

**RECOMMENDER SYSTEM MENGGUNAKAN METODE
ANALYTIC HIERARCHY PROCESS (AHP)
(Studi Kasus Pemilihan Program Studi di Universitas
Brawijaya)**

SKRIPSI

oleh:
**NURAINI
0410960042**



**PROGRAM STUDI ILMU KOMPUTER
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN
ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2009**







**RECOMMENDER SYSTEM MENGGUNAKAN METODE
ANALYTIC HIERARCHY PROCESS (AHP)
(Studi Kasus Pemilihan Program Studi di Universitas Brawijaya)**

SKRIPSI

Sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer
dalam bidang Ilmu Komputer

oleh:
NURAINI
0410960042



**PROGRAM STUDI ILMU KOMPUTER
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2009**



LEMBAR PENGESAHAN SKRIPSI

**RECOMMENDER SYSTEM MENGGUNAKAN METODE
ANALYTIC HIERARCHY PROCESS (AHP)
(Studi Kasus Pemilihan Program Studi di Universitas Brawijaya)**

oleh:
NURAINI
0410960042

Setelah dipertahankan di depan Majelis Penguji
pada tanggal 20 Januari 2009
dan dinyatakan memenuhi syarat untuk memperoleh gelar
Sarjana Komputer dalam bidang Ilmu Komputer

Pembimbing I,

Pembimbing II,

Drs. Muh Arif Rahman, M.Kom
NIP. 131 971 481

Dra. Ani Budi Astuti, M.Si
NIP. 131 993 385

Mengetahui,
Ketua Jurusan Matematika
Fakultas MIPA Universitas Brawijaya

Dr. Agus Suryanto, MSc
NIP. 132 126 049





LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Nuraini
NIM : 0410960042
Jurusan : Matematika
Program Studi : Ilmu Komputer
Penulis Skripsi berjudul : *Recommender System* menggunakan
Metode *Analytic Hierarchy Process*
(AHP) (Studi Kasus Pemilihan Program
Studi di Universitas Brawijaya)

Dengan ini menyatakan bahwa :

1. Isi dari skripsi yang saya buat adalah benar-benar karya sendiri dan tidak menjiplak karya orang lain, selain nama-nama yang termaktub di isi dan tertulis di daftar pustaka dalam skripsi ini.
2. Apabila dikemudian hari ternyata skripsi yang saya tulis terbukti hasil jiplakan, maka saya akan bersedia menanggung segala resiko yang akan saya terima.

Demikian pernyataan ini dibuat dengan segala kesadaran.

Malang, 20 Januari 2009
Yang menyatakan,

Nuraini
NIM. 0410960042



**RECOMMENDER SYSTEM MENGGUNAKAN METODE
ANALYTIC HIERARCHY PROCESS (AHP)
(Studi Kasus Pemilihan Program Studi Di Universitas Brawijaya)**

ABSTRAK

Kasus pemilihan jurusan atau program studi kuliah bukanlah urusan yang mudah. Banyak faktor atau kriteria yang harus diperhitungkan dan dipikirkan dengan masak. Memilih secara tergesa-gesa tanpa memperhitungkan segala aspek akan berakibat fatal mulai dari kesadaran yang terlambat bahwa program studi yang diambil tidak sesuai dengan kepribadian, sampai akhirnya pada tahap *drop out* (DO) atau dikeluarkannya seorang mahasiswa atau mahasiswi karena dinyatakan tidak mampu mengikuti pendidikan yang diikutinya. Penelitian ini bertujuan untuk mengimplementasikan metode *Analythic Hierarchy Process* (AHP) dalam perancangan *recommender system* untuk memberikan rekomendasi alternatif terbaik dari suatu permasalahan khususnya studi kasus yang sedang diteliti.

Implementasi metode AHP dalam *recommender system* ini menggunakan skala penilaian 0-9 (skala AHP) dalam pemberian nilai pertimbangan terhadap relatif pentingnya suatu indikator di atas yang lainnya, berkenaan dengan suatu sifat elemen yang sama. Kriteria yang digunakan dalam penerapan implementasi terhadap kasus pemilihan program studi adalah kriteria aspek pribadi, aspek dukungan dan aspek informasi. Kemudian dilakukan pengujian terhadap hirarki analisa permasalahan yang telah dimasukkan ke dalam sistem dengan responden adalah mahasiswa Universitas Brawijaya angkatan 2007 yang masih aktif pada semester genap tahun ajaran 2007/2008.

Pengujian dilakukan dengan mencocokkan hasil yang diberikan oleh sistem dengan alternatif riil (program studi) yang sedang dijalani oleh responden saat dilakukan pengujian. Berdasarkan hasil uji coba diketahui bahwa tingkat kesesuaian antara hasil yang diberikan oleh sistem dengan program studi (alternatif riil) responden sebesar 65% dari jumlah sampel penelitian yang telah ditentukan yaitu sebanyak 144 responden. Sehingga disimpulkan bahwa metode AHP dapat digunakan untuk memberikan rekomendasi alternatif pemilihan program studi di perguruan tinggi.



**RECOMMANDER SYSTEM BY USING
ANALYTIC HIERARCHY PROCESS (AHP)
(Case Studies On Program Choosing In Brawijaya University)**

Abstract

Many problems can be found in subjects or study program choosing. There are many factors and criteria should become further considerations. The wrong decision will cause a fatal effect such as inappropriateness between programs and the student's personality. In the end, the student will not be able to follow the subject and the worst is the students get dropped out (DO). The objective of this study is to implement the analytical hierarchy process in the construction of recommender system to give the best alternative recommendation on a problem especially for the case which have been researched.

The implementation of AHP method used in recommender system use measurement scale 0-9 (AHP scale) in the scoring toward the relative important of an indicator upon the others, related to the characteristics of the same elements. There are some criteria used, such as priority aspects, supportive aspects, and information aspects. Then evaluation is done towards problem's analytical hierarchy which is built in to the system the responden is students of Brawijaya University class of 2007 who are still active in the even semester academic year 2007/2008.

The evaluation is done by adjusted the result given by the system with the real alternative (study program) being taken by the responden in the mean time. Based on the test result, it is noted that the appropriateness between the result given by the system with the alternative real by the responden is 65% from the determined sample. So it is interfered that AHP method can be employed in order to give alternative recommendation of study program choosing in university.



x

KATA PENGANTAR

Puji syukur kehadirat Allah SWT, atas segala limpahan rahmat dan hidayah-Nya kepada penulis sehingga dapat menyelesaikan skripsi dengan judul “*Recommender System Menggunakan Metode Analytic Hierarchy Process (AHP)* (Studi Kasus Pemilihan Program Studi di Universitas Brawijaya)”.

Menyadari bahwa dalam penyusunan skripsi ini, telah banyak pihak yang membantu memberikan bimbingan dan motivasi, maka penulis ingin menyampaikan ucapan terima kasih dan penghargaan setinggi-tingginya kepada:

1. Bapak Drs. Muh. Arif Rahman, M. Kom, selaku dosen pembimbing I yang telah banyak memberi bantuan berupa bimbingan, masukan dan arahan dalam penyelesaian Laporan Akhir ini.
2. Ibu Dra. Ani Budi Astuti, M.Si, selaku dosen pembimbing II yang telah banyak memberi bantuan berupa bimbingan, masukan dan arahan tata cara penulisan Laporan Akhir ini.
3. Bapak Wayan Firdaus Mahmudy, S.Si., M.Kom, selaku Ketua Program Studi Ilmu Komputer, Jurusan Matematika, FMIPA Universitas Brawijaya.
4. Dr. Agus Suryanto, MSc selaku ketua Jurusan Matematika Fakultas Mipa Universitas Brawijaya Malang.
5. Ayah, Ibu, Kakak dan Adik. Terima kasih atas cinta, kasih sayang, doa, dukungan dan semangat yang tiada henti.
6. Semua *crew* Gank herry dan teman-teman ilkomers. Terima kasih atas senyuman, semangat, dukungan, do'a dan hari-hari kita.
7. Pihak lain yang telah membantu terselesaikannya Skripsi ini yang tidak bisa penulis sebutkan satu-persatu.

Semoga penulisan laporan skripsi ini bermanfaat bagi pembaca sekalian. Dengan tidak lupa kodratnya sebagai manusia, penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan, dan mengandung banyak kekurangan, sehingga dengan segala kerendahan hati penulis mengharapkan kritik dan saran yang membangun dari pembaca.

Malang, Januari 2009

Penulis



DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
HALAMAN PENGESAHAN	iii
LEMBAR PERNYATAAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL.....	xix
DAFTAR LAMPIRAN.....	xxi

BAB I PENDAHULUAN

1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Batasan masalah	2
1.4. Tujuan Penelitian	3
1.5. Manfaat Penelitian	3

BAB II TINJAUAN PUSTAKA

2.1. Pengertian Sistem	5
2.2. Sistem Rekomendasi	6
2.3. Konsep Basis Data	6
2.4. Pemodelan Sistem	7
2.5. Antar Muka Pengguna (<i>User Interface</i>)	9
2.6. <i>Analythic Hierarchy Process</i> (AHP)	9
2.6.1. Pengertian AHP	9
2.6.2. Penyusunan Hirarki	12
2.6.3. Penyusunan matriks perbandingan berpasangan	13
2.6.4. Sintesis	16
2.6.4.1. Nilai eigen dan vektor eigen	16
2.6.4.2. Pengambilan keputusan	17
2.6.4.3. Konsistensi matriks perbandingan berpasangan	18
2.6.5. Penarikan sampel acak sederhana	20

BAB III METODE PENELITIAN

3.1. Sumber Data	21
3.2. Deskripsi Sistem	23
3.3. Metode	23
3.3.1. Perancangan Database	24
3.3.2. Perancangan Sistem	25
3.3.3. Perancangan Uji Coba	32
3.3.4. Pengujian Dan Analisa Hasil	33
3.3.5. Perancangan Antar Muka	33

BAB IV IMPLEMENTASI DAN PEMBAHASAN

4.1. Lingkungan Implementasi	35
4.2. Implementasi Program	35
4.2.1. Implementasi Pembentukan Hirarki	35
4.2.2. Implementasi Penyusunan Matriks Perbandingan Berpasangan	37
4.2.3. Implementasi Perhitungan Nilai Eigen	39
4.2.4. Implementasi Pengecekan Nilai Konsistensi Matriks	39
4.2.4.1. Perhitungan Nilai Lambda, CI, CR	40
4.2.4.2. Pengecekan Konsistensi	41
4.2.5. Implementasi Perhitungan Prioritas Lokal	44
4.2.6. Implementasi Perhitungan Nilai Global	45
4.2.7. Implementasi Penyimpanan Hasil (<i>Save Report</i>)....	47
4.3. Implementasi Antarmuka	49
4.3.1. Menu Penyusunan Hirarki	49
4.3.2. Menu Sintesis	50
4.3.3. Menu <i>Help</i>	54
4.4. Pengujian Sistem	54
4.4.1. Studi Kasus	54
4.4.2. Analisa Hasil	57
4.4.2.1. Perhitungan Tingkat Kesesuaian Hasil Pengujian Sistem	57
4.4.2.2. Analisa Prioritas Global	58
4.4.2.3. Analisa Prioritas Lokal Pada Level Kriteria (Level 2)	60
4.4.2.4. Analisa Hasil Prioritas Lokal Pada Level Sub Kriteria (Level 3).....	61

BAB V KESIMPULAN DAN SARAN

5.1. Kesimpulan 65
5.2. Saran 65

DAFTAR PUSTAKA 67

LAMPIRAN 71





DAFTAR GAMBAR

	Halaman
Gambar 2.1. Hirarki analisa permasalahan.....	13
Gambar 3.1. Struktur permasalahan (hirarki analisa permasalahan) pada contoh kasus pemilihan program studi.....	22
Gambar 3.2. Tahapan – tahapan penelitian.....	24
Gambar 3.3. Diagram alir mekanisme kerja sistem.....	27
Gambar 3.4. Diagram alir proses input data <i>rule</i>	28
Gambar 3.5. Diagram alir proses pembentukan hirarki.....	29
Gambar 3.6. Diagram alir penyusunan matriks perbandingan berpasangan.....	30
Gambar 3.7. Diagram alir proses perhitungan prioritas global....	31
Gambar 3.8. Menu utama program.....	34
Gambar 4.1. <i>Procedure</i> pembentukan hirarki.....	36
Gambar 4.2. <i>Procedure</i> pemilihan level dan penyeleksian indikator asal pada level terseleksi	37
Gambar 4.3. <i>Procedure</i> penyusunan matriks berpasangan.....	38
Gambar 4.4. <i>Procedure</i> perhitungan nilai eigen	39
Gambar 4.5. <i>Procedure</i> perhitungan nilai lambda, CI, CR.....	40
Gambar 4.6. <i>Procedure</i> pengecekan konsistensi matriks	42
Gambar 4.7. Fungsi untuk mengecek apakah masih ada entri matriks yang masih kosong pada segitiga atas matriks	42
Gambar 4.8. <i>Procedure</i> untuk inisialisasi matriks, menentukan nilai awal, nilai konsisten dan ordo matriks.....	42
Gambar 4.9. <i>Procedure</i> untuk mengeset data matriks (entri matriks)	43
Gambar 4.10. <i>Procedure</i> untuk mengisikan data pada matriks ..	43
Gambar 4.11. <i>Procedure</i> untuk memperbaiki matriks yang tidak konsisten	44
Gambar 4.12. <i>Procedure</i> perhitungan prioritas lokal	45
Gambar 4.13. <i>Procedure</i> untuk mendapatkan level maksimum ..	45
Gambar 4.14. <i>Procedure</i> perhitungan nilai global.....	46
Gambar 4.15. Fungsi untuk mengecek apakah masih ada nilai eigen dari indikator yang masih kosong pada database.....	47

Gambar 4.16. <i>Listing</i> program untuk melakukan export ke ms.excell.....	48
Gambar 4.17. <i>Listing</i> program untuk menyimpan data hasil evaluasi ke dalam bentuk excel	49
Gambar 4.18. Form menu deskripsi masalah (input data <i>rule</i> dan penyusunan hirarki)	50
Gambar 4.19. Form menu sintesis pada proses pemilihan level dan indikator asal.....	51
Gambar 4.20. Form menu sintesis pada proses pengecekan konsistensi matriks	52
Gambar 4.21. Form menu sintesis pada proses pengecekan prioritas global.....	53
Gambar 4.22. Form menu <i>help</i>	54
Gambar 4.23. Data nilai prioritas global hasil uji coba sistem dengan 2 alternatif program studi	59
Gambar 4.24. Data nilai prioritas global hasil uji coba sistem dengan 3 alternatif program studi	61
Gambar 4.25. Rata – rata prioritas lokal indikator pada level kriteria.....	62
Gambar 4.26. Nilai rata – rata indikator pada sub kriteria aspek pribadi.....	63
Gambar 4.27. Nilai rata – rata indikator pada sub kriteria aspek dukungan	63
Gambar 4.28. Nilai rata – rata indikator pada sub kriteria aspek informasi.....	64



DAFTAR TABEL

	Halaman
Tabel 2.1. Simbol dasar pada <i>flowchart</i>	8
Tabel 2.2. Simbol untuk program <i>flowchart</i>	8
Tabel 2.3. Contoh tabel matriks perbandingan berpasangan	13
Tabel 2.4. Contoh tabel matriks perbandingan berpasangan ($a_{ij} = w_i / w_j$)	14
Tabel 2.5. Skala banding berpasangan.....	15
Tabel 2.6. Tabel prioritas menyeluruh.....	18
Tabel 2.7. <i>Random consistency index</i> (RI)	19
Tabel 3.1. Tabel aturan (<i>rule</i>)	24
Tabel 3.2. Tabel tipe inputan sistem	32
Tabel 3.3. Tabel rancangan uji coba evaluasi nilai eigen	32
Tabel 3.4. Tabel rancangan uji coba evaluasi prioritas global....	32
Tabel 4.1. Tabel Input Data Rule Hirarki Permasalahan dengan 3 alternatif program studi (prodi).....	55
Tabel 4.2. Tabel Input Data Rule Hirarki Permasalahan dengan 2 alternatif program studi (prodi).....	56
Tabel 4.3. Data hasil pengujian sistem	58
Tabel 4.4. Data nilai prioritas global hasil uji coba sistem dengan 2 alternatif program studi.	59
Tabel 4.5. Data nilai prioritas global hasil uji coba sistem dengan 3 alternatif program studi.	60
Tabel 4.6. Nilai rata-rata indikator pada level 2 (kriteria)	62
Tabel 4.7. Nilai rata – rata prioritas lokal indikator pada level 3 subkriteria aspek pribadi.	62
Tabel 4.8. Nilai rata – rata prioritas lokal indikator pada level 3 subkriteria aspek dukungan.....	63
Tabel 4.9. Nilai rata – rata prioritas lokal indikator pada level 3 subkriteria aspek informasi.	64



xx

DAFTAR LAMPIRAN

	Halaman
Lampiran 1. Data Program Studi yang Diteliti	71
Lampiran 2. Data Hasil Pengujian Sistem Pada Survei Pendahuluan.....	73
Lampiran 3. Data Keseluruhan Hasil Pengujian Sistem	77





BAB I PENDAHULUAN

1.1. Latar Belakang

Para pengambil keputusan hampir selalu membuat keputusan, bahkan setiap detik dari hidupnya. Ketika membuat keputusan, ada suatu proses yang terjadi dalam otak manusia yang akan menentukan kualitas keputusan yang dibuat. Ketika keputusan yang akan diambil bersifat kompleks dengan resiko yang besar, pengambil keputusan sering memerlukan suatu alat bantu dalam bentuk analisis yang bersifat ilmiah, logis, terstruktur dan konsisten. Salah satu alat analisa tersebut adalah berupa model pembuatan keputusan yang memungkinkan mereka membuat keputusan untuk masalah yang bersifat kompleks. Metode AHP merupakan salah satu model pengambilan keputusan yang sering digunakan (Susila, Wayan dan Munadi, Ernawati, 2007).

Pada dasarnya, metode AHP ini memecah-mecah suatu situasi kompleks dan tak terstruktur, ke dalam bagian-bagian komponennya; menata bagian atau variabel ini ke dalam suatu susunan hierarki; memberi nilai numerik pada pertimbangan subyektif tentang relatif pentingnya setiap variabel; dan mensistesis berbagai pertimbangan ini untuk menetapkan variabel mana yang memiliki prioritas paling tinggi dan bertindak untuk mempengaruhi hasil pada situasi tersebut (Saaty, T. L, 1993).

Penggunaan AHP dalam sebuah alat bantu pengambilan keputusan dengan multi kriteria sangat mudah dipahami dan dimengerti, sehingga penerapannya telah meluas sebagai model alternatif untuk menyelesaikan bermacam – macam masalah (Kurniawan, F, 2004).

Untuk mempermudah evaluasi perhitungan prioritas tertinggi pemilihan alternatif suatu masalah dengan menggunakan AHP, dimanfaatkan kemajuan teknologi dalam bidang pengolahan data elektronik dengan membangun sebuah perangkat lunak berupa *recommender system* (sistem rekomendasi). *Recommender system* ini bertujuan untuk memberikan informasi yang sesuai dengan keinginan atau kebutuhan seseorang berdasarkan data yang sudah ada. Sehingga *recommender system* ini diharapkan bisa diterapkan untuk memberikan rekomendasi alternatif suatu permasalahan.

Dalam penelitian kali ini yang digunakan sebagai contoh kasus adalah tentang pemilihan program studi pada perguruan tinggi. Pada

umumnya siswa yang telah lulus dari SMA, SMEA, SMK dan jenjang sederajat lainnya akan memutuskan untuk melanjutkan studi ke Perguruan Tinggi baik Perguruan Tinggi Negeri (PTN) maupun Perguruan Tinggi Swasta (PTS). Memilih jurusan atau program studi kuliah bukan urusan yang mudah. Banyak faktor atau kriteria yang harus diperhitungkan dan dipikirkan dengan masak. Memilih secara tergesa-gesa tanpa memperhitungkan segala aspek akan berakibat fatal mulai dari kesadaran yang terlambat bahwa program studi yang diambil tidak sesuai dengan kepribadian, sampai akhirnya pada tahap *drop out* (DO) atau dikeluarkannya seorang mahasiswa atau mahasiswi karena dinyatakan tidak mampu mengikuti pendidikan yang diikutinya. Maka dari itu pemilihan program studi sedini mungkin harus mulai dipertimbangkan.

Berdasarkan uraian latar belakang di atas maka judul yang diambil penulis dalam penelitian ini adalah ***“Recommender System Menggunakan Metode AHP (Analytic Hierarchy Process) Studi Kasus Pemilihan Program Studi di Universitas Brawijaya Malang”***.

1.2. Rumusan masalah

Berdasarkan uraian dari latar belakang di atas maka dapat disimpulkan suatu rumusan masalah yaitu bagaimana mengimplementasikan metode AHP dalam perancangan *recommender system* dan menerapkannya dalam contoh kasus.

1.3. Batasan Masalah

Ruang lingkup dalam penelitian ini dibatasi pada:

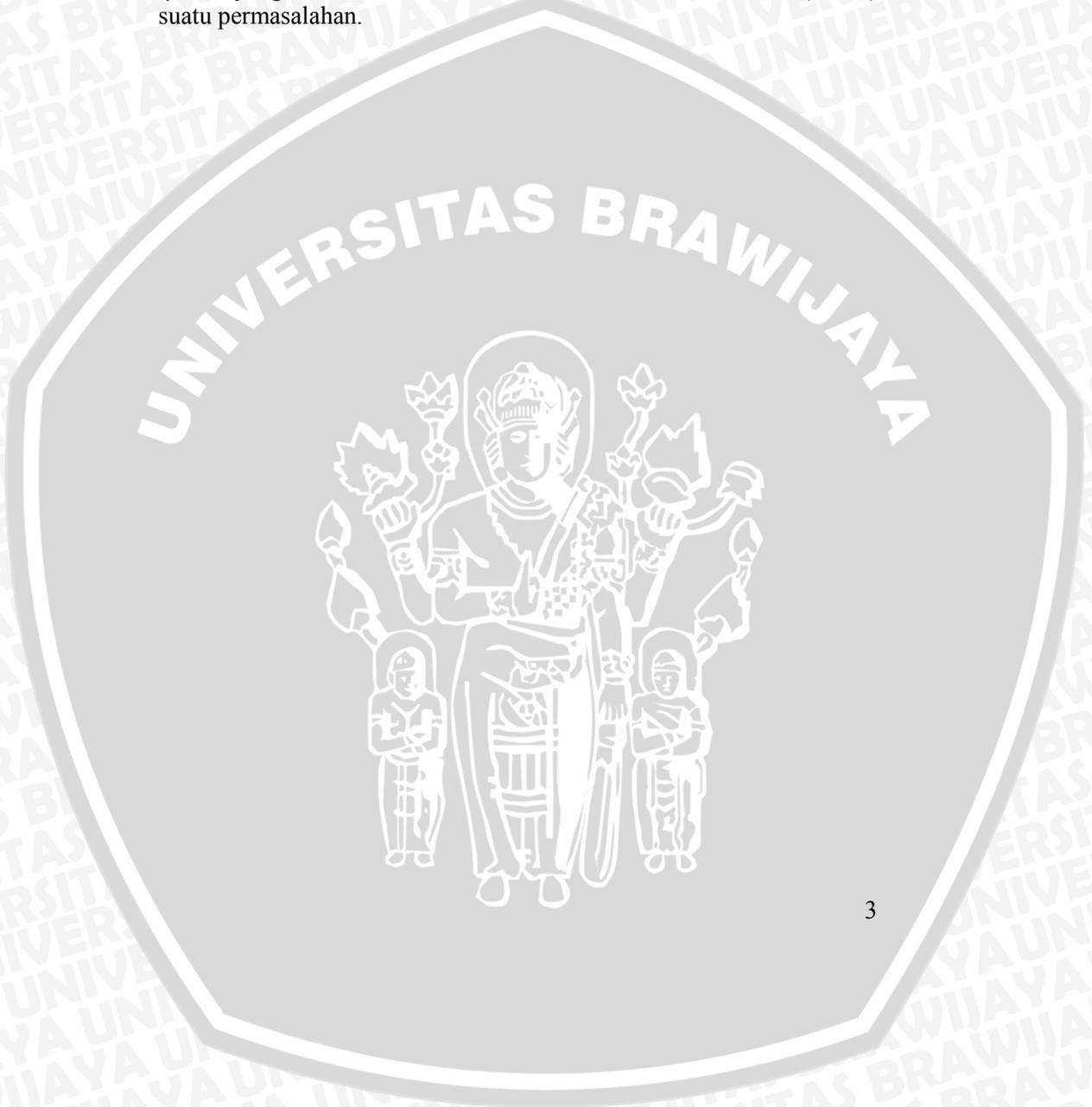
1. Penggunaan skala penilaian 0-9 (skala AHP) dalam pengisian matriks perbandingan berpasangan untuk menggambarkan relatif pentingnya suatu indikator terhadap indikator lainnya, berkenaan dengan suatu sifat elemen yang sama.
2. Contoh kasus yang digunakan dalam penelitian dan pengujian sistem adalah kasus pemilihan program studi di perguruan tinggi.
3. Pengujian sistem dilakukan dengan mencocokkan hasil rekomendasi yang diberikan oleh sistem terhadap prodi yang dijalani oleh responden saat melakukan pengujian.
4. Responden penelitian adalah mahasiswa S1 Universitas Brawijaya angkatan 2007 yang masih aktif pada semester genap tahun ajaran 2007/2008.

1.4. Tujuan

Tujuan yang ingin dicapai dalam penelitian ini adalah untuk mengimplementasikan metode *analytic hierarchy process (AHP)* dalam perancangan *recommender system* dan menerapkannya pada contoh kasus.

1.5. Manfaat

Manfaat dari penelitian ini adalah terciptanya sebuah *recommender system* yang bisa memberikan rekomendasi alternatif terbaik (solusi) suatu permasalahan.





BAB II TINJAUAN PUSTAKA

2.1. Pengertian Sistem

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu. (Jogiyanto, HM, 1988).

Suatu sistem harus mempunyai karakteristik atau sifat yang tertentu, diantaranya adalah sebagai berikut :

1. Komponen (*Components*), merupakan suatu sistem yang terdiri dari sejumlah komponen yang saling berinteraksi untuk membentuk suatu kesatuan komponen juga dapat berupa suatu sub sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.
2. Batas sistem (*Boundary*), berfungsi sebagai pembatas antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem tersebut juga menunjukkan ruang lingkup atau scope dari sistem tersebut.
3. Lingkungan luar sistem (*Environment*), adalah lingkungan luar sistem yang dapat mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat pula bersifat merugikan sistem tersebut. Lingkungan luar sistem yang menguntungkan merupakan energi yang harus tetap dijaga dan dipelihara. Sedangkan lingkungan luar sistem yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsunagn hidup dari sistem.
4. Penghubung (*Interface*), merupakan media penghubung antara sub sistem dengan sub sistem yang lain.
5. Masukan (*Input*), merupakan semua elemen yang masuk ke sistem.
6. Keluaran (*output*), merupakan suatu hasil dari energi yang diolah dan diklasifikasikan menjadi pengeluaran yang berguna.
7. Pengolahan (*Processor*), merupakan tempat yang digunakan untuk mengolah input menjadi output.
8. Sasaran atau tujuan (*Goal*), suatu sistem pasti mempunyai tujuan (*goal*) atau sasaran (*objective*) kalau sistem tersebut tidak mempunyai sasaran maka sistem tersebut tidak akan berguna. Suatu sistem dikatakan berhasil, apabila berhasil mengenai sasaran atau tujuannya. (Jogiyanto, HM, 2001).

Sistem yang dikembangkan memerlukan orang-orang yang terdidik. Manusia merupakan faktor yang utama yang menentukan berhasil atau tidaknya suatu sistem yang baik dalam proses pengembangan, penerapan maupun dalam proses operasinya.

2.2. Sistem Rekomendasi

Recommender system (sistem rekomendasi) adalah teknologi yang digunakan untuk memprediksi apakah seseorang akan tertarik dengan suatu barang atau dapat juga digunakan untuk mengidentifikasi barang mana yang sesuai dengan kebutuhan seseorang (Han dan Larypis, 2005). Sedangkan menurut Haruechaiyasak (2004), sistem rekomendasi adalah program atau agen yang secara otomatis membuat sebuah daftar yang berisi sekumpulan informasi yang sesuai dengan kebutuhan seseorang.

Dapat disimpulkan bahwa tujuan dari sistem rekomendasi adalah memberikan informasi yang sesuai dengan keinginan atau kebutuhan. (Purnawinandar, 2007).

2.3. Konsep Sistem Basis Data

Sistem basis data didefinisikan sebagai sekumpulan subsistem yang terdiri atas basis data dengan para pemakai yang menggunakan basis data secara bersama-sama, personal-personal yang merancang dan mengelola basis data, teknik-teknik untuk merancang dan mengelola basis data, serta sistem komputer untuk mendukungnya (Sutanta, 2004).

Database adalah kumpulan data (arsip) yang saling berhubungan yang disimpan secara bersama dalam media penyimpanan elektronik sedemikian rupa tanpa pengulangan (*redudancy*) yang tidak perlu untuk memenuhi berbagai kebutuhan (Fathansyah, 1999).

Sebuah sistem basis data dapat memiliki beberapa basis data. Setiap basis data dapat berisi/memiliki sejumlah objek basis data (seperti file/tabel, indeks, dan lain-lain). Disamping berisi atau menyimpan data, setiap basis data juga mengandung atau menyimpan definisi struktur baik untuk basis data maupun objek-objeknya secara detail (Fathansyah, 1999).

Perancangan basis data seringkali diasosiasikan dengan pembuatan model *Entity Relationship* (model E-R). E-R Model adalah sebuah bagan yang merupakan gambaran data dari seluruh proyek. E-R Model berisi komponen-komponen himpunan entitas dan himpunan relasi

yang masing-masing dilengkapi dengan atribut-atribut yang mempresentasikan seluruh fakta. Adapun komponen pembentuk model E-R adalah sebagai berikut :

- 1 Entitas (*entity*)
Entitas adalah suatu yang dapat dibedakan dalam dunia nyata, dimana informasi yang berkaitan dengannya dikumpulkan.
- 2 Relasi
Relasi adalah hubungan yang terjadi antara satu atau lebih entitas. Relasi tidak mempunyai keberadaan fisik kecuali yang mewarisi dari hubungan antar entitas tersebut.
- 3 Atribut (*field*)
Atribut adalah karakteristik dari entitas atau relasi, yang menyediakan penjelasan detail tentang entitas atau relasi tersebut.
- 4 Kunci (*Key*)
Key adalah identitas pembeda antara entitas satu dengan yang lain. Dalam hubungan antar tabel dikenal dua kunci data penghubung yaitu :
 - a. *Primary Key* (Kunci Utama)
Suatu atribut/*field* atau satu set atribut yang mengidentifikasi secara unik suatu kejadian yang spesifik pada entitas.
 - b. *Foreign Key* (Kunci Tamu)
Suatu atribut/*field* atau satu set atribut yang melengkapi suatu hubungan yang menunjukkan ke entitas induknya. Kunci tamu berada pada entitas anak.

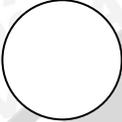
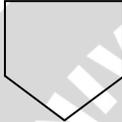
2.4. Pemodelan Sistem

Dalam pemodelan sistem terdapat sejumlah cara untuk menggambarkan sistem melalui diagram salah satu contohnya adalah menggunakan *flowchart*.

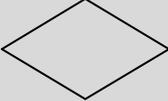
Flowchart adalah bagan-bagan yang mempunyai arus (urutan intruksi) yang menggambarkan langkah-langkah penyelesaian suatu masalah (<http://lecturer.eepis-its.edu>). Urutan-urutan intruksi tersebut diwakili simbol-simbol tertentu.

Program *flowchart* adalah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program, yang menyatakan arah alur program tersebut (<http://faculty.petra.ac.id>). Simbol dasar dan simbol program pada *flowchart* dijelaskan dalam Tabel 2.2. dan Tabel 2.3 (<http://lecturer.eepis-its.edu>).

Tabel 2.2. Simbol Dasar Pada *Flowchart*.

Simbol	Keterangan
	<p>PROSES</p> <ul style="list-style-type: none"> - proses/pengolahan - untuk program flowchart berarti <ul style="list-style-type: none"> - perhitungan - mengubah harga
	<p>INPUT-OUTPUT OPERATION</p> <ul style="list-style-type: none"> - operasi input-output (membaca dan menulis data)
	<p>CONNECTOR</p> <ul style="list-style-type: none"> - penghubung di dalam satu halaman
	<p>OFF-PAGE CONNECTOR</p> <ul style="list-style-type: none"> - penghubung antar halaman
	<p>ARROW</p> <ul style="list-style-type: none"> - tanda panah untuk menunjukkan arah <i>flowchart</i> dibaca dari kiri ke kanan atau dari atas ke bawah, selain itu harus diberi panah

Tabel 2.3. Simbol Untuk Program *Flowchart*.

Simbol	Keterangan
	<p>DECISION</p> <ul style="list-style-type: none"> - pertanyaan - penentuan keputusan (ya atau tidak)

	<p>PREPARATION</p> <p>- memberi harga awal</p>
	<p>TERMINAL</p> <p>- menyatakan permulaan dan akhir dari Proses</p>

2.5. Antarmuka Pengguna (*User Interface*)

User interface merupakan bagian perangkat lunak yang menyediakan sarana bagi pengguna agar bisa berkomunikasi dengan sistem. *User interface* akan mengajukan pertanyaan dan menyediakan sarana komunikasi jawaban atau solusi bila masalahnya sudah diketemukan. Setiap komunikasi selama proses pemecahan masalah dikendalikan oleh *user interface* (Setyawan, A, 1993). Menurut McLeod (1995), pada bagian ini terjadi dialog antara program dan pemakai, yang memungkinkan program menerima instruksi dan informasi (input) dari pemakai dan memberikan informasi (output) kepada pemakai.

2.6. *Analytical Hierarchy Process* (AHP)

2.6.1. Pengertian AHP

Sumber kerumitan masalah pengambilan keputusan bukan hanya pada tidak lengkapnya informasi. Penyebab lainnya dapat berupa beragamnya kriteria (parameter) yang ada. Jika sumber kerumitan berasal dari beragamnya kriteria (parameter), maka *Analytical Hierarchy Process* (AHP) merupakan teknik untuk membantu menyelesaikan masalah ini. AHP diperkenalkan oleh Thomas L. Saaty pada periode 1971-1975. Metode AHP mampu mengakomodir berbagai kriteria (parameter) yang saling mempengaruhi (kriteria majemuk)

Kriteria majemuk adalah suatu metode pengambilan keputusan terhadap suatu masalah atau tujuan berdasarkan beberapa kriteria (parameter) yang berpengaruh terhadap masalah tersebut. Mengingat suatu tujuan yang bersifat umum, maka perlu dijabarkan dalam beberapa kriteria yang masing-masing mempunyai sub parameter dan

sub-sub parameter sehingga diperoleh suatu struktur hirarki model evaluasi untuk tujuan tertentu.

Pada dasarnya AHP, adalah suatu teori umum tentang pengukuran. Metode yang digunakan untuk menemukan skala rasio baik dari perbandingan pasangan yang diskrit atau kontinyu. Perbandingan ini diambil dari ukuran aktual atau dari suatu skala dasar yang mencerminkan kekuatan perasaan atau intuisi seorang pengambil keputusan.

Menurut Saaty (1993), dalam menyelesaikan persoalan dengan AHP ada prinsip-prinsip yang harus dipahami diantaranya adalah sebagai berikut :

1. Penyusunan Hirarki

Setelah persoalan didefinisikan, maka perlu dilakukan *decomposition* yaitu memecah masalah atau persoalan yang utuh menjadi unsur-unsurnya. Jika ingin mendapatkan hasil yang akurat, pemecahan juga dilakukan terhadap unsur - unsur sampai tidak mungkin dilakukan pemecahan lebih lanjut, sehingga didapatkan beberapa tingkatan dari persoalan tadi. Karena alasan ini, maka proses analisis ini dinamakan hirarki.

2. Menentukan Prioritas

Prinsip ini berarti membuat penilaian tentang kepentingan relatif dua elemen pada suatu tingkat tertentu yang dalam kaitannya dengan tingkat di atasnya. Penilaian ini merupakan inti dari AHP, karena ia akan berpengaruh terhadap prioritas elemen-elemen. Hasil dari penilaian akan tampak lebih enak bila disajikan dalam bentuk matriks yang dinamakan *matriks pairwise comparison* (matriks perbandingan berpasangan).

Agar diperoleh skala bermanfaat ketika membandingkan dua elemen, seseorang yang akan memberikan jawaban perlu pengertian menyeluruh tentang elemen-elemen yang akan dibandingkan dan relevansinya terhadap kriteria atau tujuan yang dipelajari.

AHP melakukan perbandingan berpasangan antar dua elemen pada tingkat yang sama berdasarkan satu elemen pada tingkat yang lebih tinggi. Dalam hal ini elemen pada tingkat yang lebih tinggi berfungsi sebagai kriteria. Kedua elemen tersebut dibandingkan dengan menimbang tingkat preferensi elemen yang satu terhadap yang lain menurut suatu kriteria tertentu. Hasil dari perbandingan ini adalah

vektor prioritas atau relatif pentingnya suatu elemen berdasarkan kriteria tertentu. Perbandingan berpasangan dilakukan untuk setiap elemen dalam setiap tingkat. Selanjutnya adalah memberi bobot untuk setiap elemen dengan prioritasnya, kemudian menggabungkan prioritas yang telah diperoleh pada setiap hirarki untuk menghasilkan prioritas menyeluruh.

3. Konsistensi Logis

Konsistensi logis merupakan prinsip rasional AHP. Konsistensi berarti dua hal, yaitu:

1. Pemikiran atau objek yang serupa dikelompokkan menurut keseragaman dan relevansinya.

Contoh: Anggur dan kelereng.

Kriteria: bulat (dapat dikelompokkan).

Kriteria : rasa (tidak dapat dikelompokkan).

2. Relasi antar objek yang didasarkan pada kriteria tertentu, saling membenarkan secara logis.

Contoh: kriteria : kemanisan

Jika madu 2x lebih manis daripada gula pasir.

Jika gula pasir 3x lebih manis daripada molasa.

Maka madu harus 6x lebih manis daripada molasa.

Saaty (1994), menyebutkan bahwa AHP mempunyai beberapa aksioma yang harus dipenuhi. Aksioma-aksioma tersebut antara lain sebagai berikut:

1. *Reciprocal* (sifat berkebalikan)

Perbandingan dua elemen memiliki sifat *reciprocal* jika memenuhi persamaan :

$$a_{ij} = 1/a_{ji} \quad (2.1)$$

maksud persamaan (2.1) adalah elemen ke-*i* lebih disukai daripada elemen ke-*j* pada skala $1/x$, atau elemen ke-*j* lebih disukai daripada elemen ke-*i* pada skala x .

2. *Homogeneity* (keseragaman)

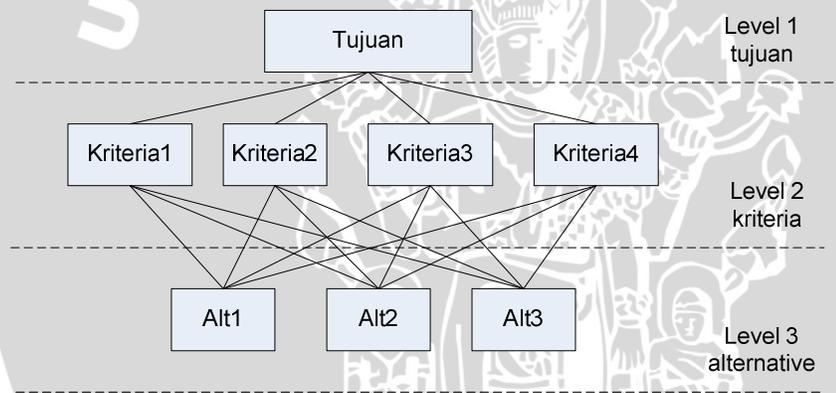
Keseragaman sangat diperlukan untuk membandingkan dua hal yang serupa. Otak manusia akan cenderung membuat kesalahan besar dalam membandingkan elemen dengan tingkat perbedaan yang tinggi (tidak homogen). Jika kondisi tidak homogen ini

terjadi maka elemen – elemen hirarki harus ditempatkan terpisah pada kelompok yang memiliki ukuran perbandingan yang seragam. Tujuan pengelompokan yang homogen pada AHP adalah untuk memudahkan perbandingan, bila dalam pengelompokan tersebut memiliki ciri umum yang dibandingkan maka akan lebih mudah mengukur objek dengan ukuran yang sebanding.

2.6.2. Penyusunan Hirarki

Menurut Saaty (1994), hirarki adalah Gambaran dari permasalahan yang kompleks dalam struktur banyak tingkat dimana tingkat paling atas adalah tujuan dan diikuti tingkat kriteria, sub kriteria dan seterusnya ke bawah sampai pada tingkat yang paling bawah adalah tingkat alternatif. Hirarki menggambarkan secara grafis saling ketergantungan elemen – elemen yang relevan, memperlihatkan hubungan antar elemen yang homogen dan hubungan dengan sistem sehingga menjadi satu kesatuan yang utuh.

Dalam membuat hirarki tidak ada batasan untuk jumlah tingkat. Jika elemen-elemen satu tingkat sulit dibandingkan, maka satu tingkat di bawahnya yang lebih sederhana dengan perbedaan yang halus harus diciptakan. Hirarki harus bersifat luwes, selalu dapat diubah guna menampung adanya kriteria baru yang muncul. Contoh sistem hirarki permasalahan 3 (tiga) level dapat dilihat pada Gambar 2.1.



Gambar 2.1. Hirarki analisa permasalahan

2.6.3. Matriks Perbandingan Berpasangan

Untuk menentukan susunan prioritas elemen, langkah awal adalah menyusun perbandingan berpasangan, yaitu membandingkan secara berpasangan seluruh elemen pada tingkat yang sama berdasarkan elemen tertentu yang berada satu tingkat di atasnya dalam hirarki. Perbandingan tersebut kemudian ditransformasikan ke dalam bentuk matriks yang digunakan dalam analisis numerik. Menurut Anton (1995), sebuah matriks adalah susunan segiempat siku-siku dari bilangan-bilangan dan bilangan tersebut dinamakan entri dalam matriks.

Menurut Saaty (1993), pada perbandingan berpasangan bentuk matriks merupakan bentuk yang paling diminati, karena matriks merupakan alat yang sederhana dan biasa digunakan, memberi kerangka untuk menguji konsistensi, memperoleh informasi tambahan dengan jalan membuat segala perbandingan yang mungkin dan menganalisis kepekaan prioritas menyeluruh terhadap perubahan dalam perbandingan.

Elemen kolom sebelah kiri, selalu dibandingkan dengan elemen baris puncak, dan nilainya diberikan kepada kolom sebelah kiri. Jika elemen dalam kolom sebelah kiri dianggap kurang menguntungkan, penilaian merupakan suatu pecahan. Nilai kebalikan diberikan kepada elemen baris ketika tampil sebagai elemen kolom dan elemen kolom tampil sebagai elemen baris, sesuai persamaan (2.1). Dalam matriks ini terdapat perbandingan dengan elemen itu sendiri pada diagonal utama dan bernilai 1.

Misalkan kriteria C memiliki beberapa elemen di bawahnya, yaitu: A_1, A_2, \dots, A_n . Tabel matriks perbandingan berpasangan berdasarkan kriteria C dapat dilihat pada Tabel 2.4.

Tabel 2.4. Contoh Tabel Matriks Perbandingan Berpasangan (a_{ij})

C	A_1	A_2	A_n
A_1	1	a_{12}	a_{1n}
A_2	a_{21}	1	a_{2n}
.....
A_n	a_{n1}	a_{n2}	1

Jika w_i dan w_j berturut-turut merupakan bobot atau prioritas elemen ke $-i$ dan ke $-j$, maka pada matriks perbandingan berpasangan yang

konsisten $a_{ij} = w_i/w_j$, dimana $\sum_{i=1}^n w_i = 1$ dan $\sum_{j=1}^n w_j = 1$. Tabel matriks perbandingan berpasangan yang konsisten dapat dilihat pada Tabel 2.5.

Tabel 2.5. Contoh Tabel Matriks Perbandingan Berpasangan ($a_{ij} = w_i / w_j$).

C	A_1	A_2	A_n
A_1	1	w_1 / w_2	w_1 / w_n
A_2	w_2 / w_1	1	w_2 / w_n
.....
A_n	w_n / w_1	w_n / w_2	1

Jika ada n elemen yang akan dibandingkan maka terdapat $n(n-1)/2$ perbandingan antar dua elemen yang harus dilakukan. Perbandingan ini sebagai entri matriks segitiga di atas diagonal utama, sedangkan entri matriks segitiga di bawah diagonal utama merupakan kebalikan dari entri matriks segitiga di atas diagonal utama yang bersesuaian. Pedoman untuk penelitian dalam perbandingan berpasangan dapat dilihat pada Tabel 2.6.



Tabel 2.6. Skala Banding Berpasangan

Tingkat Kepentingan	Definisi	Keterangan
1	Sama penting	Dua elemen menyumbangny sama besar pada sifat itu.
3	Moderat lebih penting	Pengalaman dan pertimbangan sedikit menyokong satu elemen atas yang lainnya.
5	Lebih penting	Pengalaman dan pertimbangan dengan kuat menyokong satu elemen atas yang lainnya.
7	Sangat lebih penting	Satu elemen lebih disukai dan secara praktis dominasinya sangat nyata dibandingkan dengan pasangannya.
9	Mutlak lebih penting	Satu elemen mutlak lebih disukai dibandingkan pasangannya pada tingkat keyakinan tertinggi.
2,4,6,8		Diberikan bila terdapat penilaian antara dua penilaian terdekat.
Kebalikan	$a_{ij} = 1/a_{ji}$	Jika untuk aktifitas ke-i mendapatkan satu angka bila dibandingkan dengan aktifitas ke-j, maka j memiliki nilai kebalikannya bila dibandingkan dengan i.

Sumber : Saaty (1993).

2.6.4. Sintesis

2.6.4.1. Nilai Eigen Dan Vektor Eigen

$A = (a_{ij})$ merupakan suatu matriks A yang elemen-elemennya a_{ij} . Dimana i menyatakan baris ke $-i$ dan j menyatakan kolom ke- j dari A , dimana $i, j = 1, 2, \dots, n$. Untuk mencari nilai eigen dari A menurut Anton (1995) adalah sebagai berikut:

$$A \mathbf{x} = \lambda \mathbf{x}, \mathbf{x} \neq \mathbf{0} \quad (2.2)$$

$$A \mathbf{x} = \lambda \mathbf{I} \mathbf{x}, \mathbf{x} \neq \mathbf{0} \\ (A - \lambda \mathbf{I}) \mathbf{x} = \mathbf{0}, \mathbf{x} \neq \mathbf{0} \quad (2.3)$$

persamaan (2.3) akan ada penyelesaian jika dan hanya jika:

$$|A - \lambda \mathbf{I}| = 0 \quad (2.4)$$

persamaan (2.4) dinamakan persamaan karakteristik untuk A dan memiliki n akar. Akar – akar persamaan karakteristik yang dinyatakan dalam $\lambda_i, i = 1, \dots, n$ disebut nilai eigen dari A . $\mathbf{x} \neq \mathbf{0}$ pada persamaan (2.2) merupakan vektor eigen dari A .

Pada AHP $\text{rank}(A) = 1$, hal ini dikarenakan setiap kolom A yang konsisten adalah kelipatan kolom pertama. Akibat elemen – elemen diagonal utama bernilai 1, maka $\text{trace}[A] = n = \sum_{i=1}^n \lambda_i$. Dari sifat –

sifat tersebut maka dapat disimpulkan bahwa nilai eigen terbesar dari A sama dengan n , sedangkan $n - 1$ nilai eigen selainnya bernilai nol. Dalam AHP nilai eigen terbesar dinyatakan dengan λ_{maks} . Maka diperoleh persamaan:

$$A \mathbf{w} = \lambda_{maks} \mathbf{w}$$

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \begin{pmatrix} w_1 \\ w_2 \\ \dots \\ w_n \end{pmatrix} = \lambda_{maks} \begin{pmatrix} w_1 \\ w_2 \\ \dots \\ w_n \end{pmatrix}$$

$$\begin{pmatrix} a_{11}w_1 + a_{12}w_2 + \dots + a_{1n}w_n \\ a_{21}w_1 + a_{22}w_2 + \dots + a_{2n}w_n \\ \dots \\ a_{n1}w_1 + a_{n2}w_2 + \dots + a_{nn}w_n \end{pmatrix} = \begin{pmatrix} \lambda_{maks} w_1 \\ \lambda_{maks} w_2 \\ \dots \\ \lambda_{maks} w_n \end{pmatrix}$$

$$\begin{pmatrix} \sum_{j=1}^n a_{1j} w_j \\ \sum_{j=1}^n a_{2j} w_j \\ \dots \\ \sum_{j=1}^n a_{nj} w_j \end{pmatrix} = \begin{pmatrix} \lambda_{maks} w_1 \\ \lambda_{maks} w_2 \\ \dots \\ \lambda_{maks} w_n \end{pmatrix}$$

pada matriks berordo $n \times n$ untuk semua i diperoleh nilai λ_{maks} sebagai berikut :

$$\sum_{i=1}^n \lambda_{maks} = \sum_{i=1}^n \sum_{j=1}^n a_{ij} \frac{w_j}{w_i}$$

$$n \lambda_{maks} = \sum_{i=1}^n \sum_{j=1}^n a_{ij} \frac{w_j}{w_i}$$

$$\lambda_{maks} = \frac{1}{n} \sum_{i=1}^n \sum_{j=1}^n a_{ij} \frac{w_j}{w_i}$$

jadi nilai eigen terbesar atau λ_{maks} didapat dengan menjumlahkan hasil perkalian jumlah kolom dengan eigen vektor utama.

Vektor eigen diperoleh dengan normalisasi B terlebih dahulu, sehingga $\sum_{i=1}^n w_i = 1$. Normalisasi ini dilakukan dengan membagi

entri suatu kolom matriks dengan jumlah seluruh entri dalam kolom tersebut. Setelah dinormalisasi, entri dari kolom – kolom dijumlahkan menurut baris. Untuk mendapatkan vektor eigen, entri masing – masing baris dihitung rata – ratanya. Secara matematis entri vektor eigen dapat ditulis sebagai berikut (Ekawati. 2006) :

$$w_i = \frac{1}{n} \sum_{j=1}^n \frac{a_{ij}}{\sum_{i=1}^n a_{ij}} \quad (2.5)$$

2.6.4.2. Pengambilan Keputusan

Untuk memperoleh prioritas menyeluruh, harus menghitung prioritas lokal terlebih dahulu. Prioritas lokal merupakan elemen – elemen dalam satu tingkat dengan memperhatikan satu kriteria saja. Prioritas lokal ini adalah vektor eigen.

Prioritas menyeluruh bagi alternatif – alternatif di tingkat paling bawah merupakan gabungan dari prioritas lokal, sehingga seluruh kriteria diperhatikan. Prioritas menyeluruh diperoleh dengan cara

mengalikan prioritas lokal elemen alternatif dengan prioritas lokal elemen satu tingkat di atasnya yang digunakan sebagai dasar dalam perbandingan berpasangan, kemudian menjumlahkan menurut baris yang bersesuaian. Untuk lebih jelasnya dapat dilihat pada Tabel 2.7.

Tabel 2.7. Tabel Prioritas Menyeluruh

	C_1	C_2	...	C_m	Prioritas menyeluruh
A_1	$b_{11}P_1$	$b_{12}P_2$...	$b_{1m}P_m$	u_1
A_2	$b_{21}P_1$	$b_{22}P_2$...	$b_{2m}P_m$	u_2
...
A_n	$b_{n1}P_1$	$b_{n2}P_2$...	$b_{nm}P_m$	u_n

Keterangan :

- C_j = elemen kriteria
- A = elemen alternatif
- b_{ij} = prioritas lokal elemen alternatif ke $-i$ berdasarkan elemen kriteria ke $-j$ yang berada satu tingkat di atasnya
- P_j = prioritas lokal elemen kriteria
- u_i = prioritas menyeluruh alternatif ke $-i$, $u_i = \sum_{j=1}^m b_{ij} p_j$
- i = 1,2,...,n
- j = 1,2,...,m.

2.6.4.3. Konsistensi Matriks Perbandingan Berpasangan

Apabila A adalah matriks perbandingan berpasangan yang konsisten maka semua nilai eigen bernilai nol kecuali jika yang bernilai sama dengan n . Tetapi bila A adalah matriks tak konsisten, variasi kecil atas a_{ij} akan membuat nilai eigen terbesar λ_{maks} selalu lebih besar atau sama dengan n : $\lambda_{maks} \geq n$.

Perbedaan antara λ_{maks} dengan n dapat digunakan untuk meneliti seberapa besar ketidak konsistenan yang ada dalam A , dimana rata-ratanya dinyatakan sebagai berikut:

$$\mu = \frac{\lambda_{maks} - n}{n - 1} \quad (2.6)$$

Menurut Saaty (1994) persamaan (2.3) didefinisikan sebagai *consistency indeks* (CI), dengan demikian :

$$CI = \frac{\lambda_{maks} - n}{n - 1} \tag{2.7}$$

Suatu matriks perbandingan berpasangan dinyatakan konsisten apabila nilai *consistency ratio* (CR) $\leq 5\%$ untuk $n = 3$, $(CR) \leq 8\%$ untuk $n = 4$ dan $(CR) \leq 10\%$ untuk $n \geq 5$. (Saaty,1994). Nilai CR yang lebih besar dari kriteria tersebut mengidentifikasi adanya ketidakkonsistenan, sehingga perlu dilakukan perbaikan untuk memperoleh matriks perbandingan berpasangan yang konsisten. CR dapat dihitung menggunakan rumus sebagai berikut :

$$CR = \frac{CI}{RI} \tag{2.8}$$

Dimana RI = *random consistency index*.

Nilai RI disajikan dalam Tabel 2.8.

Tabel 2.8. *Random Consistency Index* (RI)

n	1	2	3	4	5	6	7	8	9	10
RI	0	0	0.52	0.89	1.11	1.25	1.35	1.40	1.45	1.49

Sumber : Saaty (2002).

Saaty (1994) menyatakan bahwa a_{ij} yang memiliki nilai γ_{ij} paling jauh menyimpang dari 1 adalah entri penyebab ketidakkonsistenan matriks perbandingan berpasangan, dimana untuk menghitung γ_{ij} digunakan rumus :

$$\gamma_{ij} = a_{ij} w_j / w_i \tag{2.9}$$

cara untuk memperbaiki ketidakkonsistenan tersebut adalah dengan merubah a_{ij} menjadi w_i / w_j .

2.6.5. Penarikan Sampel Acak Sederhana

Sampel adalah sebagian dari populasi. Artinya tidak akan ada sampel jika tidak ada populasi. Populasi adalah keseluruhan elemen atau unsur yang akan kita teliti. Penelitian yang dilakukan atas seluruh elemen dinamakan sensus. Idealnya, agar hasil penelitiannya lebih bisa dipercaya, seorang peneliti harus melakukan sensus. Namun karena sesuatu hal peneliti bisa tidak meneliti keseluruhan elemen tadi, maka yang bisa dilakukannya adalah meneliti sebagian dari keseluruhan elemen atau unsur tadi.

Penarikan sampel acak sederhana adalah suatu metode untuk memilih n unit dari N sehingga setiap unit ${}_N C_n$ sampel yang berbeda mempunyai kesempatan yang samasa untuk dipilih (Cochran, 1991). Dalam prakteknya, penarikan sampel acak sederhana dipilih unit per unit. Pada proses penarikan, proses yang digunakan harus memberikan kesempatan terpilih yang sama untuk setiap unit dalam populasi. Unit yang terpilih sebanyak n merupakan sampel. Menurut Supranto (1992), penarikan sampel acak sederhana akan memberikan hasil data perkiraan yang dapat mewakili populasi jika populasi di mana sampel dipilih adalah homogen.

Jika suatu observasi termasuk atau tidak termasuk kategori tertentu yaitu kategori yang sedang diamati, proporsi kategori tersebut menunjukkan suatu karakteristik atau eksperimen Binomial. Rumus yang digunakan untuk menentukan banyaknya sampel (Cochran, 1991):

$$n = \frac{(Z_{\alpha/2})^2 pq}{d^2} \quad (2.10)$$
$$1 + \frac{1}{N} \left(\frac{Z_{\alpha/2} pq}{d^2} - 1 \right)$$

Keterangan :

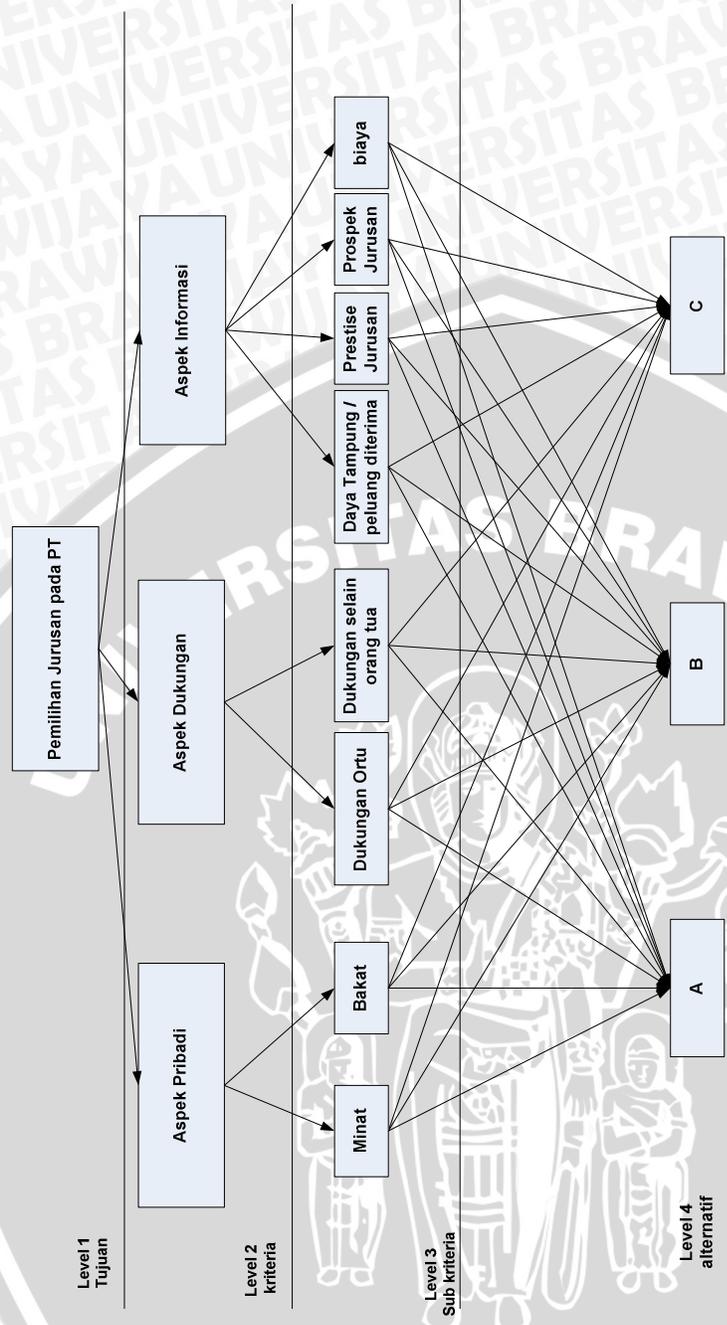
- n = ukuran sampel
- N = ukuran populasi
- d = sampling error
- p = proporsi sukses
- q = proporsi tidak sukses
- $Z_{\alpha/2}$ = 1.96

BAB III METODE PENELITIAN

3.1. Sumber Data

Dalam penerapan *recommender system* menggunakan metode AHP, contoh kasus yang diteliti adalah pemilihan program studi di perguruan tinggi. Untuk menganalisa indikator – indikator apa saja yang digunakan dalam pemilihan program studi pada perguruan tinggi maka dilakukan wawancara dengan pihak – pihak yang terkait dalam hal ini bekerja sama dengan LBBK Nurul Fikri, mahasiswa Universitas Brawijaya, pakar psikologi Universitas Negeri Malang dan studi literatur. Berdasarkan hasil wawancara dan studi literatur diperoleh data berupa persepsi mahasiswa terhadap pemilihan program studi pada perguruan tinggi yang didasarkan pada 3 kriteria, yaitu aspek pribadi, aspek dukungan, aspek informasi. Masing – masing kriteria terdiri dari beberapa sub kriteria. Untuk kriteria aspek pribadi, terdiri dari sub kriteria: minat, bakat; kriteria aspek dukungan terdiri dari sub kriteria dukungan orang tua atau keluarga, dan dukungan orang lain; kriteria aspek informasi terdiri dari sub kriteria: daya tampung atau peluang diterima, prestise jurusan, prospek jurusan, dan biaya. Berdasarkan hasil wawancara, maka diperoleh suatu ilustrasi analisa masalah yang digambarkan dalam suatu bentuk hirarki seperti Gambar 3.1.

Responden penelitian ini adalah mahasiswa S1 Universitas Brawijaya angkatan 2007 yang masih aktif pada semester genap tahun ajaran 2007/2008. Banyaknya responden (sampel) yang diteliti dihitung dengan menggunakan teori penarikan sampel acak sederhana berdasarkan persamaan (2.10). Program studi yang diteliti adalah semua program studi S1 yang terdapat di Universitas Brawijaya yang berjumlah 48 program studi seperti pada Lampiran 1.



Gambar 3.1. Struktur permasalahan (hirarki analisa permasalahan) pada contoh kasus pemilihan program studi

3.2. Deskripsi Sistem

Recommender system menggunakan AHP ini merupakan sistem yang dibangun untuk melakukan evaluasi perhitungan nilai prioritas global alternatif terbaik suatu permasalahan berdasarkan indikator yang mempengaruhi dalam permasalahan tersebut. Nilai prioritas global (menyeluruh) tertinggi dari alternatif yang dievaluasi merupakan alternatif terbaik yang direkomendasikan oleh sistem.

Sistem ini dirancang supaya dapat diimplementasikan dalam memberikan sebuah rekomendasi yang bersifat flexibel (mudah diterapkan) untuk kasus dengan multi indikator dan multi alternatif yang mempengaruhi.

Input yang dibutuhkan dalam *Recommender system* menggunakan metode AHP ini berupa:

1. Data aturan.

Data aturan digunakan untuk menggambarkan hirarki permasalahan yang ingin diuji. Data ini merupakan data pendefinisian dari keterkaitan antara suatu indikator dengan indikator lainnya. Data aturan ini berisi aturan – aturan yang nantinya digunakan untuk proses penghitungan nilai prioritas alternatif

2. Nilai prioritas (bobot) tiap – tiap indikator

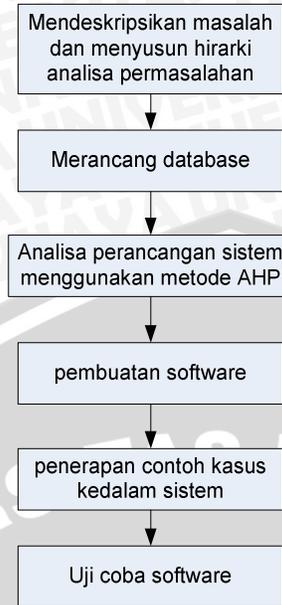
Merupakan nilai yang digunakan untuk menggambarkan relatif pentingnya suatu indikator terhadap indikator yang lainnya, berkenaan pada sifat elemen yang sama. Dalam hal ini menggunakan nilai 0-9 (skala AHP) yang ditetapkan sebagai pertimbangan dalam perbandingan antar indikator.

3.3. Metode

Tahapan – tahapan yang dilakukan dalam penelitian ini adalah sebagai berikut:

1. Mendeskripsikan masalah yang diteliti dan merancang suatu hirarki analisa permasalahan.
2. Merancang database.
3. Menganalisa dan melakukan perancangan *recommender system* menggunakan metode AHP.
4. Membuat perangkat lunak (*software*) berdasarkan analisis dan rancangan yang telah dilakukan.
5. Penerapan contoh kasus ke dalam sistem.
6. Uji coba perangkat lunak.

Langkah – langkah yang dilakukan dapat dijelaskan kembali dalam bentuk alur yang ditunjukkan dalam Gambar 3.2.



Gambar 3.2. Tahapan – tahapan penelitian

3.3.1. Perancangan Database

Dalam penelitian ini, untuk menghasilkan sistem rekomendasai menggunakan metode AHP maka dibutuhkan data aturan (*rule*) yang disimpan dalam data aturan. Rancangan dari tabel aturan yaitu berupa *id_rule*, *indikator*, *indikator_asal*, *level*, *nilai_eigen*, dan *rule_asal* seperti dijelaskan pada Tabel 3.1.

Tabel 3.1. Tabel Aturan (*RULE*)

Field	Type	Description
<i>id_rule</i>	autonumber	Primary Key, Not Null
<i>indikator</i>	text (50)	Not Null
<i>indikator_asal</i>	text (50)	Allow Null
<i>level</i>	Number	Not Null
<i>Nilai_eigen</i>	Number	Not Null
<i>Rule_asal</i>	Number	Allow Null

3.3.2. Perancangan Sistem

Pada tahapan ini akan dijelaskan mengenai proses-proses yang dilakukan dalam perancangan *recommender system* menggunakan AHP dalam memberikan rekomendasi alternatif terbaik suatu permasalahan.

Setelah *user* melakukan pendeskripsian masalah dan menyusun suatu hirarki analisa permasalahan maka untuk melakukan proses evaluasi nilai prioritas masing – masing alternatif, langkah – langkah yang dilakukan adalah sebagai berikut:

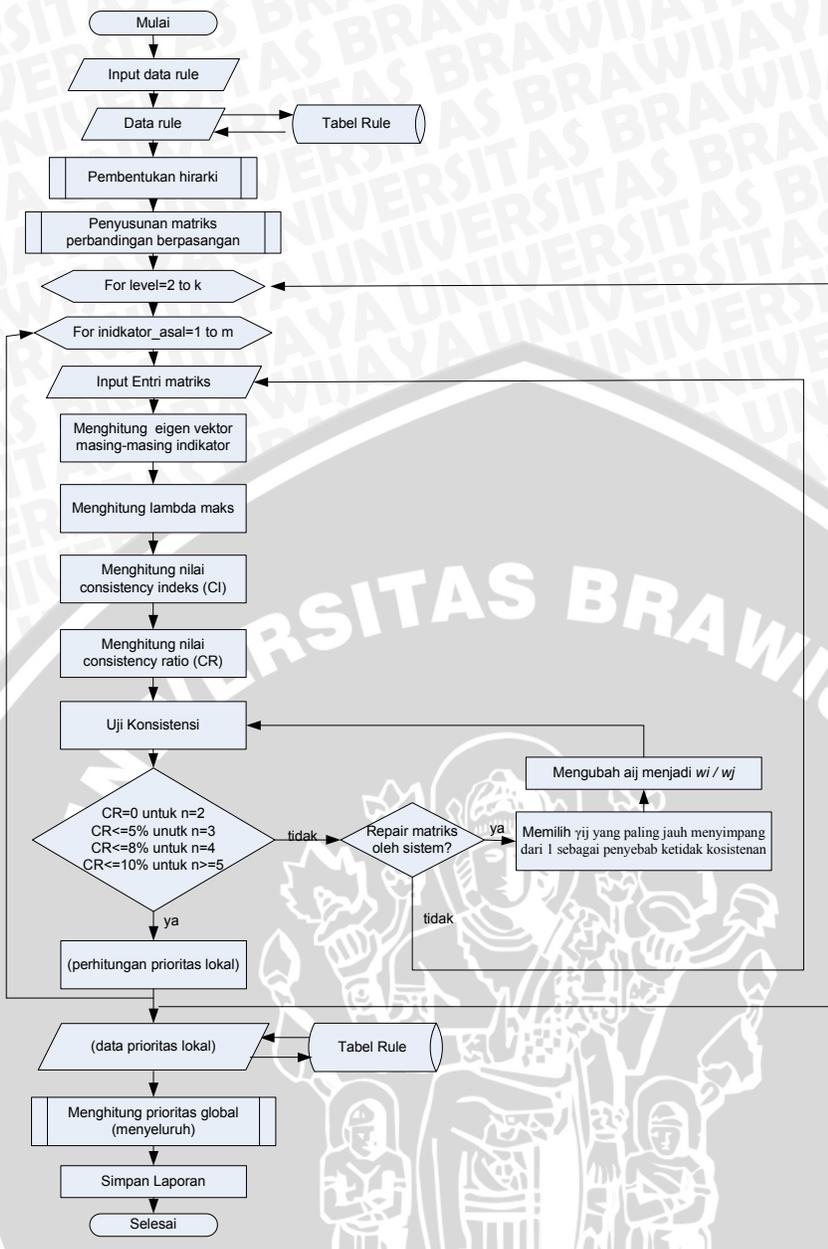
1. *User* menyusun aturan (*rule*) keterkaitan antar indikator dalam hirarki permasalahan
2. Menyusun matriks perbandingan berpasangan. *User* memberikan nilai (0-9) relatif pentingnya suatu indikator terhadap indikator yang lain berkenaan dengan sifat elemen yang sama.
3. Menghitung eigen vektor untuk tiap – tiap perbandingan. Perhitungan nilai eigen vektor sesuai dengan persamaan (2.5)
4. Melakukan uji konsistensi pada tiap matriks perbandingan berpasangan.

- Menghitung *consistency index* (CI) sesuai persamaan (2.7)
- Menghitung *consistency ratio* (CR) sesuai persamaan (2.8)
- Jika $CR = 0$ untuk $n = 2$, $CR \leq 5\%$ untuk $n = 3$, $CR \leq 8\%$ untuk $n = 4$ dan $CR \leq 10\%$ untuk $n \geq 5$. Maka matriks tersebut konsisten.

Jika terdapat matriks perbandingan berpasangan yang tidak konsisten maka dilakukan perbaikan perbandingan berpasangan.

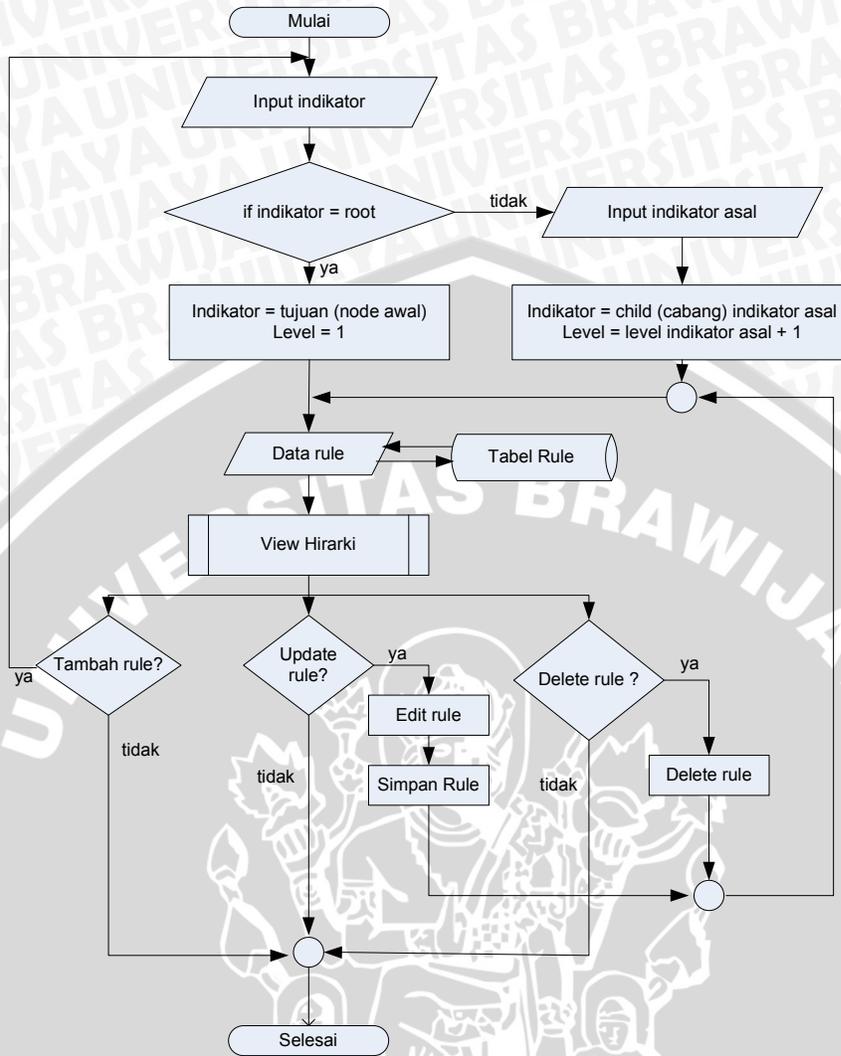
- Menghitung γ_{ij} sesuai persamaan (2.9) kemudian memilih a_{ij} yang memiliki nilai γ_{ij} paling jauh menyimpang dari 1 sebagai entri penyebab ketidak konsistenan.
 - Mengubah a_{ij} penyebab ketidak konsistenan menjadi w_i / w_j
5. Pengambilan keputusan
 - Menghitung prioritas lokal (update nilai eigen) untuk setiap matriks perbandingan berpasangan sehingga didapatkan prioritas elemen – elemen dalam satu tingkat dengan memperhatikan satu kriteria saja.
 - Menghitung prioritas menyeluruh bagi elemen alternatif
 6. Menampilkan nilai prioritas masing – masing alternatif hasil evaluasi dari matriks perbandingan berpasangan.

Alur dari proses evaluasi (perhitungan) prioritas global tiap-tiap alternatif serta mekanisme kerja sistem digambarkan kembali seperti pada Gambar 3.3.



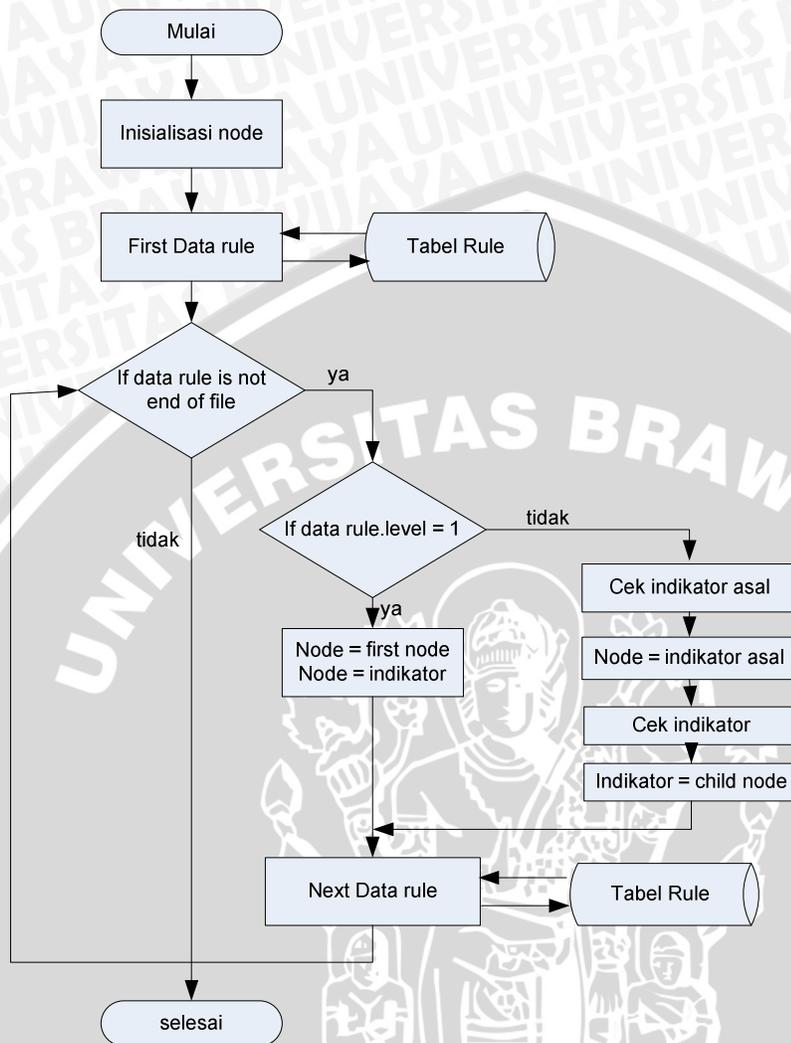
Gambar 3.3. Diagram alir mekanisme kerja sistem.

Dari proses input data rule pada diagram alir pada Gambar 3.3 dapat dijabarkan lagi dalam bentuk diagram alir seperti pada Gambar 3.4.



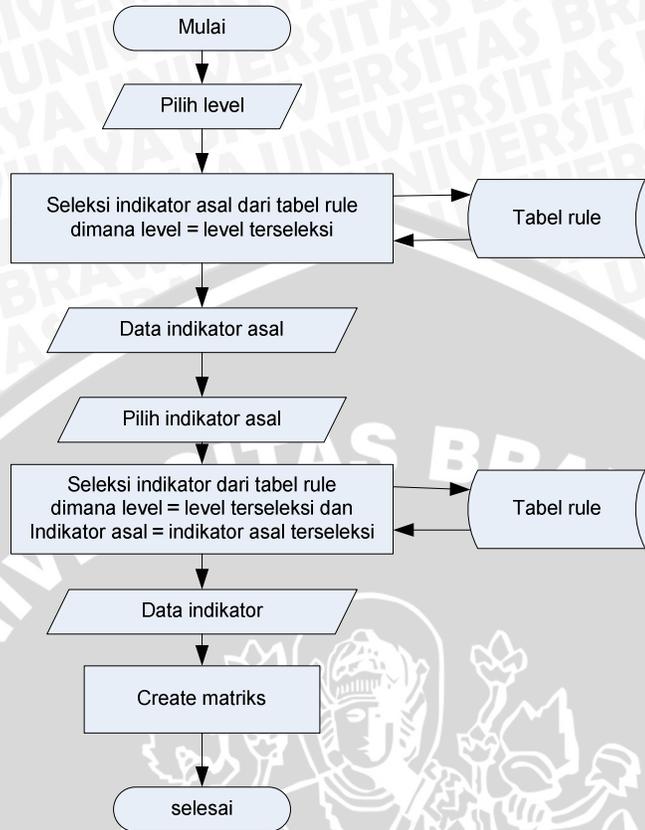
Gambar 3.4. Diagram alir proses input data rule

Pada diagram alir input data rule seperti pada Gambar 3.4 terdapat proses *view* hirarki (pembentukan hirarki) dari tabel *rule*. Proses pembentukan hirarki ini bisa dijelaskan kembali dalam bentuk diagram alir seperti pada Gambar 3.5.



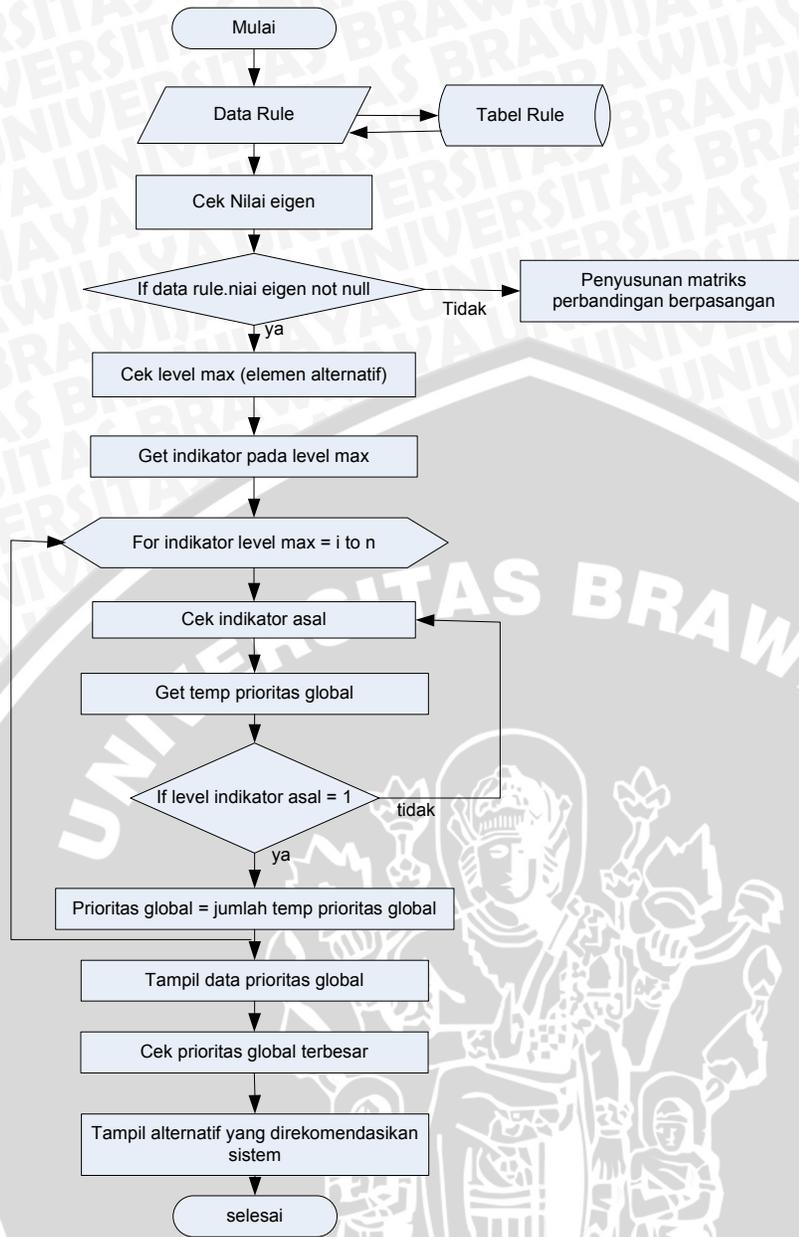
Gambar 3.5. Diagram alir proses pembentukan hirarki

Proses penyusunan matriks perbandingan berpasangan yang ada pada diagram alir proses evaluasi pada Gambar 3.3 dapat dijelaskan kembali dalam bentuk diagram alir seperti pada Gambar 3.6.



Gambar 3.6. Diagram alir penyusunan matriks perbandingan berpasangan.

Proses perhitungan prioritas global pada Gambar 3.3 dapat dijelaskan kembali ke dalam bentuk diagram alir proses perhitungan prioritas global seperti pada Gambar 3.7.



Gambar 3.7. Diagram alir proses perhitungan prioritas global.

3.3.3. Perancangan Uji Coba

Uji coba akan dilakukan terhadap perangkat lunak untuk mengetahui hasil dalam penyelesaian permasalahan multi indikator dan multi alternatif. Pengujian akan dilakukan terhadap sistem dengan menginputkan data aturan dari analisa permasalahan, serta nilai (bobot) relatif pentingnya suatu indikator terhadap indikator lain yang memiliki sifat elemen yang sama. Tipe inputan dari sistem dapat dilihat pada Tabel. 3.2.

Tabel 3.2. Tabel Tipe Inputan Sistem.

No	Nama Atribut	Tipe Atribut	Tipe Data
1.	Indikator	kontinyu	text
2.	indikator asal	diskrit	text
3.	level	diskrit	number
4.	nilai (bobot)	diskrit	number

Setiap uji coba akan dicari nilai dari vektor eigen (prioritas lokal) dari tiap-tiap indikator dan nilai global (prioritas global) dari tiap-tiap alternatif berdasarkan elemen indikator yang berada satu level (tingkat) di atasnya. Nilai global tertinggi dari alternatif yang dievaluasi merupakan alternatif yang direkomendasikan oleh sistem.

Tabel rancangan uji coba dari evaluasi nilai eigen (prioritas lokal) tiap – tiap indikator dan evaluasi prioritas global untuk tiap – tiap alternatif dapat dilihat pada Tabel 3.3 dan Tabel 3.4.

Tabel 3.3. Tabel Rancangan Uji Coba Evaluasi Nilai Vektor Eigen

Id_rule	Indikator	Indikator Asal	Nilai vektor eigen
1			
2			
...			
n			

Tabel 3.4. Tabel Rancangan Uji Coba Evaluasi Prioritas Global

	C_1	C_2	...	C_m	Prioritas global
A_1	$b_{11}P_1$	$b_{12}P_2$...	$b_{1m}P_m$	u_1
A_2	$b_{21}P_1$	$b_{22}P_2$...	$b_{2m}P_m$	u_2
...
A_n	$b_{n1}P_1$	$b_{n2}P_2$...	$b_{nm}P_m$	u_n

Keterangan :

- C_j = elemen kriteria
- A = elemen alternatif (indikator pada level paling bawah)
- b_{ij} = prioritas lokal elemen alternatif ke $-i$ berdasarkan elemen kriteria ke $-j$ yang berada satu tingkat di atasnya
- P_j = prioritas lokal elemen kriteria
- u_i = prioritas menyeluruh alternatif ke $-i$
- i = 1,2,...,n
- j = 1,2,...,m.

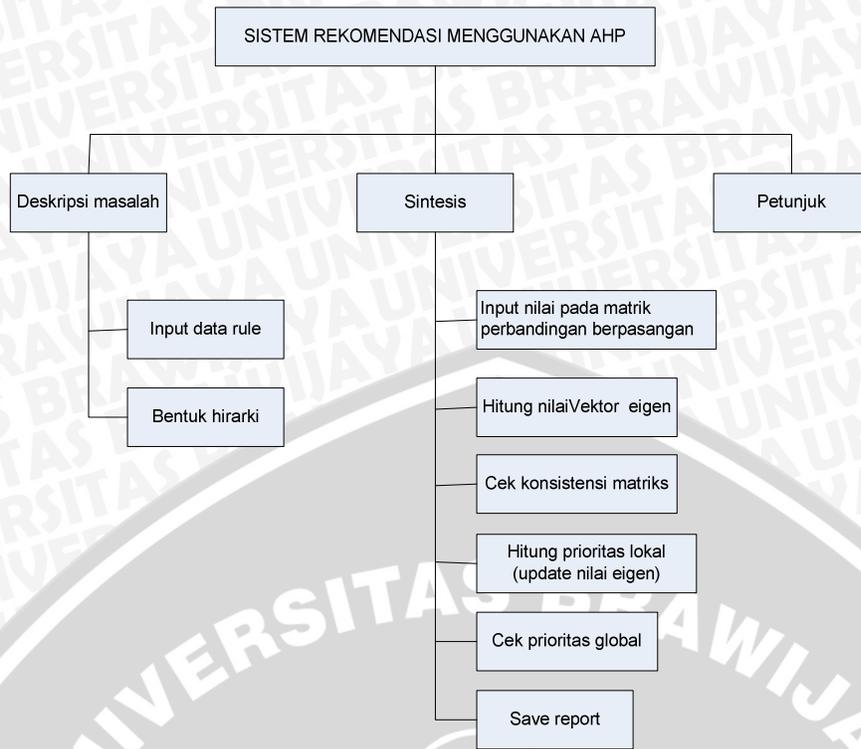
3.3.4. Pengujian Dan Analisa Hasil

Pengujian dilakukan dengan menguji kesesuaian antara hasil rekomendasi program studi yang diberikan oleh sistem melalui proses perhitungan prioritas global dengan program studi dari *user* yang sudah diterima pada suatu perguruan tinggi. Hasil rekomendasi dari sistem dibandingkan dengan hasil kenyataan dari *user*. Untuk setiap rekomendasi prodi yang dihasilkan melalui proses perhitungan prioritas global oleh sistem yang sesuai dengan prodi kenyataan dari *user* maka bernilai 1 (satu), sedangkan yang tidak sesuai maka bernilai 0 (nol). Setiap hasil yang sesuai atau tidak sesuai dengan hasil asli dari *user* akan dijumlahkan sehingga dapat diketahui kesesuaian antara hasil rekomendasi dari sistem dengan hasil yang terjadi pada *user* secara kenyataan.

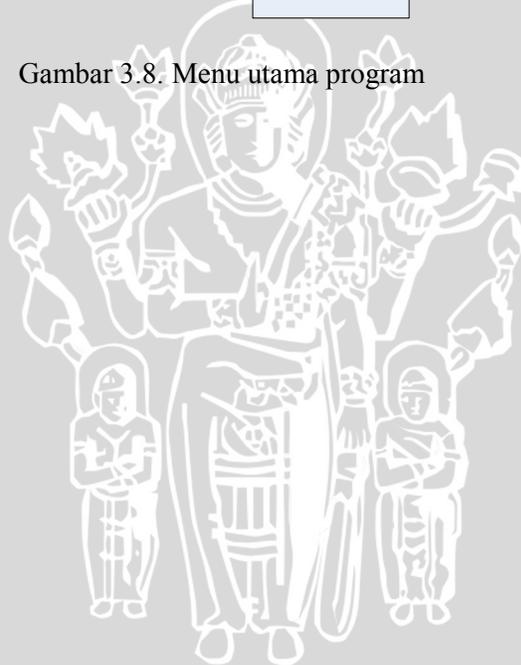
3.3.5. Perancangan Antar Muka

Rancangan antar muka (*interface*) untuk *recommender system* menggunakan AHP dapat dilihat seperti pada Gambar 3.8.





Gambar 3.8. Menu utama program





BAB IV IMPLEMENTASI DAN PEMBAHASAN

4.1. Lingkungan Impelementasi

Perangkat keras yang digunakan dalam pengembangan *recommender system* ini adalah sebagai berikut:

1. Prosesor AMD Sempron 2600+ 1.6Ghz.
2. Memori 960 Mb.
3. Hardisk dengan kapasitas 40 Gb.
4. Monitor 15".
5. Keyboard.
6. Mouse.

Sedangkan perangkat lunak yang digunakan dalam pengembangan *recommender system* ini adalah sebagai berikut:

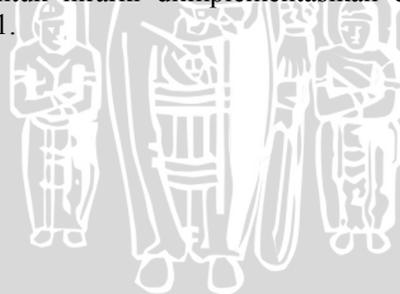
1. Sistem operasi Windows XP SP2.
2. Borland Delphi 2007.
3. Microsoft Access 2003.

4.2. Implementasi Program

Berdasarkan rancangan sistem pada Subbab 3.3.2 maka akan dijelaskan implementasi dari masing – masing proses tersebut.

4.2.1. Implementasi Pembentukan Hirarki

Proses pembentukan hirarki permasalahan merupakan awal dari semua proses yang akan dilakukan. Proses ini dimulai dengan menginputkan elemen pokok dari suatu permasalahan dan bagian – bagian dari elemen – elemen tersebut. Elemen pokok permasalahan disebut dengan indikator asal dan bagian – bagiannya disebut dengan indikator. Sistem akan menyimpan inputan dari *user* ke dalam database dan menampilkannya dalam bentuk hirarki (*tree*). Proses untuk menampilkan data *rule* dari database ke dalam bentuk hirarki diimplementasikan dalam *procedure* seperti pada Gambar 4.1.



```

procedure TFormUtama.init_tree;
begin
    node:=tvData.Items.GetFirstNode;
    tvData.Items.Clear;
end;
function TFormUtama.get_node (rule_asal : string) : TTreeNode;
var node : TTreeNode;
    i : integer;
begin
    node := tvData.Items.GetFirstNode;
    i:=0;
    while i<>tvData.Items.Count do
    begin
        if node.Text<>rule_asal then
        begin
            node := tvData.items.item[i];
        end;
        inc(i);
    end;
    result:=node;
end;
procedure TFormUtama.isi_tree;
var temp : string;
begin
    dm.TbRule.First;
    while not dm.TbRule.Eof do
    begin
        if dm.TbRule.FieldName('in level').AsInteger=1 then
        tvData.Items.Addfirst (node,dm.TbRule.FieldName('indikator').AsString)
        else
        begin
            temp := dm.TbRule.FieldName('indikator_asal').AsString;
            node := get_node(temp);
            tvData.Items.AddChildFirst (node,dm.TbRule.FieldName('indikator').AsString);
        end;
        dm.TbRule.Next;
    end;
end;
end;

```

Gambar 4.1. *Procedure* pembentukan hirarki.

Procedure `init_tree ()` berfungsi untuk inialisasi *node* dan membersihkan isi dari *tree* atau hirarki. Procedure `isi_tree ()` berfungsi untuk menyusun ulang *tree* sesuai dengan isi di database. Function `get_node (rule_asal : String)` berfungsi untuk mencari *node* dan melakukan pengecekan masing-masing *child* (anak) dari *node* berdasarkan *rule_asal*.

4.2.2. Implementasi Penyusunan Matriks Perbandingan Berpasangan

Setelah mendeskripsikan masalah dalam bentuk hirarki, maka langkah selanjutnya adalah melakukan perbandingan berpasangan terhadap indikator - indikator yang ada dalam hirarki permasalahan. Proses penyusunan matriks perbandingan berpasangan dimulai dengan memilih level yang akan diseleksi. Pada proses pemilihan level terdapat suatu *procedure* yang digunakan untuk menyeleksi indikator asal yang ada pada database yang memiliki level yang sama. *Procedure* untuk memilih level dan menyeleksi indikator asal yang memiliki level yang sama seperti level yang terseleksi diimplementasikan seperti pada Gambar 4.2.

```
procedure TFormUtama.DBComboLevelCloseUp(Sender: TObject);
begin
DM.ADOQAsal.Active:=false;
DM.ADOQAsal.SQL.Text:='select indikator asal from trule where
in_level =
'+IntToStr(dm.ADOQLevel.fieldbyname('in_level').AsInteger)+' group
by indikator_asal';
DM.ADOQAsal.Active:=true;
end;
```

Gambar 4.2. *Procedure* pemilihan level dan penyeleksian indikator asal pada level terseleksi.

Matriks akan terbentuk setelah *user* memilih indikator asal pada suatu level, sehingga ordo matriks yang terbentuk adalah sebanyak *child* (indikator) yang memiliki indikator asal dan level yang sama seperti pada indikator asal dan level yang terseleksi. Implementasi dari proses tersebut ditunjukkan seperti pada Gambar 4.3.



```

procedure TFormUtama.DBComboIndikatorAsalCloseUp(Sender: TObject);
var i,j,k:integer;
    c : PAnsiChar;
begin
  btnUpdate.Enabled:=false;
  dm.ADOQIndikator.Active:=false;
  dm.ADOQIndikator.SQL.Text:='select id,indikator from trule where
  indikator_asal like
  '+QuotedStr(dm.ADOQAsal.fieldbyname('indikator asal').AsString)+'
  and in_level =
  '+IntToStr(dm.ADOQLevel.fieldbyname('in level').AsInteger);
  dm.ADOQIndikator.Active:=true;
  y:=0;
  x:=0;
  i:=0;j:=0;k:=2;
  dm.ADOQIndikator.First;

  //buat tempat stringgrid
  //clear stringgrid;
  StringGrid1.RowCount:=1;
  StringGrid1.ColCount:=1;

  //ngisi stringgrid
  while not dm.ADOQIndikator.Eof do
  begin
    StringGrid1.Visible:=true;
    StringGrid1.RowCount:=StringGrid1.RowCount+1;
    StringGrid1.ColCount:=StringGrid1.ColCount+1;
    Inc(x);Inc(y);

    StringGrid1.Cells[0,x]:=dm.ADOQIndikator.FieldByName('indikator').
    AsString;
    if Length(dm.ADOQIndikator.FieldByName('indikator').AsString)>k
    then
      k:=
      Length(dm.ADOQIndikator.FieldByName('indikator').AsString);
      StringGrid1.ColWidths[0]:=k*8;

    StringGrid1.Cells[y,0]:=dm.ADOQIndikator.FieldByName('indikator').
    AsString;
    StringGrid1.ColWidths[y]:=k*8;
    dm.ADOQIndikator.Next;
    StringGrid1.Cells[x,y]:='1';

  end;
  StringGrid1.FixedCols:=1;
  StringGrid1.FixedRows:=1;

  for I := 1 to x do
    for j := 1 to y do
      if i<>j then
        StringGrid1.Cells[i,j]:='';
    end;
  end;

```

Gambar 4.3. *Procedure* penyusunan matriks perbandingan berpasangan.

4.2.3. Implementasi Perhitungan vektor Eigen

Setelah sistem mendapatkan input dari *user* melalui matriks perbandingan berpasangan, maka proses selanjutnya adalah mensintesis nilai inputan tersebut dengan menghitung nilai vektor eigen masing – masing indikator.

Seperti pada Persamaan 2.5 pada Subbab 2.6.4 tentang perhitungan nilai eigen, maka *procedure* untuk mengimplementasikan proses perhitungan nilai eigen tersebut dijelaskan seperti pada Gambar 4.3.

```
Procedure matriks.get_eigen;
var i,j,l : integer;
    temp : real;
begin
  //hitung jumlah kolom
  l:=ordo+1;
  for I := 1 to ordo do
  begin
    temp:=0;
    for j := 1 to ordo do
    begin
      temp := temp + data[i,j];
    end;
    data[i,l]:=temp;
  end;
  //hitung eigen
  for j := 1 to ordo do
  begin
    temp:=0;
    for i := 1 to ordo do
    begin
      temp := temp+ (data[i,j]/data[i,l]);
    end;
    temp:=temp/ordo;
    data[l,j]:=temp;
  end;
end;
end;
```

Gambar 4.4. *Procedure* perhitungan nilai eigen.

4.2.4. Implementasi Pengecekan Nilai Konsistensi Matriks

Pengecekan nilai konsistensi adalah suatu proses yang harus dilakukan dalam mensintesis nilai perbandingan dari *user*. Konsistensi dari suatu matriks ditentukan oleh nilai CR (*consistency ratio*) dari masing – masing perbandingan. Untuk mendapatkan nilai CR maka harus dihitung terlebih dahulu nilai λ dan nilai CI (*consistency index*). *Procedure* untuk mendapatkan nilai – nilai tersebut adalah sebagai berikut.

4.2.4.1. Perhitungan Nilai Lambda, CI dan CR

Nilai $\lambda_{maksimum}$ merupakan nilai eigen terbesar dari matriks berordo n . Nilai $\lambda_{maksimum}$ diperoleh dengan menjumlahkan hasil perkalian jumlah kolom dengan eigen vektor utama. Nilai CI (*consistency index*) merupakan rata-rata dari perbedaan nilai $\lambda_{maksimum}$ dengan n ordo matriks. Nilai CR menyatakan konsisten atau tidaknya suatu matriks perbandingan berpasangan. *Listing* program untuk mendapatkan nilai CI, CR dan nilai $\lambda_{maksimum}$ ditunjukkan pada Gambar 4.5.

```
procedure matriks.get_lambda;
var temp:real;
    i: integer;
    const ir : array[1..10] of double;
begin
const ir[1]:=1;
const ir[2]:=1;
const ir[3]:=0.52;
const ir[4]:=0.89;
const ir[5]:=1.11;
const ir[6]:=1.25;
const ir[7]:=1.35;
const ir[8]:=1.40;
const ir[9]:=1.45;
const ir[10]:=1.49;
temp :=0;
for i := 1 to ordo do
    temp := temp + (data[ordo+1,i]*data[i,ordo+1]);
lambda:=temp;
ci:=(lambda-ordo)/(ordo-1);
cr:=ci/const_ir[ordo];
if (ordo=2) and (cr=0) then
    konsisten:=true
else if (ordo=3) and (cr<=0.05) then
    konsisten:=true
else if (ordo=4) and (cr<=0.08) then
    konsisten:=true
else if (ordo>=5) and (cr<=0.1) then
    konsisten:=true
else
    konsisten:=false;
end;
```

Gambar 4.5. *Procedure* perhitungan nilai lambda, CI, CR.

4.2.4.2. Pengecekan Konsistensi

Diberikan satu tombol “*create matriks*” untuk memanggil *procedure* perhitungan nilai *lambda*, CI, CR di atas dan untuk mengecek apakah input yang diberikan pada matriks dinyatakan konsisten atau tidak konsisten. *Listing* program dari tombol “*create matriks*” ditunjukkan pada Gambar 4.6.

```

procedure TFormUtama.btnMatriksClick(Sender: TObject);
var i,j : integer;
begin
  if cek then      //cek apakah masih ada matrix yg kosong
  begin
    for i := 1 to x do
      for j := i to y do
        StringGrid1.Cells[i,j]:=FloatToStr(1/StrToFloat(StringGrid1.Cells[j,
        i]));
    for i := 1 to x do
      for j := 1 to y do
        begin
          try
            t.data[i,j]:=StrToFloat(StringGrid1.Cells[i,j]);
          except on EConvertError do
            begin
              MessageDlg('Data kurang lengkap...',mtWarning,[mbok],0);
            end;
          end;
        end;
      end;

m.init(x);
m.set data(t);
set_matriks;

if m.konsisten then
begin
  EditLambda.Text:=FloatToStr(m.lambda);
  EditCI.Text:=FloatToStr(m.ci);
  EditCR.Text:=FloatToStr(m.cr);
  EditKonsinten.Text:=BoolToStr(m.konsisten);
  if EditKonsinten.Text='-1' then
    labelkonsisten.Caption:='true'
  else
    labelkonsisten.Caption:='false';

  btnUpdate.Enabled:=true;
end
else
begin
  if MessageDlg('Matriks anda tidak konsisten, anda ingin sistem
mengupdate otomatis?',mtConfirmation,[mbYes,mbNo],0)=idyces then
  begin
    m.update;
    set_matriks;
    btnUpdate.Enabled:=true;
  end;
end;
end;

```

```

EditLambda.Text:=FloatToStr(m.lambda);
EditCI.Text:=FloatToStr(m.ci);
EditCR.Text:=FloatToStr(m.cr);
EditKonsinten.Text:=BoolToStr(m.konsisten);
btnUpdate.Enabled:=true;
end
else
begin
  MessageDlg('Silakan update matriks anda ...
',mtInformation,[mbok],0);
  DBComboIndikatorAsal.DropDown;
  DBComboIndikatorAsal.CloseUp(true);
end;
end;
end
else
  MessageDlg('terdapat kesalahan pada data matriks anda, periksa
kembali ...',mtWarning,[mbok],0);
end;

```

Gambar 4.6. *Procedure* pengecekan konsistensi matriks.

```

function TFormUtama.cek : boolean;
var hasil : boolean;
    m,n : integer;
begin
  hasil := true;
  for m := 1 to StringGrid1.RowCount-1 do
    for n := 1 to StringGrid1.RowCount-1 do
      if (n<m) and (StringGrid1.Cells[m,n]='') then
        hasil:=false;
      result := hasil;
    end;
  end;
end;

```

Gambar 4.7. Fungsi untuk mengecek apakah masih ada entri matriks yang masih kosong pada segitiga atas matriks.

```

procedure matriks.init (d:integer);
var i,j : integer;
begin
  konsisten:=false;
  ordo:=d;
  for i := 1 to ordo do
    for j := 1 to ordo do
      begin
        data[i,j]:=1;
      end;
    end;
  end;
end;

```

Gambar 4.8. *Procedure* untuk inisialisasi matriks, menentukan nilai awal, nilai konsisten dan ordo matriks.

```

procedure matriks.set_data (d : matriks);
var i,j : integer;
begin
for i := 1 to ordo do
  for j := 1 to ordo do
    data[i,j]:=d.data[i,j];
  end;
end;

```

Gambar 4.9. *Procedure* untuk mengeset data matriks (entri matriks).

```

procedure TFormUtama.set matriks;
var i,j : integer;
begin
m.get_eigen;
m.get_lambda;
StringGrid1.RowCount:=m.ordo+2;
StringGrid1.ColCount:=m.ordo+2;
for I := 1 to m.ordo+1 do
  for j := 1 to m.ordo+1 do
    StringGrid1.Cells[i,j]:=FloatToStr(m.data[i,j]);
  end;
end;

```

Gambar 4.10. *Procedure* untuk mengisi data pada matriks.

Apabila dari proses pengecekan didapatkan nilai CR melebihi batas ketentuan maka nilai konsisten bernilai 0 (nol) yang berarti matriks tersebut tidak konsisten, dan apabila nilai CR kurang dari batas ketentuan maka nilai konsisten bernilai 1 (satu) yang berarti matriks tersebut konsisten. Sistem memberikan pilihan bagi *user* apabila mendapatkan matriks yang diproses bernilai tidak konsisten yaitu melakukan perbandingan ulang atau sistem yang akan memperbaiki matriks yang tidak konsisten tadi.

Matriks yang tidak konsisten harus dicari nilai *gamma* penyebab ketidakkonsistennya, yaitu nilai *gamma* yang melenceng jauh dari 1 (satu). Kemudian dilakukan perbaikan terhadap entri matriks yang memiliki nilai *gamma* paling besar. *Procedure* untuk pencarian nilai *gamma* dan perbaikan entri penyebab ketidakkonsistenan ditunjukkan pada Gambar 4.11.

```

procedure matriks.update;
var i,j,m,n : integer;
    max : double;
begin
max := 1;
m:=1;
n:=1;
for j := 1 to ordo do //row
    for i := j+1 to ordo do //kol
        begin
gama.data[i,j]:=data[i,j]*(data[ordo+1,i]/data[ordo+1,j]);
        end;
for j := 1 to ordo do //row
    for i := j+1 to ordo do //kol
        begin
if gama.data[i,j]>max then
            begin
max:=gama.data[i,j];
m:=j;n:=i;
            end;
        end;
gama.data[n,m]:=data[ordo+1,m]/data[ordo+1,n];
data[n,m]:=gama.data[n,m];
end;
end;

```

Gambar 4.11. *Procedure* untuk memperbaiki matriks yang tidak konsisten.

4.2.5. Implementasi Perhitungan Prioritas Lokal

Prioritas lokal suatu indikator didapatkan dengan cara memperbaharui nilai eigen dari suatu matriks perbandingan berpasangan yang telah dinyatakan konsisten, oleh karena itu diberikan satu tombol “*update eigen*” untuk melakukan proses perhitungan nilai lokal tadi. Proses ini diimplementasikan dalam *procedure* seperti pada Gambar 4.12.



```

procedure TFormUtama.btUpdateClick(Sender: TObject);
var j,k : integer;
begin
j:=1;
k:=m.ordo+1;
query_indikator.First;
while not query_indikator.Eof do
begin
  komand.CommandText:='update TRULE set NILAI EIGEN =
'+quotedstr(FloatToStr(m.data[k,j]))+' where id =
'+query_indikator.FieldName('id').AsString; //'update TRULE set
NILAI_EIGEN = '+FloatToStr(m.data[m.ordo+1,j]) +' where id =
'+query_indikator.FieldName('id').AsString;
  komand.Execute;
  Inc(j);
  query_indikator.Next;
end;
dm.TbRule.Refresh;
end;

```

Gambar 4.12. *Procedure* perhitungan prioritas lokal.

4.2.6. Implementasi Perhitungan Prioritas Global

Prioritas global (menyeluruh) diperoleh dengan cara mengalikan prioritas lokal elemen alternatif atau elemen yang berada pada level paling bawah (level maksimum) dengan prioritas elemen satu tingkat di atasnya yang digunakan dalam perbandingan berpasangan, kemudian menjumlahkan menurut baris yang bersesuaian.

Proses untuk mendapatkan level maksimum atau untuk menemukan elemen alternatif diimplementasikan pada *procedure* seperti pada Gambar 4.13.

```

procedure TFormUtama.getmax;
begin
max_level:=0;
dm.TbRule.First;
while not dm.TbRule.Eof do
begin
  if dm.TbRule.FieldName('in level').AsInteger>max_level then
    max_level:=dm.TbRule.FieldName('in_level').AsInteger;
  DM.TbRule.Next;
end;
end;

```

Gambar 4.13. *Procedure* untuk mendapatkan level maksimum

Untuk menghitung nilai global dari elemen alternatif maka diberikan tombol “*Check global priority*” yang memiliki *listing* program seperti pada Gambar 4.14.

```

procedure TFormUtama.btnGlobalClick(Sender: TObject);
var g : global;

```

```

i : integer;
temp : double;
temp : string;
maks_ : string;
maks : double;
begin
Memol.Lines.Clear;
if cek_eigen then
begin
getmax;
dm.ADOQAlter.Active:=false;
dm.ADOQAlter.SQL.Text:='select indikator from trule where in_level =
'+IntToStr(max_level)+' group by indikator';
dm.ADOQAlter.Active:=true;
dm.ADOQAlter.First;
maks := 0;
while not dm.ADOQAlter.Eof do
begin
temp:=0;
temp_ := dm.ADOQAlter.FieldByName('indikator').AsString;
dm.ADOQTabel.Active:=false;
dm.ADOQTabel.Active:=true;
dm.ADOQTabel.First;
while not dm.ADOQTabel.Eof do
begin
if
StrComp(pchar(temp_),pchar(dm.ADOQTabel.FieldByName('indikator').AsS
tring))=0 then
begin
g.sets;
i :=dm.ADOQTabel.FieldByName('id').AsInteger;
g.get_nilai_global(i);
temp := temp + g.nilai_prior;
end;
if temp>maks then
begin
maks := temp;
maks_:= temp;
end;
dm.ADOQTabel.Next;
end;
Memol.Lines.Add(' '+temp );
Memol.Lines.Add(' '+FloatToStr(temp));
dm.ADOQAlter.Next;
end;
Memol.Lines.Add(' '+Alternatif yang direkomendasikan');
Memol.Lines.Add(' '+ maks );
end
else
MessageDlg('nilai eigen masih ada yang kosong, harap periksa
kembali...',mtWarning,[mbok],0);
end;

```

Gambar 4.14. *Procedure* perhitungan nilai global.

```
function TFormUtama.cek_eigen:boolean;
```

```

var hasil:boolean;
begin
hasil := true;
dm.TbRule.First;
while not dm.TbRule.Eof do
begin
if dm.TbRule.FieldName('nilai_eigen').AsString='' then
hasil:=false;
dm.TbRule.Next;
end;
result:=hasil;
end;

```

Gambar 4.15. Fungsi Untuk mengecek apakah masih ada nilai eigen dari indikator yang masih kosong pada database.

4.2.7. Implementasi Penyimpanan Hasil (Save Report)

Sistem memberikan pilihan bagi *user* untuk menyimpan hasil dari evaluasi yang telah dilakukan. Hasil evaluasi dapat disimpan dalam bentuk Excell atau *worksheet*. Fungsi untuk melakukan export ke dalam bentuk excell ditunjukkan dalam Gambar 4.16.

```

function TFormUtama.IsObjectAcive(className : string):boolean;
var
ClassID: TCLSID;
Unknown: IUnknown;
begin
try
ClassID := ProgIDToClassID(className);
result := GetActiveObject(ClassID, nil, Unknown) = S_OK;
except
// raise;
result := false;
end;
end;
procedure TFormUtama.aktivasi(nmfile : string);
const xlWBATWorksheet = -4167;
begin
try
XLS := CreateOleObject('Excel.Application');
//XLBook := XLS.Workbooks.Open(nmfile);
XLS.Workbooks.add(xlWBATWorksheet);
except
MessageBeep(MB_ICONEXCLAMATION);
MessageDlg('file not available!',mtError, [mbOK], 0);
if IsObjectAcive('Excel.Application') then
deaktivasi;
end;
end;
procedure TFormUtama.deaktivasi;
Begin

```

```

Try
  XLS.Quit;
  XLS := Unassigned;
Except
  MessageDlg('Unable to Close Excel', mtError, [mbOK], 0);
End;
End;

```

Gambar 4.16. Listing program untuk melakukan export ke MS.Excel.

Sistem memberikan satu tombol “*Save report*” untuk memanggil fungsi dan *procedure* diatas kemudian melakukan penyimpanan hasil evaluasi ke dalam bentuk excel atau *worksheet*. Tombol “*Save report*” ini diberi *listing* program seperti pada Gambar 4.17.

```

//data hasil
procedure TFormUtama.buka(namafilename:string);
var nm_sheet : string;
    i,j,k : integer;
begin
try
begin
  aktivasi(namafilename);
  //header
  XLS.cells[1,1].value:='AHP RECOMENDER SYSTEM REPORT';
  XLS.cells[2,1].value:='TABEL RULE';
  //column header
  XLS.cells[3,1].value:='ID';
  XLS.cells[3,2].value:='INDIKATOR';
  XLS.cells[3,3].value:='INDIKATOR ASAL';
  XLS.cells[3,4].value:='LEVEL';
  XLS.cells[3,5].value:='EIGEN';
  //content
  dm.TbRule.First;
  i:=4;
  while not dm.TbRule.Eof do
  begin
    XLS.cells[i,1].value:=dm.TbRule.FieldName('id').AsString;
    XLS.cells[i,2].value:=dm.TbRule.FieldName('indikator').AsString;
    XLS.cells[i,3].value:=dm.TbRule.FieldName('indikator_asal').AsString;
    XLS.cells[i,4].value:=dm.TbRule.FieldName('in_level').AsString;
    XLS.cells[i,5].value:=dm.TbRule.FieldName('nilai_eigen').AsString;
    dm.TbRule.Next;
    Inc(i);
  end;
  //global priority
  XLS.cells[i+2,1].value:='GLOBAL PRIORITY';
  i:=i+3;
  k:=1;
  for j := 0 to memo2.Lines.Count do
  begin
    XLS.cells[i,k].value:=memo2.Lines.ValueFromIndex[j];

```

```
if k=1 then
    k:=2
else
begin
    Inc(I);
    k:=1;
end;
end;
XLS.visible:=true;
XLS.ActiveWorkBook.SaveAs(namafile);
XLS := Unassigned;
end;
except
begin
    MessageDlg('Kesalahan File Excel',mterror,[mbok],0);
    Deaktivasi;
end;
end;
end;
```

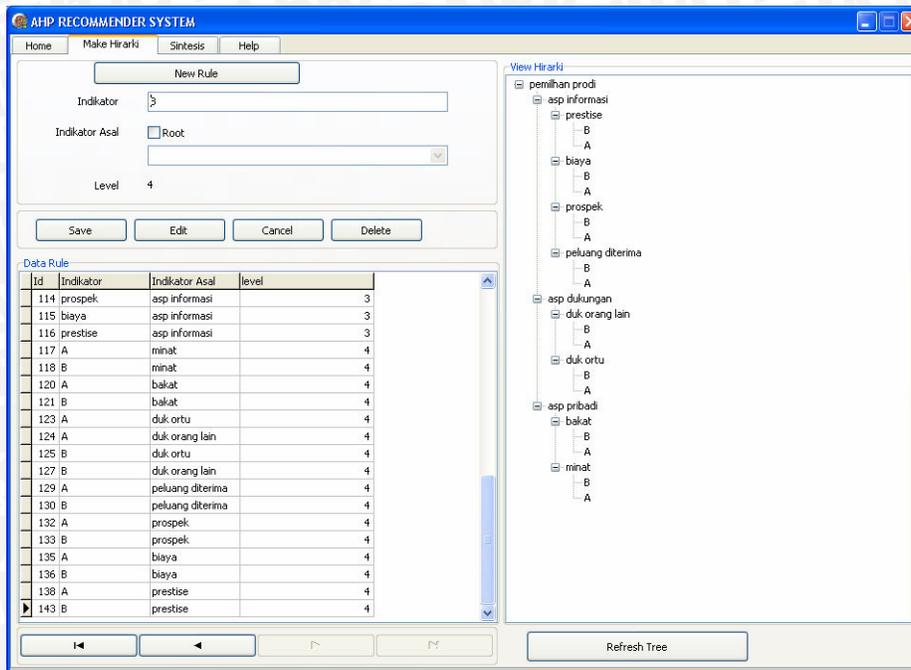
Gambar 4.17. Listing program untuk menyimpan data hasil evaluasi ke dalam bentuk excel.

4.3. Implementasi Antarmuka

4.3.1. Menu Penyusunan Hirarki (Deskripsi Masalah)

Pada form awal terdapat 4 (empat) menu utama, yaitu menu penyusunan hirarki, sintesis, *help*, dan *home*. Dalam menu penyusunan hirarki, *user* akan melakukan pendeskripsian masalah dengan menginputkan *rule* permasalahan dan sistem akan menampilkannya dalam bentuk hirarki. Tampilan form input data *rule* dan penyusunan hirarki dapat dilihat pada Gambar 4.18.





Gambar 4.18. Form menu deskripsi masalah (input data *rule* dan penyusunan hirarki).

4.3.2. Menu Sintesis

Pada tab menu sintesis ini, *user* akan melakukan perbandingan berpasangan menurut level dan indikator asal. Perbandingan berpasangan akan dilakukan terhadap semua indikator pada semua level yang telah didefinisikan pada form menu penyusunan hirarki.

Langkah pertama *user* harus memilih level dan indikator asal pada tombol yang tersedia pada menu sintesis. Proses ini dapat dilihat pada Gambar 4.19.

The screenshot shows the 'AHP RECOMMENDER SYSTEM' interface. The 'View Data Eigen' section contains the following table:

Indikator	Indikator Asal	Level	Eigen Vektor
biaya	asp informasi	3	4140280422935
prestise	asp informasi	3	8712448899401
A	minat	4	6666666666667
B	minat	4	3333333333333
A	bakat	4	0,75
B	bakat	4	0,25
A	duk ortu	4	3333333333333
A	duk orang lain	4	6666666666667
B	duk ortu	4	6666666666667
B	duk orang lain	4	3333333333333
A	peluang diterima	4	0,2
B	peluang diterima	4	0,8
A	prospek	4	0,8
B	prospek	4	0,2
A	biaya	4	6666666666667
B	biaya	4	3333333333333
A	prestise	4	0,75
B	prestise	4	0,25

Gambar 4.19. Form menu sintesis pada proses pemilihan level dan indikator asal.

Dalam memberikan nilai, *user* bisa memberikan nilai 1 sampai 9 atau memberikan nilai kebalikannya sesuai dengan data yang tertera pada skala penilaian, kemudian tekan tombol “*Create Matriks*” untuk mendapatkan nilai eigen dan konsistensi matriks dari perbandingan yang telah dilakukan. Apabila matriks dinyatakan tidak konsisten maka *user* perlu melakukan perbandingan ulang terhadap inputan matriks atau memilih sistem yang melakukan perbaikan (*repair*) terhadap matriks. Proses perbandingan ulang terhadap matriks yang tidak konsisten tadi diperlukan, karena dapat mempengaruhi kualitas proses perhitungan global nantinya. Gambar 4.20 adalah contoh matriks yang telah dinyatakan konsisten dan nilai eigen diperbaharui (di *update*).

AHP RECOMMENDER SYSTEM

Home Make Hirarki Sintesis Help

Pilih Level: 2
Indikator Asal: pemilihan prodi

Matriks Perbandingan Berpasangan

Create Matriks Record Eigen

	asp pribadi	asp dukungan	asp informasi	Nilai Eigen
asp pribadi	1	4	5	0,690202517788725
asp dukungan	0,25	1	1	0,160645867542419
asp informasi	0,2	1	1	0,149151614668056
	1,45	6	7	1

Detail Uji Konsistensi

Lambda: 3,00873015873016 CR: 0,00839438339438337

CI: 0,00436507936507935 Konsisten: -1 true

Skala Penilaian

Skala	Keterangan (nilai relatif elemen baris(A) terhadap elemen puncak(B))
1	A sama pentingnya dengan B
3	A sedikit lebih penting daripada B
5	A lebih penting daripada B
7	A sangat Lebih Penting daripada B
9	A mutlak lebih penting daripada B
(2,4,6,8)	menunjukkan arti diantara skala2 ganjil
(1/3,1/5,1/7,1/9)	B sedikit lebih penting daripada A

View Data Eigen

Indikator	Indikator Asal	Level	Eigen Vektor
biaya	asp informasi	3	4140280422935
prestise	asp informasi	3	8712448899401
A	minat	4	6666666666667
B	minat	4	3333333333333
A	bakat	4	0,75
B	bakat	4	0,25
A	duk ortu	4	3333333333333
A	duk orang lain	4	6666666666667
B	duk ortu	4	6666666666667
B	duk orang lain	4	3333333333333
A	peluang diterima	4	0,2
B	peluang diterima	4	0,8
A	prospek	4	0,8
B	prospek	4	0,2
A	biaya	4	6666666666667
B	biaya	4	3333333333333
A	prestise	4	0,75
B	prestise	4	0,25

Cek Prioritas Global

Data Prioritas Global

Save Report Exit Application

Gambar 4.20. Form menu sintesis pada proses pengecekan konsistensi matriks.

Setelah *user* selesai melakukan perbandingan berpasangan terhadap semua indikator asal pada semua level, maka langkah terakhir adalah melakukan pengecekan atau perhitungan nilai prioritas global. *User* dapat melihat hasil dari proses pengecekan berupa nilai dari masing – masing alternatif yang disintesis. Alternatif dengan nilai global terbesar merupakan alternatif yang direkomendasikan oleh sistem. Gambar 4.21 merupakan contoh hasil dari pengecekan nilai prioritas untuk tiap – tiap alternatif.

Matriks Perbandingan Berpasangan

	asp pribadi	asp dukungan	asp informasi	Nilai Eigen
asp pribadi	1	4	5	0,690202517788725
asp dukungan	0,25	1	1	0,160645867542419
asp informasi	0,2	1	1	0,149151614668856
	1,45	6	7	1

Detail Uji Konsistensi

Lambda: 3,00873015873016 CR: 0,00839438339438337
 CI: 0,00436507936507935 Konsisten: -1 true

Skala Penilaian

Skala	Keterangan (nilai relatif elemen baris(A) terhadap elemen puncak(B))
1	A sama pentingnya dengan B
3	A sedikit lebih penting daripada B
5	A lebih penting daripada B
7	A sangat Lebih Penting daripada B
9	A mutlak lebih penting daripada B
{2,4,6,8}	menunjukkan arti diantara skala2 ganjil
1/2, (0,3)	B sedikit lebih penting daripada A
1/3, (0,33)	B lebih penting daripada A

View Data Eigen

Indikator	Indikator Asal	Level	Eigen Vektor
biaya	asp informasi	3	4140280422935
prestise	asp informasi	3	8712448899401
A	minat	4	6666666666667
B	minat	4	3333333333333
A	bakat	4	0,75
B	bakat	4	0,25
A	duk ortu	4	3333333333333
A	duk orang lain	4	6666666666667
B	duk orang lain	4	6666666666667
B	duk orang lain	4	3333333333333
A	peluang diterima	4	0,2
B	peluang diterima	4	0,8
A	prospek	4	0,8
B	prospek	4	0,2
A	biaya	4	6666666666667
B	biaya	4	3333333333333
A	prestise	4	0,75
B	prestise	4	0,25

Data Prioritas Global

A: 0,695474976505261
 B: 0,304525023494739
 Alternatif yang direkomendasikan: A

Gambar 4.21. Form menu sintesis pada proses pengecekan prioritas global.

Tombol “Save report” pada Gambar 4.21 digunakan untuk menyimpan hasil evaluasi dari user ke dalam bentuk excell atau *worksheet*. Apabila user merasa tidak yakin dengan alternatif yang diberikan oleh sistem, maka user bisa mengulang kembali proses mulai dari perbandingan berpasangan hingga pengecekan nilai prioritas global. Apabila setelah dilakukan sintesis ulang kemudian didapatkan alternatif yang direkomendasikan oleh sistem tidak berbeda secara signifikan dengan proses yang pertama, maka memang itulah alternatif terbaik bagi user.

4.3.3. Menu Help

Langkah pertama - Deskripsi Hirarki

Level 1 tujuan

Level 2 kriteria

Langkah yang harus dilakukan =

Home Make Hirarki Sintesis Help

New Rule

Indikator: tujuan

Indikator Asal: Root

Level: 1

Save Edit Cancel Delete

ID	Indikator	Indikator Asal	level
149	tujuan		1

Input Hirarki :

1. input indikator pada level 1 pada kota indikator, kemudia beri tanda cek pada root (menandakan bahwa indikator tsb, sebagai induk awal dari hirarki)
2. tekan tombol save.
3. tekan tombol new rule untuk membuat node baru.
4. untuk indikator pada level selanjutnya, masukkan indikator pada kotak indikator dan indikator asal bisa dilihat pada kotak lookup combobox indikator asal.
5. tekan tombol save.
6. lakukan hal yang sama untuk semua indikator pada hirarki yang ingin dibuat.
7. untuk melihat hasil dari hirarki yang terbentuk bisa dilihat pada kolom sebelah kanan form make hirarki.
8. Gunakan tombol edit, atau delete untuk mengubah rule yang ada di hirarki

Langkah kedua - Sintesis

1. pilih tab "sintesis" - kemudian pilih level dan indikator asal

Home Make Hirarki Sintesis Help

Pilih Level: 2

Indikator Asal: tujuan

2. akan terbentuk matriks - kemudian lakukan perbandingan berpasangan dengan memberi nilai 0 - 9

Matriks Perbandingan Berpasangan

	kriteria1	kriteria2	kriteria3	Nilai Eigen
kriteria1	1	3	2	0,540404040404
kriteria2	0,3333333333	1	1	0,210606060606
kriteria3	0,5	1	1	0,240909090909
	1,8333333333	5	4	1

3. tekan tombol "create matriks" untuk melakukan pengecekan konsistensi matriks dan didapat hasil seperti :

Detail Uji Konsistensi

Lambda: 3,02222222222222 CR: 0,0213675213675213

CI: 0,0111111111111111 Konsisten: true

Keterangan :

- lakukan perbandingan berpasangan untuk semua elemen pada semua level.
- tekan tombol "record eigen" untuk merecord prioritas lokal masing2 indikator pada perbandingan matriks yang konsisten.

4. tekan tombol "cek global priority" untuk mendapat alternatif yang memiliki nilai prioritas paling tinggi

Cek Prioritas Global	Data Prioritas Global
	alt1
	0,775811688311688
	alt2
	0,224188311688312
	Alternatif yang direkomendasikan
	alt1

Gambar 4.22. Form menu sintesis pada proses pengecekan prioritas global.

Dalam tab menu *help*, berisi petunjuk atau langkah – langkah penggunaan sistem. Dalam tab menu *help* ini dijelaskan cara mendeskripsikan hirarki, cara melakukan perbandingan berpasangan, pengecekan konsistensi matriks, perhitungan prioritas lokal hingga pada pengecekan prioritas global.

4.4. Pengujian Sistem

Pengujian sistem dilakukan untuk menguji coba sistem dalam menyelesaikan masalah dari studi kasus yang diteliti berdasarkan data inputan dari *user*.

4.4.1. Studi Kasus

Pengujian awal dilakukan dengan memasukkan *rule* pembentuk hirarki dari studi kasus yang diteliti yaitu kasus pemilihan program studi pada perguruan tinggi. Pengujian dilakukan dengan menggunakan 2 alternatif dan 3 alternatif program studi, karena pada umumnya *user* hanya memiliki pilihan dengan menggunakan 2 alternatif atau 3 alternatif program studi pada saat mengikuti test masuk perguruan tinggi.

Rule (aturan) pembentuk hirarki analisa permasalahan pemilihan program studi dengan menggunakan 2 alternatif dan 3 alternatif dijelaskan pada Tabel 4.1 dan Tabel 4.2.

Tabel 4.1. Tabel Input Data *Rule* Hirarki Permasalahan dengan 3 Alternatif Program Studi (Prodi).

ID	Indikator	Indikator Asal	Level
1	Pemilihan prodi		1
2	Asp.pribadi	Pemilihan prodi	2
3	Asp.dukungan	Pemilihan prodi	2
4	Asp.informasi	Pemilihan prodi	2
5	Minat	Asp.pribadi	3
6	Bakat	Asp.pribadi	3
7	Nilai try out	Asp.pribadi	3
8	Nilai raport	Asp.pribadi	3
9	Duk.orang tua	Asp.dukungan	3
10	Duk.orang lain	Asp.dukungan	3
11	Daya tampung	Asp.informasi	3
12	Prestise jurusan	Asp.informasi	3
13	Prospek jurusan	Asp.informasi	3
14	biaya	asp.informasi	3
15	Prodi A	Minat	4
16	Prodi A	Bakat	4
17	Prodi A	Duk.orang tua	4
18	Prodi A	Duk.orang lain	4
19	Prodi A	Daya tampung	4
20	Prodi A	Prestise jurusan	4

21	Prodi A	Prospek jurusan	4
22	Prodi A	Biaya	4
23	Prodi B	Minat	4
24	Prodi B	Bakat	4
25	Prodi B	Duk.orang tua	4
26	Prodi B	Duk.orang lain	4
27	Prodi B	Daya tampung	4
28	Prodi B	Prestise jurusan	4
29	Prodi B	Prospek jurusan	4
30	Prodi B	Biaya	4
31	Prodi C	Minat	4
32	Prodi C	Bakat	4
33	Prodi C	Duk.orang tua	4
34	Prodi C	Duk.orang lain	4
35	Prodi C	Daya tampung	4
36	Prodi C	Prestise jurusan	4
37	Prodi C	Prospek jurusan	4
38	Prodi C	Biaya	4

Tabel 4.2. Tabel Input Data *Rule* Hirarki Permasalahan dengan 2 Alternatif Program Studi (Prodi).

ID	Indikator	Indikator Asal	Level
1	Pemilihan prodi		1
2	Asp.pribadi	Pemilihan prodi	2
3	Asp.dukungan	Pemilihan prodi	2
4	Asp.informasi	Pemilihan prodi	2
5	Minat	Asp.pribadi	3
6	Bakat	Asp.pribadi	3
7	Nilai try out	Asp.pribadi	3
8	Nilai raport	Asp.pribadi	3
9	Duk.orang tua	Asp.dukungan	3
10	Duk.orang lain	Asp.dukungan	3
11	Daya tampung	Asp.informasi	3
12	Prestise jurusan	Asp.informasi	3
13	Prospek jurusan	Asp.informasi	3
14	Biaya	Asp.informasi	3
15	Prodi A	Minat	4
16	Prodi A	Bakat	4

17	Prodi A	Duk.orang tua	4
18	Prodi A	Duk.orang lain	4
19	Prodi A	Daya tampung	4
20	Prodi A	Prestise jurusan	4
21	Prodi A	Prospek jurusan	4
22	Prodi A	Biaya	4
23	Prodi B	Minat	4
24	Prodi B	Bakat	4
25	Prodi B	Duk.orang tua	4
26	Prodi B	Duk.orang lain	4
27	Prodi B	Daya tampung	4
28	Prodi B	Prestise jurusan	4
29	Prodi B	Prospek jurusan	4
30	Prodi B	Biaya	4

Jumlah keseluruhan populasi yaitu mahasiswa S1 Universitas Brawijaya angkatan 2007 yang masih aktif pada semester genap tahun 2007/2008 adalah sebanyak 5563 mahasiswa. Dari survei pendahuluan yang telah dilakukan diketahui proporsi sukses mahasiswa dalam pengujian sistem sebesar 80%. Hasil pengujian sistem pada survei pendahuluan dapat dilihat pada Lampiran 2. Berdasarkan pada persamaan (2.7), dengan tingkat keyakinan 95% dan *sampling error* 10% maka ukuran sampel yang digunakan sebesar :

$$n = \frac{(1.96)^2 \cdot 0.8 \cdot 0.2}{(0.1)^2} \cdot \frac{1}{1 + \frac{1}{5563} \left(\frac{(1.96)^2 \cdot 0.8 \cdot 0.2}{(0.1)^2} - 1 \right)}$$

$$n = 111,4898$$

$$n = 112$$

Jadi jumlah minimum sampel yang diambil dari seluruh populasi adalah sebesar 112 responden.

Jumlah mahasiswa sebagai responden ditentukan dengan memberikan proporsi yang sama untuk tiap program studi. Dengan demikian jumlah responden untuk tiap – tiap program studi adalah sebanyak 3 responden. Hasil pengujian sistem selengkapnya dapat dilihat pada Lampiran 3.

4.4.2. Analisa Hasil

4.4.2.1. Perhitungan Tingkat Kesesuaian Hasil Pengujian Sistem

Pengujian dilakukan dengan mencocokkan hasil alternatif berupa program studi yang dihasilkan sistem dengan hasil riil dari responden (prodi responden saat melakukan pengujian sistem) baik dari pengujian yang menggunakan 2 alternatif maupun 3 alternatif program studi. Dari hasil pengujian diperoleh data kesesuaian antara hasil yang diberikan oleh sistem seperti pada Tabel 4.3.

Tabel 4.3. Data Hasil Pengujian Sistem

Hasil yang diberikan sistem	Jumlah	Prosentase (%)
Sesuai dengan alternatif riil	94	66%
Tidak sesuai dengan alternatif riil	40	28%
Sampel <i>error</i>	10	6%

Dari Tabel 4.3 dapat diketahui data hasil yang sesuai dengan alternatif riil dimaksudkan bahwa nilai prioritas global dari alternatif riil responden memiliki nilai global paling tinggi diantara alternatif lain yang dievaluasi. Data hasil yang tidak sesuai dengan alternatif riil dimaksudkan bahwa alternatif riil dari responden ternyata memiliki nilai yang cukup kecil dibandingkan dengan alternatif lain yang dievaluasi. Sampel *error* dimaksudkan bahwa responden dalam menginputkan alternatif yang akan dievaluasi atau diujicobakan ke sistem tidak menginputkan prodi yang sedang dijalani saat ini, sehingga hasil yang diberikan oleh sistem tidak bisa dibandingkan dengan hasil riil atau prodi dari responden saat pengujian.

Dari data hasil yang tidak sesuai dengan hasil yang diberikan oleh sistem dapat diketahui beberapa alasannya diantaranya dikarenakan responden mengalami sedikit kesulitan pada penggunaan skala penilaian (0-9) dalam memberikan nilai perbandingan pada proses perbandingan berpasangan, responden dalam memberikan data penilaian tidak melakukannya dengan sungguh – sungguh atau memberikan penilaian secara acak, banyaknya inputan nilai yang harus diisikan oleh *user* pada saat melakukan perbandingan berpasangan, dan responden secara jujur memang tidak menginginkan prodi yang sedang dijalani.

4.4.2.2. Analisa Prioritas Global

Dari data hasil pengujian sistem pada Tabel 4.3 dapat dianalisa nilai prioritas global dari tiap – tiap alternatif riil responden baik yang menggunakan 2 alternatif maupun 3 alternatif program studi dan dari hasil yang sesuai maupun tidak sesuai dengan yang rekomendasikan oleh sistem.

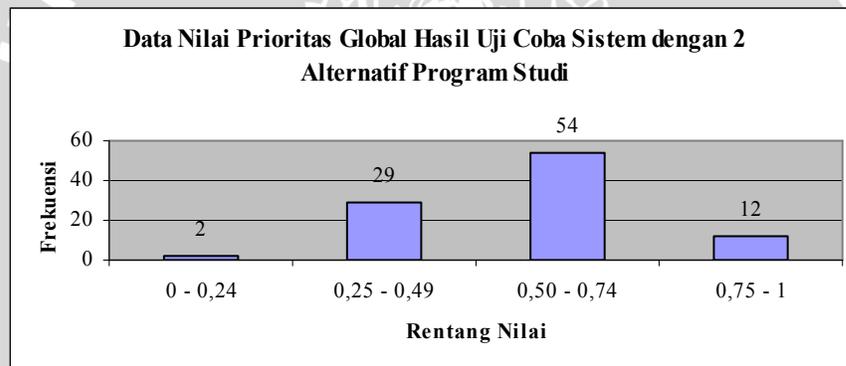
a. Pengujian dengan Menggunakan 2 Alternatif Program Studi

Dari hasil pengujian sistem dapat diketahui prioritas global dari alternatif (program studi) riil dari responden yang menggunakan 2 alternatif program studi seperti dijelaskan pada Tabel 4.4.

Tabel 4.4. Data Nilai Prioritas Global Hasil Uji Coba Sistem dengan 2 Alternatif Program Studi.

No	Rentang Nilai Prioritas Global	Frekuensi
1	0 - 0,24	2
2	0,25 - 0,49	29
3	0,50 - 0,74	54
4	0,75 - 1	12
	Total	97

Dari Tabel 4.5 dapat dibuat grafik seperti digambarkan pada Gambar 4.23.



Gambar 4.23. Data nilai prioritas global hasil uji coba sistem dengan 2 alternatif program studi.

Nilai prioritas global dari alternatif yang dievaluasi memiliki *range* antara 0 – 1. Dengan menggunakan 2 alternatif pemecahan masalah dapat

diketahui bahwa rentang nilai prioritas global dari suatu alternatif yang direkomendasikan oleh sistem berada pada rentang nilai 0.5 – 1 dan sebaliknya alternatif yang memiliki prioritas global pada rentang nilai 0 – 0.49 tidak direkomendasikan oleh sistem sebagai alternatif terbaik.

Dari Gambar 4.23 dan Tabel 4.4 dapat diketahui bahwa rentang nilai prioritas global antara 0 – 0.49 merupakan nilai prioritas global alternatif prodi riil dari responden yang tidak sesuai dengan hasil yang direkomendasikan oleh sistem. Sedangkan rentang nilai 0.50 – 1 merupakan rentang nilai prioritas global alternatif prodi riil responden yang sesuai dengan hasil yang direkomendasikan oleh sistem.

Dari pengujian dengan menggunakan 2 alternatif program studi dapat diketahui bahwa mayoritas responden memiliki nilai prioritas global dari alternatif riil diatas 0.5, sehingga hasil yang diberikan oleh sistem adalah sesuai dengan alternatif riil responden. Nilai prioritas global alternatif prodi riil yang berada para rentang nilai di bawah 0.5 atau tidak sesuai dengan hasil yang diberikan oleh sistem hanya memiliki frekuensi (jumlah) yang kecil.

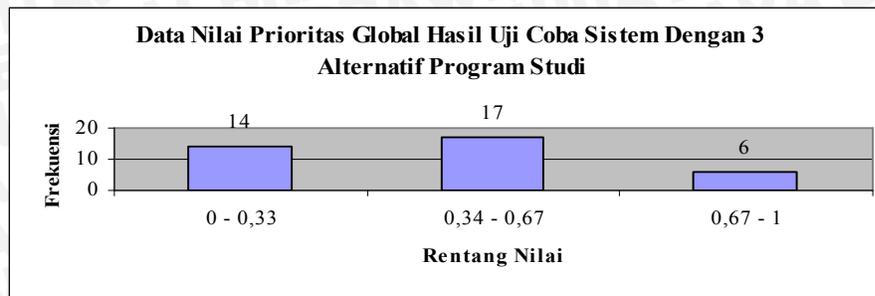
b. Pengujian dengan Menggunakan 3 Alternatif Program Studi

Dari hasil pengujian sistem dapat diketahui prioritas global dari alternatif (program studi) riil dari responden yang menggunakan 3 alternatif program studi seperti dijelaskan pada Tabel 4.5.

Tabel 4.5. Data Nilai Prioritas Global Hasil Uji Coba Sistem dengan 3 Alternatif Program Studi.

No	Rentang Nilai Prioritas Global	Frekuensi
1	0 - 0,33	14
2	0,34 - 0,67	17
3	0,67 - 1	6
	Total	37

Dari Tabel 4.5 dapat dibuat grafik seperti digambarkan pada Gambar 4.24.



Gambar 4.24. Data nilai prioritas global hasil uji coba sistem dengan 3 alternatif program studi.

Dari hasil pengujian dengan menggunakan 3 alternatif pemecahan masalah dapat diketahui bahwa rentang nilai prioritas global dari suatu alternatif yang direkomendasikan oleh sistem berada pada rentang nilai 0.34 – 1 dan sebaliknya alternatif yang memiliki prioritas global pada rentang nilai 0 – 0.33 tidak direkomendasikan oleh sistem sebagai alternatif terbaik.

Dari Gambar 4.24 dan Tabel 4.5 dapat diketahui bahwa rentang nilai antara 0 – 0.33 adalah rentang nilai prioritas global dari alternatif prodi riil responden yang tidak sesuai dengan hasil yang direkomendasikan oleh sistem. Rentang nilai prioritas global antara 0.34 – 1 adalah rentang nilai prioritas global dari alternatif prodi riil responden yang sesuai dengan hasil yang direkomendasikan oleh sistem.

Dari hasil pengujian sistem dengan menggunakan 3 alternatif program studi dapat diketahui bahwa mayoritas responden memiliki nilai prioritas global alternatif prodi riil yang cukup tinggi yaitu diatas 0.33 sehingga hasil yang diberikan oleh sistem adalah sesuai dengan alternatif riil responden. Nilai prioritas global dari alternatif riil responden yang berada pada rentang nilai di bawah 0.33 atau tidak sesuai dengan hasil yang direkomendasikan oleh sistem hanya memiliki frekuensi yang cukup kecil.

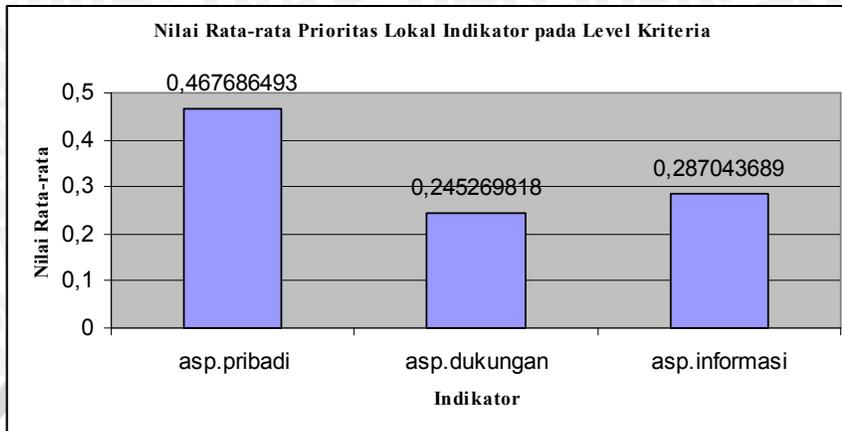
4.4.2.3. Analisa Prioritas Lokal Pada Level Kriteria (Level 2)

Dari hasil pengujian juga bisa dicari nilai rata-rata prioritas lokal dari masing-masing indikator dari kasus pemilihan program studi pada perguruan tinggi. Nilai rata – rata priortias lokal ini menunjukkan tingkat prioritas (kepentingan) dari masing-masing indikator menurut preferensi dan pengalaman pribadi responden. Nilai rata – rata prioritas lokal dari indikator pada Level 2 (level kriteria) dapat dilihat pada Tabel 4.6.

Tabel 4.6. Nilai Rata-rata Indikator Pada Level 2 (Kriteria)

Indikator	Rata-Rata Prioritas Lokal
asp.pribadi	0,467686493
asp.dukungan	0,245269818
asp.informasi	0,287043689

Dari Tabel 4.6 dapat dibuat grafik seperti digambarkan pada Gambar 4.25.



Gambar 4.25. Nilai rata – rata prioritas lokal indikator pada level kriteria.

Dari Tabel 4.6 dan Gambar 4.25 dapat diketahui bahwa dari indikator pada level kriteria, aspek pribadi memiliki nilai rata – rata paling tinggi sehingga aspek pribadi lebih diprioritaskan oleh responden. Menyusul peringkat kedua adalah aspek informasi dan peringkat ketiga adalah aspek dukungan.

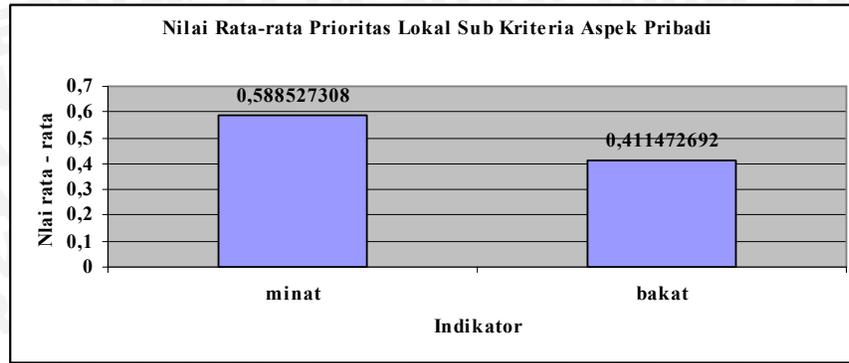
4.4.2.4. Analisa hasil prioritas lokal pada level sub kriteria (Level 3)

Nilai rata – rata prioritas lokal pada level sub kriteria baik sub kriteria aspek pribadi, aspek dukungan ataupun aspek informasi dapat dijelaskan pada Tabel 4.7, Tabel 4.8, dan Tabel 4.9.

Tabel 4.7. Nilai Rata-rata Prioritas Lokal Indikator Pada Level 3 Sub Kriteria Aspek Pribadi.

Indikator	Rata-rata Prioritas Lokal
Minat	0,588527308
Bakat	0,411472692

Dari Tabel 4.7 dapat dibuat grafik seperti digambarkan pada Gambar 4.26.



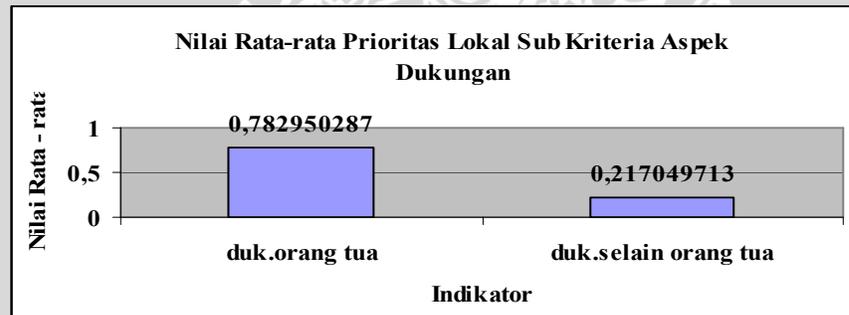
Gambar 4.26. Nilai rata – rata prioritas lokal indikator pada sub kriteria aspek pribadi.

Dari Tabel 4.7 dan Gambar 4.26 dapat diketahui bahwa dari indikator pada sub kriteria aspek pribadi, ternyata indikator minat memiliki rata – rata prioritas lokal paling tinggi, sehingga indikator minat paling diprioritaskan oleh responden dibandingkan dengan bakat.

Tabel 4.8. Nilai Rata – rata Prioritas Lokal Indikator Pada Level 3 Sub Kriteria Aspek Dukungan.

Indikator	Rata-rata Prioritas Lokal
Duk.orang tua	0,782950287
Duk.selain orang tua	0,217049713

Dari Tabel 4.8 dapat dibuat grafik seperti digambarkan pada Gambar 4.27.



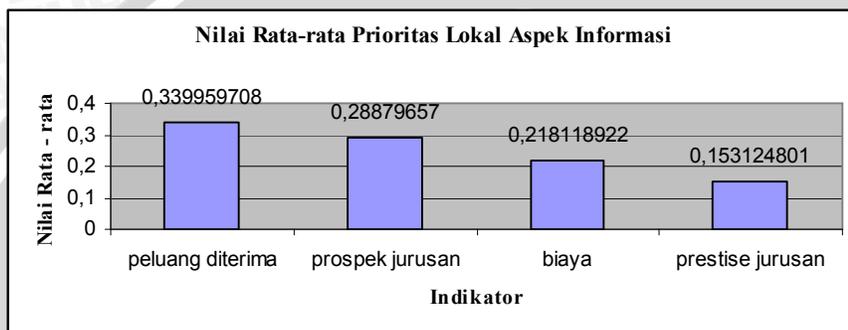
Gambar 4.27. Nilai rata – rata indikator pada sub kriteria aspek dukungan.

Dari Tabel 4.8 dan Gambar 4.27 dapat diketahui bahwa dari indikator pada sub kriteria aspek dukungan, ternyata indikator dukungan orang tua memiliki nilai rata - rata prioritas lokal paling tinggi sehingga indikator dukungan orang tua lebih diprioritaskan dari pada dukungan selain orang tua.

Tabel 4.9. Nilai Rata – rata Prioritas Lokal Indikator Pada Level 3 Sub Kriteria Aspek Informasi.

Indikator	Rata-rata Prioritas Lokal
Peluang diterima	0,339959708
Prospek jurusan	0,28879657
Biaya	0,218118922
Prestise jurusan	0,153124801

Dari Tabel 4.9 dapat dibuat grafik seperti digambarkan pada Gambar 4.28.



Gambar 4.28. Nilai rata – rata prioritas lokal indikator pada sub kriteria aspek informasi.

Dari Tabel 4.9 dan Gambar 4.28 dapat diketahui bahwa dari indikator pada sub kriteria aspek informasi, ternyata peluang diterima memiliki nilai prioritas paling tinggi. dan menyusul di peringkat kedua adalah indikator prospek ke depan dari jurusan, indikator biaya dan prestise jurusan (program studi) memiliki tingkat prioritas paling rendah atau tidak memiliki pengaruh yang cukup signifikan dalam proses pemilihan program studi.

BAB V KESIMPULAN DAN SARAN

5.1. Kesimpulan

Kesimpulan yang dapat diambil dari penelitian ini adalah :

1. Metode *Analythic Hierarchy Process* (AHP) dapat digunakan dalam penentuan program studi/jurusan pada perguruan tinggi.
2. Mayoritas alternatif yang dihasilkan oleh sistem pada waktu pengujian adalah sesuai dengan hasil alternatif (program studi) yang dijalani responden pada saat pengujian dilakukan dengan prosentase tingkat kesesuaian sebesar 65% dari 144 responden.
3. Dari hasil penelitian diketahui bahwa dalam pemilihan program studi di perguruan tinggi ternyata kriteria yang paling mempengaruhi atau diprioritaskan adalah aspek pribadi yaitu dengan memperhatikan minat, diikuti pada peringkat kedua adalah aspek informasi yaitu dengan memperhatikan peluang diterima suatu jurusan/prodi kemudian di peringkat ketiga adalah aspek dukungan dengan memperhatikan dukungan orang tua.

5.2. Saran

Saran yang dapat diberikan untuk pengembangan sistem rekomendasi menggunakan metode AHP lebih lanjut adalah:

1. penggunaan skala penilaian yang lebih memudahkan *user*. Dalam hal pengembangan lebih lanjut bisa dikembangkan dengan menggunakan skala penilaian yaitu skala likert (1-5), karena skala likert lebih memudahkan *user* dalam memberikan penilaian dan lebih umum digunakan walaupun agak rumit dalam proses perhitungan prioritas global.
2. Perancangan antar muka (*user interface*) yang lebih *user friendly* atau lebih memudahkan *user* dalam melakukan perbandingan berpasangan.
3. Sebelum melakukan pengujian sistem terhadap responden, perlu dilakukan suatu analisis sensitivitas terhadap kriteria – kriteria (semua elemen) yang mempengaruhi permasalahan yang diteliti.



DAFTAR PUSTAKA

- Alam, M. Agus, J. 2001. *Belajar Sendiri Borland Delphi 7.0*. ElekMedia Komputindo Kelompok Gramedia, Jakarta.
- Anton, H. 1995. *Aljabar Linier Elementer*. Edisi ke-5. Alih bahasa : Silaban,P dan Susila, N. Erlangga. Jakarta.
- Cochran, W. G. 1991. *Teknik Penarikan Sampel*. Penerjemah : Rudiansyah. Universitas Indonesia Press, Jakarta.
- Ekawati, H. 2006. *Penentuan Prioritas Mahasiswa dalam Memilih Simcard Prabayar GSM Menggunakan AHP (Analytic Hierarchy process) (Studi Kasus di Universitas Brawijaya)*. Skripsi S1. Fakultas MIPA Universitas Brawijaya. Malang. Tidak dipublikasikan.
- Fathansyah, Ir. 1999. *Basis Data*. CV. Informatika, Bandung
- Gay, L.R. 1976. *Educational Research : Competencies for Analysis and Application*. Bell & Howell Company, Ohio.
- Hameed. 2007. *Tips dan Cara Memilih Jurusan Kuliah Di Perguruan Tinggi Yang Baik*.
<http://hameedfinder.blogspot.com/2007/08/tips-cara-memilih-jurusan-kuliah-di.html>. Tanggal Akses 22 Maret 2008.
- Han, Eui-Hong. dan G. Karypis. 2005. *Feature Based Recommendation System*. Internet: <http://glaros.dtc.umn.edu/gkhome/fetch/papers/fbrsCIKM05.pdf>. Tanggal akses 10 Mei 2008.
- Haruechaiyasak, C., Mei-Ling Shyu, dan Shu-Ching Chen. 2004. *A Data Mining Framework for Building A Web-Page Recommender System*. Internet: <http://www.eng.miami.edu/~shyu/Paper/2004/iri04-rec.pdf>. Tanggal akses 10 Mei 2008.
- <http://faculty.petra.ac.id>, 2006. Algoritma dan Flowchart. Tanggal akses : 10 Juni 2008.

http://lecturer.eepis-its.edu. 2007. *Algoritma dan Flowchart: Dasar Pemrograman dan Algoritma*. Tanggal akses: 10 Juni 2008.

Jogiyanto, HM. 1988. *Analisa dan Desain Sistem Informasi*. Andi Offset, Yogyakarta.

Jogiyanto, HM. 2001. *Analisa dan Desain Sistem Informasi*. PT Andi Offset, Yogyakarta

Latifah, S. 2005. *Prinsip – prinsip Dasar AHP (Analytic Hierarchy process)*. E-USU Repository, Sumatera Utara.

Lydia, Anastasia. dan Dewi, Dian.RS. *Penelitian Memetics Pada Pengumpulan Informasi Untuk Mendukung Keputusan Pemilihan Jurusan / Perguruan Tinggi Bagi Seorang Calon Mahasiswa Dari Smu-Surabaya*. <http://www.mail-archive.com/its@pandawa.com/index.html>. Tanggal Akses 22 Maret 2008.

Madcoms. 2003. *Seri Panduan Pemrograman Borland Delphi 7.0 (Jilid 2)*. ANDI, Yogyakarta.

Maholtra, N. K. 2002. *Basic Marketing Research Applications to Contemporary Issues*. Prentice Hall, New Jersey.

Noviyanti, D. 2004. *Penentuan Prioritas Stasiun Radio pilihan Mahasiswa dengan AHP (Analytic Hierarchy process) (Studi Kasus di F.MIPA Universitas Brawijaya)*. Skripsi S1. Fakultas MIPA Universitas Brawijaya. Malang. Tidak dipublikasikan.

Purnawinandar, A. 2007. *Sistem Rekomendasi Jenis Asuransi Menggunakan Metode Klasifikasi Naïve Bayes Classifier*. Skripsi S1. Fakultas MIPA Universitas Brawijaya. Malang. Tidak dipublikasikan.

Saaty, T. L. 1993. *Pengambilan Keputusan bagi para Pemimpin*. Cetakan kedua. Penerjemah : Setiono, L. Gramedia, Jakarta.

Singarimbun, m. dan S. Effendi. 1995. *Metode Penelitian Survei*. Lembaga Penelitian dan Penerangan Ekonomi dan Sosial, Jakarta.

Supranto, J. 1992. *Teknik Sampling untuk Survei dan Eksperimen*. Cetakan pertama. Rineka Cipta, Jakarta.

Yitnosumarno, S. 1994. *Dasar – dasar Statistika dengan Penekanan Terapan dalam Bidang Agrokompleks, Teknologi dab Sosial*. Rajagrafindo Persada, Jakarta.





Lampiran 1. Program Studi yang Diteliti.

NO	PROGRAM STUDI
1	PS Ilmu Hukum
2	PS Ilmu Ekonomi dan Studi Pembangunan
3	PS Manajemen
4	PS Akuntansi
5	PS Ilmu Administrasi Publik
6	PS Ilmu Administrasi Bisnis
7	PS Agronomi
8	PS Hortikultura
9	PS Pemuliaan Tanaman
10	PS Ilmu Tanah
11	PS Agrobisnis
12	PS Penyuluhan dan Komunikasi Pertanian
13	PS Ilmu Hama dan Penyakit Tumbuhan
14	PS Nutrisi dan Makanan Ternak
15	PS Produksi Ternak
16	PS Sosial Ekonomi Peternakan
17	PS Teknologi Hasil Ternak
18	PS Teknik Sipil
19	PS Teknik Mesin
20	PS Teknik Industri
21	PS Teknik Pengairan
22	PS Teknik Elektro
23	PS Teknik Perangkat Lunak
24	PS Teknik Arsitektur
25	PS Perencanaan Wilayah dan Kota
26	PS Pendidikan Dokter
27	PS Ilmu Keperawatan
28	PS Ilmu Gizi
29	PS Pemanfaatan Sumberdaya Perikanan
30	PS Teknologi Hasil Perikanan
31	PS Sosial Ekonomi Perikanan
32	PS Budidaya Perairan
33	PS Manajemen Sumberdaya Perairan
34	PS Kimia
35	PS Fisika

Lanjutan Lampiran 1.

NO	PROGRAM STUDI
36	PS Biologi
37	PS Ilmu Komputer
38	PS Matematika
39	PS Statistika
40	PS Teknologi Industri Pertanian
41	PS Teknologi Hasil Pertanian
42	PS Teknik Pertanian
43	PS Sosiologi
44	PS Ilmu Komunikasi
45	PS Psikologi
46	PS Hubungan Internasional
47	PS Sastra Inggris
48	PS Sastra Jepang

UNIVERSITAS BRAWIJAYA



