritma genetika adalah simulasi dari proses evolusi Darwin dan operasi genetika atas kromosom (Kusumadewi, 2003: 279).

Ada beberapa istilah yang perlu dimengerti dalam algoritma genetika, yaitu:

Kromosom: individu yang terdapat dalam satu populasi. Kromosom ini merupakan solusi yang masih berbentuk simbol.

Individu: menyatakan satu nilai atau keadaan yang menyatakan salah satu solusi yang mungkin dari permasalahan yang diangkat.

Genotype (Gen): sebuah nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan *gen* yang dinamakan kromosom. Dalam algoritma genetika, *gen* ini bisa berupa nilai *biner*, *float*, *integer* maupun karakter.

Allele: nilai yang berada dalam *gen*.

Locus: letak suatu *gen* berada dalam suatu kromosom.

Fungsi *fitness*: alat ukur dalam proses evaluasi yang dilalui kromosom. Nilai *fitness* akan menunjukkan kualitas dari suatu kromosom dalam populasi tersebut.

Offspring: anak (generasi berikutnya) yang terbentuk dari gabungan 2 kromosom generasi sekarang yang bertindak sebagai induk (*parent*) dengan menggunakan operator penyilangan (*crossover*) maupun operator mutasi.

Populasi: sejumlah solusi yang mungkin. Populasi awal dibangun secara acak, sedangkan populasi berikutnya merupakan hasil evolusi kromosom-kromosom melalui iterasi yang disebut dengan generasi. **Generasi:** iterasi yang dilakukan untuk menentukan populasi berikutnya.

Secara umum sebuah penerapan algoritma genetika akan melalui siklus yang terdiri dari 4 langkah, yaitu :

- a. Membangun sebuah populasi yang terdiri dari beberapa kromosom.
- b. Evaluasi masing-masing kromosom (Fungsi *fitness*).
- c. Proses seleksi agar didapat kromosom yang terbaik.
- d. Manipulasi genetika untuk menciptakan populasi baru dari kromosom.

2.3.2 Struktur Umum Algoritma Genetika

Pada algoritma ini, teknik pencarian dilakukan sekaligus atas sejumlah solusi yang mungkin yang dikenal dengan istilah populasi. Individu 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 istilah generasi. Pada setiap generasi, kromosom akan melalui proses evaluasi dengan menggunakan alat ukur yang disebut dengan *fitness*. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas kromosom dalam populasi tersebut. Generasi berikutnya dikenal dengan istilah anak (*offspring*) terbentuk dari gabungan 2 kromosom generasi sekarang yang bertindak sebagai induk (*parent*) dengan menggunakan operator *crossover*. Selain operator *crossover*, 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. Setelah melalui beberapa generasi, maka algoritma ini akan konvergen ke *kromosom* terbaik (Kusumadewi, 2003: 280).

Secara umum, algoritma genetika dapat diilustrasikan dalam blok diagram berikut ini:



Gambar 2.1 Blok Diagram Algoritma Genetika

Proses dimulai dari tahap pengkodean untuk dibangkitkan populasi awal, kemudian kromosom dengan nilai *fitness* tertinggi akan menjadi *parent* untuk diseleksi dan dihasilkan kromosom anak untuk melakukan proses *crossover* dan mutasi sehingga dihasilkan generasi baru. Jika kondisi terminasi tercapai, maka generasi baru menjadi generasi selanjutnya. Jika tidak, ulangi proses seleksi *parent*.

2.3.3 Komponen-komponen Utama Algoritma Genetika

Ada enam komponen utama dalam algoritma genetika, yaitu (Kusumadewi, 2003: 280-283):

a. Teknik Pengkodean

Teknik pengkodean disini meliputi pengkodean *gen* dari kromosom. *Gen* merupakan bagian dari kromosom. Satu *gen* biasanya akan mewakili satu variabel.

Algoritma genetika merepresentasikan *gen* sebagai bilangan real atau desimal, disini digunakan *discrete decimal encoding*, yaitu setiap *gen* berupa deretan bilangan bulat dalam interval [0,9].

b. 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.

c. Fungsi Evaluasi

Ada 2 hal yang harus dilakukan dalam melakukan evaluasi kromosom, yaitu: evaluasi fungsi objektif (fungsi tujuan) dan konversi fungsi objektif dengan nilai yang tidak negatif. Apabila ternyata fungsi objektif memiliki nilai negatif, maka perlu ditambahkan suatu konstanta C agar nilai *fitness* yang terbentuk menjadi tidak negatif.

d. Seleksi

Seleksi ini bertujuan untuk memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang paling fit. Seleksi akan

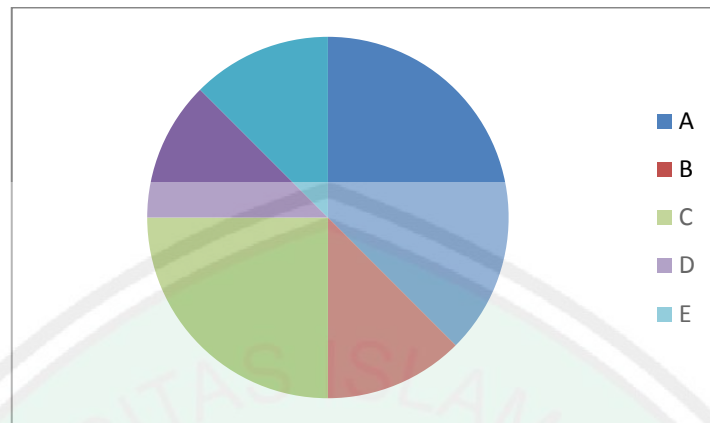
menentukan individu-individu mana saja yang akan dipilih untuk dilakukan rekombinasi dan bagaimana *offspring* terbentuk dari individu-individu terpilih tersebut. Langkah pertama yang dilakukan dalam seleksi ini adalah pencarian nilai *fitness*. Masing-masing individu dalam suatu wadah seleksi akan menerima probabilitas reproduksi yang tergantung pada nilai objektif dirinya sendiri terhadap nilai objektif dari semua individu dalam wadah seleksi tersebut. Nilai *fitness* inilah yang nantinya akan digunakan pada tahap-tahap seleksi berikutnya.

Metode *roulette wheel selection* atau seleksi roda *roulette* merupakan metode yang paling sederhana. Pada metode ini, individu-individu dipetakan dalam suatu segmen garis secara berurutan sedemikian sehingga tiap-tiap segmen individu memiliki ukuran yang sama dengan ukuran *fitness*-nya.

Pada metode ini induk dipilih berdasarkan nilai *fitness*-nya, semakin besar nilai *fitness* maka semakin besar kemungkinan untuk terpilih menjadi induk. Diandaikan semua kromosom diletakkan pada sebuah roda *roulette*, besarnya kemungkinan bagi setiap kromosom adalah tergantung dari nilai *fitness*-nya seperti pada contoh berikut:

Kromosom	<i>Fitness</i>
A	15
B	5
C	10
D	5
E	5

Gambar 2.2 Contoh populasi dengan 5 kromosom



Gambar 2.3 Probabilitas kromosom dalam roda *roulette*

Pada Gambar 2.8 merupakan contoh dalam satu populasi terdiri dari lima kromosom. Pada tiap kromosom memiliki nilai *fitness* yang berbeda-beda. Pada Gambar 2.9 dapat diketahui probabilitas terpilihnya masing-masing kromosom untuk menjadi induk. Pada kromosom A memiliki nilai *fitness* 15 dan nilai tersebut nilai *fitness* tertinggi pada populasi tersebut, sehingga kromosom A memiliki probabilitas terbesar untuk terpilih menjadi induk.

e. Operator Genetika

Ada 2 operator Genetika, yaitu:

1. Operator untuk melakukan rekombinasi, rekombinasi yang digunakan adalah rekombinasi *one-point crossover*
2. Operator untuk mutasi, mutasi yang digunakan adalah *swapping mutation*

f. Penentuan Parameter

Parameter di sini adalah parameter kontrol algoritma genetika, yaitu: ukuran populasi (*popsize*), peluang *crossover* (P_c), dan peluang

mutasi (P_m). Nilai parameter ini ditentukan juga berdasarkan permasalahan yang akan dipecahkan. Ada beberapa rekomendasi yang bisa digunakan, antara lain (Kusumadewi, 2003: 283):

Untuk permasalahan yang memiliki kawasan solusi cukup besar, De Jong merekomendasikan untuk nilai parameter kontrol:

$$(P_{\text{popsize}}; P_c; P_m) = (50; 0,6; 0,001)$$

Bila rata-rata *fitness* setiap generasi digunakan sebagai indikator, maka Grefensette merekomendasikan:

$$(P_{\text{popsize}}; P_c; P_m) = (30; 0,95; 0,01)$$

Bila *fitness* dari individu terbaik dipantau pada setiap generasi maka usulannya adalah:

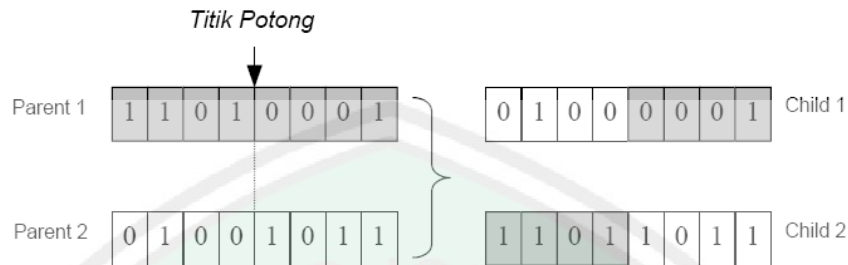
$$(P_{\text{popsize}}; P_c; P_m) = (80; 0,45; 0,01)$$

Ukuran populasi sebaiknya tidak lebih kecil dari 30, untuk sembarang jenis permasalahan.

2.3.4 *Crossover*

Crossover (perkawinan silang) adalah operator genetika yang utama. Operator bekerja dengan mengambil dua individu dan memotong *string* kromosom mereka pada posisi yang terpilih secara acak, untuk memproduksi dua *segment head* dan dua *segment tail* (Son, 2007: 185). *Crossover* bertujuan menambah keanekaragaman *string* dalam satu produksi dengan penyilangan antar *string* yang diperoleh dari reproduksi sebelumnya (Anita dan Muhammad, 2006: 196).

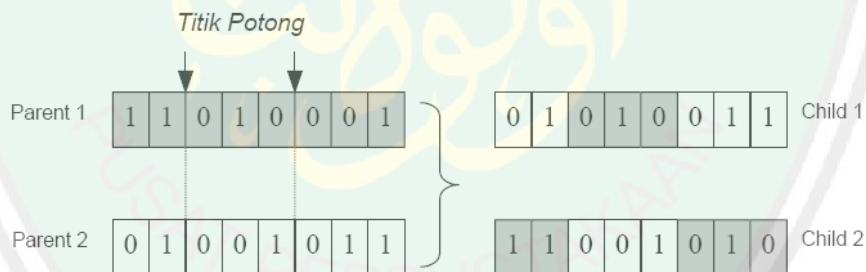
a. *Crossover* satu titik.



Gambar 2.4 Crossover satu titik
(Suyanto, 2011:212)

Crossover satu titik dengan menentukan titik potong tertentu secara acak dari kromosom, kemudian menukarnya dengan kromosom yang lainnya sehingga menghasilkan kromosom anak.

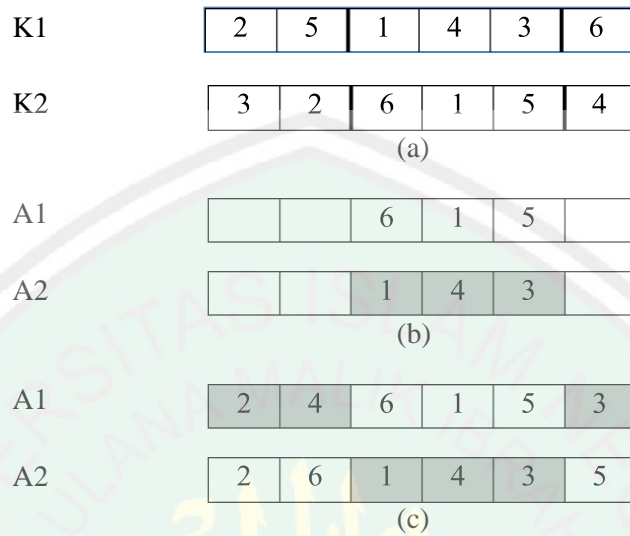
b. *Crossover* dua titik.



Gambar 2.5 Crossover dua titik
(Suyanto, 2011:212)

Operator *crossover* dapat dilakukan dengan lebih dari dua titik. Tetapi jumlah titik potong yang semakin banyak akan memperendah kualitas solusi yang didapatkan. Hal ini disebabkan operasi *crossover* terlalu sering merusak kromosom yang baik.

c. *Order crossover* (OX)



Gambar 2.6 *Order Crossover*

(Suyanto, 2005:66)

Crossover menggunakan skema *order crossover* yaitu pertama, dibangkitkan dua titik pindah silang pada dua parent K1 dan K2 (a). Gen-gen yang berada di antara kedua titik silang ditukarkan (b). Gen-gen pada K1 yang belum ada pada A1 dimasukkan ke tempat yang kosong secara berurutan. Hal ini sama juga dilakukan untuk A2.

2.3.5 Mutasi

Mutasi merupakan proses mengubah nilai dari satu atau beberapa *gen* dalam suatu kromosom (Anita dan Muhammad, 2006:197). Mutasi ini berperan untuk menggantikan *gen* yang hilang dari populasi akibat proses seleksi yang memungkinkan munculnya kembali *gen* yang tidak muncul pada inisialisasi populasi (Kusumadewi, 2003: 296). Mutasi diterapkan dengan probabilitas sangat kecil. Jika mutasi dilakukan terlalu sering, maka akan menghasilkan individu

yang lemah karena konfigurasi bit pada kromosom yang unggul akan dirusak (Suyanto, 2011: 213). Mutasi ini bukanlah operator yang utama, yang dilakukan secara acak pada *gen* dengan kemungkinan yang kecil. (P_m sekitar 0,001) (Son, 2007: 185). Peluang mutasi (P_m) mengendalikan banyaknya *gen* baru yang akan dimunculkan untuk dievaluasi. Jika peluang mutasi terlalu kecil, banyak *gen* yang mungkin berguna tidak pernah dievaluasi. Tetapi bila peluang mutasi terlalu besar, maka akan terlalu banyak gangguan acak, sehingga anak akan kehilangan kemiripan dari induknya.

a. Mutasi Real

Pada mutasi bilangan real, ukuran langkah mutasi biasanya sangat sulit ditentukan. Ukuran yang sangat kecil biasanya sering mengalami kesuksesan, namun ada kalanya ukuran yang lebih besar akan berjalan lebih cepat.

b. Mutasi Biner

Cara sederhana untuk mendapatkan mutasi biner adalah dengan mengganti satu atau beberapa nilai *gen* dari kromosom. Langkah langkah mutasi ini adalah :

- (1) Hitung jumlah *gen* pada populasi (panjang kromosom dikalikan jumlah populasi)
- (2) Pilih secara acak *gen* yang akan dimutasi
- (3) Tentukan kromosom dari *gen* yang akan dimutasi.
- (4) Ganti nilai *gen* (0 ke 1 atau 1 ke 0) dari kromosom yang akan dimutasi tersebut.

Berdasarkan bagian yang termutasi, proses mutasi dapat dibedakan menjadi tiga bagian:

- a. Tingkat Kromosom, semua gen dalam kromosom berubah.



Gambar 2.7 Mutasi tingkat kromosom

(Suyanto, 2011: 213)

Pada contoh ini, gen yang tadinya bernilai 1 menjadi 0, sebaliknya gen yang bernilai 0 menjadi 1.

- b. Tingkat Gen, semua bit dalam satu gen akan berubah.



Gambar 2.8 Mutasi tingkat gen

(Suyanto, 2011: 213)

- c. Tingkat Bit, hanya satu bit yang berubah.



Gambar 2.9 Mutasi tingkat bit

(Suyanto, 2011: 213)

2.3.6 Kriteria Penghentian

Terdapat berbagai macam kriteria penghentian yang bisa digunakan, tiga di antaranya adalah (Suyanto, 2011: 214-215):

- a. Memberikan batasan jumlah iterasi. Apabila batas iterasi tersebut dicapai, iterasi dihentikan dan laporkan individu bernilai *fitness* tertinggi sebagai solusi terbaik.
- b. Memberikan batasan waktu proses algoritma genetika. Kriteria ini digunakan pada sistem waktu nyata (*real time systems*), dimana solusi harus ditemukan paling lama, misalkan 3 menit. Dengan demikian, algoritma genetika bisa dihentikan ketika proses sudah berlangsung selama hampir 3 menit.
- c. Menghitung kegagalan penggantian anggota populasi yang terjadi secara berurutan sampai jumlah tertentu. Misalkan setelah 100 iterasi tidak ada penggantian individu dalam populasi karena individu anak yang dihasilkan selalu memiliki nilai *fitness* lebih rendah daripada orang tuanya. Dalam kondisi seperti ini, kita bisa menghentikan iterasi.

2.4 UML (*Unified Modelling Language*)

Dalam melakukan pemodelan sistem, dapat digunakan notasi-notasi UML yang digambarkan secara elektronik. Notasi UML sendiri dibuat oleh Grady Booch, DR. James Rumbaugh, Ivar Jacobson, Rebecca Wirfs-Brock, Peter Yourdon dkk. Jacobson menulis tentang pendefinisian tentang persyaratan-persyaratan sistem yang disebut *use case*, juga mengembangkan sebuah metode untuk perancangan sistem yang disebut *Object-Oriented Software Engineering (OOSE)* yang berfokus pada analisis.

UML menyediakan beberapa diagram visual yang menunjukkan berbagai aspek dalam sistem. Ada beberapa diagram yang disediakan dalam UML, antara lain :

2.4.1 Use Case Diagram

Use case diagram menyediakan interaksi antara *use case* dan *actor*. *Actor* dapat berupa orang, peralatan atau sistem lain yang berinteraksi dengan sistem yang sedang dibangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan pemakai.

2.4.2 Activity Diagram

Activity diagram menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktifitas dapat digunakan untuk menunjukkan aliran kerja bisnis (*business workflow*). Dapat juga digunakan untuk menggambarkan aliran kejadian (*workflow of events*) dalam *use case*.

2.4.3 Sequence Diagram

Sequence diagram digunakan untuk menunjukkan aliran fungsionalitas dalam *use case*.

2.4.4 Collaboration Diagram

Collaboration diagram menunjukkan informasi yang sama persis seperti *sequence diagram*, akan tetapi dalam bentuk dan tujuan yang berbeda. Pada *sequence diagram* keseluruhan interaksi berdasarkan urutan waktu, akan tetapi pada *collaboration diagram* interaksi antar obyek atau *actor* di tunjukkan arah panah tanpa keterangan waktu.

2.4.5 Class Diagram

Class diagram menunjukkan interaksi antar kelas dalam sistem. Para analis menggunakan diagram ini untuk menunjukkan detail sistem, sedangkan arsitek sistem mempergunakan diagram ini untuk melihat rancangan sistem.

2.4.6 Statechart Diagram

Statechart diagram menyediakan sebuah cara untuk memodelkan bermacam-macam keadaan yang mungkin dialami oleh sebuah obyek. Jika dalam diagram kelas menunjukkan gambaran statis kelas-kelas dan relasinya, *statechart diagram* digunakan untuk memodelkan tingkah laku dinamik sistem.

2.4.7 Component Diagram

Component diagram menunjukkan model secara fisik komponen perangkat lunak dalam sistem dan hubungannya. Ada dua tipe komponen yaitu *executable* dan *libraries code*. Masing-masing kelas dalam model akan dipetakan ke sebuah komponen kode pustaka. Setelah komponen dibuat, kemudian ditambahkan dalam diagram komponen dengan memberikan relasi antara komponen-komponennya. Relasi yang terjadi antar komponen hanya satu tipe relasi yaitu dependensi yang menunjukkan ketergantungan *compile-time* dan *runtime* antara komponen-komponen tersebut.

2.4.8 Deployment Diagram

Deployment Diagram menampilkan rancangan fisik jaringan yang terdapat berbagai komponen.

Diagram-diagram tersebut tidak harus digunakan semua dalam pengembangan perangkat lunak, penggunaan diagram disesuaikan dengan

kebutuhan. Secara konteks, penerapan dari diagram-diagram tersebut dapat digambarkan menggunakan *Rational Rose*.

Rational Rose merupakan sebuah tool pemodelan visual, yang menggunakan UML (*Unified Modeling Language*) sebagai bahasanya. *Rational Rose* mendukung *roundtrip engineering*, yang berarti Anda dapat mengenerate kode dari model yang Anda buat, dan dapat melakukan *reverse engineering* dari *source code* yang ada untuk menghasilkan gambaran arsitektur dari software aplikasi Anda.

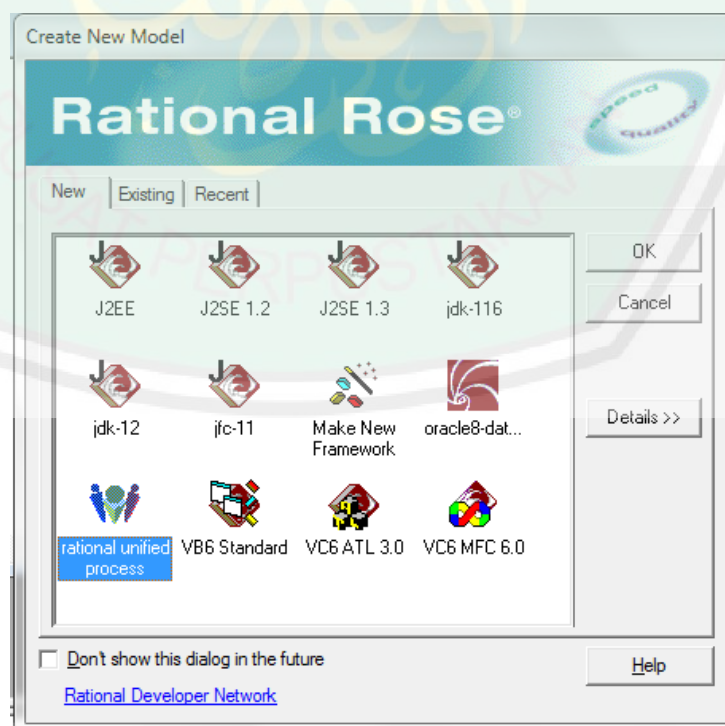
Rational Rose merupakan salah satu software yang paling banyak digunakan untuk melakukan design software melalui pendekatan UML (*Unified Modelling Language*). *Rational Rose* merupakan software yang menyediakan banyak fungsi-fungsi seperti: design proses, *generate code*, *reverse engineering*, serta banyak fungsi-fungsi yang lain. *Rational Rose* merupakan tool yang sangat mudah karena sudah menyediakan contoh-contoh design dari beberapa software. Sebelum menggunakannya, terlebih dahulu pahami tentang UML (*Unified Modelling Language*). Tutorial ini ditujukan untuk pembaca yang sudah memiliki pengetahuan tentang UML.

Edisi *Rational Rose* :

- a. *Rose Modeler* – Tidak mendukung bahasa pemrograman apapun.
- b. *Rose Profesional* – Mendukung satu bahasa pemrograman.
- c. *Rose Enterprise* – Mendukung banyak bahasa, yaitu CORBA, VC++, VB, Java, dan sebagainya.

Untuk membuat file *Rational Rose*, buka aplikasi *Rational Rose* sehingga anda akan dihadapkan pada beberapa pilihan seperti terlihat pada gambar 1, contohnya: jdk-12, jdk-116, jenterprise, jfc-11, Oracle Database, *Rational Unified Process* (RUP), VB6 Standard serta model-model yang lain. Model-model ini merupakan pilihan apakah ingin membuat file *Rational Rose* yang mengandung komponen Java, Visual Basic atau Visual C++. Hal ini dibutuhkan ketika nanti akan melakukan *generate code* (membuat contoh kode) dari design yang telah dibuat.

Rational Unified Process (RUP) merupakan pilihan yang tepat untuk membuat suatu design yang lengkap dengan cara mudah karena model ini merupakan template, sehingga kita dapat langsung melakukan perubahan dengan mengganti *use case*, *actors*, *class diagram* yang telah ada pada template RUP.



Gambar 2.10 Tampilan awal *Rational Rose*

Langkah membangun suatu aplikasi:

- a. Melihat keseluruhan sistem.
- b. Membuat suatu model bisnis.
- c. Menentukan *Requirements* yang harus dipenuhi oleh software yang akan dibuat. (*Use case Model*)
- d. Membuat desain mengenai arsitektur software. (Model Desain)
- e. Merealisasikan ke dalam *source code*

Terdapat empat view, yang tiap-tiap view tersebut menjelaskan penegasan aspek yang berbeda mengenai sistem yang dimodelkan.

- a. *Use Case View* untuk memahami dan menggunakan sistem yang dimodelkan (Bagaimana *actor* dan *use case* berinteraksi). Terdapat beberapa diagram dalam view ini, yaitu :

- *Use Case Diagram*
- *Sequence Diagram*
- *Collaboration Diagram*
- *Activity Diagram*

- b. *Logical View* mengarah pada persyaratan fungsional sistem (kelas- kelas dan hubungan antarkelas tersebut). Terdapat beberapa diagram dalam view ini, yaitu :

- *Class Diagram*
- *Sequence Diagram*
- *Collaboration Diagram*
- *Statechart Diagram*

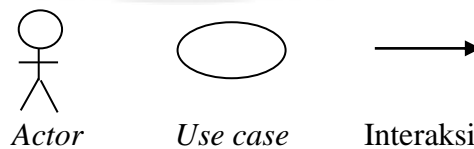
- c. *Componen View* Pengaturan *software* (informasi komponen *software*, komponen tereksekusi dan library untuk sistem yang dimodelkan). Satu jenis diagram yang terdapat view ini, yaitu *Component Diagram*.
- d. *Deployment View* Pemetaan setiap proses ke dalam hardware. Satu jenis diagram yang terdapat pada view ini, yaitu *Deployment Diagram*.

Macam-macam Diagram dalam *Rational Rose*:

- a. *Use case diagram* menjelaskan manfaat sistem jika dilihat dari sudut pandang orang yang berada di luar sistem (*actor*). Digunakan untuk menangkap *requirement system* dan untuk memahami bagaimana sistem seharusnya bekerja.

Sebuah use case diagram mengandung :

- *Actor* : Pengguna *software* aplikasi, berupa manusia, *hardware* atau sistem informasi yang lain. *Actor* dapat memasukan informasi ke sistem, menerima informasi dari sistem, atau juga keduanya.
- *Use case* : Perilaku (apa yang pengguna kerjakan) *software* aplikasi, termasuk interaksi antara *actor* dengan *software* aplikasi tersebut.



Gambar 2.11 Komponen *use case diagram*
(Sholiq, 2006:8)

Terdapat 2 macam tipe relasi :

1. *include*, menggambarkan bahwa suatu *use case* seluruhnya meliputi fungsionalitas dari *use case* lainnya.
2. *extends*, suatu *use case* merupakan tambahan fungsionalitas dari *use case* lain jika kondisi atau syarat tertentu terpenuhi.

Relasi menggambarkan hubungan antara *actor* dan *use case*.

Relasi-relasi tersebut dapat dibagi menjadi: *Unidirectional Association*, *Generalization* dan *Dependency*.

- b. *Class Diagram* memperlihatkan hubungan antarkelas dari suatu sistem dan penjelasan detil tiap-tiap kelas di dalam model desain (dalam *logical view*) dari suatu sistem.
- c. *Sequence Diagram* menjelaskan interaksi objek yang disusun dalam satu urutan waktu. *Sequence diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu di dalam *use case*. Tipe diagram ini menekankan pada urutan kejadian.
- d. *Collaboration Diagram* digunakan sebagai alat untuk menggambarkan interaksi yang mengungkapkan keputusan mengenai perilaku sistem. Tipe diagram ini menekankan pada hubungan antar objek. Dalam satu diagram ini terdapat beberapa *object*, *link* dan *message*.
- e. *Activity Diagram* memodelkan alur kerja sebuah proses bisnis dan urutan aktivitas dalam sebuah proses. Diagram ini mirip dengan *flowchart*. *Activity diagram* terdiri dari beberapa elemen :

1. Sebuah keadaan awal (*start state*) dan keadaan akhir (*end state*)
2. Aktivitas-aktivitas, menggambarkan satu tahapan dalam *workflow* tersebut.
3. Transisi, menggambarkan keadaan apa yang mengikuti suatu keadaan lain.
4. Keputusan (*decision*), elemen yang menyediakan pilihan alur dalam *workflow*
5. Batang penyelarasan (*synchronization bar*), memperlihatkan subalur paralel.
6. *Swimlane*, menjelaskan pemeran bisnis yang bertanggungjawab terhadap aktivitas yang dikandungnya.



Gambar 2.12 Komponen *Activity diagram*

- f. *Statechart Diagram* memodelkan perilaku dinamis satu kelas atau objek. *Statechart* memperlihatkan urutan keadaan sesaat (*state*) yang dilalui sebuah objek, kejadian yang menyebabkan sebuah transisi dari suatu *state* kepada yang lain dan aksi yang menyebabkan perubahan suatu *state* atau aktivitas.
- g. *Component Diagram* menggambarkan alokasi semua kelas dan objek ke dalam komponen-komponen fisik sistem *software*. Diagram ini memperlihatkan pengaturan dan ketergantungan antara komponen-

komponen *software*, seperti *source code*, *binary code*, dan komponen tereksekusi.

- h. *Deployment Diagram* memperlihatkan pemetaan *software* kepada *hardware*. Setiap model hanya memiliki satu diagram ini.



BAB III

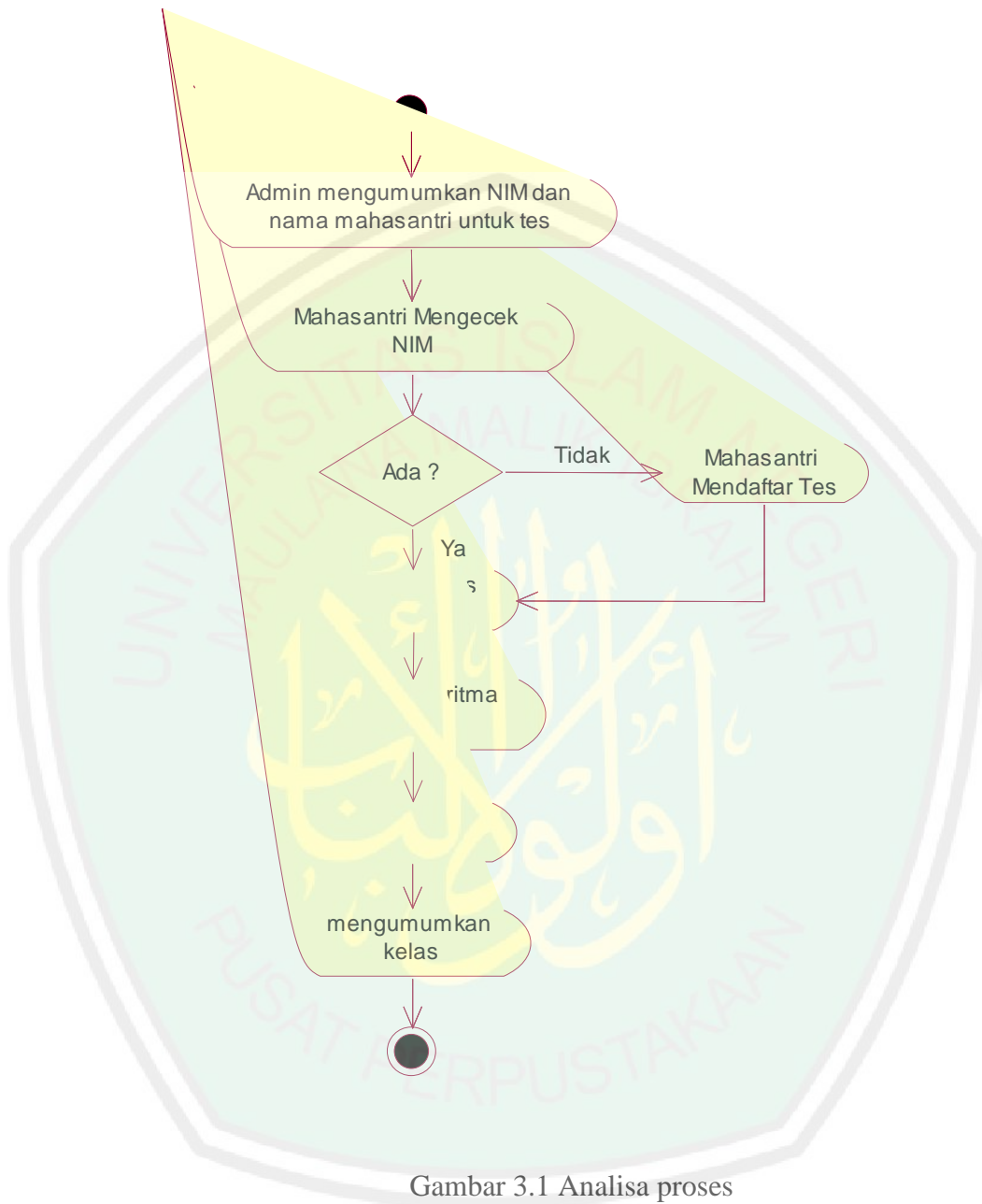
ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

3.1.1 Analisa Proses

Proses penempatan kelas *ta'lim afkar* mahasiswa baru di *ma'had* selama ini dilakukan dengan sederhana, yaitu dengan *mensorting* hasil dari nilai *placement test* sehingga dalam satu kelas bisa terdapat lebih dari dua mahasiswa yang berada di kelas yang bukan tingkatannya, karena yang diperhatikan adalah nilai tes saja. Dalam aplikasi ini, pemrosesan tidak hanya berdasarkan hasil tes saja, akan tetapi ada kriteria lain yang digunakan, yaitu asal sekolah. Kedua kriteria ini nantinya akan di proses dalam algoritma genetika untuk menghasilkan penempatan kelas.

Berikut gambar analisa proses, pada gambar 3.1 diketahui proses dimulai mahasiswa mengecek NIM dan namanya untuk mengikuti tes sampai pengumuman penempatan kelas.



Gambar 3.1 Analisa proses

3.1.2 Analisa Keluaran

Keluaran yang dihasilkan dari sistem ini berupa daftar nama mahasiswa sesuai dengan kelas yang ditempati. Daftar nama tersebut dicocokkan dengan nama dan NIM mahasiswa yang bersangkutan.

3.1.3 Analisa Masukan

Masukan yang digunakan adalah data mahasantri baru yang tinggal di *mabna* masing-masing beserta nilai *placement test*.

3.1.4 Identifikasi Kebutuhan

Dari analisa tersebut, perlu adanya sebuah sistem yang dapat mengelola pengklasifikasian kelas *ta'lim afkar* mahasantri baru di *ma'had* dengan baik dan dapat memberikan solusi yang terbaik dari permasalahan tersebut. Untuk itu, berikut ini akan dijelaskan kebutuhan-kebutuhan dari sistem yang ingin dicapai:

- a. Parameter-parameter yang digunakan dalam pengklasifikasian kelas *ta'lim afkar* mahasantri baru di *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang.
- b. Program khusus untuk menangani masalah tersebut.
- c. Masukan berupa data mahasantri baru dan nilai *placement test*.

3.1.5 Kebutuhan Sistem

Komponen-komponen yang dibutuhkan untuk membangun sistem tersebut mulai tahap penelitian sampai tahap implementasi menggunakan perangkat komputer dengan spesifikasi sebagai berikut.:

- a. *Hardware* dan *software* untuk pembuatan sistem.

Hardware:

- 1 unit laptop
- *Processor* Intel Core 2 Duo 2,00GHz
- *Memory* 2 GB
- *Harddisk* 250 GB

- *Mouse, Keyboard*

Software:

- *Windows 7 Ultimate* sebagai sistem operasi
- *Xampp 1.7.4* sebagai webserver
- *Notepad++*
- *Mozilla Firefox*
- *Adobe Photoshop*

b. *Hardware dan software minimal untuk menjalankan sistem.*

Hardware:

- *Processor Pentium III*
- *Memory 128 MB*
- *Harddisk 20 GB*
- *Mouse, Keyboard dan Monitor*

Software:

- *Windows XP Professional*
- *Mozilla Firefox*

3.2 Pemodelan UML

3.2.1 Use Case Diagram

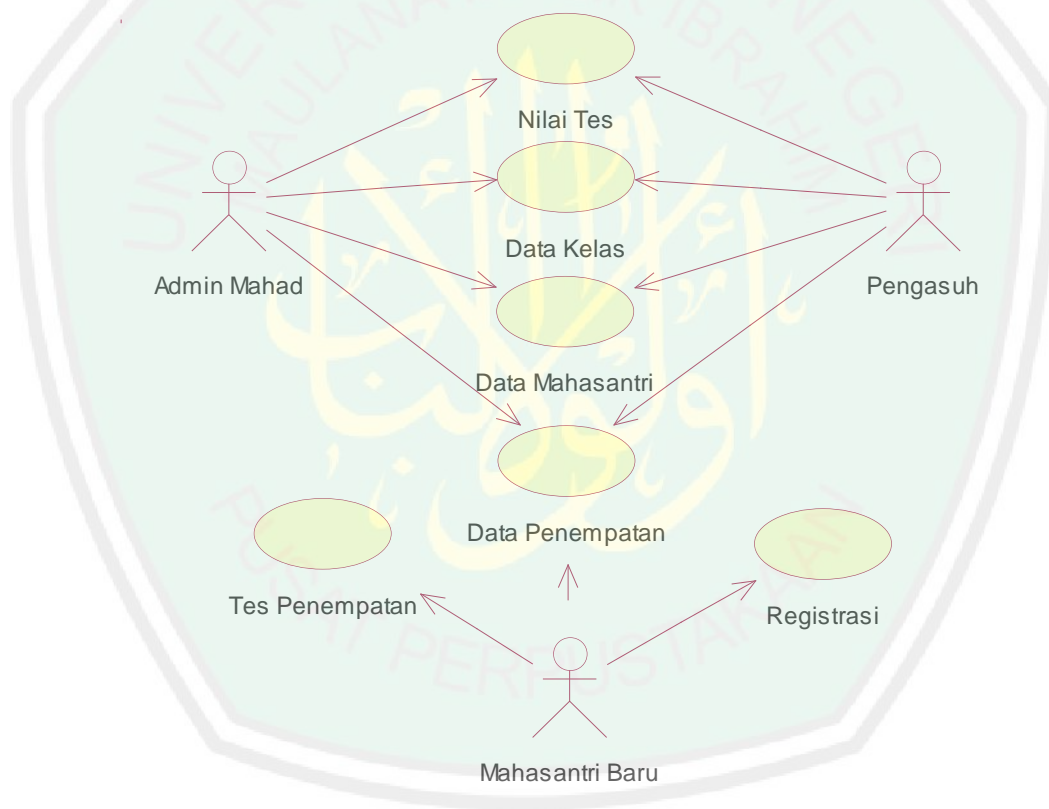
Use case diagram digunakan untuk menggambarkan seluruh proses dalam sistem dan memodelkannya, Dengan *use case diagram* ini dapat diketahui proses yang terjadi pada aktivitas sistem penempatan kelas mahasantri baru. Dengan

diagram ini juga dapat diketahui fungsi yang digunakan oleh sistem yang digunakan. Sebuah use case terdiri dari:

Actor : Pengguna software aplikasi, berupa manusia, hardware atau sistem informasi yang lain. Dalam sistem ini, *actor* meliputi admin dan mahasantri.

Use case : Perilaku, interaksi antara *actor* dengan *software* aplikasi tersebut.

Dalam hal ini meliputi operator-operator algoritma genetika.



Gambar 3.2 Use Case Diagram

Deskripsi use case diagram dari gambar 3.2

a. *Use case* : Registrasi

Actor : Mahasantri Baru

Deskripsi : Mahasantri baru registrasi untuk mengikuti tes

b. *Use case* : Tes penempatan

Actor : Mahasantri Baru

Deskripsi : Mahasantri baru mengikuti *placement tes*

c. *Use case* : Data penempatan

Actor : Mahasantri baru, Admin *ma'had*, Pengasuh

Deskripsi : Mahasantri baru dan pengasuh melihat pengumuman hasil *placement tes* beserta penempatan kelas, Admin maintain data penempatan

d. *Use case* : Data Mahasantri

Actor : Pengasuh, Admin *ma'had*

Deskripsi : Pengasuh melihat data seluruh mahasantri baru, Admin maintain data mahasantri

e. *Use case* : Data Kelas

Actor : Pengasuh, Admin *ma'had*

Deskripsi : Pengasuh melihat data kelas *ta'lim*, Admin maintain data kelas

f. *Use case* : Data nilai tes

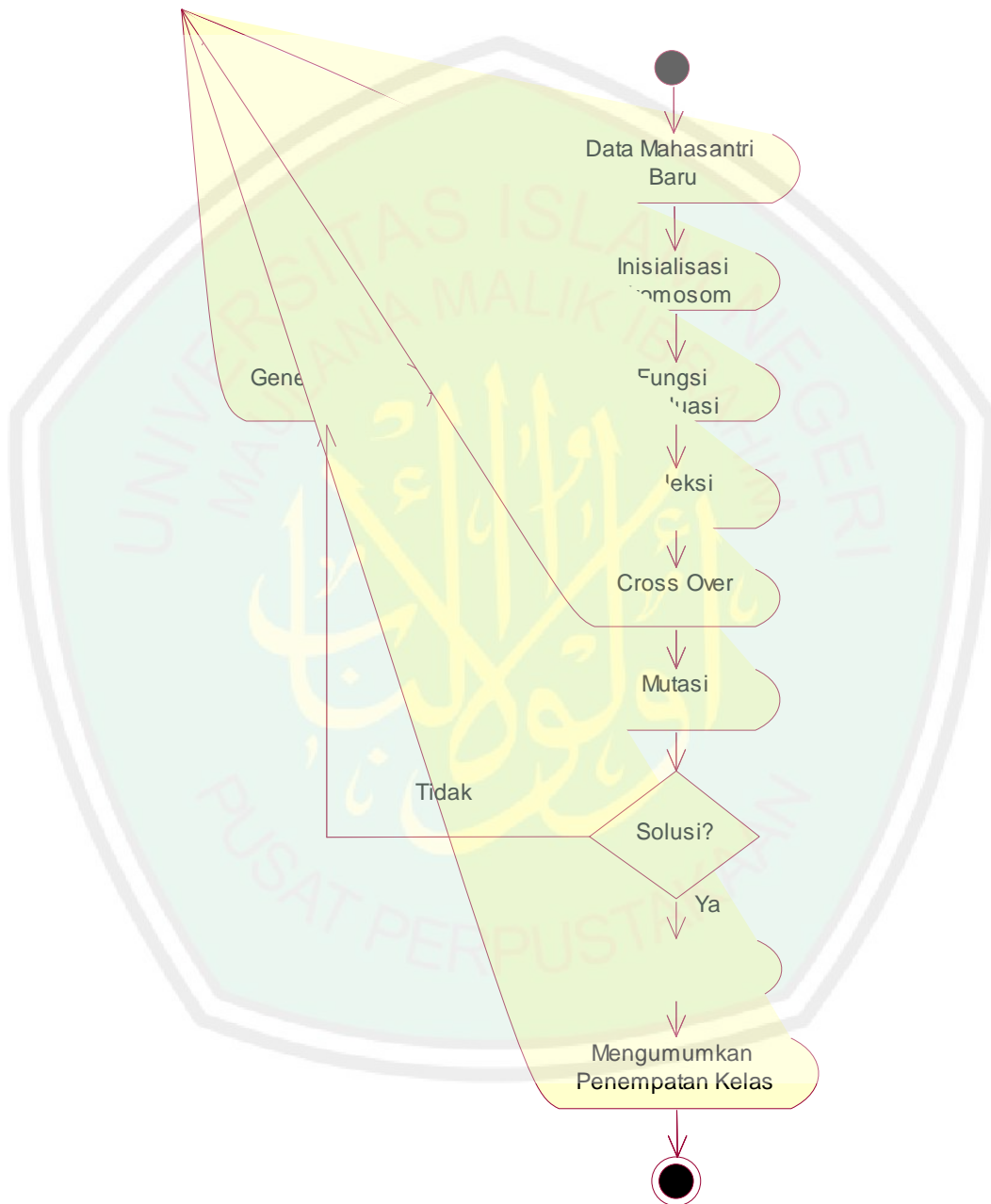
Actor : Pengasuh, Admin *ma'had*

Deskripsi : Pengasuh melihat nilai tes mahasantri baru, Admin maintain nilai tes

3.2.2 Activity Diagram

Activity diagram merupakan suatu bentuk *flow diagram* yang memodelkan alur kerja sebuah proses dan urutan aktivitas sebuah proses. Diagram ini mirip

dengan *flowchart* karena kita dapat memodelkan sebuah alur kerja dari suatu aktivitas ke aktivitas lainnya.



Gambar 3.3 Activity diagram alur algoritma genetika

Berikut gambaran *activity diagram* dalam proses penempatan kelas mahasantri baru:

a. *Activity diagram* tes penempatan



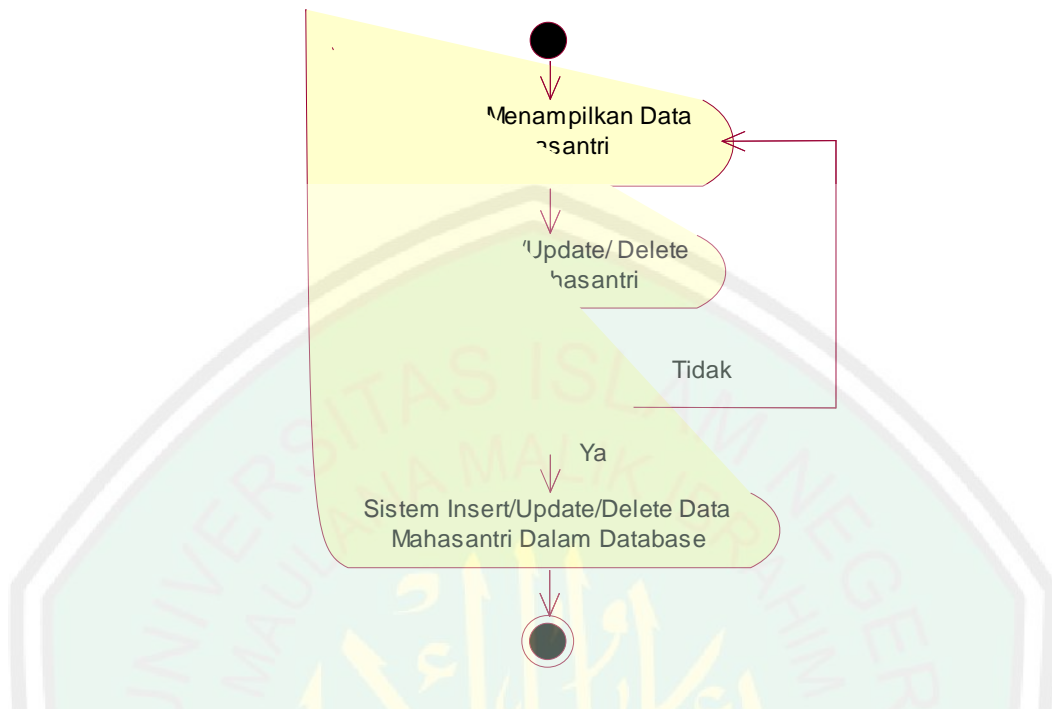
Gambar 3.4 *Activity diagram* tes penempatan

b. *Activity diagram* mengumumkan hasil penempatan



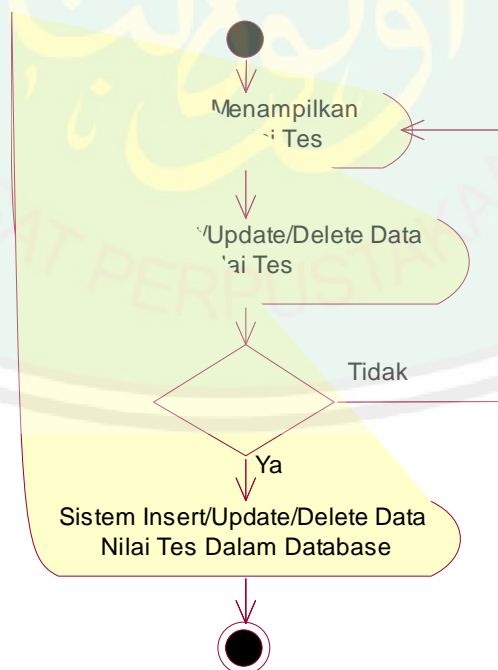
Gambar 3.5 *Activity diagram* mengumumkan hasil penempatan

c. *Activity diagram* maintain data mahasantri



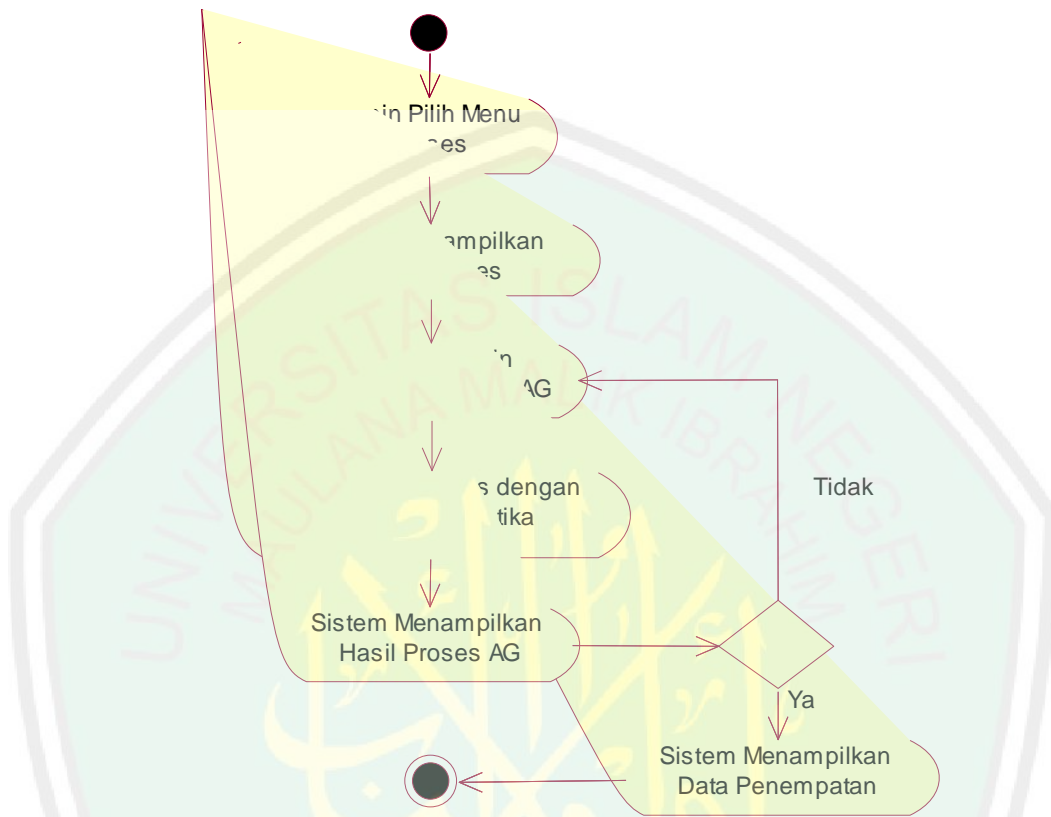
Gambar 3.6 *Activity diagram* maintain data maha-santri

d. *Activity diagram* maintain nilai tes



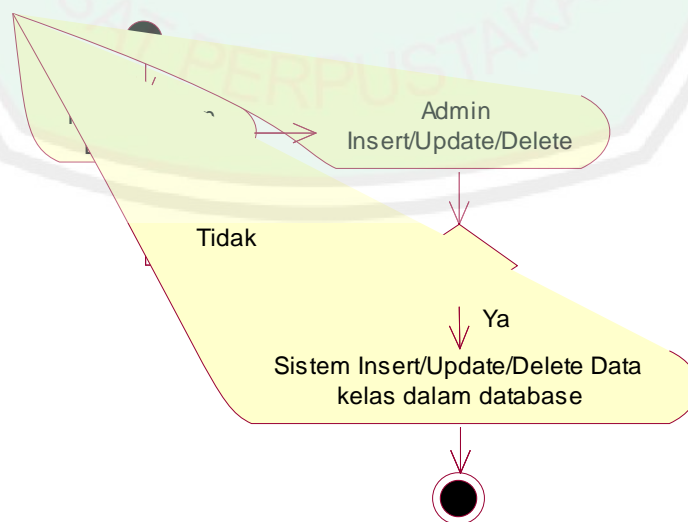
Gambar 3.7 *Activity diagram* maintain nilai tes

e. *Activity diagram* maintain data penempatan



Gambar 3.8 *Activity diagram* maintain data penempatan

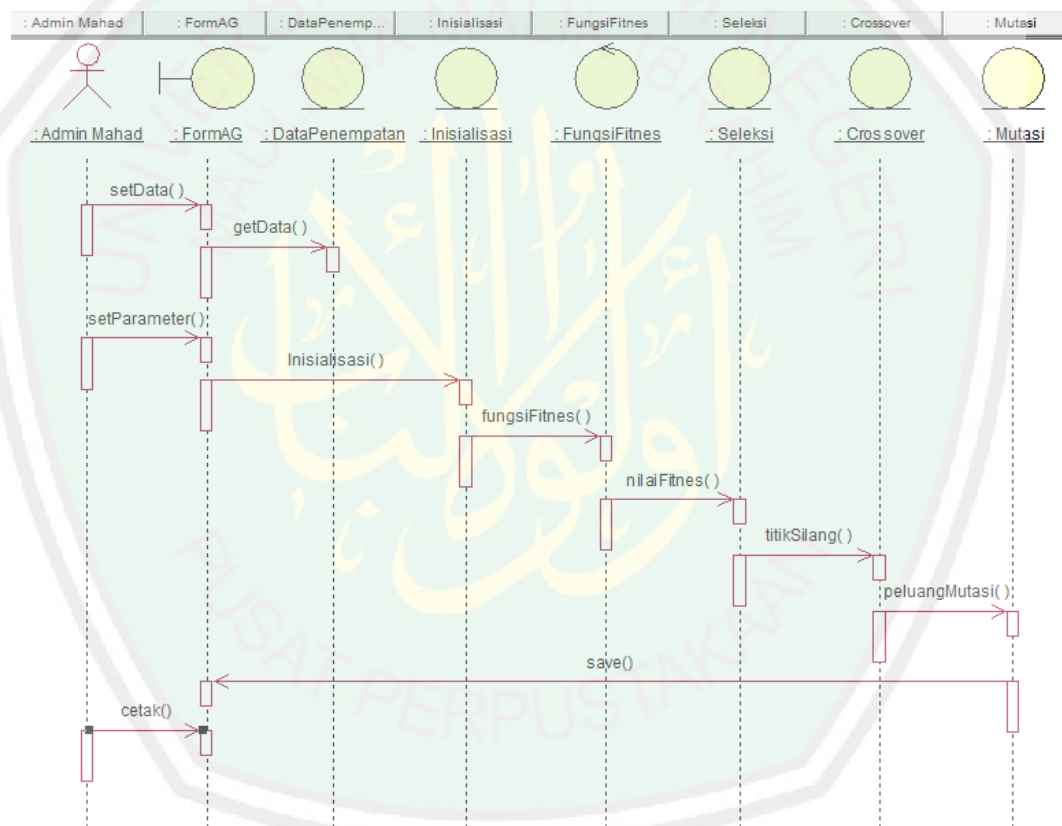
f. *Activity diagram* maintain data kelas



Gambar 3.9 *Activity diagram* maintain data kelas

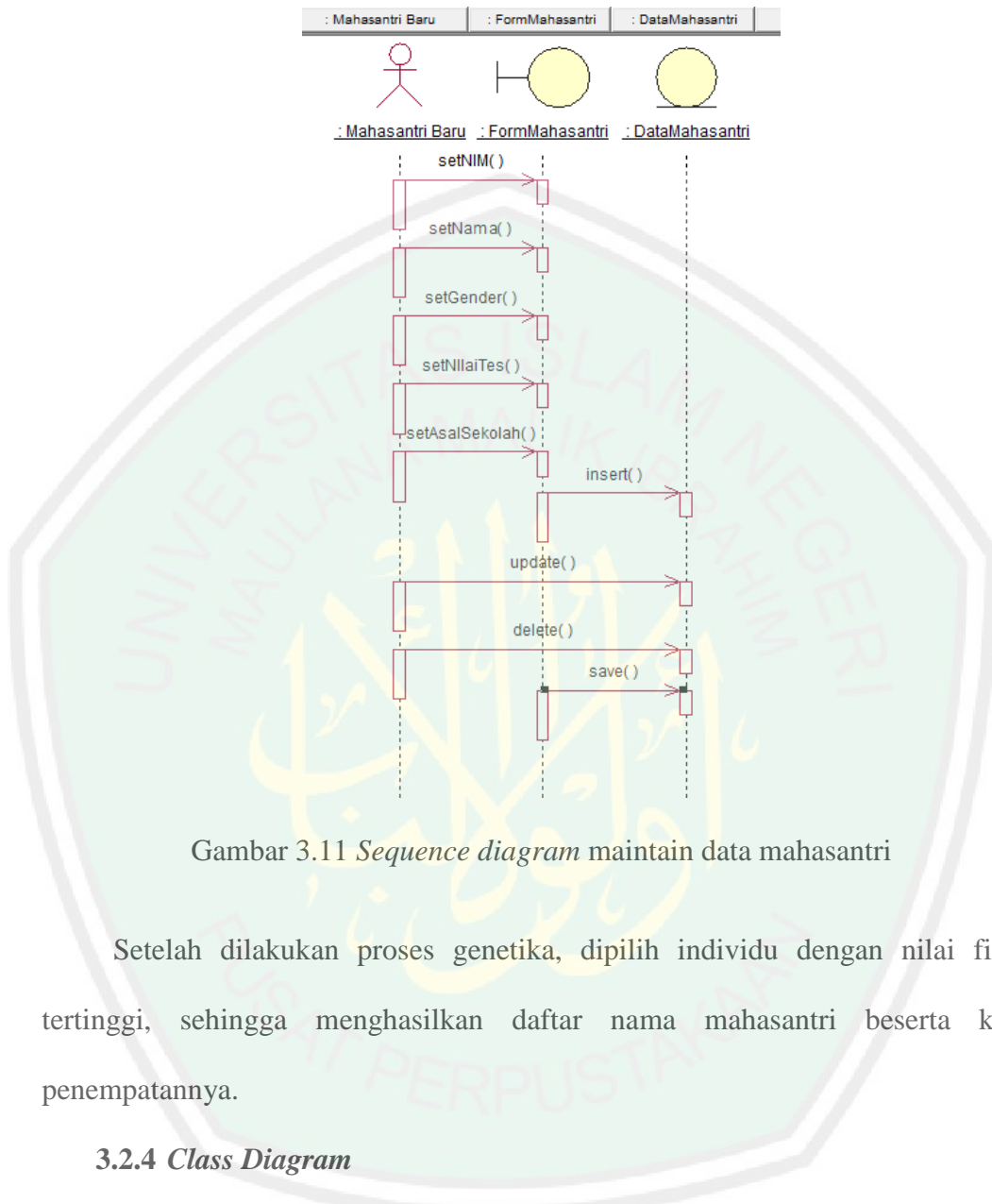
3.2.3 Sequence Diagram

Sequence diagram menjelaskan interaksi object yang disusun dalam suatu urutan waktu. Diagram ini berasosiasi dengan *use case*. *Sequence diagram* memperlihatkan tahap demi tahap apa yang seharusnya terjadi untuk menghasilkan sesuatu dalam *use case*. Berikut *sequence diagram* penempatan kelas mahasantri.



Gambar 3.10 *Sequence diagram* proses genetika

Gambar tersebut menggambarkan *sequence diagram* tahapan yang terjadi pada proses algoritma genetika, tahapan ini diproses dalam halaman admin.

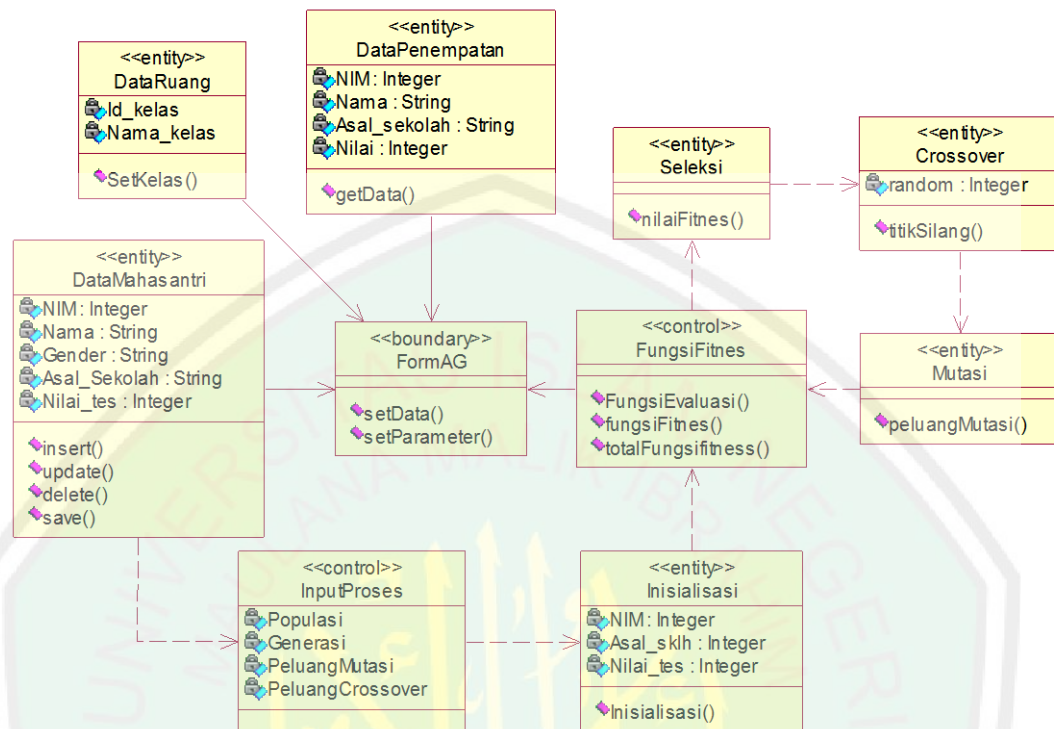


Gambar 3.11 *Sequence diagram* maintain data mahasantri

Setelah dilakukan proses genetika, dipilih individu dengan nilai *fitnes* tertinggi, sehingga menghasilkan daftar nama mahasantri beserta kelas penempatannya.

3.2.4 *Class Diagram*

Class diagram merupakan visualisasi struktur kelas-kelas dari suatu sistem, merupakan tipe diagram yang paling banyak dipakai. *Class diagram* memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas di dalam model desain dari suatu sistem, seperti gambar 3.12 berikut ini:



Gambar 3.12 Class diagram

3.3 Perancangan Sistem

3.3.1 Perancangan Algoritma

Pada bagian ini akan dijelaskan bagaimana proses algoritma berjalan mulai dari pengkodean sampai mutasi.

a. Teknik Pengkodean

Pengkodean yang akan digunakan pada penempatan kelas *ta'lim* mahasiswa adalah *discrete decimal encoding* dimana tiap gen berisi bilangan bulat dalam interval [0,9]. Pemilihan pengkodean secara desimal ini karena lebih efisien daripada pengkodean biner yang akan membuat barisan bit yang panjang sehingga diperlukan pengkodean ulang untuk mendapatkan nilai sebenarnya.

Berikut pengkodean asal sekolah dan nilai tes :

Tabel 3.1 Pengkodean asal sekolah

Asal Sekolah	Nilai Pengkodean
SD, SMP, SMA/SMK	1
SD, MTs, SMA/SMK	2
SD, SMP, MA	3
SD, MTs, MA	4
MI, SMP, SMA/SMK	2
MI, MTs, SMA/SMK	3
MI, SMP, MA	4
MI, MTs, MA	5
SD, SMP, SMA/SMK, Pontren	2
SD, MTs, SMA/SMK, Pontren	3
SD, SMP, MA, Pontren	4
SD, MTs, MA, Pontren	5
MI, SMP, SMA/SMK, Pontren	3
MI, MTs, SMA/SMK, Pontren	4
MI, SMP, MA, Pontren	5
MI, MTs, MA, Pontren	6

Pengkodean tersebut ditambahkan nilai 1 (satu) bagi mahasantri yang pernah mengenyam pendidikan di pondok pesantren (pontren).

Tabel 3.2 Pengkodean *Range* nilai tes

Nilai tes	Nilai Pengkodean	Kriteria
81-100	3	Tinggi
61-80	2	Sedang
0-60	1	Rendah

Range nilai tes pada tabel tersebut berdasarkan buku monitoring yang dikeluarkan *ma'had*.

Algoritma genetika bekerja dengan menggunakan pendekatan *random*, sehingga nilai-nilai yang dihasilkan adalah nilai *random*. Pada pengklasifikasian kelas *ta'lim afkar* mahasantri baru di *ma'had* dengan algoritma genetika akan terjadi banyak iterasi. Hal tersebut dikarenakan diperlukan suatu nilai yang sesuai agar mendapatkan kombinasi yang tepat dari variabel mahasantri. Semakin banyak iterasi yang dilakukan, maka waktu yang dibutuhkan akan semakin lama.

Oleh karena itu maka penyelesaian masalah pengklasifikasian kelas *ta'lim afkar* mahasantri baru di *ma'had* ini akan diselesaikan melalui dua tahap. Tahap pertama adalah mengkombinasikan gen-gen pembentuk suatu kromosom mahasantri dengan menggunakan algoritma genetika. Tahap kedua adalah menempatkan mahasantri pada kelas *ta'lim* yang sesuai dan pada tahap ini diselesaikan dengan pemrograman penelusuran biasa berdasarkan hasil tahap pertama.

b. Inisialisasi Kromosom

Inisialisasi kromosom direpresentasikan dalam bentuk *array* dengan tipe data *record* yang berisi data yang mendukung proses pengklasifikasian kelas, panjang dari kromosom adalah sebanyak jumlah maksimal mahasantri yang diikuti dalam proses penempatan, dalam setiap gen yaitu mahasantri yang membawa nilai kriteria pengklasifikasian.

Berikut ini merupakan ilustrasi inisialisasi kromosom dengan contoh 24 kromosom dengan populasi 4. Parameter yang digunakan adalah asal sekolah/tamatan dan nilai tes dari pengkodean sebelumnya.

Tabel 3.3 Inisialisasi kromosom

NIM	Asal Sekolah	Nilai Tes
12120001	3	2
12130091	5	4
12660222	2	2
12140129	5	3
12220010	3	3
12230030	6	4
12240057	3	3
12310101	5	4
12640017	3	3
12320113	5	2
12630073	2	1
12330043	5	3
12620057	2	2
12410074	4	3
12420099	6	4
12430082	3	2
12510008	2	1
12520036	4	3
12650028	2	1
12610026	2	2
12140023	5	4
12210068	4	3
12320121	3	3
12630129	4	3

Berikut ilustrasi inialisasi kromosom berdasarkan tabel 3.3 :

Tabel 3.4 Ilustrasi inialisasi kromosom

Individu 1						
NIM	12120001	12130091	12660222	12140129	12220010	12230030
X1	3	5	2	5	3	6
X2	2	4	2	3	3	4

Individu 2						
NIM	12240057	12310101	12640017	12320113	12630073	12330043
X1	3	5	3	5	2	5
X2	3	4	3	2	1	3

Individu 3						
NIM	12620057	12410074	12420099	12430082	12510008	12520036
X1	2	4	6	3	2	4
X2	2	3	4	2	1	3

Individu 4						
NIM	12650028	12610026	12140023	12210068	12320121	12630129
X1	2	2	5	4	3	4
X2	1	2	4	3	3	3

c. Fungsi Evaluasi (*Fitness*)

Langkah selanjutnya adalah menghitung nilai *fitness* setiap individu telah terbentuk dalam populasi. Proses seleksi dilakukan dengan cara membuat kromosom yang mempunyai nilai *fitness* terbesar mempunyai kemungkinan terpilih yang besar atau mempunyai nilai probabilitas yang tinggi untuk menjadi induk dalam generasi berikutnya.

Dalam hal ini, untuk mencari suatu nilai *fitness* digunakan fungsi objektif sebagai berikut:

$$f(\mathbf{k}) = \{(\mathbf{X1}+\mathbf{X2}) + (\mathbf{X1}+\mathbf{X2}) + \dots \} \quad (1)$$

K adalah kromosom ke-i yang sedang dicari nilai objektifnya, X1 adalah inisialisasi asal sekolah mahasantri yang bersangkutan, sedangkan X2 adalah nilai tesnya. Nilai X1 dan X2 diambil dari nilai paling tinggi dalam tiap kromosom, yaitu asal sekolah dan nilai tes.

Misalkan kita hitung nilai objektif dari susunan kromosom tersebut adalah sebagai berikut, f adalah fungsi objektif (*fitness*) :

Tabel 3.5 Nilai *fitness*

Individu	Fitness
1	$F = (6+4) = 10$
2	$F = (5+4) = 9$
3	$F = (6+4) = 10$
4	$F = (5+4) = 9$

Panjang dari fungsi objektif (Panjang $f(\mathbf{k})$) tersebut adalah jumlah mahasantri yang diikutkan dalam proses pengklasifikasian (N) dibagi jumlah mahasantri dalam tiap kelas (r). Sehingga diperoleh persamaan sebagai berikut:

$$\text{Panjang } f(\mathbf{k}) = N/r \quad (2)$$

Dari persamaan (1) tersebut kemudian ditentukan nilai *fitness* dengan persamaan berikut:

$$\text{Fitness}[\mathbf{k}] = f(\mathbf{k}) \quad (3)$$

Karena nilai *fitness* dari sebuah kromosom adalah nilai objektif dari kromosom itu sendiri, maka solusi yang dicari adalah memaksimalkan sebuah fungsi objektif.

d. Seleksi

Seleksi mempunyai peranan penting dalam algoritma genetika, pada proses ini dipilih induk yang digunakan untuk menghasilkan individu baru. Seleksi yang digunakan adalah seleksi roda *roulette*. Pada seleksi roda *roulette*, semakin tinggi nilai *fitness* maka semakin besar kemungkinan untuk terpilih menjadi induk.

Rumus untuk mencari probabilitas adalah sebagai berikut:

$$P[k] = \text{Fitness}[k] / \text{total_Fitness} \quad (4)$$

Untuk proses seleksi *roulette wheel*, kita harus mencari terlebih dahulu nilai probabilitas kumulatifnya. Setelah dihitung probabilitas kumulatifnya maka proses seleksi menggunakan *roulette wheel* dapat dilakukan dengan membangkitkan bilangan acak dalam *range* 0-1.

Berdasarkan hasil perhitungan persamaan (1). Diperoleh nilai probabilitas tiap individu.

Tabel. 3.6 Probabilitas tiap individu

Individu	Fitness	Probabilitas
1	10	$10/38 = 0.26$
2	9	$9/38 = 0.24$
3	10	$10/38 = 0.26$
4	9	$9/38 = 0.24$

Berdasarkan tabel tersebut dapat kita ketahui bahwa individu ke-1 dan ke-3 mempunyai probabilitas individu yang paling tinggi memiliki kemungkinan

terbesar dari individu lainnya untuk terpilih menjadi *parent* pada generasi selanjutnya.

e. *Crossover*

Crossover dilakukan antara 2 kromosom untuk menghasilkan keturunan baru. Metode *crossover* yang banyak digunakan adalah *one point crossover*. Pada skema ini, ditentukan sembarang bilangan acak untuk menentukan posisi persilangan, kemudian menukar satu bagian kromosom dari kedua *parent* kromosom untuk menghasilkan kromosom anak dengan tetap menjaga nilai gen dan susunan yang bukan bagian dari kromosom tersebut.

Berikut ilustrasi *crossover* berdasarkan tabel 3.6.

Tabel 3.7 Ilustrasi *crossover*

Sebelum <i>crossover</i>						
Individu 1						
NIM	12120001	12130091	12660222	12140129	12220010	12230030
Individu 3						
NIM	12620057	12410074	12420099	12430082	12510008	12520036
Setelah <i>crossover</i>						
Individu 1						
NIM	12120001	12430082	12660222	12140129	12220010	12230030
Individu 3						
NIM	12620057	12410074	12420099	12130091	12510008	12520036

Berdasarkan tabel tersebut, diketahui individu 1 dan individu 3 terpilih untuk proses *crossover*, selanjutnya gen terpilih ditukar dengan menjaga nilainya.

f. Mutasi

Mutasi dilakukan dengan cara melakukan perubahan pada sebuah gen atau lebih dari sebuah individu. Tujuan dari mutasi adalah agar individu-individu yang ada dalam populasi semakin bervariasi. Mutasi akan sangat berperan jika pada populasi awal hanya ada sedikit solusi yang mungkin terpilih. Sehingga, operasi mutasi sangat berguna dalam mempertahankan keanekaragaman individu dalam populasi meskipun dengan mutasi tidak dapat diketahui apa yang terjadi pada individu baru.

Skema mutasi yang digunakan adalah *swapping mutation*. *Swapping mutation* adalah mutasi yang dilakukan dengan menukar langsung nilai dari gen. Pemilihan individu yang akan dimutasi dilakukan secara random.

Tabel 3.8 Ilustrasi mutasi

Sebelum mutasi						
Individu 1						
NIM	12120001	12430082	12660222	12140129	12220010	12230030
Setelah mutasi						
Individu 1						
NIM	12120001	12230030	12660222	12140129	12220010	12430082

Setelah individu ke-1 terpilih untuk dimutasi, kemudian secara random ditentukan gen yang akan di mutasi.

Jumlah kromosom yang mengalami mutasi dalam satu populasi ditentukan oleh peluang mutasi (P_m). Peluang mutasi mengendalikan banyaknya gen baru yang akan dimunculkan untuk dievaluasi. Jika peluang mutasi terlalu kecil,

banyak gen yang mungkin berguna tidak pernah dievaluasi. Tetapi bila peluang mutasi ini terlalu besar, maka akan terlalu banyak gangguan acak, sehingga anak akan kehilangan kemiripan dari induknya.

3.3.2 Output

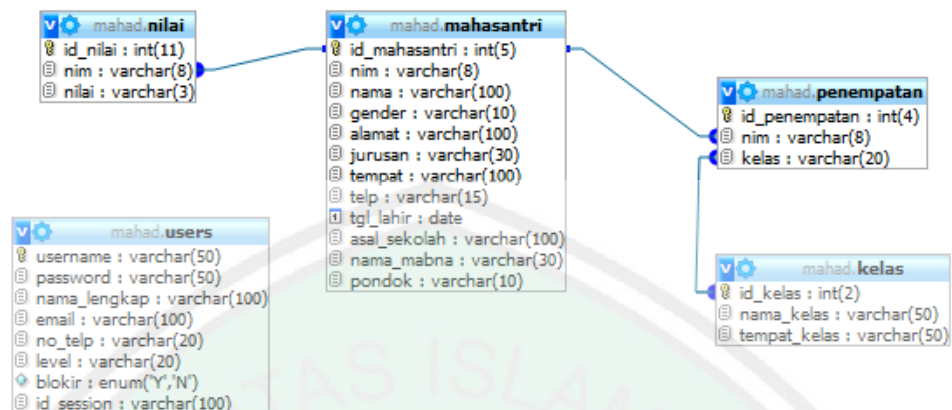
Pembagian kelas dilakukan setelah proses algoritma genetika selesai, yaitu ketika individu sebagai solusi yang paling optimal telah didapatkan. Pembagian ruang ini dilakukan dengan membagi jumlah gen dalam kromosom solusi dengan jumlah mahasantri yang ditetapkan dalam tiap kelas, dalam hal ini jumlah mahasantri dalam tiap kelas adalah seluruh jumlah mahasantri dibagi jumlah kelas. Berikut ilustrasi output dengan membagi menjadi 4 kelas

Tabel 3.9 Ilustrasi output pengklasifikasian kelas

Kelas A	Kelas B	Kelas C	Kelas D
12120001	12240057	12620057	12650028
12230030	12310101	12410074	12610026
12660222	12640017	12420099	12140023
12140129	12320113	12130091	12210068
12220010	12630073	12510008	12320121
12430082	12330043	12520036	12630129

3.4 Perancangan Database

Database yang akan dibangun terdiri dari komponen utama penempatan mahasantri di *ma'had* dan juga pengembangan dari beberapa komponen utama. Komponen utama antara lain mahasantri, parameter nilai dan kelas. Sedangkan pengembangan dari parameter nilai terdiri dari nilai *placement test* dan asal sekolah/tamatan. Berikut ini adalah relasi antar tabel dalam *database*:



Gambar 3.13 Relasi antar tabel

Aplikasi *database* yang digunakan adalah MySQL, nama *databasenya* mahad. Berikut ini nama-nama tabel yang digunakan beserta *field-field* yang terdapat pada masing-masing tabel:

3.4.1 Tabel Mahasantri, digunakan untuk menyimpan data mahasantri

Tabel 3.10 Tabel mahasantri

No.	Field Name	Data Type	Primary key
1.	Id_mahasantri	Integer(11)	Yes
2.	Nim	Integer(8)	No
3.	Nama	Varchar(100)	No
4.	Gender	Varchar(10)	No
5.	Alamat	Varchar(100)	No
6.	Jurusan	Varchar(30)	No
7.	Tempat_lahir	Varchar(100)	No
8.	Tgl_lahir	Date	No
9.	Telp	Varchar(15)	No
10	Asal_sekolah	Varchar(100)	No
11	Nama_mabna	Varchar (30)	No
12	Pondok	Enum ('Y', 'N')	No

3.4.2 Tabel *users* digunakan untuk menyimpan data *users* admin untuk login

Tabel 3.11 Tabel *users*

No.	Field Name	Data Type	Primary key
1.	Id_user	Integer(11)	Yes
2.	Username	Varchar(50)	No
3.	Password	Varchar(50)	No
4.	Nama_lengkap	Varchar(100)	No
5.	Email	Varchar(100)	No
6.	No_telp	Varchar(50)	No
7.	Level	Varchar(50)	No
8.	Blokir	Enum ('Y', 'N')	No

3.4.3 Tabel Kelas, digunakan untuk menyimpan data kelas yang di tempati mahasiswa dalam proses penempatan kelas

Tabel 3.12 Tabel kelas

No.	Field Name	Data Type	Primary key
1.	Id_kelas	Integer(2)	Yes
2.	Nama_kelas	Varchar(50)	No
3.	Tempat_kelas	Varchar(50)	No

3.4.4 Tabel Penempatan, digunakan untuk menyimpan data hasil proses penempatan yang telah dilakukan dengan algoritma genetika

Tabel 3.13 Tabel Penempatan

No.	Field Name	Data Type	Primary key
1.	Id_penempatan	Integer(2)	Yes
2.	NIM	Integer(8)	No
3.	Kelas	Varchar(20)	No

3.4.5 Tabel Nilai, digunakan untuk menyimpan nilai *placement test*

Tabel 3.14 Tabel Nilai

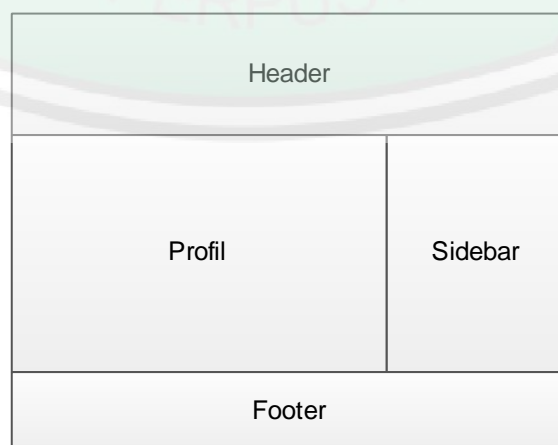
No.	Field Name	Data Type	Primary Key
1.	Id_nilai	Integer(11)	Yes
2.	Nim	Integer(8)	No
3.	Nilai	Integer(3)	No

3.5 Perancangan *Interface*

3.5.1 Halaman Utama



Gambar 3.14 Desain Halaman utama



Gambar 3.15 Desain Halaman Profil

3.5.2 Halaman Admin

Login

username

Password

Gambar 3.16 Desain login admin

[Lihat Website](#)

Data Mahasantri

No.	NIM	Nama	Jurusan	Ttl	Telp/HP	Asal Sklah	Aksi

first < 1 2 3 > last

- > Berita
- > Mahasantri
- > Kelas
- > Nilai tes
- > Kriteria
- > Parameter
- > Hasil Penempatan
- > Keluar

Gambar 3.17 Desain *view* data mahasantri

[Lihat Website](#)

Berita

No.	Judul	Gambar	Tgl Post	Aksi

first < 1 2 3 > last

- > Berita
- > Mahasantri
- > Kelas
- > Nilai tes
- > Kriteria
- > Parameter
- > Hasil Penempatan
- > Keluar

Gambar 3.18 Desain *view* berita

[Lihat Website](#)

- > Berita
- > Mahasantri
- > Kelas
- > Nilai tes
- > Kriteria
- > Parameter
- > Hasil Penempatan
- > Keluar

Kelas

No.	Kelas	Tempat	Ket	Aksi

Gambar 3.19 Desain view kelas

[Lihat Website](#)

- > Berita
- > Mahasantri
- > Kelas
- > Nilai tes
- > Kriteria
- > Parameter
- > Hasil Penempatan
- > Keluar

Nilai Tes

No.	NIM	Nama	Nilai	Aksi

first < 1 2 3 > last

Gambar 3.20 Desain view nilai tes

[Lihat Website](#)

- > Berita
- > Mahasantri
- > Kelas
- > Nilai tes
- > Kriteria
- > Parameter
- > Hasil Penempatan
- > Keluar

Kriteria

asal sekolah	inisialisasi	Nilai tes	inisialisasi

Gambar 3.21 Desain Kriteria

[Lihat Website](#)

- > Berita
- > Mahasantri
- > Kelas
- > Nilai tes
- > Kriteria
- > Parameter
- > Hasil Penempatan
- > Keluar

Input Parameter

Parameter

Generasi

Populasi

Peluang Crossover

Peluang Mutasi

Gambar 3.22 Desain parameter genetika

[Lihat Website](#)

- > Berita
- > Mahasantri
- > Kelas
- > Nilai tes
- > Kriteria
- > Parameter
- > Hasil Penempatan
- > Keluar

Hasil Penempatan

Kelas Aly

No.	NIM	Nama

first < 1 2 3 > last

Gambar 3.23 Desain view hasil penempatan

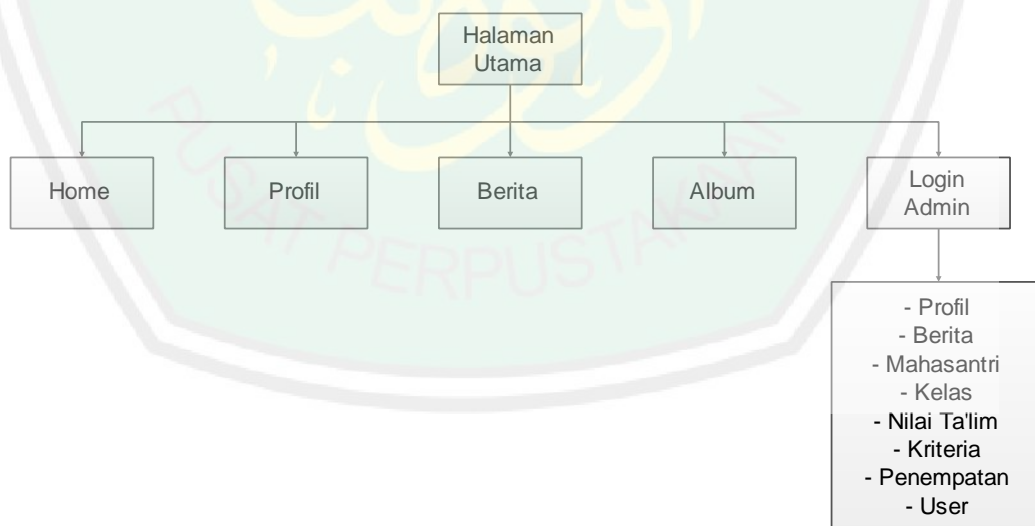
BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

Implementasi sistem merupakan tahapan meletakkan sistem untuk di uji serta analisa hasil dari program yang telah dibuat. Implementasi sistem juga merupakan sebuah proses pembuatan dan penerapan sistem secara utuh baik dari sisi perangkat keras maupun perangkat lunaknya. Tujuan dari pengujian ini adalah untuk mengetahui apakah aplikasi yang telah dibuat sesuai dengan perancangannya serta untuk pengembangan dan perbaikan lebih lanjut.

Berikut ini *sitemap* program pengklasifikasian kelas *ta'lim afkar Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang.



Gambar 4.1 *Sitemap* program

4.1.1 Interface

Pada sub bab ini dijelaskan mengenai fungsi masing-masing *form* yang ada dalam program aplikasi ini beserta desain *interfacenya*.

a. Halaman Beranda

Merupakan halaman awal dalam website. Halaman ini adalah halaman awal dimana setiap pengunjung bisa melihat *update* berita dan informasi tentang *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang.



Gambar 4.2 Halaman utama

b. Halaman Profil

Halaman ini berisi tentang profil dan sejarah tentang *Ma'had* Sunan Ampel Al-‘Ali UIN Maulana Malik Ibrahim Malang.



Gambar 4.3 Halaman profil

c. Login

Form login digunakan untuk mengakses halaman admin. Berikut desain *interfacenya*.


























Gambar 4.4 Form Login

d. Mahasantri

Pada menu mahasantri menampilkan data seluruh mahasantri. Pada menu ini digunakan untuk mengolah informasi tentang mahasantri baru yang akan diproses dalam program ini. Pada menu ini, data mahasantri yang ada di database bisa ditambah, edit, atau dihapus. Berikut ini adalah desain *interfacenya*:



The screenshot shows a web application interface for managing students (Mahasantri). On the left is a navigation menu with options: Dashboard, Profil, Berita, Mahasantri (selected), Kelas, Nilai Talim Afkar, Kriteria, Parameter, and User. The main content area is titled 'Mahasantri' and includes a search bar with the text 'Masukkan Nama Mahasantri:' and a 'Cari' button. Below the search bar is a 'Tambah Mahasantri' button. The main part of the interface is a table with the following data:

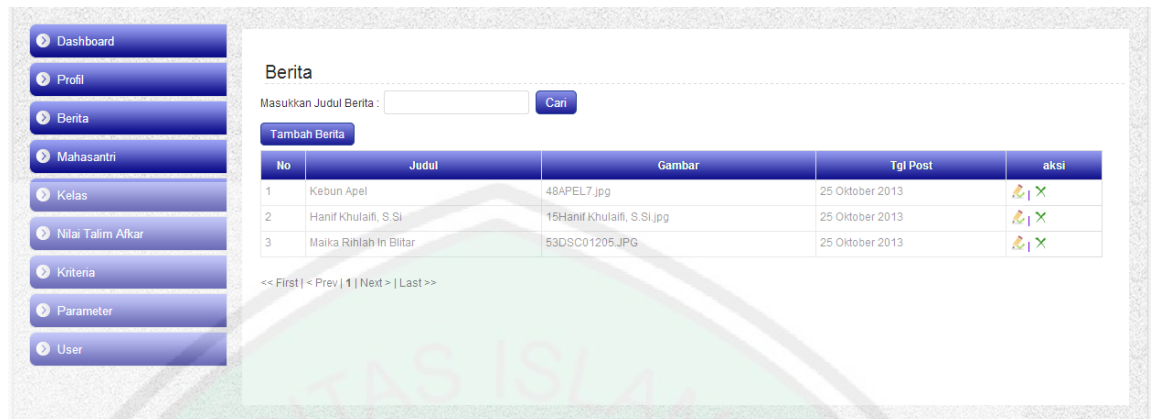
No	NIM	Nama	Jenis Kelamin	Jurusan	Asal Sekolah	Telp	aksi
76	12510159	A ALFAN AULIYA ILLAH	Pria	MANAJEMEN	SMKN 1 Pasuruan Kota Pasuruan		  
77	12520101	AHMAD RIZA FAHTAROMI	Pria	AKUNTANSI	MAN Rejoso Peterongan Kab. Jombang		  
78	12210080	ACHMAD OOMARUDIN	Pria	AHWALUS SYAHSIYAH	MA Manbaus Sholihin Kab. Gresik		  
79	12640048	SYAIFUL BAHRI	Pria	FISIKA	MA Al-Mardiyah Kab. Pamekasan		  
80	12110179	M. ALFAN SANTOSO	Pria	PENDIDIKAN AGAMA ISLAM	MA 03 Ma'arif Kab. Jember		  
81	12330123	MOHAMMAD FATHUL HSAN	Pria	PENDIDIKAN BAHASA ARAB	MA Al-Maarif 1 Singosari Kab. Malang		  
82	12660079	WAHYU SETIAWAN	Pria	TEKHNIK ARSITEKTUR			  
83	12110177	DARUL FAROKI NUR WILDANI	Pria	PENDIDIKAN AGAMA ISLAM			  
84	12110176	ANDRE SEPTIAN	Pria	PERBANKAN SYARIAH	MA 03 Ma'arif Kab. Jember		  
85	12530016	ZAHRUL AZZI	Pria	PENDIDIKAN GURU MADRASAH BTD	SMA Lab UM Kota Malang		  
86	12140107	RIFKI DWI ALFIANTO	Pria	PERBANKAN SYARIAH	MA Ihyaul Ulum Kab. Gresik		  
87	12530015	M. KHORUL FAHMI	Pria	PENDIDIKAN AGAMA ISLAM	MAN 1 Malang Kota Malang		  
88	12110176	HADI BSA	Pria	HUKUM BISNIS SYARIAH	MAS Al-Amien Jambu Kab. Sumenep		  
89	12220104	NIURTAMIN	Pria	BAHASA DAN SASTRA INGGRES	MAN Gondanglegi Kab. Malang		  
90	12320092	ACHMAD NIAMIL CHAMM	Pria	BAHASA DAN SASTRA ARAB	SMAN Khusus Jeneponto		  

At the bottom of the table, there is a pagination control: << First | < Prev | ... 4 | 5 | 6 | 7 | 8 | ... | 18 | Next > | Last >>

Gambar 4.5 Data Mahasantri

e. Berita

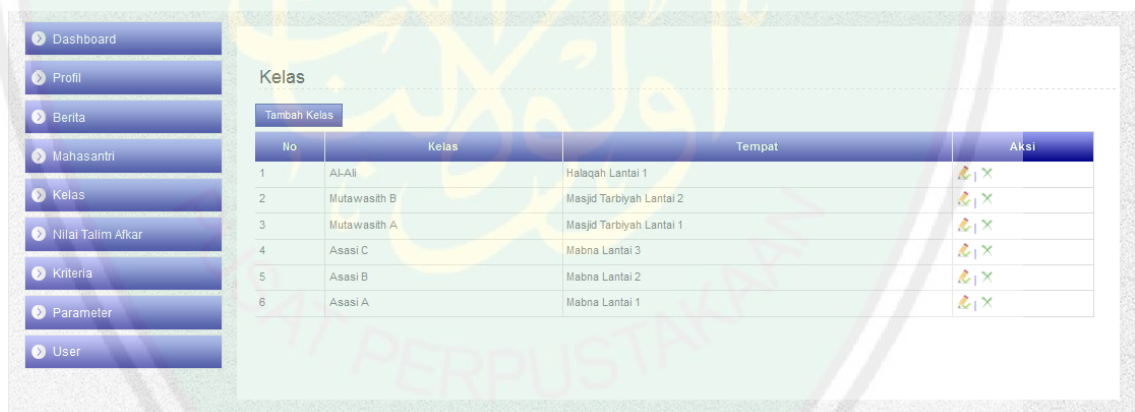
Menampilkan semua berita yang berhubungan dengan Ma'had Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang. Admin bisa menambah, mengedit, atau menghapus berita.



Gambar 4.6 Berita

f. Kelas

Menu kelas menampilkan data kelas. Data pada menu ini bisa ditambah, edit, atau dihapus. Berikut ini adalah desain *interfacenya*:



Gambar 4.7 Data Kelas

g. Nilai

Pada menu nilai menampilkan data nilai seluruh mahasantri. Admin bisa menambah, mengedit, atau menghapus nilai mahasantri. Berikut ini adalah desain *interfacenya*:

No	NIM	Nama	Nilai	Aksi
1	12310082	ASEP IRWAN SUTIAWAN	31	
2	12220131	WAHID RIZKICA AKBAR	22	
3	12510119	DECKY MAULANA	51	
4	12140123	M. TAMRIN AMRULLOH	14	
5	12210096	MOCH. NIKO AL AYUBI	21	
6	12410164	M HAQIQI RACHMANSYAH	41	
7	12620078	TRI ANGGA DENI ISTANTO	62	
8	12410165	SUBHANALLAH RAMDHAN	41	
9	12330089	DIQO ULL AZMI	33	
10	12510113	MUKHAMMAD ILHAM MARGA F	51	
11	12130124	FREDY HARYONO	13	
12	12210095	SUNJAYA NUR APRIRIAN T	21	
13	12330126	SAEFUDIN	33	
14	12510112	ABDUL BARI' EL LAUDZA'I	51	
15	12220126	AH SHOLHUDDIN	22	

Gambar 4.8 Data Nilai

h. Kriteria

Pada menu kriteria menampilkan kriteria. Admin bisa mengedit kriteria penilaian pada algoritma genetika. Berikut ini adalah desain *interfacenya*:

Asal sekolah	Inisialisasi	Aksi	Nilai Test	Inisialisasi	Aksi
Pontren	3		81 - 100	4	
MA	2		61 - 80	3	
SMK	1		41 - 60	2	
SMA	1		0 - 40	1	

Gambar 4.9 Kriteria

i. Parameter Genetika

Pada menu parameter genetika, Admin bisa mengedit parameter genetika yang akan digunakan pada proses algoritma genetika. Berikut ini adalah desain *interfacenya*:

Gambar 4.10 Parameter Genetika

j. Data user admin

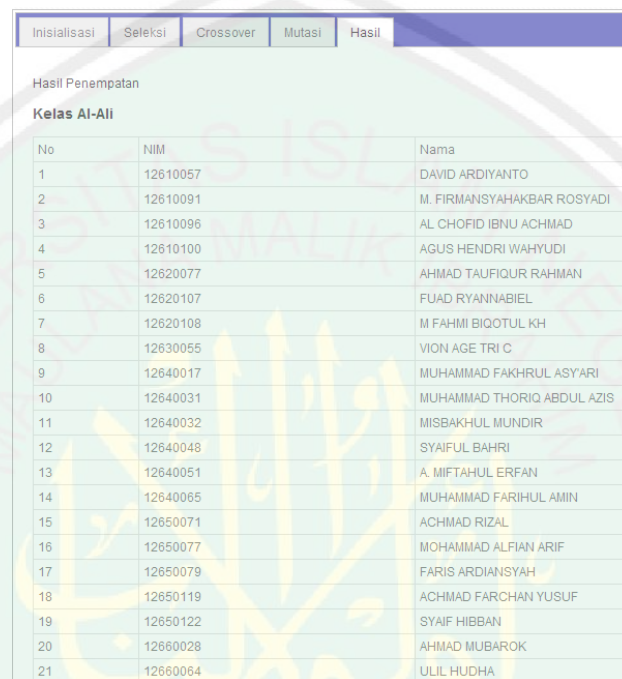
Pada menu admin, berisi data admin. Berikut ini adalah desain *interfacenya*:

No.	Nama Lengkap	Username	Email	Telepon	Aksi
1	andhika	andhika	andhika.tri@gmail.com		
2	Administrator	admin			

Gambar 4.11 Data user admin

k. Hasil Penempatan

Pada menu hasil penempatan, menampilkan hasil penempatan proses algoritma genetika. Berikut ini adalah desain *interfacenya*:



No	NIM	Nama
1	12610057	DAVID ARDIYANTO
2	12610091	M. FIRMANSYAHAKBAR ROSYADI
3	12610096	AL CHOFID IBNU ACHMAD
4	12610100	AGUS HENDRI WAHYUDI
5	12620077	AHMAD TAUFIQUR RAHMAN
6	12620107	FUAD RYANNABIEL
7	12620108	M FAHMI BIQOTUL KH
8	12630055	VION AGE TRI C
9	12640017	MUHAMMAD FAKHRUL ASYARI
10	12640031	MUHAMMAD THORIQ ABDUL AZIS
11	12640032	MISBAKHUL MUNDIR
12	12640048	SYAIFUL BAHRI
13	12640051	A. MIFTAHUL ERFAN
14	12640065	MUHAMMAD FARIHUL AMIN
15	12650071	ACHMAD RIZAL
16	12650077	MOHAMMAD ALFIAN ARIF
17	12650079	FARIS ARDIANSYAH
18	12650119	ACHMAD FARCHAN YUSUF
19	12650122	SYAIF HIBBAN
20	12660028	AHMAD MUBAROK
21	12660064	ULIL HUDHA

Gambar 4.12 Hasil penempatan

4.1.2 Deskripsi Program

Program terdiri dari enam proses utama, yaitu inisialisasi kromosom, evaluasi, seleksi, *crossover*, mutasi dan proses pengklasifikasian kelas. Berikut penjelasan tiap proses:

a. Inisialisasi Kromosom

Inisialisasi kromosom dalam program adalah dengan membangkitkan nilai random Nomor Induk Mahasantri (NIM) sebanyak jumlah mahasantri baru yang akan ditempatkan dalam kelas *ta'lim*. Berikut listing program untuk inisialisasi:

```

function inisial () {
    $ini = mysql_query ("SELECT * FROM mahasantri m, nilai n WHERE m.nim=n.nim
ORDER BY RAND()");
    $individu = array();
    $kromosom = array();
    while ($row = mysql_fetch_array($ini)){
        $a=strtolower($row['asal_sekolah']);
        $a=strtolower($row['asal_sekolah']);
        $b=strtolower($row['pondok']);
        $sd = strtolower($row['asal_sd']);
        $smp = strtolower($row['asal_smp']);
        if (substr($b,0,2)=='y'){
            if (substr($sd,0,2)=='sd' && substr($smp,0,3)=='smp' &&
substr($a,0,2)=='sm') {
                $asal='2';
            } else if (substr($sd,0,2)=='sd' && substr($smp,0,3)=='mts' &&
substr($a,0,2)=='sm') {
                $asal='3';
            } else if (substr($sd,0,2)=='sd' && substr($smp,0,3)=='smp' &&
substr($a,0,2)=='ma') {
                $asal='4';
            } else if (substr($sd,0,2)=='sd' && substr($smp,0,3)=='mts' &&
substr($a,0,2)=='ma') {
                $asal='5';
            } else if (substr($sd,0,2)=='mi' && substr($smp,0,3)=='smp' &&
substr($a,0,2)=='sm') {
                $asal='3';
            } else if (substr($sd,0,2)=='mi' && substr($smp,0,3)=='mts' &&
substr($a,0,2)=='sm') {
                $asal='4';
            } else if (substr($sd,0,2)=='mi' && substr($smp,0,3)=='smp' &&
substr($a,0,2)=='ma') {
                $asal='5';
            } else if (substr($sd,0,2)=='mi' && substr($smp,0,3)=='mts' &&
substr($a,0,2)=='ma') {
                $asal='6';
            } else {
                $asal='2';
            }
        }
        } else if (substr($b,0,2)=='t'){
            if (substr($sd,0,2)=='sd' && substr($smp,0,3)=='smp' &&
substr($a,0,2)=='sm') {
                $asal='1';
            } else if (substr($sd,0,2)=='sd' && substr($smp,0,3)=='mts' &&
substr($a,0,2)=='sm') {
                $asal='2';
            } else if (substr($sd,0,2)=='sd' && substr($smp,0,3)=='smp' &&
substr($a,0,2)=='ma') {
                $asal='3';
            } else if (substr($sd,0,2)=='sd' && substr($smp,0,3)=='mts' &&
substr($a,0,2)=='ma') {

```

```

        $asal='4';
    } else if (substr($sd,0,2)=='mi' && substr($smp,0,3)=='smp' &&
substr($a,0,2)=='sm') {
        $asal='2';
    } else if (substr($sd,0,2)=='mi' && substr($smp,0,3)=='mts' &&
substr($a,0,2)=='sm') {
        $asal='3';
    } else if (substr($sd,0,2)=='mi' && substr($smp,0,3)=='smp' &&
substr($a,0,2)=='ma') {
        $asal='4';
    } else if (substr($sd,0,2)=='mi' && substr($smp,0,3)=='mts' &&
substr($a,0,2)=='ma') {
        $asal='5';
    }
    } else {
        $asal='1';
    }
}
if($row['nilai']>=80){
    $nilai =4;
} else if($row['nilai']<80 && $row['nilai']>=60 ){
    $nilai =3;
} else if($row['nilai']<60 && $row['nilai']>=40 ){
    $nilai =2;
} else {
    $nilai =1;
}

$jumlah = 0;
$jumlah += $asal;
$jumlah += $nilai;

$a=array("nim"=>$row['nim'], "asal"=>$asal, "tes"=>$nilai, "fit"=>$jumlah);
array_push($individu, $a);
$this->arr_individu = $individu;
}

return ($individu);
}

```

b. Evaluasi

Fungsi evaluasi dimaksudkan untuk mencari nilai *fitness* dari suatu kromosom. Setelah kromosom dibangkitkan, kemudian tiap kromosom dicari nilai *fitnessnya* dengan membuat modulo tujuh dari panjang kromosom. Dibuat modulo tujuh karena diharapkan akan terbentuk susunan

yang berjumlah rata-rata 38 gen dari panjang kromosom yang bersangkutan. Kemudian dalam tiap susunan tujuh tersebut dicari nilai maksimum untuk kriteria asal sekolah dan nilai tes, misalkan masing-masing kriteria diwakili oleh variabel x_1 dan x_2 . Setelah itu antara susunan tujuh yang ada dalam panjang kromosom itu akan dijumlahkan dan akhirnya akan didapatkan nilai fitness dari kromosom itu. Karena masalah pengklasifikasian ini merupakan masalah yang tergolong maksimisasi, maka nilai *fitness* adalah nilai objektif (nilai evaluasi) itu sendiri. Berikut cuplikan program fungsi evaluasi.

```
function hitungFitnes ($indiv){
    $jmlmhs = $this->jumlahMhs();
    $jmlkls = $this->jumlahKls();
    $mhsperkelas = $jmlmhs/$jmlkls;

    $fitnes = 0;
    $x1 = 0;
    $x2 = 0;
    $k = 0;
    for ($j=0;$j<$mhsperkelas;$j++){

        $i = 0;
        while ($i<$jmlkls){
            $asal[$i] = $indiv[$k]['asal'];
            $tes [$i] = $indiv[$k]['tes'];

            $k++;
            $i++;
        }
        $x1 = max($asal);
        $x2 = max($tes);
        $fitnes += $x1 + $x2;
    }
    return ($fitnes);
}
```

c. Seleksi

Seleksi dimulai dengan dibangkitkan sebuah nilai secara random. Kemudian lakukan penelusuran dimulai dengan indeks $j=1$ sampai j kurang dari banyaknya populasi. Jika nilai random lebih besar dari probabilitas kumulatif kromosom indeks ke j naikan nilai j . Cek ulang sampai nilai j lebih besar dari populasi atau nilai random kurang dari probabilitas kumulatif. Dapatkan kromosom indeks ke- j sebagai orang tua. Berikut cuplikan program untuk memilih induk:

```
function seleksi () {
    $mhs = $this->jumlahMhs();
    $fit = 0;
    $totfitnes = 0;
    $pk = 0;
    $Qk = 0;

    for ($i=0;$i<$this->populasi;$i++){
        $fit = $this->arr_kromosom [$i]['fitnes'];
        $totfitnes += $fit;
    }

    for ($i=0; $i<count($this->arr_inisial); $i++){

        $id = $this->arr_inisial [$i]['id'];
        $fit = $this->arr_inisial [$i]['fitnes'];
        $individu = $this->arr_inisial[$i]['individu'];

        $pk = $fit/$totfitnes;

        $Qk += $pk;
        $data=array('id'=>$id, 'fitnes'=>$fit, 'pk'=>$pk, 'kumulatif'=>$Qk,
        'individu'=>$individu);
        array_push($this->arr_seleksi,$data);

    for ($ra=0;$ra<$this->populasi;$ra++){
        $r = rand(10,100)/100;

        $found=false;
        $i = 0;

        while ($i<$this->populasi){
```

```
        if ($this->arr_seleksi[$i]['kumulatif'] > $r && !$found){
            array_push($this->arr_cross,$this->arr_seleksi[$i]);
            $found = true;
        }
        $i++;
    }
    return $this->arr_seleksi;
}
```

d. *Crossover*

Crossover di mulai dengan membangkitkan nilai random, setelah itu dipilih kromosom kandidat orang tua dengan membandingkan dengan peluang *crossover* (P_c), jika nilai random tersebut kurang dari P_c maka kromosom kedua orang tua yang telah dipilih tersebut berpeluang untuk dilakukan proses *crossover* yang kemudian hasilnya disimpan dalam kromosom anak, jika tidak demikian maka kromosom anak adalah kromosom orang tua itu sendiri.

Pada perkawinan silang dimulai dengan menyimpan kromosom orang tua pada sebuah variabel. Setelah itu ditentukan titik potong perkawinan silang secara random. Dalam masalah ini perkawinan silang diimplementasikan dengan skema *single point crossover*, di mana dalam skema ini satu bagian kromosom dipertukarkan dengan tetap menjaga nilai gen dan susunan yang bukan bagian dari kromosom tersebut. Berikut cuplikan program untuk *crossover*:

```

function crossOver (){
    $mhs = $this->jumlahMhs();
    $jml=0;
    $arrSeleksi = array();
    for ($c=0;$c<$this->populasi;$c++){
        // menentukan indek mana yang di cross
        $rn = rand(10,100)/100;

        if($rn <= $this->peluangCros){
            array_push($arrSeleksi,$c);
            $c = $c+1;
            $jml++;
        }
    }

    for ($cr=0;$cr<$jml;$cr++){
        if($jml>1){
            if ($jml%2!=0){
                $jml--;
            }
        }
        $spotong = rand(0,$mhs);
        for ($s=0;$s<$jml;$s++){
            if ($s%1==0 && $s!=0){
                for ($i=0;$i<$spotong;$i++){
                    $anak1=$this->arr_cross[$arrSeleksi[$s1]]['individu'][$i];
                    $anak2= $this->arr_cross[$arrSeleksi[$s]]['individu'][$i];

                    $tuker = false;

                    for ($si=$spotong;$si<$mhs;$si++){
                        if($anak1['nim']==$this->arr_cross[$arrSeleksi[$s]]['individu'][$si]['nim']) {
                            $tuker = true;
                        }
                    }
                    if ($tuker == false){
                        $this->arr_cross[$arrSeleksi[$s-1]]['individu'][$i] = $anak2;
                        $this->arr_cross[$arrSeleksi[$s]]['individu'][$i] = $anak1;
                    }
                }
            }
        }
    }
    for ($cr=0;$cr<$jml;$cr++){
        if($jml>1){
            if ($jml%2!=0){
                $jml--;
            }
        }
        $fitcross = $this->hitungFitnes($individu);
    }
}

```

```

        $arrc=array('id'=>$idc,'fitnes'=>$fitcross, 'individu'=>$individu);
        array_push($this->arr_mutasi,$arrc);
    }
}
for ($scr=0;$scr<$this->populasi;$scr++){

    $idc = $this->arr_cross[$scr]['id'];
    $individu = $this->arr_cross[$scr]['individu'];
    for($si=0;$si<$mhs;$si++){

        $fitcross = $this->hitungFitnes($individu);
        $arrc=array('id'=>$this->arr_cross[$scr]['id'], 'fitnes'=>$fitcross,
'individu'=>$individu);
        array_push($this->arr_mutasi,$arrc);
    }
    return $this->arr_cross;
}

```

e. Mutasi

Proses mutasi bisa tidak terjadi, karena hal ini bergantung pada peluang mutasi (P_m). Mutasi dimulai dengan membangkitkan nilai secara random. Jika nilai random tersebut kurang dari P_m maka mutasi terjadi. Pada proses mutasi ini dilakukan dengan cara *swapping mutation*, yaitu dengan menukar nilai dua gen. Langkah awal adalah menentukan dua gen yang akan dimutasi, kemudian langsung menukar nilai kedua gen tersebut. Berikut cuplikan program untuk mutasi:

```

function mutasi () {
    for ($m=0;$m<$this->populasi;$m++){
        $rIndividu = rand(0,100)/100;
        if($rIndividu <= $this->peluangMutasi){
            $rNIM1 = rand(0,($mhs-1));
            $rNIM2 = rand(0,($mhs-1));
            $mut1=$this->arr_mutasi[$rIndividu]['individu'][$rNIM1];
            $mut2=$this->arr_mutasi[$rIndividu]['individu'][$rNIM2];
            $this->arr_mutasi[$rIndividu]['individu'][$rNIM1]=$mut2;
            $this->arr_mutasi[$rIndividu]['individu'][$rNIM2]=$mut1;
        }
    }
}

```

f. Penentuan individu terpilih

Individu terpilih adalah kromosom yang memiliki nilai *fitness* tertinggi. Jika ada 2 atau lebih individu dengan nilai *fitness* tertinggi, maka ditentukan secara acak.

```

foreach ($this->arr_mutasi[0]['individu'] as $key => $row) {
    $nim[$key] = $row['nim'];
    $fit[$key] = $row['fit'];
}

array_multisort($fit, SORT_DESC, $nim, SORT_DESC, $this->arr_mutasi[0]['individu']);

print("<table class='data'>");
print("<tr class='data'><td class='data'><b>Nim</b></td>");
    for($i=0;$i<$mhs;$i++){
        print("<td>" . $this->arr_mutasi[0]['individu'][$i]['nim']. "</td>");
    }
    print("</tr>\n");
print("<tr class='data'><td class='data'><b>x1</b></td>");
    for($i=0;$i<$mhs;$i++){
        print("<td class='data'>" . $this->arr_mutasi[0]['individu'][$i]['asal']. "</td>");
    }
    print("</tr>\n");
print("<tr class='data'><td class='data'><b>x2</b></td>");
    for($i=0;$i<$mhs;$i++){
        print("<td class='data'>" . $this->arr_mutasi[0]['individu'][$i]['tes']. "</td>");
    }
    print("</tr>\n");
print("<tr class='data'><td class='data'><b>fit</b></td>");
    for($i=0;$i<$mhs;$i++){
        print("<td class='data'>" . $this->arr_mutasi[0]['individu'][$i]['fit']. "</td>");
    }
print("</tr>\n");
print("</tr>\n");
print("</table>\n");

```

g. Pembagian kelas

Pembagian kelas dilakukanurut mulai kelas yang paling tinggi sampai terendah. Jumlah mahasiswa di tiap-tiap kelas disamaratakan antara kelas

yang satu dengan kelas yang lainnya sehingga semua mahasiswa dalam satu *mabna* mendapatkan kelas. Posisi kelas tertinggi ditempati oleh individu dengan nilai *fitness* tertinggi sesuai hasil dari proses algoritma genetika.

Berikut cuplikan program untuk pengklasifikasian kelas:

```

$mhs = $this->jumlahMhs();
$kls = $this->jumlahKls();
$mhsperkls = $mhs/$kls;

foreach ($this->arr_mutasi[0]['individu'] as $key => $row) {
    $nim[$key] = $row['nim'];
    $asal[$key] = $row['asal'];
    $fit[$key] = $row['fit'];
}

array_multisort($fit, SORT_DESC, $asal, SORT_DESC, $this->arr_mutasi[0]['individu']);

echo "<h3>Kelas Al-Ali</h3>";
<table class='data'>
<tr class='data'><td class='data'>No</td>
<td class='data'>NIM</td>
<td class='data'>Nama</td>
<td class='data'>Fit</td> </tr>";

$no = 1;
for($i=0;$i<$mhsperkls;$i++){
    $nim = $this->arr_mutasi[0]['individu'][$i]['nim'];
    $fit = $this->arr_mutasi[0]['individu'][$i]['fit'];

    $stampil=mysql_query("SELECT nim, nama FROM mahasantri WHERE nim = $nim
LIMIT $mhsperkls ");

    while($r=mysql_fetch_array($stampil)){
        echo "<tr><td class='data'>$no</td>
<td class='data'>$r[nim]</td>
<td class='data'>$r[nama]</td>
<td class='data'>$fit</td>
</tr>";

        $no++;
    }
}

echo "</table>\n";

```

4.2 Uji Coba

Aplikasi diterapkan dengan memasukkan data mahasiswa baru UIN Maulana Malik Ibrahim Malang Angkatan 2013/2014 yang sudah menempati *ma'had*. Sampel data mahasantri yang digunakan adalah mahasantri baru *Ma'had* Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang yang menempati *mabna* Ibnu Rusdy. Berikut datanya :

Tabel 4.1 Sampel Data Mahasantri

Mahasantri Baru	267
Kelas	7
Kriteria	<ul style="list-style-type: none"> ➤ Asal Sekolah ➤ Nilai

Pada tabel tersebut diketahui bahwa terdapat 267 mahasantri baru, 7 kelas dan 2 kriteria yang akan diproses dalam algoritma genetika.

Langkah selanjutnya dalam proses ini adalah menentukan parameter genetika. kondisi *default* kombinasi parameter genetika yang digunakan adalah :

Tabel 4.2 Parameter Genetika

Generasi	10
Populasi	30
Peluang <i>crossover</i>	0.45
Peluang mutasi	0.01

Pada tabel tersebut, secara *default* dapat dijelaskan bahwa jumlah generasi atau *iterasi* yang dihasilkan adalah sepuluh generasi. Setiap generasi menghasilkan 30 populasi atau 30 kromosom. Peluang *crossover* (P_c) adalah 0.45, diharapkan 45% dari 30 kromosom mengalami *crossover*. Peluang mutasi sebesar

0,01, berarti jika nanti pada proses mutasi nilai random yang dihasilkan kurang dari peluang mutasi, maka proses mutasi terjadi pada kromosom tersebut.

Berikut hasil proses genetika berdasarkan parameter tersebut.

Generasi 1																				
Nim	13220001	13320113	13410140	13660039	13510126	13510063	13110013	13210002	13140109	13530006	13510135	13210098	13510132	13320013	13320111	13110020	13610069	13650006	13410076	13510125
X1	5	4	2	1	2	4	6	5	2	2	2	2	1	2	5	3	2	1	2	1
X2	2	2	2	2	2	1	2	2	2	3	2	2	1	2	2	2	3	2	2	1
Fitness 303																				
Nim	13220120	13320112	13510135	13410012	13330099	13330087	13670035	13110102	13410011	13330058	13510008	13110014	13140059	13670004	13210018	13110106	13630066	13220124	13130003	13110118
X1	2	2	2	1	2	4	2	1	2	2	5	3	5	2	4	2	2	2	2	3
X2	2	2	2	2	3	2	2	2	1	3	3	2	1	2	3	3	2	1	2	2
Fitness 296																				
Nim	13110164	13220004	13650003	13610059	13410014	13660038	13130046	13310074	13410012	13510132	13650006	13660061	13130005	13650047	13210098	13330007	13210018	13540015	13620073	13110020
X1	2	2	2	2	4	5	5	5	1	1	1	4	1	5	2	6	2	2	2	3
X2	2	2	2	2	1	3	2	3	2	1	2	2	2	2	2	3	2	2	2	2
Fitness 304																				
Nim	13220115	13660063	13650074	13520005	13410074	13110110	13670017	13640004	13650004	13410071	13310081	13630058	13630062	13310010	13610064	13220070	13220063	13650007	13510066	13320117
X1	2	5	2	2	5	5	5	3	1	4	4	5	6	6	1	5	4	1	2	5
X2	1	2	1	1	3	2	2	1	1	2	3	2	2	2	1	2	2	2	1	1
Fitness 301																				
Nim	13110116	13130001	13660037	13540014	13410074	13510006	13650050	13210070	13310010	13110168	13130104	13220010	13650077	13610069	13510066	13310008	13220111	13630062	13660058	13410011
X1	5	4	5	3	5	2	2	2	6	4	2	2	5	2	2	6	2	6	2	1
X2	2	1	2	2	3	2	2	2	2	2	2	2	3	3	1	3	2	2	2	1

Gambar 4.13 Inisialisasi awal

Pada inisialisasi awal, individu di bangkitkan secara acak kemudian dihitung nilai *fitness*nya untuk proses seleksi. Individu dengan *fitness* tertinggi terpilih menjadi calon induk untuk proses *crossover*.

Generasi ke-2
Sebelum Seleksi

id : 1, fitness : 303, pk : 0.033711615487316 Qk : 0.033711615487316
id : 2, fitness : 296, pk : 0.032932799287939 Qk : 0.066644414775256
id : 3, fitness : 304, pk : 0.03382287494437 Qk : 0.10046728971963
id : 4, fitness : 301, pk : 0.033489096573209 Qk : 0.13395638629283
id : 5, fitness : 304, pk : 0.03382287494437 Qk : 0.16777926123721
id : 6, fitness : 296, pk : 0.032932799287939 Qk : 0.20071206052514
id : 7, fitness : 298, pk : 0.033155318202047 Qk : 0.23386737872719
id : 8, fitness : 305, pk : 0.033934134401424 Qk : 0.26780151312862
id : 9, fitness : 301, pk : 0.033489096573209 Qk : 0.30129060970182
id : 10, fitness : 298, pk : 0.033155318202047 Qk : 0.33444592790387
id : 11, fitness : 299, pk : 0.033266577659101 Qk : 0.36771250556297
id : 12, fitness : 298, pk : 0.033155318202047 Qk : 0.40086782376502
id : 13, fitness : 297, pk : 0.033044058744993 Qk : 0.43391188251001
id : 14, fitness : 298, pk : 0.033155318202047 Qk : 0.46706720071206
id : 15, fitness : 304, pk : 0.03382287494437 Qk : 0.50089007565643
id : 16, fitness : 294, pk : 0.032710280373832 Qk : 0.53360035603026
id : 17, fitness : 302, pk : 0.033600356030263 Qk : 0.56720071206053
id : 18, fitness : 297, pk : 0.033044058744993 Qk : 0.60024477080552
id : 19, fitness : 308, pk : 0.034267912772586 Qk : 0.6345126835781
id : 20, fitness : 299, pk : 0.033266577659101 Qk : 0.66777926123721
id : 21, fitness : 289, pk : 0.032153983088563 Qk : 0.69993324432577
id : 22, fitness : 298, pk : 0.033155318202047 Qk : 0.73308856252781
id : 23, fitness : 298, pk : 0.033155318202047 Qk : 0.76624388072986
id : 24, fitness : 305, pk : 0.033934134401424 Qk : 0.80017801513129
id : 25, fitness : 296, pk : 0.032932799287939 Qk : 0.83311081441923
id : 26, fitness : 299, pk : 0.033266577659101 Qk : 0.86637739207833
id : 27, fitness : 299, pk : 0.033266577659101 Qk : 0.89964396973743
id : 28, fitness : 301, pk : 0.033489096573209 Qk : 0.93313306631064
id : 29, fitness : 297, pk : 0.033044058744993 Qk : 0.96617712505563
id : 30, fitness : 304, pk : 0.03382287494437 Qk : 1

Gambar 4.14 Nilai *fitness* tiap individu

crossover Generasi ke-2
 Crossover individu ke-1
 Crossover individu ke-12
 Crossover individu ke-17
 Crossover individu ke-20
 Crossover individu ke-27
 Crossover individu ke-29

Gambar 4.15 Individu yang terpilih untuk *crossover*

Pada gambar 4.14 diketahui nilai *fitness* masing-masing individu untuk dihitung *fitness* kumulatifnya. setelah itu dibangkitkan bilangan *random*, untuk memilih individu yang akan di *crossover*.

Nim	13140014	13210016	13650071	13330050	13650014	13530007	13110120	13510069	13140006	13110006	13660038	13310007	13650077	13630062	13510012	13510073
x1	1	2	2	3	2	5	4	1	3	5	3	5	6	2	1	
x2	2	1	2	3	2	1	2	2	1	3	3	2	3	2	2	2
fit	3	2	4	5	5	3	7	6	2	6	8	5	8	8	4	3

fitness setelah mutasi: 304

Nim	13610066	13650075	13220118	13620038	13650053	13630013	13110168	13220062	13310011	13310011	13110018	13130012	13660008	13220014	13110013	13640055
x1	2	3	6	4	2	2	4	3	6	6	4	2	3	1	6	2
x2	2	2	2	1	1	1	2	2	2	2	1	1	2	1	2	1
fit	4	5	8	5	3	3	6	5	8	8	5	3	5	2	8	3

fitness setelah mutasi: 297

Nim	13110164	13520004	13660063	13220010	13410014	13510071	13140062	13130004	13410012	13510069	13210070	13320111	13220014	13220016	13320120	13110022
x1	2	2	5	2	4	2	6	4	1	4	2	5	1	3	2	5
x2	2	2	2	2	1	1	3	1	2	2	2	2	1	1	2	3
fit	4	4	7	4	5	3	9	5	3	6	4	7	2	4	4	8

fitness setelah mutasi: 300

Nim	13130051	13210098	13650003	13220001	13510066	13210009	13510072	13310074	13640044	13510132	13650006	13660061	13130005	13650047	13210098	13330007
x1	6	2	2	5	2	5	6	5	1	1	4	1	5	2	6	
x2	2	2	2	2	1	2	1	3	2	1	2	2	2	2	2	
fit	8	4	4	7	3	7	7	8	3	2	3	6	3	7	4	8

fitness setelah mutasi: 302

Nim	13510129	13650076	13220070	13610059	13320013	13660038	13130046	13330058	13220070	13630065	13310044	13410074	13310011	13510063	13210002	13530002
x1	1	5	5	2	2	5	5	2	5	2	5	5	6	4	5	2
x2	1	2	2	2	2	3	2	3	2	2	2	3	2	1	2	2
fit	2	7	7	4	4	8	7	5	7	4	7	8	8	5	7	4

Gambar 4.16 Hasil mutasi

Proses mutasi dipengaruhi oleh probabilitas mutasi, semakin kecil nilai probabilitas mutasi, maka semakin kecil pula suatu individu dalam suatu generasi mengalami mutasi. Hal ini untuk menjaga individu terbaik dalam generasi tidak rusak atau hilang karena proses mutasi terlalu sering.

Kelas Mutawasith A

No	NIM	Nama
1	13130046	IMAM SYAIFUDIN
2	13130050	AHMAD SYIHABUDDIN ASRORI
3	13210002	M. NUR AZIS
4	13210009	M. SHODRI FALAHUDDIN
5	13210013	MUHAMMAD KHALILURRAHMAN
6	13220001	NUR FAHMI AZHARI
7	13220070	M MUHSIN R
8	13310044	ANUGERAH FEBRIANSYAH
9	13320111	MOH NUR HIDAYATULLAH
10	13320114	FADLY SHOLEHUDIN A
11	13330053	MUAMMAR NUR ISLAMI
12	13330086	TRI FAHMI AMIRUDDIN
13	13410135	MOHAMMAD KHALIL
14	13510124	REZKY YOGA SYAHPUTRA
15	13520068	MUHAMMAD BAHTIAR
16	13530001	NURUDDIN FIRDAUS
17	13620006	LUTHFI HAKIM SUDRAJAT
18	13620036	AHMAD ALAMUL YAQIN
19	13630058	AHMAD SULTON AKBARUL FALAH
20	13640009	ISMAN HALIS
21	13640050	ARYZA ARTA DHANIAR
22	13650047	MUHAMMAD ISMAIL HASAN
23	13650076	YOGI PRADANA
24	13660037	AHMAD FAOJAN
25	13660063	MOHAMMAD SAICHUL ABBAS

Gambar 4.17 Hasil Penempatan

4.3 Analisa Hasil

Hasil penempatan diperoleh dengan mengambil kromosom yang memiliki *fitness* tertinggi sebagai individu terbaik pada sebuah generasi. Secara umum, jumlah generasi berpengaruh terhadap nilai *fitness* dan proses. Semakin banyak generasi maka nilai *fitness* yang diperoleh semakin baik. Akan tetapi semakin lama proses genetika yang terjadi, karena dalam setiap generasi terjadi proses seleksi, *crossover* dan mutasi. Banyaknya generasi juga tidak bisa menaikkan nilai *fitness* terus menerus, karena pada generasi tertentu nilai *fitness* akan konvergen sehingga nilai *fitness* tidak mengalami perubahan.

Berikut nilai *fitness* hasil uji coba :

Tabel 4.3 Nilai *fitness* inisialisasi awal

Kromosom	Nilai Fitness	Kromosom	Nilai Fitness
1	303	16	294
2	296	17	302
3	304	18	297
4	301	19	308
5	304	20	299
6	296	21	289
7	298	22	298
8	305	23	298
9	301	24	305
10	298	25	296
11	299	26	299
12	298	27	299
13	297	28	301
14	298	29	297
15	304	30	304

Nilai *Fitness* dari inisialisasi tersebut, selanjutnya mengalami proses seleksi.

Tabel 4.4 Nilai *fitness* Setelah Seleksi

Kromosom	Nilai Fitness	Kromosom	Nilai Fitness
1	301	16	298
2	298	17	299
3	304	18	298
4	302	19	297
5	299	20	304
6	297	21	301
7	308	22	304
8	301	23	299
9	298	24	297
10	305	25	308
11	298	26	296
12	298	27	298
13	297	28	289
14	298	29	297
15	299	30	304

Tabel 4.5 Kromosom yang mengalami *crossover*

<i>Parent</i>	Nilai <i>Fitness</i>	Kromosom Anak	Nilai <i>Fitness</i>
Kromosom ke-1	301	Kromosom ke-1	304
Kromosom ke-12	298	Kromosom ke-12	297
Kromosom ke-17	299	Kromosom ke-17	300
Kromosom ke-20	304	Kromosom ke-20	302
Kromosom ke-27	298	Kromosom ke-27	297
Kromosom ke-29	297	Kromosom ke-29	297

Berdasarkan Tabel 4.3 diketahui nilai *fitness* tiap-tiap kromosom pada inisialisasi awal. Kromosom dengan nilai *fitness* tertinggi di seleksi menjadi

parent untuk *crossover*. Tabel 4.5 menunjukkan kromosom yang menjadi *parent* dan mengalami *crossover* yang menghasilkan kromosom anak.

Tabel 4.6 Kromosom yang mengalami mutasi

Kromosom	Gen yang dimutasi
28	25 dan 86

Pada tabel tersebut menunjukkan gen yang dimutasi pada kromosom yang dipilih secara acak. Karena peluang mutasi kecil yaitu 0,001 atau hanya 1 persen, maka proses ini kadang tidak terjadi.

Tabel 4.7 Populasi *fitness* pada generasi ke-2

Kromosom	Nilai Fitness	Kromosom	Nilai Fitness
1	304	16	305
2	297	17	298
3	300	18	297
4	302	19	297
5	297	20	298
6	297	21	299
7	304	22	298
8	298	23	300
9	304	24	298
10	302	25	297
11	299	26	302
12	297	27	301
13	308	28	304
14	301	29	299
15	298	30	297

Pada tabel tersebut menunjukkan hasil dari proses yang menghasilkan generasi ke-2. Untuk generasi ke-3 dan selanjutnya dilakukan proses yang sama. Berikut hasil setelah generasi ke-10.

Tabel 4.8 Populasi pada generasi ke-10

Kromosom	Nilai <i>Fitness</i>	Kromosom	Nilai <i>Fitness</i>
1	304	16	304
2	297	17	298
3	300	18	297
4	302	19	297
5	297	20	298
6	297	21	299
7	304	22	298
8	298	23	300
9	304	24	298
10	302	25	297
11	299	26	302
12	297	27	301
13	304	28	304
14	301	29	299
15	298	30	297

Pada analisa hasil akan dilakukan uji coba dengan mengganti nilai parameter genetika. Hal ini dilakukan untuk mengetahui keakuratan *output* yang dihasilkan. individu terbaik adalah kromosom yang memiliki nilai *fitness* tertinggi dalam satu generasi. Semakin banyak generasi maka nilai *fitness* yang didapatkan akan semakin baik, namun bukan berarti banyaknya generasi dapat menghasilkan nilai *fitness* yang tinggi. Pada generasi tertentu nilai *fitness* akan konvergen sehingga nilai *fitness* tidak mengalami perubahan.

Tabel 4.9 Hasil uji coba

Generasi	Pop	PC	PM	Waktu	Memory	Σ Data Lama	Σ Data Baru	Data yang sama	Persentase Kesesuaian
2	3	0.45	0.01	0.61 s	0.99 MB	267	267	232	86.89 %
2	5	0.45	0.01	0.78 s	1.02 MB	267	267	234	87.64 %
2	10	0.45	0.01	0.93 s	1.13 MB	267	267	233	87.26 %
5	10	0.45	0.01	1.95 s	1.17 MB	267	267	229	85.76 %
2	3	0.6	0.01	0.60 s	0.99 MB	267	267	237	88.76 %
2	5	0.6	0.01	0.70 s	1.05 MB	267	267	236	88.38 %
2	10	0.6	0.01	0.95 s	1.16 MB	267	267	237	88.76 %
5	10	0.6	0.01	2.18 s	1.32 MB	267	267	235	88.01 %
2	3	0.6	0.05	0.64 s	0.99 MB	267	267	235	88.01 %
2	5	0.6	0.05	0.64 s	1.05 MB	267	267	232	86.89 %
2	10	0.6	0.05	1.06 s	1.16 MB	267	267	243	91.01 %
5	10	0.6	0.05	2.11 s	1.32 MB	267	267	235	88.01 %
5	20	0.45	0.01	2.74 s	1.61 MB	267	267	238	89.13 %
10	20	0.45	0.01	6.90 s	2.06 MB	267	267	241	90.26 %
5	30	0.45	0.01	5.22 s	1.79 MB	267	267	232	86.89 %
10	30	0.45	0.01	10.27 s	2.61 MB	267	267	232	86.89 %
5	20	0.6	0.01	3.36 s	1.63 MB	267	267	233	87.26 %
10	20	0.6	0.01	7.14 s	2.30 MB	267	267	234	87.64 %
5	30	0.6	0.01	5.16 s	2.08 MB	267	267	244	91.38 %
10	30	0.6	0.01	2.18 s	2.90 MB	267	267	238	89.13 %
5	20	0.45	0.05	2.96 s	1.61 MB	267	267	234	87.64 %
10	20	0.45	0.05	6.47 s	1.96 MB	267	267	236	88.38 %
5	30	0.45	0.05	4.83 s	1.93 MB	267	267	232	86.89 %
10	30	0.45	0.05	12.90 s	2.49 MB	267	267	239	89.51%
5	20	0.6	0.05	2.83 s	1.70 MB	267	267	234	87.64 %
10	20	0.6	0.05	4.11 s	2.21 MB	267	267	230	86.14 %
5	30	0.6	0.05	5.86 s	1.98 MB	267	267	236	88.38 %
10	30	0.6	0.05	13.81 s	2.87 MB	267	267	234	87.64 %

Tabel 4.9 merupakan hasil uji coba dengan mengganti parameter genetika dengan membandingkan waktu dan memori yang digunakan saat proses genetika berlangsung, kemudian dibandingkan dengan hasil manual. Hasil yang didapat bahwa semakin besar generasi dan atau populasi semakin lama waktu yang diperlukan dan semakin banyak memory yang digunakan.

4.4 Pengklasifikasian Menurut Islam

Pengklasifikasian merupakan cara untuk mengelompokkan. Banyaknya cara yang digunakan untuk mengklasifikasi hendaknya memperhatikan kemampuan mahasiswa baru yang akan belajar sehingga berada di kelas yang tepat.

Firman Allah dalam Al-Qur'an surat An-Nahl ayat 90.

إِنَّ اللَّهَ يَأْمُرُ بِالْعَدْلِ وَالْإِحْسَانِ وَإِيتَايَ ذِي الْقُرْبَىٰ وَيَنْهَىٰ عَنِ الْفَحْشَاءِ
وَالْمُنْكَرِ وَالْبَغْيِ ۚ يَعِظُكُمْ لَعَلَّكُمْ تَذَكَّرُونَ ﴿٩٠﴾

Artinya :

Sesungguhnya Allah menyuruh (kamu) Berlaku adil dan berbuat kebajikan, memberi kepada kaum kerabat, dan Allah melarang dari perbuatan keji, kemungkaran dan permusuhan. Dia memberi pengajaran kepadamu agar kamu dapat mengambil pelajaran. (QS. An-Nahl:90).

Sesungguhnya Allah secara terus menerus memerintahkan siapapun diantara hamba hamba-Nya untuk berlaku adil dalam sikap, ucapan dan tindakan, walau terhadap diri sendiri dan menganjurkan berbuat ihsan yang lebih utama dari keadilan. Dengan perintah dan larangan Dia memberi pengajaran dan bimbingan kepada kamu semua menyangkut segala aspek kebajikan agar selalu ingat dan mengambil pelajaran yang berharga (Quraish Shihab 2002:323-324).

Berdasarkan ayat tersebut, kita harus melakukan sesuatu dengan adil, musuhpun harus kita perlakukan adil seperti yang telah dijelaskan dalam surat Al-Maidah ayat 8.

يَتَأَيُّهَا الَّذِينَ ءَامَنُوا كُونُوا قَوَّامِينَ لِلَّهِ شُهَدَاءَ بِالْقِسْطِ ۚ وَلَا يَجْرِمَنَّكُمْ شَنَاٰنُ قَوْمٍ عَلَىٰ ءَلَّا تَعْدِلُوا ۚ اَعْدِلُوا هُوَ اَقْرَبُ لِلتَّقْوَىٰ ۚ وَاتَّقُوا اللَّهَ ۚ اِنَّ اللَّهَ خَبِيْرٌۢ بِمَا تَعْمَلُوْنَ ﴿٨﴾

Artinya :

Hai orang-orang yang beriman hendaklah kamu jadi orang-orang yang selalu menegakkan (kebenaran) karena Allah, menjadi saksi dengan adil. dan janganlah sekali-kali kebencianmu terhadap sesuatu kaum, mendorong kamu untuk berlaku tidak adil. Berlaku adillah, karena adil itu lebih dekat kepada takwa. dan bertakwalah kepada Allah, Sesungguhnya Allah Maha mengetahui apa yang kamu kerjakan.

(QS. Al-Maidah:8).

Ayat tersebut menyuruh kita untuk selalu berlaku adil terhadap diri sendiri ataupun musuh. Adil berarti menempatkan sesuatu pada tempat yang semestinya. Seorang muslim wajib berlaku adil dalam menyelesaikan suatu masalah, karena adil itu lebih dekat kepada takwa yang sempurna. Karena itu, dalam program pengklasifikasian ini digunakan prinsip keadilan yaitu dengan menempatkan mahasiswa kedalam kelas yang sesuai kemampuannya berdasarkan nilai tes dan asal sekolah.

BAB V

PENUTUP

5.1 Kesimpulan

Penerapan program ini untuk penerapan pengklasifikasian kelas mahasiswa baru menggunakan algoritma genetika sudah mencapai hasil optimal, akan tetapi perlu penambahan data untuk kriteria yang diberikan agar bisa mencapai hasil maksimal, untuk itu perlu dilakukan penelitian lebih lanjut.

Hasil uji coba memperoleh hasil yang baik pada generasi 5, populasi 30, peluang *crossover* 0,6 dan peluang mutasi 0,01 dengan waktu 5.16 detik, *memory* yang digunakan mencapai 2.08 MB dengan tingkat kesesuaian komposisi mahasiswa 91% dengan cara manual.

5.2 Saran

Perlu diadakan penelitian lebih lanjut mengenai komposisi pengklasifikasian kelas *ta'lim afkar* agar lebih efektif dan efisien demi tercapainya visi, misi dan tujuan Ma'had Sunan Ampel Al-'Ali UIN Maulana Malik Ibrahim Malang.

Program pengklasifikasian kelas *ta'lim afkar* menggunakan algoritma genetika ini masih jauh dari sempurna, oleh karena itu perlu saran dan kritik untuk perbaikan dan pengembangan program lebih lanjut.

DAFTAR PUSTAKA

- Al-Qur'an al-Karim.
- Departemen Agama. 2003. *Pondok Pesantren dan Madarrasah Diniyah: Pertumbuhan dan Perkembangannya*.
- Desiani, Anita dan Muhammad Arhami. 2006. *Konsep Kecerdasan Buatan*. Yogyakarta: Andi.
- Kusumadewi, Sri. 2003. *Artificial Intelegence (Teknik dan Aplikasinya)*. Yogyakarta: Graha Ilmu.
- Kusumadewi, Sri. dan Hari Purnomo. 2005. *Aplikasi Logika Fuzzy Untuk Pendukung Keputusan*. Yogyakarta: Graha Ilmu.
- Kuswadi, Son. 2007. *Kendali Cerdas, Teori dan Aplikasi Praktisnya*. Yogyakarta: Andi.
- Ridwan, Mujib. 2009. Optimasi Penempatan Mahasiswa Baru Di Ma'had Sunan Ampel Al-Ali Universitas Islam Negeri (UIN) Malang Menggunakan Algoritma Genetika. *Skripsi Tidak Diterbitkan*. Malang: Jurusan Teknik Informatika Fakultas Sainstek UIN Maliki Malang.
- Shihab, M. Quraish. 2003. *Tafsir Al-Misbah: Pesan, Kesan dan Keserasian Al-Qur'an, Vol. 1*. Jakarta: Lentera Hati.
- Shihab, M. Quraish. 2003. *Tafsir Al-Misbah: Pesan, Kesan dan Keserasian Al-Qur'an, Vol. 14*. Jakarta: Lentera Hati.
- Sholihq. 2006. *Pemodelan Sistem Informasi Berorientasi Obyek dengan UML*. Yogyakarta: Graha Ilmu.
- Suyanto. 2005. *Algoritma Genetika dalam Matlab*. Yogyakarta: Andi.
- Suyanto. 2011. *Artificial Intelligence: Searching, Reasoning, Planning dan learning, edisi revisi*. Bandung: Informatika.
- Universitas Islam Negeri (UIN) Malang. 2006. *Guidebook of Ma'had Sunan Ampel Al-Aly (The State Islamic University) 2006-2007*. Malang: UIN Malang Press.
- Zahro, Hani Zulfia. 2011. Sistem Pendukung Keputusan Pembagian Kelas Reguler Siswa Baru Menggunakan Algoritma Genetika di SMA Negeri 1 Lawang. *Skripsi Tidak Diterbitkan*. Malang: Jurusan Teknik Informatika Fakultas Sainstek UIN Maliki Malang.

Lampiran 1 perbandingan hasil algoritma genetika

Kelas Al-Aly Hasil Algoritma Genetika			Kelas Al-Aly		
No	NIM	Nama	No	NIM	Nama
1	13110111	M. ROFI'UR RUTABI	1	13410005	AGUNG DARMANSYAH
2	13140062	HERMAN	2	13110006	AHMAD NUR
3	13310008	HAYYIN NALAL F.	3	13510008	REZA ARIFianto
4	13310077	FAIS FATAYANI	4	13320004	MOH. ZA'IMIL ALIVIN
5	13310078	M. KHOLILUR R	5	13210001	AGUNG WIRAYUDA
6	13110012	MAS FARIZAL SIDIK	6	13630001	M. HIDAYATUL M
7	13110013	M. MAKHRUS SALIM	7	13630009	M SHOBIHUL KHOIR
8	13130051	MIFTACHUL TAUFIQI	8	13110015	RIZKY ALDOTRIO W
9	13130096	BHIMA MAHARDITIA	9	13410013	MOHAMMAD FIKRI Z
10	13220013	AFIFUDDIN	10	13310008	HAYYIN NALAL F
11	13220118	HAYAT	11	13110019	FAHMI FARDIANSYAH
12	13310010	M. KHISNULLOH ALI	12	13210018	RISKON AS SHIDDIQIE
13	13310011	M. BAADIYUS SURUR	13	13110022	A CHIZAM BAIHAQY
14	13330007	M. SURGO FIRDAUS	14	13220017	IDKHAM KHALID
15	13510155	M. FAIDLUN NI'AM	15	13110106	M ABDUL HAMID A
16	13540014	FIRMAN SYAHRUL H.	16	13140062	HERMAN
17	13110022	A. CHIZAM BAIHAQY	17	13110111	M. ROFI'UR RUTABI
18	13110160	WAHAB SULTAN	18	13410074	MINAN NUR ROHMAN
19	13210099	A. DERMAWAN M.	19	13530006	M ABDUL AZIS
20	13310074	MUHAMMAD AYYUB A	20	13110113	SAIFUL RIJAL
21	13410074	MINAN NUR ROHMAN	21	13330050	BIMA SUGENG P
22	13630009	M. SHOBIHUL KHOIR	22	13660038	DARARI TAUFIQ F
23	13650077	M TAQUIDDIN ISLAMI	23	13330058	M BAGUS A N
24	13650082	M FAHRUDIN MAHDI	24	13210099	A DERMAWAN M
25	13660038	DARARI TAUFIQ F	25	13310074	M AYYUB ASYHARI
26	13320117	HERMAWAN PETRUS	26	13650077	M TAQUIDDIN ISLAMI
27	13320119	M. DZIA UL HAQ	27	13110160	WAHAB SULTAN

28	13510063	M. FARKHAN	28	13630060	REZA FAJAR SHOLEH
29	13510123	HADIYAN NURBAINA T	29	13310077	FAIS FATAYANI
30	13620033	SUHARTONO	30	13310078	M KHOLILUR R
31	13650056	FAISAL BRILIANSYAH	31	13510134	M AQIL MAULANA
32	13110014	M. NAUFAL HUMAM	32	13330099	TRI AULIA ADNAN
33	13110107	ZAINAL ARIFIN	33	13650082	M FAHRUDIN MAHDI
34	13110110	M. ALIQODIN	34	13610069	FATHULLAH FUADY
35	13110116	M. ALFAN HUDA	35	13310081	MUHAMMAD ROJI
Kelas Mutassith A Hasil Algoritma Genetika			Kelas Mutassith A		
No	NIM	Nama	No	NIM	Nama
1	13110120	BAGUS WIBAWA K.	1	13210005	IIF BHRUL ARIFIN
2	13130046	IMAM SYAIFUDIN	2	13330001	W NURSIRANDA PUTRA
3	13130050	A. SYIHABUDDIN A	3	13210009	M. SHODRI F
4	13210002	M. NUR AZIS	4	13110009	A FAIZ M R
5	13210009	M. SHODRI F.	5	13220009	AHMAD SYAIFUR RIZAL
6	13210013	M. KHALILURRAHMAN	6	13110012	MAS FARIZAL SIDIK
7	13210068	ZAINUL MUSTOFA	7	13220010	DENNI WIDJAKSONO P
8	13220001	NUR FAHMI AZHARI	8	13210002	M. NUR AZIS
9	13220009	AHMAD SYAIFUR RIZAL	9	13110013	M. MAKHRUS SALIM
10	13220070	M MUHSIN R	10	13330007	M. SURGO FIRDAUS
11	13310044	ANUGERAH F.	11	13310007	SYAUQI ABRORI BADRI
12	13320111	MOH NUR H	12	13110014	M NAUFAL HUMAM
13	13330053	MUAMMAR NUR I	13	13110016	IMAM SIROJUDDIN
14	13330086	TRI FAHMI AMIRUDDIN	14	13410012	EKO BAYU KRISNUR P
15	13410078	HARIS HANIFAH	15	13410017	ARI ISWAHYUDI
16	13410135	MOHAMMAD KHALIL	16	13620010	MOCH. FAIZUL HUDA
17	13520068	MUHAMMAD BAHTIAR	17	13110020	AMIR FAHMI A
18	13530001	NURUDDIN FIRDAUS	18	13140014	AINUR FIRMANSYAH
19	13610033	A. SUKRON JAZULI	19	13310010	M KHISNULLOH ALI W
20	13620006	LUTHFI HAKIM S	20	13310011	M BAADIYUS SURUR
21	13620036	AHMAD ALAMUL	21	13660008	FAISHOL ROZIQL

22	13640009	ISMAN HALIS	22	13210068	ZAINUL MUSTOFA
23	13650047	M. ISMAIL HASAN	23	13110103	AGUS BUDI SUSILO
24	13660037	AHMAD FAOJAN	24	13130046	IMAM SYAIFUDIN
25	13660058	A. HARIZUL HAKAM W	25	13220062	FAUZI ABDILAH
26	13660063	M SAICHUL ABBAS	26	13530001	NURUDDIN FIRDAUS
27	13670004	M ZULKHAQ V	27	13110107	ZAINAL ARIFIN
28	13670017	ANIS AKHWAN DHAFIN	28	13620034	IMAM SUBANDI
29	13110106	M. ABD. HAMID A	29	13650049	IMDAD RABBANI
30	13220017	IDKHAM KHALID	30	13620035	RUDINI
31	13310081	MUHAMMAD ROJI	31	13140061	SAFAK
32	13320004	MOH. ZA'IMIL ALIVIN	32	13130050	A SYIHABUDDIN A
33	13410005	AGUNG DARMANSYAH	33	13110110	M ALIQODIN
34	13110002	M. IRFAN ZAMZAMI	34	13130051	MIFTACHUL TAUFIQI
35	13130004	TIO BUKI	35	13310044	ANUGERAH F
36	13130012	M. SIFAU MASRURI	36	13310045	M HANIF AZIZI SM
37	13210007	FATHUR RAHMAN	37	13110114	A MUMTAZ ABDULLAH
Kelas Mutassith B Hasil Algoritma Genetika			Kelas Mutassith B		
1	13310079	MAULANA ISMAIL AZIS	1	13410075	M. MU'IZADIN NUR F
2	13320066	MUHAMAD RIZKY H	2	13410076	ABD NASIH ULWAN
3	13510071	REFRIGIANTO KUSUMA	3	13610033	A SUKRON JAZULI
4	13530007	PRATAMA ARISNA P	4	13650050	AHMAD SYAIFUDDIN Z
5	13640006	IRHAM NAJMUDIN	5	13660039	DIMAS BAGUS TRI S
6	13110016	IMAM SIROJUDDIN	6	13410081	KHOIRUL AFIFUDDIN
7	13110103	AGUS BUDI SUSILO	7	13330053	MUAMMAR NUR I
8	13110168	MOCHAMAD NUR H S	8	13660040	IMAM ALI RIZKI
9	13130054	HENDRA	9	13110116	M ALFAN HUDA
10	13210069	AHMAD SIDDIQ RIDHA	10	13510069	M ICHSAN SAUQI
11	13220063	ARDIAN YAZID	11	13660041	HAFIDH HAMDHAN
12	13320113	ATIQR RAHMAN	12	13320070	ABIYU HAIDAR R
13	13330001	WAHYULIANSYAH N P	13	13110120	BAGUS WIBAWA K
14	13330087	MUH. NUZULUL MUBTAAJIN	14	13660055	MOH BAHRUL ARIFIN

15	13410071	WILDAN HABIBULLOH	15	13330087	MUH. NUZULUL M
16	13410081	KHOIRUL AFIFUDDIN	16	13330088	NAZARUDIN ALFATH
17	13510011	MOHAMMAD BAGIR A	17	13510155	M FAIDLUN NI'AM
18	13510015	M NIZAR KHARIS M	18	13330091	AHMAD ZAHUDA
19	13510069	M ICHSAN SAUQI	19	13110161	M AFI YUDDIN N
20	13510070	A SYAIKHU NUR HILLAN	20	13110164	ABDUL HADI
21	13510124	REZKY YOGA S	21	13210100	M ALIF ILHAM R
22	13620034	IMAM SUBANDI	22	13520068	M BAHTIAR
23	13620035	RUDINI	23	13220118	HAYAT
24	13650006	ALFAN NAWAZIRU Z	24	13220119	MISBAHUDIN
25	13650076	YOGI PRADANA	25	13110166	M LUTFI HAMIDI
26	13660041	HAFIDH HAMDHAN	26	13110168	M NUR HADI SAPUTRO
27	13660061	FADEL RACHMANA A	27	13140109	SYAFI'IN
28	13110006	AHMAD NUR	28	13210101	AGUS IDNUDDIN
29	13210001	AGUNG WIRAYUDA	29	13540014	FIRMAN SYAHRUL H
30	13630001	M. HIDAYATUL M	30	13630064	M FAHMI FUADUL L
31	13110018	ABDAL MALIK FAJAR	31	13540015	AULIYA AKBAR R
32	13130001	AFIF ALFIKRI SU'AIDI	32	13410140	FAIZAL RAMADHAN
33	13410014	SLAMET	33	13130105	HANIF BAHTIAR R
34	13410073	ABDUL MUCHITH	34	13110174	ARIF RAHMAN HAKIM
35	13510074	FIRMANSYAH	35	13220128	INDRA PRASTA
36	13610010	SOLICHIN M	36	13210102	ANAS RONIYADI
37	13620038	MUHAMMAD RUSYDI	37	13620073	M IHSANUDDIN
Kelas Asasi A Hasil Algoritma Genetika			Kelas Asasi A		
1	13650051	FAIQ NUKHA	1	13660002	ALFIAN
2	13110020	AMIR FAHMI A	2	13220001	NUR FAHMI AZHARI
3	13110118	M KHOLISIN	3	13520001	M SYAHRUL ABIDIN
4	13220062	FAUZI ABDILAH	4	13510006	M YASIR ARAFAT P
5	13220071	YANUAR FAJRI A	5	13140006	RIDWAN NURIL FAUZI
6	13310007	SYAUQI ABRORI BADRI	6	13130004	TIO BUKI
7	13520010	DEDI DWI SETYAWAN	7	13210006	ALVIN CHAIR

8	13650014	JUNI ARAMIKO	8	13640004	ANSYORI
9	13650075	M.NABIL FAHD A	9	13210007	FATHUR RAHMAN
10	13660008	FAISHOL ROZIQI	10	13510009	ABROR SUYUDI Y
11	13660055	MOH BAHRUL ARIFIN	11	13520005	BUDI DHARMA A
12	13110015	RIZKY ALDOTRIO W	12	13650003	NURDIANSYAH
13	13210018	RISKON AS SHIDDIQIE	13	13510011	M BAGIR ASSEGAFF
14	13330050	BIMA SUGENG P	14	13650004	ARDIANSYAH SURYA P.
15	13330058	MUHAMMAD B	15	13640006	IRHAM NAJMUDIN
16	13330099	TRI AULIA ADNAN	16	13620007	EMAN SUHERMAN
17	13410013	MOHAMMAD FIKRI Z	17	13620002	DANANG HADI UTOMO
18	13510008	REZA ARIFANTO	18	13110002	M. IRFAN ZAMZAMI
19	13510134	MOHAMAD AQIL M	19	13130001	AFIF ALFIKRI SU'AIDI
20	13530006	MOHAMAD ABDUL A	20	13650007	M ARDI ZULFIAN
21	13610069	FATHULLAH FUADY	21	13630008	MIZANUL UKHROWI R
22	13630060	REZA FAJAR SHOLEH	22	13640009	ISMAN HALIS
23	13140059	APEHANSA R A	23	13650011	SWANDARU WISMOYO
24	13140108	MALIK ARIFIN	24	13410011	IMAM AKBAR W
25	13220016	MUHAMMAD B GHONI	25	13210016	ADI CANDRA IBRAHIM
26	13410024	NAUFAL MAFAZI	26	13650014	JUNI ARAMIKO
27	13510136	ZIDNY FITRA ZULFIKAR	27	13410014	SLAMET
28	13520041	ARDI PRIBADI	28	13670004	M ZULKHAQ V
29	13620007	EMAN SUHERMAN	29	13610010	SOLICHIN M
30	13630033	FIRDAUS ATAKA FAZA	30	13220013	AFIFUDDIN
31	13640004	ANSYORI	31	13220014	M LUKMAN IBRAHIM
32	13650011	SWANDARU WISMOYO	32	13110018	ABDAL MALIK FAJAR A
33	13650053	WACHIT WAHYU N	33	13320012	M ALLAM BAHARUDIN
34	13660002	ALFIAN	34	13630013	MAFTUH HANANI
35	13660006	RIZAL QOMARUZ Z	35	13620011	ARIS ABDUL HALIM
36	13110009	A FAIZ M R	36	13410020	EKO BAMBANG K
37	13110021	ALFIAN FIRMAN HALA	37	13110021	ALFIAN FIRMAN HALA
38	13110102	RESTU MAULANA MP	38	13640052	FAHRURRIJAL AZIZ
39	13110114	AHMAD MUMTAZ	39	13640056	MUHAMMAD JAMROZI

Kelas Asasi B Hasil Algoritma Genetika			Kelas Asasi B		
1	13110161	MOH.AFI YUDDIN N	1	13410073	ABDUL MUCHITH
2	13110164	ABDUL HADI	2	13660037	AHMAD FAOJAN
3	13110166	M LUTFI HAMIDI	3	13630033	FIRDAUS ATAKA FAZA
4	13110174	ARIF RAHMAN HAKIM	4	13530002	M. ANDRI PRASETYA
5	13130003	NANDA FADILA IKHSAN	5	13530007	PRATAMA ARISNA P
6	13130006	NENO UBAYDILAH	6	13530008	KIN MCCLLOUDS A M
7	13130104	DERI INDRA SETIAWAN	7	13510066	M FAHMI ANWAR
8	13130105	HANIF BAHTIAR R	8	13670017	ANIS AKHWAN DHAFIN
9	13140009	ANDRIAN YUFA B	9	13210069	AHMAD SIDDIQ RIDHA
10	13140061	SAFAK	10	13650051	FAIQ NUKHA
11	13140075	RIKZA AKMAL FARUQI	11	13130054	HENDRA
12	13140109	SYAFI'IN	12	13410078	HARIS HANIFAH
13	13210005	IIF BAHRUL ARIFIN	13	13620038	M RUSYDI AMIN
14	13210067	WAHID NUGROHO	14	13130057	M ARIF SETIAWAN
15	13210070	FAHMI BAHAR P	15	13320066	M RIZKY HERMAW
16	13210098	AHMAD FAJAR QOLBIN	16	13510070	A SYAIKHU NUR HILLAN
17	13210100	MUHAMMAD ALIF I	17	13510071	REFRIGIANTO KUSUMA
18	13210101	AGUS IDNUDIN	18	13510072	AIDIL SYAHRIN
19	13210102	ANAS RONIYADI	19	13650053	WACHIT WAHYU N
20	13220010	DENNI WIDJAKSONO P	20	13220071	YANUAR FAJRI A
21	13220110	YODHI SATRIA	21	13510074	FIRMANSYAH
22	13220111	ADITYA RAHMAN M	22	13410085	M. SULTHON DZUL H
23	13220119	MISBAHUDIN	23	13110118	M KHOLISIN
24	13220128	INDRA PRASTA	24	13520041	ARDI PRIBADI
25	13310045	MUHAMMAD HANIF A	25	13650055	TEGAR SWASONO
26	13320013	NAILUS SURUR	26	13650056	FAISAL BRILIANSYAH
27	13320070	ABIYU HAIDAR R	27	13140075	RIKZA AKMAL FARUQI
28	13320114	FADLY SHOLEHUDIN A	28	13220110	YODHI SATRIA
29	13320120	AHMAD MAULIDI A	29	13510123	HADIYAN NURBAINA T
30	13330091	AHMAD ZAHUDA	30	13510122	BAYU MISBAHUL

31	13410017	ARI ISWAHYUDI	31	13510124	REZKY YOGA S
32	13410075	M. MU'IZADIN NUR F	32	13210098	A FAJAR QOLBIN S
33	13410076	ABDULLOH NASIH U	33	13220111	ADITYA RAHMAN M
34	13510004	HAKAM ALFAQIH AZIZ	34	13640044	KHOIRUL ANAM
35	13510006	MUHAMMAD YASIR A	35	13330086	TRI FAHMI AMIRUDDIN
36	13510012	MUHAMMAD LUTHFI H	36	13510125	RIZALDI UMAR SIDIQ
37	13510065	ALDIKA	37	13320111	MOH NUR H
38	13510073	PRATAMA FAJAR R	38	13630066	AHMAD AAN SAID D
39	13510122	BAYU MISBAHUL U	39	13660063	M SAICHUL ABBAS
Kelas Asasi C Hasil Algoritma Genetika			Kelas Asasi C		
1	13510126	ZAINUN AL FIKRI	1	13410133	ERIC HERMANSYAH
2	13520004	WILDAN KHISBULLAH S	2	13610064	MOH ALI ROSYDIN
3	13520069	RAHMAD NUR M	3	13320119	M DZIA UL HAQ
4	13530002	M. ANDRI PRASETYA	4	13310079	MAULANA ISMAIL AZIS
5	13540015	AULIYA AKBAR R	5	13510132	AFIFUDDIN
6	13610057	M TAQIYUDDIN	6	13320120	A MAULIDI AGUS DIAN
7	13610059	M SIHABUDDIN	7	13540013	M MAGHFUR AL M
8	13610066	ANANG MAULANA	8	13320121	M LUTHFILLAH
9	13610084	IMAM BASORI	9	13630062	ARIF KHASANUDIN
10	13620010	MOCH. FAIZUL HUDA	10	13140108	MALIK ARIFIN
11	13620073	M IHSANUDDIN	11	13610066	ANANG MAULANA
12	13630062	ARIF KHASANUDIN	12	13660061	FADEL RACHMANA A
13	13630064	M FAHMI FUADUL	13	13130103	M SYAFIQ HASYWAF A
14	13630065	MUKSIN MAULANA	14	13130104	DERI INDRA SETIAWAN
15	13640052	FAHRURRIJAL AZIZ	15	13640050	ARYZA ARTA DHANIAR
16	13640056	M JAMROZI FARID	16	13630065	MUKSIN MAULANA
17	13650003	NURDIANSYAH	17	13220124	A FITHRUL MUBIN
18	13650049	IMDAD RABBANI	18	13520070	RIO ADAM Y
19	13650050	A SYAIFUDDIN Z	19	13510136	ZIDNY FITRA ZULFIKAR
20	13650071	SAIFULLAH	20	13640055	ILHAM ZAKIYAL ALBA
21	13650081	NUR HASAN	21	13620006	LUTHFI HAKIM SUBRIAT

22	13670035	ALMAS FIRAS KANZI J	22	13510004	HAKAM ALFAQIH AZIZ
23	13110019	FAHMI FARDIANSYAH	23	13130003	NANDA FADILA IKHSAN
24	13110113	SAIFUL RIJAL	24	13140009	ANDRIAN YUFA B
25	13130057	M ARIF SETIAWAN	25	13520004	WILDAN KHISBULLAH S
26	13140006	RIDWAN NURIL FAUZI	26	13130005	BUDI HARTONO
27	13210006	ALVIN CHAIR	27	13510012	M LUTHFI HAMDANI
28	13210066	A WILDAN DIMYATI	28	13130006	NENO UBAYDILAH
29	13220061	AGUS AGUNG SUSILO	29	13650006	ALFAN NAWAZIRU Z
30	13220115	AKHMAD MURTADA	30	13210013	M KHALILURRAHMAN
31	13220124	A FITHRUL MUBIN	31	13320013	NAILUS SURUR
32	13320012	M ALLAM BAHARUDIN	32	13410071	WILDAN HABIBULLOH
33	13320115	MOH.FAKHRURROZI	33	13510065	ALDIKA
34	13320121	M LUTHFILLAH	34	13620036	A ALAMUL YAQIN
35	13510009	ABROR SUYUDI Y	35	13210070	FAHMI BAHAR P
36	13510066	M FAHMI ANWAR	36	13130059	NANANG EDY LUGITO
37	13510072	AIDIL SYAHRIN	37	13630034	GURUH PRASETYA Y
38	13520001	M SYAHRUL ABIDIN	38	13510073	PRATAMA FAJAR R
39	13520005	BUDI DHARMA A	39	13670035	ALMAS FIRAS KANZI J
40	13540013	M MAGHFUR AL M	40	13520069	RAHMAD NUR M
Kelas Asasi D Hasil Algoritma Genetika			Kelas Asasi D		
1	13610032	ICHWAN ALIFUDIN	1	13520010	DEDI DWI SETYAWAN
2	13620011	ARIS ABDUL HALIM	2	13220016	M BUSTHOMI ABDUL G
3	13630008	MIZANUL UKHROWI R	3	13660006	RIZAL QOMARUZ Z
4	13630013	MAFTUH HANANI	4	13130012	M. SIFAU MASRURI
5	13640055	ILHAM ZAKIYAL ALBA	5	13510015	M NIZAR KHARIS M
6	13650074	MOH RIZAL FAIZUN	6	13410024	NAUFAL MAFAZI
7	13650079	AFRIZAL DWI KUSUMA	7	13110102	RESTU MAULANA MP
8	13130005	BUDI HARTONO	8	13220061	AGUS AGUNG SUSILO
9	13130059	NANANG EDY LUGITO	9	13210066	A WILDAN DIMYATI
10	13130103	MUHAMMAD SYAFIQ	10	13210067	WAHID NUGROHO
11	13140014	AINUR FIRMANSYAH	11	13510063	M. FARKHAN

12	13320112	MOH. HAIKAL ASLIKH R	12	13610032	ICHWAN ALIFUDIN
13	13330088	NAZARUDIN ALFATH	13	13650047	M ISMAIL HASAN
14	13410012	EKO BAYU KRISNUR P	14	13620033	SUHARTONO
15	13410085	M. SULTHON DZUL H	15	13140059	APEHANSA R
16	13410140	FAIZAL RAMADHAN	16	13220063	ARDIAN YAZID
17	13510127	AHMAD REZA PAHLEVI	17	13610059	M SIHABUDDIN
18	13510135	M. FAHRUDDIN	18	13320113	ATIQRUR RAHMAN
19	13630034	GURUH PRASETYA Y	19	13650074	MOH RIZAL FAIZUN
20	13630058	AHMAD SULTON A	20	13510126	ZAINUN AL FIKRI
21	13630066	AHMAD AAN SAID D	21	13510127	A REZA PAHLEVI
22	13640044	KHOIRUL ANAM	22	13650075	M.NABIL FAHD A
23	13640050	ARYZA ARTA DHANIAR	23	13630058	A SULTON A
24	13650007	M ARDI ZULFIAN	24	13650076	YOGI PRADANA
25	13650055	TEGAR SWASONO	25	13510129	ARIF ZHAMRONNY
26	13660039	DIMAS BAGUS TRI S	26	13320114	FADLY SHOLEHUDIN A
27	13660040	IMAM ALI RIZKI	27	13220115	AKHMAD MURTADA
28	13210016	ADI CANDRA IBRAHIM	28	13610084	IMAM BASORI
29	13220014	M LUKMAN IBRAHIM	29	13320115	MOH.FAKHRURROZI
30	13410011	IMAM AKBAR W	30	13660058	A HARIZUL HAKAM W
31	13410020	EKO BAMBANG K	31	13650079	AFRIZAL DWI KUSUMA
32	13410133	ERIC HERMANSYAH	32	13320117	HERMAWAN PETRUS
33	13510125	RIZALDI UMAR SIDIQ	33	13220070	M MUHSIN R
34	13510129	ARIF ZHAMRONNY	34	13650071	SAIFULLAH
35	13510132	AFIFUDDIN	35	13610057	M TAQIYUDDIN
36	13520070	RIO ADAM Y	36	13320112	MOH. HAIKAL ASLIKH R
37	13530008	KIN MCCLLOUDS A M	37	13130096	BHIMA MAHARDITIA
38	13610064	MOH ALI ROSYDIN	38	13410135	MOHAMMAD KHALIL
39	13620002	DANANG HADI UTOMO	39	13650081	NUR HASAN
40	13650004	ARDIANSYAH SURYA P.	40	13510135	M. FAHRUDDIN