

**SISTEM PAKAR BERBASIS ATURAN (*RULE BASED*)
UNTUK REKOMENDASI PROGRAM STUDI
BAGI CALON MAHASISWA**

SKRIPSI

Oleh :
WAHYU DIAN WULANDINI
0310963040-96



**PROGRAM STUDI ILMU KOMPUTER
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2008**

**SISTEM PAKAR BERBASIS ATURAN (*RULE BASED*)
UNTUK REKOMENDASI PROGRAM STUDI
BAGI CALON MAHASISWA**

SKRIPSI

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

Oleh :

WAHYU DIAN WULANDINI

0310963040-96



**PROGRAM STUDI ILMU KOMPUTER
JURUSAN MATEMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS BRAWIJAYA
MALANG
2008**

LEMBAR PENGESAHAN SKRIPSI

**SISTEM PAKAR BERBASIS ATURAN (*RULE BASED*)
UNTUK REKOMENDASI PROGRAM STUDI
BAGI CALON MAHASISWA**

Oleh :

WAHYU DIAN WULANDINI
0310963040-96

Setelah dipertahankan di depan Majelis Penguji
pada tanggal 14 Januari 2008
dan dinyatakan memenuhi syarat untuk memperoleh gelar
Sarjana Komputer dalam bidang Ilmu Komputer

Pembimbing I,

Pembimbing II,

Dian Eka R, S.Si., M.Kom
NIP. 132 300 244

Bayu Rahayudi, ST., MT
NIP. 132 318 424

Mengetahui,
a.n Ketua Jurusan Matematika
Fakultas MIPA Universitas Brawijaya
Sekretaris,

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

LEMBAR PERNYATAAN

Saya yang bertanda tangan di bawah ini :

Nama : Wahyu Dian Wulandini
NIM : 0310963040-96
Jurusan : Matematika
Program Studi : Ilmu Komputer
Penulis tugas akhir berjudul : Sistem Pakar Berbasis Aturan
(*rule Based*) untuk Rekomendasi
Program Studi bagi Calon
Mahasiswa

Dengan ini menyatakan bahwa :

1. Isi dari tugas akhir 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 Tugas Akhir ini.
2. Apabila dikemudian hari ternyata Tugas Akhir 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, 14 Januari 2008
Yang menyatakan,

Wahyu Dian Wulandini
NIM. 0310963040-96

SISTEM PAKAR BERBASIS ATURAN (*RULE BASED*) UNTUK REKOMENDASI PROGRAM STUDI BAGI CALON MAHASISWA

ABSTRAK

Dalam dunia pendidikan saat ini, Universitas / Perguruan Tinggi menawarkan banyak pilihan program studi. Di sisi lain, banyak siswa sekolah menengah atas kurang mengerti dan memahami karakteristik dari masing-masing program studi yang ada di Perguruan Tinggi. Hal tersebut menyebabkan siswa tersebut bingung dan terkadang salah dalam memilih program studi yang sesuai dengan minat dan kemampuan yang dimilikinya. Dengan adanya sistem pakar ini, diharapkan dapat membantu dan mengarahkan siswa sekolah (calon mahasiswa) untuk memilih program studi yang sesuai dengan minat dan kemampuan yang dimilikinya tanpa mengecilkan peran Bimbingan dan Konseling Sekolah.

Tujuan pembuatan Skripsi ini adalah mengklasifikasikan karakteristik setiap program studi dengan menggunakan sistem pakar berbasis aturan dan menentukan program studi berdasarkan fakta – fakta yang ada, serta mengimplementasikan sistem pakar berbasis aturan untuk rekomendasi program studi agar dapat mempermudah dan membantu calon mahasiswa.

Sistem pakar ini dibangun dengan menggunakan Kaidah Produksi sebagai representasi pengetahuan dengan metode inferensi *Forward Chaining* dan *Backward Chaining* yang digunakan secara terpisah. *Forward Chaining* digunakan jika calon mahasiswa belum memiliki gambaran tentang program studi yang diminatinya, sedangkan *Backward Chaining* digunakan jika calon mahasiswa sudah memiliki / mengetahui apa program studi yang diminatinya. Dalam hal ini, siswa hanya ingin mengetahui seberapa besar tingkat kecocokan dirinya dengan jurusan yang diminatinya. Metode akuisisi yang digunakan adalah metode manual (wawancara). Selain itu, metode *certainty factor* juga digunakan agar hasil rekomendasi yang diberikan sistem lebih akurat.

Selain itu juga dilakukan permintaan pendapat terhadap siswa kelas XII SMAN 2 Pasuruan tentang sistem yang dibuat, diperoleh kesimpulan bahwa mayoritas siswa menganggap sistem pakar ini cukup membantu dan mempermudah siswa dalam memilih program studi.

RULE BASED EXPERT SYSTEM STUDY PROGRAM RECOMMENDATION FOR UNDERGRADUATED STUDENT'S

ABSTRACT

In the world of present education, a lot of university/higher education institution offers so many study programs. Contrarily, a lot of senior high school students who are attending to be under graduated students, doesn't really understand what those study programs studies about. This kind of matter certainly often brings the students into confusion and sometimes leads them to the wrong choice of study program that appropriate with their interests and the abilities they have. By using this expert system, it is expected to help students in choosing study program which appropriate with their interests and abilities, without underestimating the role of school counseling.

The aim of this research is to classify every each study program characteristics using rule based expert system and to decide the study program based on available facts, and also to implement rule based expert system for recommending study program so that able to help undergraduate students.

This expert system is constructed using Production Rules as the knowledge representation with forward chaining and backward chaining inference method that used separately. Forward chaining method is used, in case that the students has not yet had any idea about the study program they are about to choose. While the backward chaining method is used in case that the students has known the study program they are willing to choose. For the second case, it is assumed that the students only want to know how suitable the study program they'll choose with their ability. The method used for aquisition is manual method (interview). Beside that, it is using certainty factor method in order to obtain more accurate recommendation result.

Beside those methods mentioned above, we also asked the third grade students of Senior High School 2 Pasuruan, about the system. And it is concluded that most of students agree that this system is helpful enough for the students in choosing study programs.

KATA PENGANTAR

Puji syukur penulis haturkan kehadiran Allah SWT yang telah memberikan rahmatNya sehingga penulis dapat menyelesaikan penulisan tugas akhir yang berjudul “Sistem Pakar berbasis aturan (*rule based*) untuk rekomendasi program studi bagi calon mahasiswa”.

Dalam penyusunan tugas akhir ini, Penulis mengucapkan terima kasih kepada :

1. Ibu Dian Eka Ratnawati, S.Si., M.Kom dan Bapak Bayu Rahayudi, ST., MT selaku pembimbing utama dan pembimbing pendamping dalam penulisan tugas akhir ini.
2. Bapak Dr. Agus Suryanto, M.Sc selaku Ketua Jurusan Matematika Fakultas MIPA Universitas Brawijaya
3. Bapak Wayan Firdaus Mahmudy, S.Si., M.Kom, selaku Ketua Program Studi Ilmu Komputer, Jurusan Matematika, FMIPA Universitas Brawijaya.
4. Bapak Drs. Muh. Arif Rahman, M.Kom, selaku dosen pembimbing akademis.
5. Segenap bapak dan ibu dosen yang telah mendidik dan mengajarkan ilmunya kepada Penulis selama menempuh pendidikan di Program Studi Ilmu Komputer
6. Segenap staf dan karyawan Jurusan Matematika Universitas Brawijaya yang telah membantu penyusunan tugas akhir ini.
7. Dan semua pihak yang telah membantu dalam penyusunan tugas akhir ini yang tidak dapat disebutkan satu per satu.

Penulis sadari bahwa masih banyak kekurangan dalam penulisan tugas akhir ini yang disebabkan oleh keterbatasan kemampuan dan pengalaman. Oleh karena itu Penulis sangat menghargai saran dan kritik yang sifatnya membangun demi perbaikan penulisan dan mutu isi tugas akhir ini untuk kelanjutan penelitian serupa di masa mendatang.

Penulis berharap semoga tugas akhir ini dapat memberikan manfaat kepada pembaca dan bisa diambil manfaatnya untuk pengembangan di masa mendatang.

Malang, 14 Januari 2008

Penulis

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	iii
LEMBAR PERNYATAAN	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvii
DAFTAR LAMPIRAN	xix
Bab I	
Pendahuluan	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Masalah	2
1.4 Batasan Masalah	3
1.5 Manfaat	3
1.6 Metodologi Pemecahan Masalah	3
1.7 Sistematika Penulisan	4
Bab II	
Tinjauan Pustaka	5
2.1 Penelitian yang ada sebelumnya	5
2.2 Pengertian Sistem Pakar	5
2.3 Ciri-ciri Sistem Pakar	6
2.4 Keuntungan dan Kelemahan Sistem Pakar	7
2.5 Struktur Sistem Pakar	8
2.5.1. Antarmuka pengguna (<i>user interface</i>)	10
2.5.2. Basis Pengetahuan (<i>Knowledge Based</i>)	10
2.5.3. Akuisisi Pengetahuan	12
2.5.4. Mesin Inferensi	13
2.5.5. <i>Workplace (Blackboard)</i>	14
2.5.6. Fasilitas Penjelasan	15
2.5.7. Perbaikan Pengetahuan	15
2.6 Representasi Pengetahuan	16
2.6.1. Kaidah Produksi (<i>Production Rules</i>)	16
2.7 Faktor Kepastian	17
2.7.1. Kombinasi Aturan	18

Bab III	Metodologi dan Perancangan	21
3.1	Deskripsi Umum Sistem	22
3.2	Perancangan Sistem	23
3.2.1.	Perancangan Blok Diagram	23
3.2.2.	Perancangan <i>Dependency Diagram</i>	24
3.2.3.	Penyusunan Tabel Keputusan	25
3.2.4.	Pengalihan Tabel Keputusan ke aturan IF-THEN	26
3.2.5.	Perancangan Basis Data	28
3.2.6.	Perancangan <i>Inference Engine</i>	34
3.2.6.1.	<i>Forward Chaining</i>	34
3.2.6.2.	<i>Backward Chaining</i>	39
3.2.7.	Contoh Cara menghitung CF	41
3.2.8.	Administrator Sistem	45
3.2.9.	Perancangan <i>User Interface</i>	46
3.2.10.	Kuisisioner	46
Bab IV	Pembahasan	49
4.1	Lingkungan Implementasi.....	49
4.1.1.	Lingkungan Perangkat Keras	49
4.1.2.	Lingkungan Perangkat Lunak.....	49
4.2	Implementasi Program.....	49
4.2.1.	<i>Forward Chaining</i>	49
4.2.1.	<i>Backward Chaining</i>	55
4.3	Implementasi Sistem	58
4.4	Kuisisioner	69
4.5	Pengujian Sistem	69
4.5.1.	Studi Kasus	69
4.5.2.	Hasil Kuisisioner.....	70
Bab V	Kesimpulan Dan Saran	75
5.1	Kesimpulan	75
5.2	Saran	75
Daftar Pustaka	77
Lampiran	79

DAFTAR GAMBAR

Gambar	2.1	Skema Sistem Pakar.....	9
Gambar	2.2	Contoh <i>Blok Diagram of Domain</i>	11
Gambar	2.3	Contoh <i>Dependency Diagram</i>	12
Gambar	2.4	<i>Forward Chaining</i>	14
Gambar	2.5	<i>Backward Chaining</i>	14
Gambar	3.1	Diagram Alir Penelitian	21
Gambar	3.2	Perancangan Sistem	23
Gambar	3.3	<i>Block Diagram Of Domain</i>	23
Gambar	3.4	<i>Block Diagram of Decission Situation</i>	24
Gambar	3.5	<i>Dependency Diagram</i>	25
Gambar	3.6	Relasi antar tabel	33
Gambar	3.7	Proses Kerja <i>inference engine</i>	34
Gambar	3.8	Proses Menampilkan Pertanyaan.....	36
Gambar	3.9	Proses Pengecekan Premis	37
Gambar	3.10	Proses Pengecekan Rule	38
Gambar	3.11	Proses Penampilan Keputusan	39
Gambar	3.12	Proses Kerja <i>inference engine</i>	39
Gambar	3.13	Proses Penampilan Hasil	40
Gambar	3.14	Cek Rule dan Premis	41
Gambar	3.15	Administrasi Sistem	45
Gambar	3.16	Menu Utama	46
Gambar	4.1	Form <i>User Login</i>	58
Gambar	4.2	Form <i>Home</i>	58
Gambar	4.3	Form Pengetahuan.....	59
Gambar	4.4	Form <i>Top 10 List</i>	59
Gambar	4.5	Form Konsultasi	60
Gambar	4.6	Form Tanya 1	61
Gambar	4.7	Form Tanya 2	61
Gambar	4.8	Form Hasil Konsultasi A	62
Gambar	4.9	Form Detail Konsultasi	62
Gambar	4.10	Form Pilih Jurusan	63
Gambar	4.11	Form Tanya 3	63
Gambar	4.12	Form Hasil Konsultasi B	64
Gambar	4.13	Form Pengaturan	65
Gambar	4.14	Form Login Admin	65
Gambar	4.15	Form Atur jurusan	66
Gambar	4.16	Form Atur Premis	66

Gambar	4.17 Form Atur Bidang Ilmu	67
Gambar	4.18 Form atur <i>rule</i>	67
Gambar	4.19 Form add/edit <i>rule</i>	68
Gambar	4.20 Form <i>Rule</i> detail	68
Gambar	4.21 Kuisisioner	69
Gambar	4.22 Rata-rata jawaban Kuisisioner siswa	72
Gambar	4.23 Hasil Penilaian pakar terhadap rekomendasi.....	73
Gambar	4.24 Persentase tingkat persetujuan pakar terhadap Rekomendasi sistem	73



DAFTAR TABEL

Tabel 2.1	Aturan Penggabungan <i>efidence</i> ...	18
Tabel 3.1	Tabel keputusan	25
Tabel 3.2	Struktur Tabel TJurusan	29
Tabel 3.3	Struktur Tabel TRule	30
Tabel 3.4	Struktur Tabel TRelasi	30
Tabel 3.5	Struktur Tabel TPremis	30
Tabel 3.6	Struktur Tabel TTipe	31
Tabel 3.7	Struktur Tabel Tilmu	31
Tabel 3.8	Struktur Tabel TempPremis.....	32
Tabel 3.9	Struktur Tabel TempRule.....	32
Tabel 3.10	Struktur Tabel TMetarule	32
Tabel 3.11	Struktur Tabel Temp2	33
Tabel 4.1	Jawaban Kuisisioner	71



DAFTAR LAMPIRAN

Lampiran 1	Aturan yang dipakai	79
Lampiran 2	Daftar Jurusan	87
Lampiran 3	Tabel Premis	89
Lampiran 4	<i>Store Procedure</i>	95
Lampiran 5	Hasil Rekomendasi Sistem Pakar Kelas Bahasa..	99
Lampiran 6	Hasil Rekomendasi Sistem Pakar Kelas IPA.....	101
Lampiran 7	Hasil Rekomendasi Sistem Pakar Kelas IPS.....	105



BAB I

PENDAHULUAN

1.1 Latar Belakang

Komputer telah berkembang menjadi alat pengolah data, penghasil informasi bahkan ikut berperan dalam pengambilan keputusan. Tidak puas dengan hanya fungsi tersebut, para ahli komputer masih terus mengembangkan kecanggihan komputer agar dapat memiliki kemampuan seperti manusia (Kusrini, 2006). Tidak dapat dipungkiri, aplikasi sistem pakar merupakan satu bagian dari bidang kecerdasan buatan yang belakangan ini semakin menarik perhatian, karena dengan sistem pakar, seseorang dapat berkonsultasi tentang suatu masalah tanpa harus bertemu langsung dengan ahlinya.

Permasalahan yang ditangani oleh seorang pakar bukan hanya permasalahan yang mengandalkan algoritma, namun terkadang juga masalah yang sulit dipahami. Permasalahan tersebut dapat diatasi oleh seorang pakar dengan pengetahuan dan pengalamannya. Oleh karena itu sistem pakar dibangun bukan berdasarkan algoritma tertentu tetapi berdasarkan basis pengetahuan dan aturan (Kusrini, 2006). *Knowledge* atau pengetahuan dalam sistem pakar tidak terbatas hanya pada seorang ahli tetapi juga dapat berupa pengetahuan umum lainnya yang terdapat pada buku, majalah, koran dan lain-lain. Pengetahuan ini bersifat khusus untuk satu domain masalah saja. Domain masalah ini adalah bidang atau suatu lingkup yang khusus seperti kedokteran, pendidikan, bisnis dan sebagainya (Kusrini, 2006).

Dalam dunia pendidikan saat ini, Universitas / Perguruan Tinggi menawarkan banyak pilihan jurusan / program studi. Di sisi lain, banyak siswa sekolah menengah atas kurang mengerti dan memahami karakteristik dari masing-masing program studi yang ada di Perguruan Tinggi. Hal tersebut menyebabkan siswa tersebut bingung dan terkadang salah dalam memilih program studi yang sesuai dengan bidang minat dan kemampuan yang dimilikinya.

Dalam wawancaranya dengan *HL Magazine*, DR. H. Arief Rachman M.Pd mengatakan, “Sebenarnya, pengetahuan mengenai pilihan jurusan/program studi di perguruan tinggi adalah tanggung jawab Orang Tua dan Sekolah...”, hal tersebut benar, tetapi yang terjadi di lapangan sangat jauh berbeda. Banyak mahasiswa yang merasa salah masuk jurusan/program studi (M.Sinar Dinarga, 2006).

Mereka tidak banyak tahu tentang jurusan-jurusan yang ada di perguruan tinggi walaupun nama jurusan tersebut sering mereka

dengar. Selama ini mereka paling banyak mendapat informasi mengenai suatu jurusan apabila di keluarga mereka pernah atau sedang kuliah di jurusan tersebut. Keluarga juga kadang tidak mendukung minat siswa apabila minat mereka tertuju pada suatu jurusan yang di cap “kurang menjual”. Sekolah sebagai pihak kedua yang bertanggung jawab juga tidak banyak membantu, karena kebanyakan murid hanya mendapat sedikit informasi dari pihak sekolah (M.Sinar Dinarga, 2006).

Namun dengan adanya sistem Pakar yang berbasiskan aturan ini, diharapkan dapat membantu dan mengarahkan siswa sekolah menengah atas (calon mahasiswa) untuk memilih jurusan/program studi yang sesuai berdasarkan bidang minat dan kemampuan yang dimilikinya serta dapat membantu memberikan gambaran tentang jurusan/program studi tersebut tanpa mengecilkan peran Bimbingan dan Konseling Sekolah.

1.2 Rumusan Masalah

Permasalahan yang ditangani dalam pembuatan tugas akhir ini adalah :

1. Bagaimana mengklasifikasikan karakteristik setiap program studi dengan menggunakan sistem pakar berbasis aturan (*rule based*).
2. Bagaimana menentukan program studi berdasarkan fakta – fakta yang ada, dimana fakta berupa minat dan kemampuan calon mahasiswa.
3. Bagaimana mengimplementasikan sistem pakar berbasis aturan untuk rekomendasi program studi agar dapat mempermudah dan membantu calon mahasiswa

1.3 Tujuan

Tujuan pembuatan Tugas Akhir ini adalah :

1. Mengklasifikasikan karakteristik masing masing program studi dengan menggunakan sistem pakar berbasis aturan.
2. Menentukan program studi berdasarkan fakta – fakta yang ada.
3. Mengimplementasikan sistem pakar berbasis aturan untuk rekomendasi program studi agar dapat mempermudah dan membantu calon mahasiswa

1.4 Batasan Masalah

Batasan masalah pada tugas akhir ini adalah :

1. Data program studi diambil hanya dari beberapa PTN di Indonesia dengan tidak mencantumkan semua program studi yang tersedia.
2. Program studi hanya terbatas pada jenjang strata satu (S1) dan diasumsikan jurusan dan program studi berada pada satu level yang sama.
3. Pengklasifikasian Program Studi dilakukan berdasarkan bidang keilmuan seperti Sains, Teknik, Kesehatan, Peternakan, Pertanian, Perikanan, Sosial, Ekonomi, Hukum dan Sastra.
4. Menggunakan Kaidah Produksi (*Production Rules*) sebagai representasi pengetahuan dengan metode inferensi *Forward Chaining* dan *Backward Chaining* yang digunakan secara terpisah tanpa membandingkan dengan metode lain.
5. Dalam penggunaan metode *Forward Chaining*, sistem akan menampilkan banyak sekali pertanyaan, tidak hanya tentang minat/keinginan tetapi juga tentang bakat dan kemampuan siswa. Selain itu, sistem juga menelusuri semua kemungkinan bidang ilmu dan program studi yang sesuai dengan siswa.
6. Implementasi sistem pakar ini dibangun dengan Delphi7 dan SQL Server 2000.

1.5 Manfaat

Manfaat yang diperoleh dari penulisan tugas akhir ini adalah :

1. Manfaat bagi masyarakat :
 - Membantu calon mahasiswa dalam memilih jurusan/program studi di perguruan tinggi.
 - Mempermudah calon mahasiswa memperoleh informasi tentang program studi di perguruan tinggi.
2. Manfaat pribadi :
 - Dapat memahami tentang sistem pakar berbasis aturan terutama tentang Kaidah Produksi (*Production Rules*) yang diterapkan pada rekomendasi program studi.

1.6 Metodologi Pemecahan Masalah

Untuk mencapai tujuan yang dirumuskan sebelumnya, maka metodologi yang digunakan dalam penulisan tugas akhir ini adalah:

1. Studi literatur
Mempelajari teori-teori yang berhubungan dengan konsep sistem pakar dari berbagai referensi.
2. Pendefinisian dan analisis masalah

Mendefinisikan dan menganalisis masalah untuk mencari solusi yang tepat.

3. Perancangan dan implementasi sistem

Membuat perancangan perangkat lunak dengan analisis terstruktur dan mengimplementasikan hasil rancangan tersebut yaitu sistem pakar rekomendasi program studi / jurusan

4. Uji coba dan analisa hasil implementasi

Menguji perangkat lunak, dan menganalisa hasil dari implementasi tersebut apakah sudah sesuai dengan tujuan yang dirumuskan sebelumnya, untuk kemudian menarik kesimpulannya.

1.7 Sistematika Penulisan

Tugas akhir ini disusun berdasarkan sistematika penulisan sebagai berikut:

1. BAB I PENDAHULUAN

Berisi latar belakang masalah, perumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi pemecahan masalah, dan sistematika penulisan.

2. BAB II TINJAUAN PUSTAKA

Menguraikan teori-teori yang berhubungan dengan sistem pakar berbasis aturan.

3. BAB III METODOLOGI DAN PERANCANGAN SISTEM

Pada bab ini akan dijelaskan mengenai metode-metode yang digunakan dalam pembuatan sistem pakar untuk rekomendasi program studi / jurusan.

4. BAB IV PEMBAHASAN

Pada bab ini akan dilakukan implementasi sistem pakar untuk rekomendasi program studi dan analisis sistem perangkat lunak yang dibangun yaitu apakah hasil dari sistem pakar menghasilkan informasi yang diinginkan.

5. BAB V KESIMPULAN DAN SARAN

Berisi kesimpulan dari seluruh rangkaian penelitian serta saran kemungkinan pengembangannya.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian sebelumnya (terdahulu)

Sistem pakar tentang rekomendasi program studi pernah dibangun oleh 2 orang mahasiswa, yaitu :

1. Riska P.Pribadi (PENS-ITS tahun 2005) dengan menggunakan kaidah produksi sebagai representasi pengetahuan, *forward chaining* sebagai metode inferensi dan pohon keputusan sebagai basis pengetahuan. Data program studi hanya diambil dari Universitas Airlangga dan ITS Surabaya tanpa menggunakan metode faktor kepastian (*certainty factor*). Aplikasi dibangun dengan menggunakan Java dan menghasilkan output berupa satu program studi. Basis Pengetahuan sistem pakar ini tidak bisa diubah (tidak dapat menambah aturan dan jurusan baru) .
2. Esti Rahayuningtyas. (D3-MITEK Unibraw tahun 2006) dengan menggunakan kaidah produksi sebagai representasi pengetahuan, *forward chaining* sebagai metode inferensi dan pohon keputusan sebagai basis pengetahuan tanpa menggunakan metode faktor kepastian. Aplikasi dibangun dengan menggunakan PHP dan juga menghasilkan output berupa satu program studi saja.

2.2. Pengertian Sistem Pakar

Martin dan Oxman (1988) mendefinisikan sistem pakar sebagai sistem berbasis komputer yang menggunakan pengetahuan, fakta dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat diselesaikan para pakar. Sedangkan Andi Setiawan (1993) mendefinisikan sistem pakar (*expert systems*) sebagai program berbasis pengetahuan yang menyediakan solusi-solusi dengan kualitas pakar untuk problema-problema dalam suatu domain yang spesifik.

Sistem pakar berbasis aturan berisi aturan sebagai basis pengetahuannya. Aturan tersebut digunakan untuk memberikan pertanyaan ke pengguna dan kemudian diproses agar dapat menyediakan rekomendasi (Turban,1995).

Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari ahlinya, karena itu dengan sistem pakar ini diharapkan orang awam pun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli (Sri Kusumadewi, 2003).

Pada dasarnya sistem pakar diterapkan untuk mendukung aktifitas pemecahan masalah. Beberapa aktifitas pemecahan masalah yang dimaksud adalah :

- Pembuatan keputusan (*Decision making*)
- Pemaduan pengetahuan (*Knowledge fusing*)
- Pembuatan desain (*Designing*)
- Perencanaan (*Planning*)
- Prakiraan / Prediksi (*Forecasting*)
- Pengaturan (*Regulation*)
- Pengendalian (*Controlling*)
- Diagnosa (*Diagnosing*)
- Penjelasan (*Explaining*)
- Pemberian nasehat (*Advising*)
- Pelatihan dan simulasi (*Tutorial*)

Selain itu, sistem pakar dapat berfungsi sebagai asisten yang pandai dari seorang pakar (Martin dan Oxman, 1988).

Muh.Arhami (2005) dalam bukunya menyatakan, istilah sistem pakar, sistem *knowledge based* atau sistem pakar *knowledge based* sering digunakan dengan arti yang sama. Kebanyakan orang menggunakan istilah sistem pakar karena lebih singkat, bahkan walau belum benar-benar pakar (hanya menggunakan pengetahuan umum). Kenyataannya, dalam dunia komersil saat ini banyak sistem pakar yang dapat secara efektif melaksanakan tugas yang tidak terlalu memerlukan pakar, seperti *help desk* otomatis, sistem pakar yang merekomendasikan restoran atau sistem pakar yang memberi nasehat tentang imunisasi yang disarankan (Turban, 1995).

2.3. Ciri - ciri Sistem Pakar

Karena disebabkan keheuristikannya dan sifatnya yang berdasar pada pengetahuan maka menurut Muhammad Arhami (2005) umumnya sistem pakar memiliki sifat :

- Memiliki informasi handal, baik dalam menampilkan langkah-langkah maupun dalam menjawab pertanyaan.
- Mudah dimodifikasi yaitu menambah atau menghapus suatu kemampuan dari basis pengetahuannya.
- Heuristik dalam menggunakan pengetahuan walau sering kali kurang sempurna untuk mendapat penyelesaiannya (heuristik merupakan suatu strategi untuk melakukan proses pencarian ruang problem secara selektif).
- Dapat digunakan dalam berbagai jenis komputer.

Kusrini (2006) menambahkan beberapa ciri lain yaitu :

- Terbatas pada bidang yang spesifik.
- Dapat memberikan penalaran untuk data-data yang tidak lengkap / tidak pasti.
- Berdasarkan aturan / kaidah tertentu.
- Dirancang untuk dapat dikembangkan secara bertahap.
- Output bersifat anjuran / nasihat.
- Output tergantung dialog dengan pengguna.

2.4. Keuntungan dan Kelemahan Sistem Pakar

Menurut Kusrini (2006), ada beberapa keuntungan dalam memakai sistem pakar, di antaranya dapat :

- Membuat seorang yang awam dapat bekerja seperti layaknya seorang pakar.
- Dapat bekerja dengan informasi yang tidak lengkap dan tidak pasti.
- Menyediakan nasihat yang konsisten dan dengan akurasi yang lebih baik.
- Handal.
- Tidak dapat lelah / bosan.
- Memiliki kemampuan memecahkan masalah yang kompleks.
- Memungkinkan pemindahan pengetahuan ke lokasi yang jauh serta memperluas jangkauan pakar karena dapat dipakai dan diperoleh dimana saja.

Selain keuntungan diatas, Muhammad Arhami (2005) menyatakan bahwa sistem pakar juga memiliki kelemahan, diantaranya adalah :

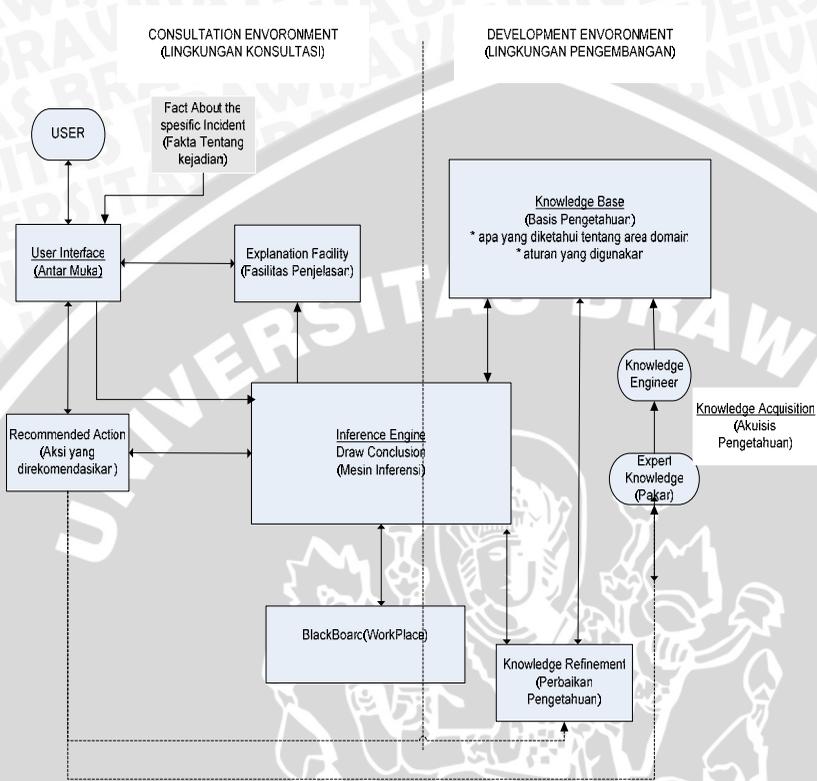
- Pengetahuan tidak selalu bisa didapatkan dengan mudah, karena ketidaktersedianya pakar dari masalah yang dibuat, walaupun ada, pendekatan yang digunakan oleh pakar bisa berbeda-beda.
- Untuk membuat sistem pakar berkualitas tinggi sangat sulit dan biaya pengembangan dan pemeliharannya mahal.
- Bisa juga sistem tidak dapat membuat keputusan.
- Sistem pakar tidak 100 % benar.

2.5. Struktur Sistem Pakar

Sistem Pakar disusun oleh dua bagian utama, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*) (Turban, 1995).

Lingkungan pengembangan digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar, sedangkan lingkungan konsultasi digunakan oleh pengguna yang bukan pakar untuk memperoleh pengetahuan dan nasehat pakar. Komponen sistem pakar dalam kedua bagian tersebut dapat dilihat pada gambar 2.1 . Skema Sistem Pakar

Dari skema sistem pakar berikut, Turban (1995) menjelaskan, tiga komponen utama yang tampak secara virtual di setiap sistem pakar yaitu basis pengetahuan, mesin inferensi dan antarmuka pengguna. Sistem pakar yang berinteraksi dengan pengguna dapat pula berisi komponen akuisisi pengetahuan, *workplace* (tempat kerja), penjelasan (*justifier*) dan perbaikan pengetahuan. Kebanyakan sistem pakar saat ini tidak berisi komponen perbaikan pengetahuan dan tidak memiliki fasilitas untuk belajar langsung dari pengalamannya sendiri (Turban, 1995).



Gambar 2.1 Skema Sistem Pakar
(Sumber : Turban, E, 1995)

Dalam penggunaannya, sistem pakar akan mengajukan berbagai pertanyaan kepada user dalam rangka mengumpulkan informasi awal tentang suatu masalah yang akan dipecahkannya. Jika sistem pakar sudah menerima input yang diperlukannya, maka ia akan melacak solusi atau kesimpulannya. Outputnya bisa berupa serangkaian rekomendasi yang harus ditempuh pengguna atau memilih salah satu alternatif terbaik dari beberapa pilihan yang disajikan sistem pakar (Andi Setyawan, 1993).

2.5.1. 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. Ia menyediakan sarana komunikasi jawaban atau solusi bila masalahnya sudah diketemukan. Setiap komunikasi selama proses pemecahan masalah dikendalikan oleh *user interface* (Andi Setyawan, 1993). Menurut McLeod (1995), pada bagian ini terjadi dialog antara program dan pemakai, yang memungkinkan sistem pakar menerima instruksi dan informasi (input) dari pemakai dan memberikan informasi (output) kepada pemakai.

2.5.2. Basis Pengetahuan (*Knowledge Base*)

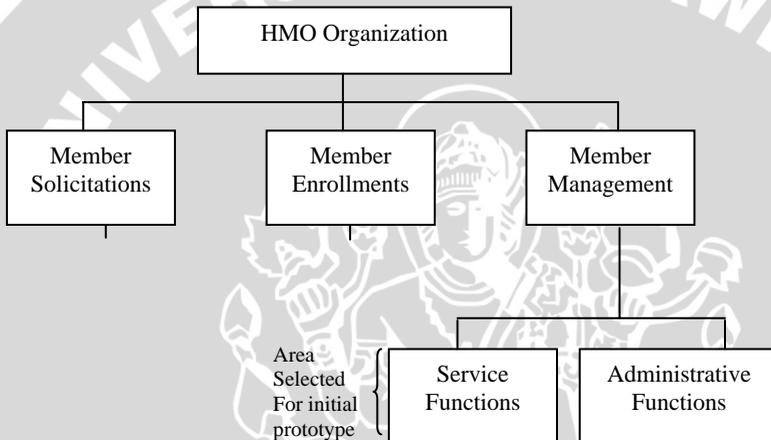
Menurut Sri Kusumadewi (2003), basis pengetahuan berisi pengetahuan - pengetahuan dalam penyelesaian masalah, tentu saja dalam domain tertentu. Ada dua bentuk pendekatan yang sangat umum digunakan, yaitu :

- **Berbasis aturan (*Rule Based*)**
Pengetahuan direpresentasikan dengan menggunakan aturan berbentuk IF-THEN. Bentuk ini digunakan jika kita memiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu. Disamping itu, bentuk ini juga digunakan apabila dibutuhkan penjelasan tentang langkah pencapaian solusi.
- **Berbasis kasus (*Case Based*)**
Basis pengetahuan akan berisi kumpulan solusi yang telah dicapai sebelumnya, kemudian akan diturunkan menjadi suatu solusi untuk keadaan yang terjadi sekarang. Bentuk ini digunakan apabila user menginginkan untuk tahu lebih banyak lagi pada kasus kasus yang hampir sama (mirip). Selain itu bentuk ini juga digunakan apabila kita telah memiliki sejumlah situasi atau kasus tertentu dalam basis pengetahuan.

Ada beberapa tahapan pengembangan basis pengetahuan (Dogolite, 1993), yaitu:

- a. Perancangan *block diagram* (*block diagram of domain* dan *block diagram of decision situation*.)

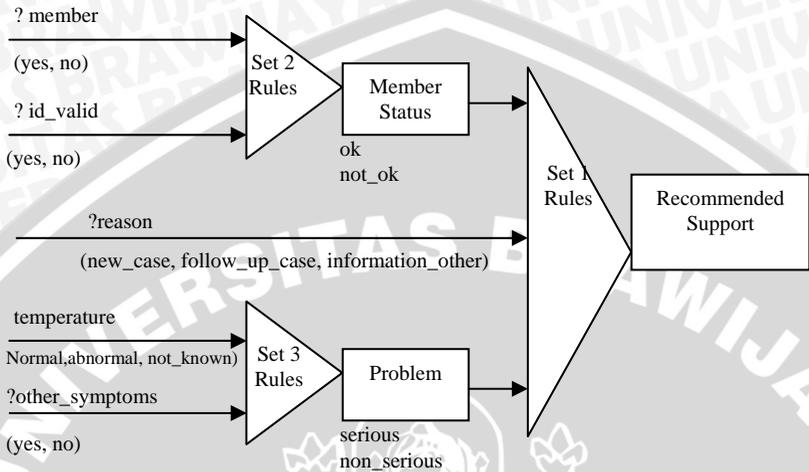
Berikut ini *block diagram of domain* yang mengindikasikan sub-area yang dipilih sebagai *prototype* awal pengembangan *knowledge based system*, yang diambil pada contoh kasus penentuan jenis perawatan bagi seorang pasien dalam sebuah organisasi kesehatan atau lebih dikenal dengan HMO (*Health Management Organization*).



Gambar 2.2 *Block diagram of domain*
(Sumber : D.G.Dogolite, 1993)

- b. Pengalihan *block diagram* ke *dependency diagram*

Dari diagram blok pada gambar 2.2, dapat dibentuk sebuah *dependency diagram*. Berikut *dependency diagram* dari kasus diatas :



Gambar 2.3 *Dependency diagram*
(Sumber : D.G.Dogolite, 1993)

c. Perancangan Tabel Keputusan (*decision table*)

d. Pengalihan table keputusan ke aturan IF-THEN

Setiap aturan dalam table keputusan dikonversi ke dalam aturan IF-THEN.

Struktur dan sintak sebuah aturan :

- RULE label : *keyword* aturan diikuti dengan sebuah nama.
- IF : mengidentifikasi awal dari kondisi aturan.
- THEN : mengidentifikasi awal kesimpulan dari sebuah aturan.
- ELSE : digunakan untuk mengidentifikasi awal dari sebuah alternatif kesimpulan pada sebuah aturan.

2.5.3. Akuisisi Pengetahuan

Akuisisi pengetahuan adalah transfer dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program komputer. Dalam tahap ini, *knowledge engineer* berusaha menyerap pengetahuan untuk selanjutnya ditransfer ke dalam basis pengetahuan. Pengetahuan/*knowledge* diperoleh dari pakar, di

lengkapi dengan buku, laporan penelitian dan pengalaman pemakai (Muh. Arhami, 2005).

Menurut Turban (1995) terdapat tiga metode dalam akuisisi pengetahuan, yaitu :

- Metode manual (wawancara, analisis protokol, observasi, *prototyping*, dan lainnya).
- Metode semiotomatis (*Repertory Grid Analysis*).
- Metode otomatis (*Machine Learning*, biasanya digunakan dalam sistem pakar berbasis kasus).

2.5.4. Mesin Inferensi

Inferensi merupakan proses untuk menghasilkan informasi dari apa yang diketahui atau diasumsikan (Kusrini, 2006). Menurut Andi Setyawan (1993), mesin inferensi adalah suatu rangkaian prosedur yang digunakan untuk menguji basis pengetahuan dengan cara yang sistematis pada saat menjawab pertanyaan, memecahkan persoalan atau membuat keputusan dalam suatu domain yang ditentukan.

Dalam Sistem pakar, proses inferensi ini dilakukan dalam suatu modul yang disebut *inference engine* (mesin inferensi). Menurut Kusrini (2006), ketika representasi pengetahuan pada bagian basis pengetahuan telah lengkap, atau pada level yang cukup akurat, maka representasi pengetahuan itu telah siap digunakan.

Diantara banyak metode inferensi, ada dua metode yang penting dan sering digunakan dalam sistem pakar, yaitu *forward chaining* dan *backward chaining*

- *Forward chaining*

Pencocokan dilakukan dari pernyataan sebelah kiri (IF terlebih dahulu), dengan kata lain, penalaran dimulai dari fakta terlebih dahulu untuk menguji kebenaran hipotesis (Kusrini,2006). *Forward chaining* adalah pendekatan yang dimotori data (*data driven*). Dalam pendekatan ini pelacakan dimulai dari informasi masukan dan selanjutnya mencoba menggambarkan kesimpulan (Muh. Arhami, 2005)

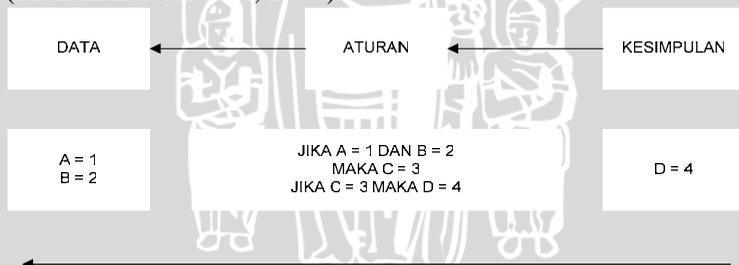


Gambar 2.4. *Forward chaining*

- *Backward chaining*

Pencocokan dilakukan dari pernyataan sebelah kanan (THEN terlebih dahulu), dengan kata lain, penalaran dimulai dari hipotesis terlebih dahulu dan untuk menguji kebenaran hipotesis tersebut harus dicari fakta – fakta yang ada dalam basis pengetahuan. (Kusrini, 2006).

Backward chaining adalah pendekatan yang dimotori tujuan (*Goal driven*) dimana pelacakan dimulai dari tujuan, selanjutnya dicari aturan yang memiliki tujuan tersebut untuk kesimpulannya. Selanjutnya proses pencatatan menggunakan premis untuk aturan tersebut sebagai tujuan baru dan mencari aturan lain dengan tujuan baru sebagai kesimpulannya. Proses ini berlanjut sampai semua kemungkinan ditemukan (Muhammad Arhami, 2005)



Gambar 2.5. *Backward chaining*

2.5.5. *WorkPlace (BlackBoard)*

WorkPlace atau *Blackboard* merupakan area kerja memori yang disimpan sebagai database untuk deskripsi persoalan terbaru yang ditetapkan oleh data input, digunakan juga untuk perekaman

hipotesis dan keputusan sementara (Turban,1995). Maksudnya, *workplace* menyimpan informasi sementara dari suatu proses konsultasi. Setiap suatu proses baru dijalankan, memori tersebut di set ke kondisi awal. Perhatikan contoh sistem pakar perbaikan mobil berikut, pada saat mobil mengalami kerusakan, anda memilih gejala kerusakan yang diprediksi dalam komputer untuk disimpan dalam *workplace*, komputer kemudian menampilkan pertanyaan selanjutnya yang harus dijawab, informasi jawaban tersebut juga disimpan dalam *workplace*.

2.5.6. Fasilitas penjelasan (*Justifier*)

Fasilitas penjelasan menyimpan data – data *history* dari suatu proses konsultasi, yaitu aturan-aturan mana saja yang berperan dalam pengambilan keputusan. Komponen tambahan dalam sistem pakar ini menggambarkan penalaran sistem pada pemakai (menjelaskan perilaku sistem pakar). Kemampuan untuk melacak tanggung jawab suatu kesimpulan terhadap sumbernya merupakan hal yang penting. Fasilitas penjelasan dapat melacak tanggung jawab tersebut dan menjelaskan perilaku sistem pakar dengan menjawab pertanyaan “bagaimana kesimpulan diperoleh” dan “mengapa pertanyaan diajukan”. Umumnya, penjelasan menampilkan aturan mana yang dipakai untuk memperoleh rekomendasi (Turban, 1995)

2.5.7. Perbaikan Pengetahuan

Pakar manusia memiliki sistem perbaikan pengetahuan, yaitu kemampuan untuk menganalisis pengetahuannya sendiri dan meningkatkan kinerjanya serta memiliki kemampuan belajar dari kinerjanya (Muh. Arhami, 2005). Evaluasi ini juga diperlukan dalam sistem pakar sehingga program dapat menganalisis alasan keberhasilan atau kegagalannya. Hal ini dapat mengarah kepada peningkatan sehingga menghasilkan basis pengetahuan yang lebih akurat dan pertimbangan yang lebih efektif. Komponen pelengkap tersebut masih dikembangkan dalam sistem pakar eksperimental pada beberapa universitas dan lembaga riset. (Turban, 1995)

2. 6. Representasi Pengetahuan

Kusrini (2006) mendefinisikan representasi pengetahuan sebagai metode yang digunakan untuk mengkodekan pengetahuan ke dalam sistem pakar. Jantung sistem pakar adalah pengetahuan. Dalam software kecerdasan buatan terdapat banyak metode yang berbeda dalam menampilkan pengetahuan.

Pengetahuan tersebut dapat direpresentasikan dalam bentuk yang sederhana ataupun yang kompleks, tergantung dari permasalahannya (Schnupp, 1989). Beberapa model representasi pengetahuan yang penting adalah:

- predikat kalkulus
- logika (*logic*)
- bingkai (*frame*)
- jaringan semantic (*semantic nets*)
- *script*
- kaidah produksi (*production rule*)

Melalui pengalaman, dapat diketahui salah satu metode terbaik untuk menampilkan pengetahuan ke dalam sistem pakar adalah kaidah produksi. Hampir semua sistem pakar komersial dan eksperimental menggunakan kaidah produksi atau yang disebut juga kaidah IF-THEN (Andi Setyawan, 1993).

2.6.1 Kaidah Produksi (*Production Rule*)

Kaidah produksi sangat populer karena formatnya sangat fleksibel. Hampir semua macam pengetahuan bisa ditulis dalam bentuk yang sesuai dengan format kaidah IF-THEN. Kaidah semacam ini umumnya sangat mudah ditulis, dan secara relatif mudah membentuk pangkalan pengetahuan yang impresif dengan cepat.

Kaidah produksi diformat dalam dua bagian. Pertama bagian IF yang menyatakan premis, kondisi atau *antecedent*. Biasanya disebut kaidah sebelah kiri. Kedua, bagian kaidah produksi THEN yang menyatakan konklusi, aksi, atau konsekuensi yang akan menggantikan kondisi sebelah kiri, bila ternyata sudah sesuai. Jika premis benar atau kondisinya cocok, maka bagian sebelah kanan benar (Muh. Arhami, 2005).

Kusrini (2006) mengklasifikasikan kaidah menjadi dua macam, yaitu kaidah derajat pertama (*first order rule*) dan kaidah meta (*meta rule*).

- Kaidah derajat pertama (*first order rule*) adalah kaidah sederhana yang terdiri dari anteseden dan konsekuen. Misalnya

IF	Suka membaca koran, buku, majalah = YES
AND	Suka menonton berita = YES
AND	Suka <i>browsing</i> internet = YES
THEN	Wawasan luas

- Kaidah meta (*meta rule*) adalah kaidah yang anteseden atau konsekuennya mengandung informasi tentang kaidah lain. Misalnya :

Aturan 1 :

IF	Selalu tepat waktu = YES
AND	Tidak suka menunda pekerjaan = YES
AND	Menepati peraturan = YES
THEN	Disiplin tinggi

Aturan 2 :

IF	Tertarik pada bidang sains = YES
AND	Disiplin tinggi = YES
AND	Daya analitis kuat = YES
AND	Mampu berpikir kritis dan logis = YES
THEN	Bidang Keilmuan = SAINS

Dalam contoh diatas, konsekuen dari aturan 1 yaitu disiplin tinggi menjadi anteseden/premis bagi aturan 2.

Keuntungan menggunakan kaidah produksi adalah aturan tersebut mudah dipahami dan aturan baru dapat ditambahkan dengan mudah ke dalam basis pengetahuan tanpa mempengaruhi aturan yang telah ada. Ketidakpastian yang dihubungkan dengan tiap aturan dapat ditambahkan untuk meningkatkan keakuratannya.

2.7. Faktor Kepastian (*Certainty Factor*)

Dalam penyusunan aturan, pengguna sering kali dihadapkan pada suatu premis yang meminta kebenaran absolut, sedangkan pada kenyataannya seringkali ditemui suatu fakta yang kebenarannya

tidak bersifat absolut. Misalnya pada sebuah pertanyaan, “apakah pasien menderita nyeri pada perut?”. Untuk memasukkan nilai/*value* pada *inference engine*, pengguna terkadang merasa bingung, rasa nyeri yang dimaksudkan adalah nyeri yang bagaimana, apakah nyeri sedikit saja sudah bisa dimasukkan sebagai *value*, atau nyeri yang hebat saja yang dimasukkan sebagai *value*?. Dua orang yang berbeda pada kasus yang sama bisa saja memberikan jawaban yang berbeda. Untuk itu, diperlukan sebuah metode untuk menyatakan ketidakpastian atas jawaban yang berbeda tersebut. Ada beberapa metode yang dapat digunakan untuk teori ketidakpastian, salah satunya adalah faktor kepastian (*certainty factor*) (Adrianto, 2003).

Dalam *certainty theory*, data data kualitatif direpresentasikan sebagai Derajat Keyakinan (*degree of belief*). Dalam mengekspresikan derajat keyakinan, *certainty theory* menggunakan suatu nilai yang disebut *Certainty Factor (CF)* untuk mengasumsikan derajat keyakinan seorang pakar terhadap suatu data. *Certainty Factor* memperkenalkan konsep *belief* / keyakinan dan *disbelief* / ketidakyakinan (Muhammad Arhami, 2005).

Adrianto (2003) mengatakan bahwa tingkat ketidakpastian CF dapat ditentukan dari validitas *knowledge base rule* dari pakar dan validitas respon pengguna. Nilai CF antara -1 sampai 1. CF positif bermakna fakta mendukung hipotesis sedangkan CF negatif menandakan negasi dari hipotesis dengan kata lain lebih beralasan untuk tidak mempercayai hipotesis.

2.7.1. Kombinasi Aturan

Metode MYCIN (salah satu sistem pakar yang terkenal) untuk menggabungkan *evidence* pada *antecedent*/premis sebuah aturan dalam menghitung CF ditunjukkan pada tabel aturan dibawah ini :

Tabel 2.1. Aturan menggabungkan *evidence*

<i>Evidence, E</i>	<i>Antecedent Ketidakpastian</i>
E1 and/dan E2	Min [CF(H,E1),CF(H,E2)]
E1 or/atau E2	Max [CF(H,E1),CF(H,E2)]
Not/tidak E	- CF(H,E)

Berikut adalah contoh kasus yang melibatkan kombinasi faktor kepastian (*certainty factor*):

IF	Selalu tepat waktu
AND	Tidak suka menunda pekerjaan
AND	Menepati peraturan
THEN	Disiplin tinggi, CF=0,7

Dengan menganggap:

E1 : Selalu tepat waktu
E2 : Tidak suka menunda pekerjaan
E3 : Menepati peraturan
H : Disiplin tinggi

Nilai CF Hipotesis pada saat *evidence* pasti adalah:

$$CF(H,E) = CF(H,E1 \cap E2 \cap E3) \\ = 0,7$$

dan disebut dengan *attenuation factor*. *Attenuation factor* didasarkan pada asumsi bahwa semua fakta E1, E2 dan seterusnya diketahui dengan pasti, yaitu :

$$CF(E1,e) = CF(E2,e) = CF(E3,e) = 1$$

dimana *partial evidence* *e* adalah observasi fakta yang didorong ke arah konklusi bahwa E1 diketahui dengan pasti. Biasanya *attenuation factor* dapat diperoleh dari Pakar.

Dalam kasus ini, kondisi pasien tidak dapat ditentukan dengan pasti. *Certainty factor evidence E* yang dipengaruhi *partial evidence e* ditunjukkan dengan nilai asumsi sbb (hasil inputan *user*):

$$CF(E1,e) = 0,5 \quad CF(E2,e) = 0,8 \quad CF(E3,e) = 0,3$$

$$\text{Sehingga } CF(E,e) = CF(E1 \cap E2 \cap E3) \\ = \min [CF(E1,e), CF(E2,e), CF(E3,e)] \\ = \min (0,5; 0,8; 0,3) = 0,3$$

Nilai *Certainty factor* hipotesisnya adalah :

$$CF(H,e) = CF(E,e) * CF(H,E) \\ = 0,3 * 0,7 = 0,21$$

Dimana :

CF(E,e) : *certainty factor Evidence E* yang dipengaruhi oleh

$CF(H,E)$: *evidence e*
 : *certainty factor* hipotesis dengan asumsi *evidence*
 dengan pasti ,yaitu ketika $CF(E,e) = 1$
 $CF(H,e)$: *certainty factor* hipotesis yang dipengaruhi oleh
evidence e
 (Kusrini,2006).

Jika ada kaidah / aturan lain yang termasuk dalam hipotesis yang sama tetapi berbeda dalam faktor kepastian, maka penghitungan CF dihitung dari penggabungan fungsi yang didefinisikan sebagai berikut :

$$CF_{combine} \begin{cases} \frac{CF_1+CF_2(1-CF_1)}{CF_1+CF_2} & \text{kedua duanya} > 0 \\ & \text{salah satu} < 0 \\ \frac{1- \text{Min}(|CF_1|,|CF_2|)}{CF_1+CF_2(1+CF_1)} & \text{kedua duanya} < 0 \end{cases}$$

Dimana rumusan $CF_{combine}$ digunakan berdasar pada apakah faktor kepastian positif atau negatif (Muh.Arhami, 2005)

Sebagai contoh, jika ada kaidah/aturan lain yang menghasilkan “disiplin tinggi” dengan $CF = 0,5$, maka penghitungan CF menggunakan $CF_{combine}$ sehingga diperoleh :

$$CF_{combine} (0,21;0,5) = 0,21+ 0,5(1-0,21) = 0,605$$

Anggap bahwa ada kaidah/aturan ketiga yang juga memiliki konklusi/konsekuensi yang sama, tetapi memiliki $CF = -0,4$, maka dengan menggunakan rumus ke dua dapat dihitung

$$CF_{combine} (0,605;-0,4) = \frac{0,605-0,4}{1-\min(|0,605|,|-0,4|)} = 0,34$$

BAB III

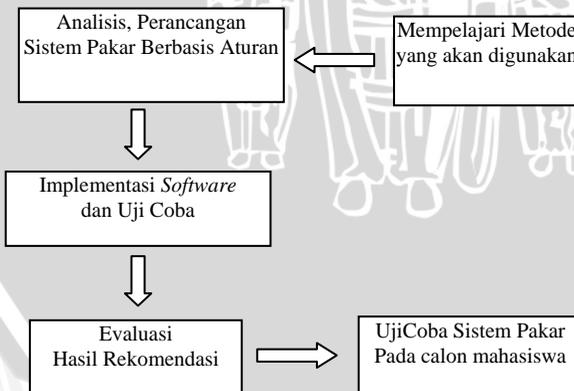
METODOLOGI DAN PERANCANGAN

Pada bab metodologi dan perancangan ini akan dibahas metode, rancangan yang digunakan dan langkah-langkah yang dilakukan dalam penelitian tentang sistem pakar berbasis aturan (*ruled based*) untuk menentukan program studi/ jurusan bagi calon mahasiswa

Penelitian dilakukan dengan tahapan – tahapan berikut ini :

1. Mempelajari metode yang akan digunakan pada sistem pakar berbasis aturan untuk menentukan program studi/ jurusan bagi calon mahasiswa. Metode tersebut telah dijelaskan pada bab 2.
2. Menganalisa dan merancang sistem pakar berbasis aturan dengan menggunakan kaidah produksi sebagai representasi pengetahuan, *Forward Chaining* dan *Backward Chaining* sebagai metode inferensi dan wawancara sebagai metode akuisisi
3. Membuat perangkat lunak berdasarkan analisis dan perancangan yang telah dilakukan.
4. Uji coba perangkat lunak sistem pakar.
5. Evaluasi hasil rekomendasi program studi yang dibuat oleh sistem, berdasarkan kecocokan antara aturan-aturan yang ada dengan rekomendasi program studinya.
6. Menguji coba sistem pakar pada calon mahasiswa untuk melihat tingkat efektivitasnya dan meminta mereka mengisi kuisioner tentang pendapat mereka terhadap sistem pakar ini.

Langkah – langkah penelitian ini dapat digambarkan seperti berikut :



Gambar 3.1 Diagram alir penelitian

3.1. Deskripsi Umum Sistem

3.2.

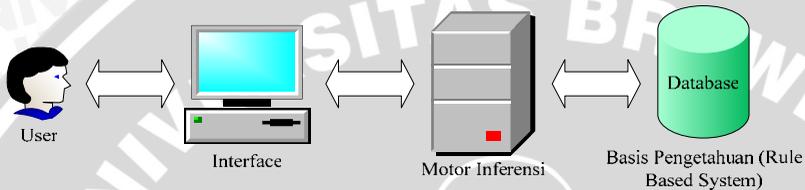
Dalam memberikan rekomendasi program studi/jurusan, sistem membutuhkan masukan dari *user* berupa jawaban jawaban dari pertanyaan yang diajukan komputer. Dengan menggunakan Sistem produksi (*production rule*), data-data jawaban tersebut akan dipelajari untuk menghasilkan rekomendasi program studi / jurusan dengan tingkat kepastian tertentu. Hasil yang akan muncul berupa jurusan yang dianjurkan beserta Universitas Negeri di Indonesia yang menawarkan pilihan jurusan tersebut beserta keterangan jurusan tersebut. Hasil dari sistem pakar ini hanya berupa rekomendasi atau saran (*advice*) yang tidak 100 % benar dan tidak harus dilaksanakan.

Untuk menjalankan sistem pakar ini, proses yang dilakukan adalah :

- pengguna (*user*) mendefinisikan permasalahan (input) dengan menggunakan antarmuka komputer (*interface*) yaitu dengan menjawab pertanyaan (*Yes/No Question*) yang diajukan komputer.
- Komputer akan mengajukan banyak pertanyaan tentang minat/keinginan, bakat, dan kemampuan user untuk mendapati informasi dari user.
- Setiap menjawab pertanyaan, pengguna juga diminta untuk memilih tingkat kepastian untuk menentukan nilai *CF user*. Ada 5 pilihan jawaban, yaitu Sangat Pasti/Sangat yakin ($CF = 1$), Pasti/Yakin ($CF = 0,8$), Kemungkinan besar ($CF = 0,6$), Mungkin ($CF = 0,4$) dan Ragu-ragu ($CF = 0,2$).
- komputer menyerahkan fakta permasalahan dari jawaban tersebut ke motor inferensi kemudian sistem akan berinteraksi dengan basis pengetahuan dalam basis data.
- Pada metode inferensi *forward chaining*, sistem akan memeriksa dan menelusuri semua kemungkinan bidang ilmu yang sesuai untuk mengetahui minat/keinginan, bakat, dan kemampuan user baik yang disadari maupun yang tidak disadari (misal bakat terpendam) agar dapat menentukan bidang studi terbaik untuk direkomendasikan.
- Pertanyaan selanjutnya yang akan muncul tergantung dari jawaban yang diberikan pengguna. Pertanyaan akan terus

muncul sampai sistem memperoleh cukup fakta untuk menentukan jurusan.

- pengguna dapat melihat hasilnya (output) melalui antarmuka komputer berupa rekomendasi / saran (sebagai hasil dari inferensi) tentang program studi/jurusan yang cocok untuk dipilih beserta universitas yang menawarkan dan lapangan kerja yang tersedia.



Gambar 3.2 Perancangan Sistem.

Sistem pakar yang dirancang ini merupakan pengembangan dari tugas akhir sistem pakar pemilihan jurusan yang dibuat oleh Riska P.Pribadi (www.eepis-its.edu).

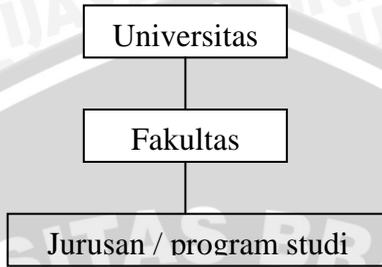
3.2. Perancangan Sistem

Perancangan sistem pakar ini dibagi menjadi 7 tahap, yaitu:

1. Perancangan *block diagram*
2. Perancangan *dependency diagram*
3. Penyusunan tabel keputusan dari *dependency diagram*
4. pengalihan tabel keputusan ke aturan IF-THEN
5. Perancangan Basis Data
6. Perancangan *inference engine*
7. Perancangan *user interface*

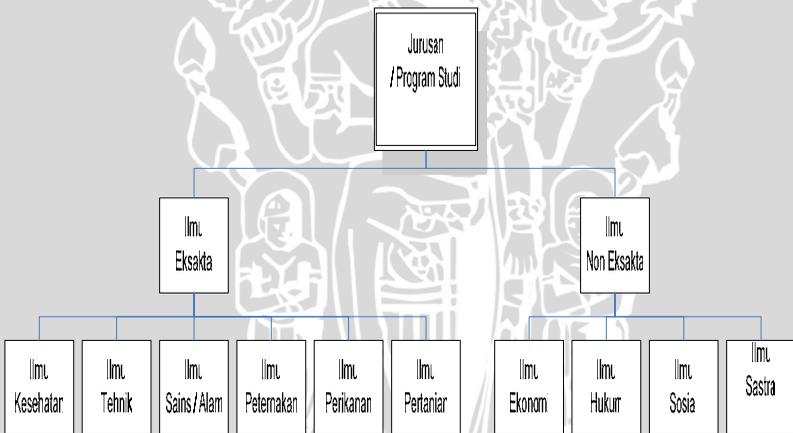
3.2.1. Perancangan *Block Diagram*

Gambar 3.3 dibawah ini menunjukkan *block diagram of domain* yang mengindikasikan sub area yang dipilih sebagai *prototype* awal pengembangan sistem. Jurusan atau program studi yang terdapat pada fakultas dikelompokkan berdasarkan ilmu eksakta atau non eksakta (M.Sinar Dinarga, 2006).



Gambar 3.3. Block diagram of domain

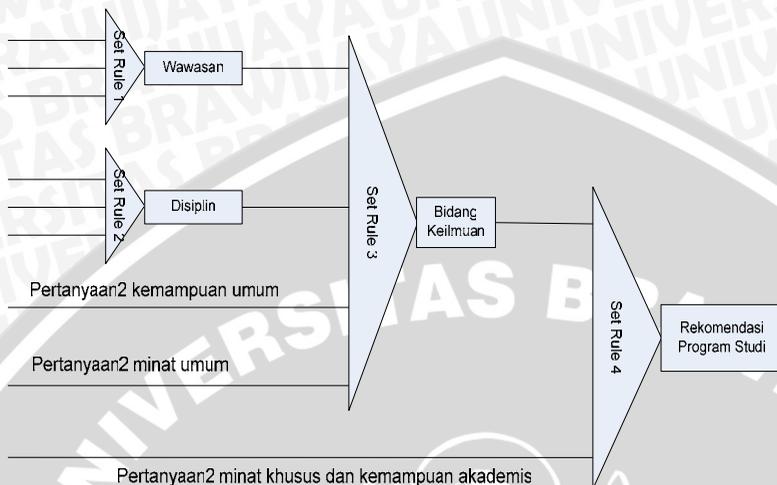
Dalam sistem pakar ini, ilmu eksakta dibagi menjadi 6 bidang keilmuan (Ilmu kesehatan, Ilmu tehnik, Ilmu Sains/Alam, Ilmu peternakan, Ilmu Perikanan, Ilmu Pertanian) sedangkan non eksakta dibagi menjadi 4 bidang keilmuan (Ilmu Ekonomi, Ilmu Hukum, Ilmu Sosial, dan Ilmu Sastra).



Gambar 3.4 Block diagram of decision situation

3.2.2. Perancangan Dependency Diagram

Dependency diagram menunjukkan hubungan antara faktor kritis (faktor yang memberikan *value* pada proses pengambilan keputusan), pertanyaan yang akan diinputkan, aturan, *value* dan rekomendasi yang dibuat sistem. Dari blok diagram yang disusun sebelumnya, disusun *dependency diagram* seperti dibawah ini :



Gambar 3.5 *Dependency diagram*

Contoh premis :

1. Minat khusus : ingin belajar cara mengobati penyakit
2. Minat umum : suka menggambar
3. Kemampuan umum : kesabaran tinggi, teliti, rajin dll
4. Kemampuan akademis : nilai matematika ≥ 7

3.2.3. Penyusunan tabel keputusan (*Decision Table*)

Dari *dependency diagram* yang telah disusun sebelumnya, dapat disusun sebuah tabel keputusan (*decision table*) berikut ini :

Table 3.1 Tabel keputusan

Premis	Program studi			
	A	B	N
PR001	*	*		*
PR002				*
PR003	*			
PR004	*			*
PR005		*		
PR006		*		
PR007	*	*		
PR008				

....				
PR n	*	*		*

Keterangan:

1. * menandakan bahwa premis ini merupakan syarat dari jurusan N.
2. Premis diperoleh dari wawancara dengan ahli yang menggunakan sumber diantaranya dari buku *Jurusan Apa Buat kamu (non eksakta)*, *Jurusan Apa Buat kamu (eksakta)*, *Panduan memilih jurusan (Tempo)* dan *Panduan memilih jurusan (Primagama)*.

3.2.4. Pengalihan tabel keputusan ke aturan IF-THEN

Setelah membuat tabel keputusan, langkah berikut adalah mengkonversikannya menjadi aturan IF – THEN. Karena tabel keputusan diatas sudah merupakan tabel tereduksi, maka dapat langsung dilakukan konversi aturan/*rule*.

Berikut adalah contoh beberapa aturan yang akan digunakan dalam sistem pakar pemilihan program studi/jurusan.

- R1 IF Suka membaca koran, buku, majalah
AND Suka menonton berita
THEN Wawasan luas, CF = 0,76
- R2 IF Suka membaca koran, buku, majalah
AND Suka *browsing* internet
THEN Wawasan luas, CF = 0,72
- R3 IF Selalu tepat waktu
AND Tidak suka menunda pekerjaan
AND Menepati peraturan
THEN Disiplin tinggi, CF = 0,81
- R4 IF Tertarik pada bidang sains
AND Wawasan luas
AND Teliti dan sabar
AND Mampu berpikir kritis dan logis
THEN Bidang Keilmuan = Sains, CF = 0,76

- R5 IF Tertarik pada bidang sains
AND Disiplin tinggi
AND memiliki daya analisis yang kuat
THEN Bidang Keilmuan = Sains, CF = 0,71
- R6 IF Tertarik pada bidang Teknik
AND Mampu bekerjasama dalam tim
AND Suka Menghitung
THEN Bidang Keilmuan = Teknik, CF = 0,77
- R7 IF Tertarik pada bidang Teknik
AND Suka kegiatan lapangan
AND Teliti dan sabar
AND Tidak Buta Warna
THEN Bidang Keilmuan = Teknik, CF = 0,83
- R8 IF Tertarik pada bidang kesehatan
AND Disiplin Tinggi
AND Tidak buta warna
AND Suka Menghafal dan daya ingat baik
THEN Bidang Keilmuan = Kesehatan, CF = 0,80
- R9 IF Tertarik pada bidang kesehatan
AND Tidak mudah jijik terhadap sesuatu
AND Teliti dan Sabar
AND Daya analisis kuat
THEN Bidang Keilmuan = Kesehatan, CF = 0,75
-
- R24 IF Bidang keilmuan Sains
AND Nilai rata-rata fisika ≥ 7
AND Suka bekerja di laboratorium
THEN Jurusan = FISIKA, CF = 0.76
- R25 IF Ingin mempelajari ilmu sains
AND Ingin belajar tentang gejala alam, elektronika dan instrumentasi, alat optik dll
THEN jurusan = FISIKA, CF = 0,80
-

R40 IF Ingin mempelajari ilmu tehnik
AND Ingin belajar seni desain bangunan dan struktur
bangunan terpadu
THEN jurusan = TEKNIK ARSITEKTUR, CF = 0.85

R41 IF Bidang keilmuan tehnik
AND Nilai rata-rata matematika >7
AND Suka menggambar
AND Kreatif dan imajinasi tinggi
THEN jurusan = TEKNIK ARSITEKTUR, CF = 0.78

.....

Confidence Faktor (CF) dari premis akan diinputkan oleh *user* karena tiap-tiap user memiliki tingkat keyakinan yang berbeda untuk masing-masing premis. Sedangkan CF untuk masing-masing aturan (*rule*) diambil dari wawancara dengan 2 orang guru Bimbingan dan Konseling.

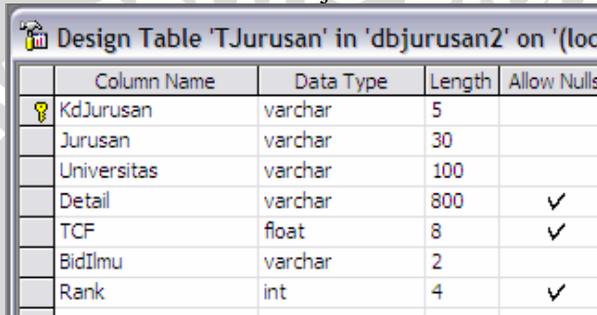
3.2.5. Perancangan Basis Data

Rencana implementasi basis data yang digunakan pada sistem pakar ini adalah SQLSERVER 2000. Basis data digunakan untuk menyimpan aturan, premis dan kesimpulan. Selain itu, *inference engine* juga menggunakan beberapa tabel kerja sebagai *workplace* dalam proses pengambilan keputusan. Tabel-tabel yang digunakan antara lain :

- Tabel TJurusan
Digunakan untuk menyimpan keterangan atau detail tentang jurusan, berisi *field-field* sebagai berikut :
 - KdJurusan, sebagai *primary key* dan sebagai kode untuk jurusan
 - Jurusan, untuk menyimpan nama jurusan
 - Universitas, untuk menyimpan universitas mana saja yang menawarkan jurusan tersebut
 - Detail, untuk menyimpan detail jurusan dan syarat apa saja yang seharusnya dipenuhi jika kita ingin masuk di jurusan tersebut, termasuk pekerjaan apa saja yang tersedia.

- TCF, untuk menyimpan hasil akhir penghitungan CF (CF Total) masing-masing jurusan.
- BidIlmu, untuk menyimpan kode bidang keilmuan masing-masing jurusan
- Rank, untuk menyimpan rangking masing-masing jurusan. Semakin banyak pencarian, semakin besar nilainya

Tabel 3.2 Struktur Tabel Tjurusan



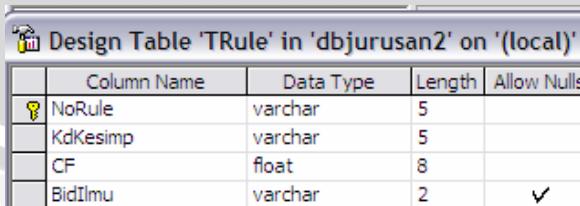
	Column Name	Data Type	Length	Allow Nulls
🔑	KdJurusan	varchar	5	
	Jurusan	varchar	30	
	Universitas	varchar	100	
	Detail	varchar	800	✓
	TCF	float	8	✓
	BidIlmu	varchar	2	
	Rank	int	4	✓

- Tabel TRule

Digunakan untuk menyimpan hasil kesimpulan suatu *rule*, yang memiliki *field-field* sebagai berikut :

- NoRule, sebagai *primary key* dan sebagai no aturan
- KdKesimp, untuk menyimpan kode kesimpulan dari aturan tersebut, bisa berupa kode jurusan atau kode premis
- CF, untuk menyimpan nilai *confident factor* (CF) dari aturan. Diperoleh dari ahli secara langsung
- BidIlmu, untuk menyimpan bidang keilmuan (berupa kode, seperti ES, EK, ET dll.)

Tabel 3.3 Struktur Tabel TRule



	Column Name	Data Type	Length	Allow Nulls
🔑	NoRule	varchar	5	
	KdKesimp	varchar	5	
	CF	float	8	
	BidIlmu	varchar	2	✓

- Tabel TRelasi

Digunakan untuk menyimpan *default value* dari suatu aturan, yang berisi *field* berikut ini :

- NoPremis, sebagai no urut premis dari suatu aturan
- NoRule, untuk menyimpan no aturan, menunjukkan bahwa premis tersebut berasal dari aturan yang mana

Tabel 3.4 Struktur Tabel TRelasi

Column Name	Data Type	Length	Allow Nulls
NoRule	varchar	5	
NoPremis	varchar	5	

- Tabel TPremis

Digunakan untuk menyimpan atribut dari suatu premis, yang berisi field berikut :

- NoPremis, sebagai *primary key* dan no urut premis dari suatu aturan, terdiri dari 13 macam kode yang berbeda, (misal : PU001 = Premis Umum no 1, ES001= Premis Eksakta Sains no 1, EK001= Premis Eksakta Kesehatan no 1, ET001= Premis Eksakta Teknik no 1, NE001= Premis NonEksakta Ekonomi no 1, dll)
- Premis, berisi premis dari suatu aturan

Tabel 3.5 Struktur Tabel TPremis

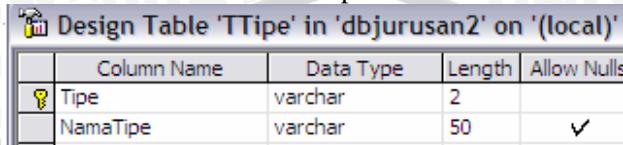
Column Name	Data Type	Length	Allow Nulls
NoPremis	varchar	5	
Premis	varchar	100	

- Tabel Ttipe

Digunakan untuk menyimpan jenis Tipe, apakah Eksakta (E) atau Non eksakta (NE)

- Tipe, untuk menyimpan kode tipe dari jurusan
- NamaTipe, untuk menyimpan nama tipe

Tabel 3.6 Struktur Tabel TType



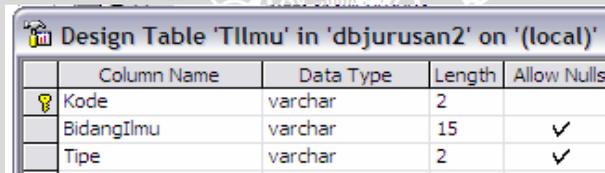
	Column Name	Data Type	Length	Allow Nulls
🔑	Tipe	varchar	2	
	NamaTipe	varchar	50	✓

- Tabel Ilmu

Digunakan untuk menyimpan Bidang Ilmu yang tersedia

- Kode, sebagai *primary key* dan untuk menyimpan kode bidang keilmuan
- Bidang Ilmu, untuk menyimpan nama bidang keilmuan
- Tipe, untuk menyimpan tipe dari bidang keilmuan

Tabel 3.7 Struktur Tabel TIllu



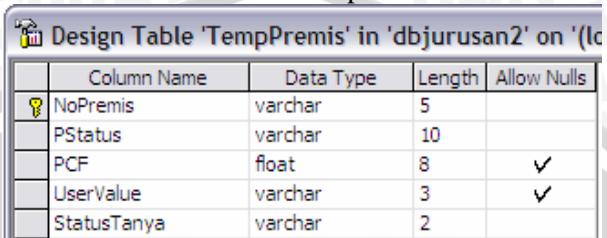
	Column Name	Data Type	Length	Allow Nulls
🔑	Kode	varchar	2	
	BidangIlmu	varchar	15	✓
	Tipe	varchar	2	✓

- Tabel TempPremis

Digunakan untuk menyimpan status suatu premis, ketika proses inference engine dilakukan, yang berisi field berikut :

- NoPremis, sebagai *primary key* dan no urut premis dari suatu aturan
- PStatus, untuk menyimpan status dari premis tersebut, apakah *free, true* atau *false*. Pada saat inialisasi tabel kerja, semua *record* pada *field* ini berisi status *free*.
- PCF, untuk menyimpan nilai *confident factor* yang diinputkan *user* untuk masing-masing premis
- UserValue, untuk menyimpan *value* yang pengguna masukkan
- StatusTanya, untuk menyimpan status premis, apakah akan ditanyakan (*Ask*) atau tidak (*Not Ask*).

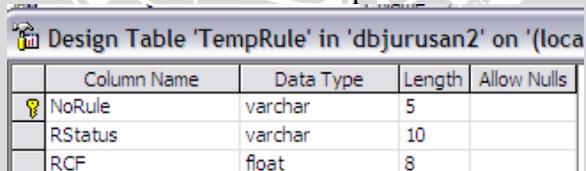
Tabel 3.8 Struktur Tabel TempPremis



	Column Name	Data Type	Length	Allow Nulls
	NoPremis	varchar	5	
	PStatus	varchar	10	
	PCF	float	8	✓
	UserValue	varchar	3	✓
	StatusTanya	varchar	2	

- Tabel TempRule
Digunakan untuk menyimpan status suatu *rule* ketika proses *inference engine* dilakukan, berisi *field-field* berikut :
 - NoRule, sebagai *primary key* dan sebagai no aturan
 - RStatus, untuk menyimpan status dari aturan tersebut, apakah *active* (*default* status), *discarded* (ditolak) atau, *triggered* (diterima). Pada saat inialisasi tabel kerja, semua *record* pada *field* RStatus akan bernilai *active*
 - RCF, untuk menyimpan nilai *confident factor* (CF) aturan tersebut. Nilai CF ini diperoleh dari hasil perhitungan.

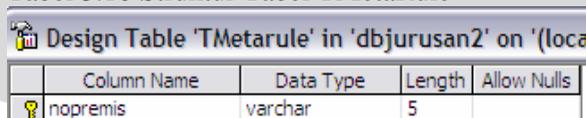
Tabel 3.9 Struktur Tabel TempRule



	Column Name	Data Type	Length	Allow Nulls
	NoRule	varchar	5	
	RStatus	varchar	10	
	RCF	float	8	

- Tabel TMetaRule
Tabel bantuan ini digunakan untuk menyimpan no premis yang bersifat metarule. Tabel ini tidak berelasi dengan tabel lain

Tabel 3.10 Struktur Tabel TMetaRule



	Column Name	Data Type	Length	Allow Nulls
	nopremis	varchar	5	

➤ Tabel Temp2

Tabel bantuan ini digunakan untuk menyimpan 'goal' dari *backward chaining* dan tidak memiliki relasi dengan tabel yang lain, berisi field sebagai berikut :

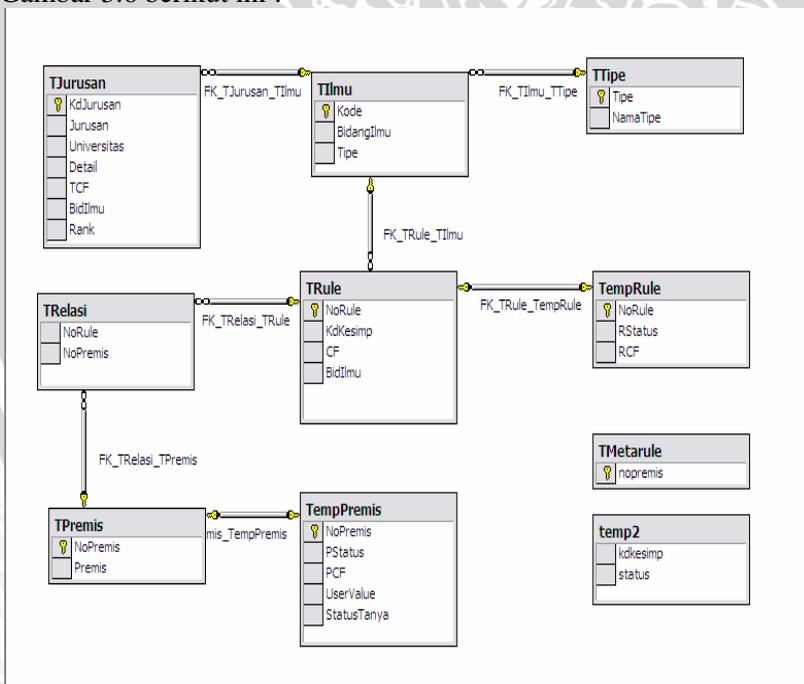
- KdKesimp, digunakan untuk menyimpan kode kesimpulan dari aturan yang menjadi 'goal' dari *backward chaining*
- Status, digunakan untuk menyimpan level status dari kode kesimpulan.

Tabel 3.11 Struktur Tabel Temp2

Design Table 'temp2' in 'dbjurusant2' on '(local)'

Column Name	Data Type	Length	Allow Nulls
kdkesimp	varchar	5	✓
status	int	4	✓

Dari beberapa tabel diatas, dapat disusun Relasi antar tabel seperti Gambar 3.6 berikut ini :



Gambar 3.6 Relasi antar tabel

3.2.6. Perancangan *inference engine*

Inference engine pada sistem pakar ini menggunakan metode *forward chaining* dan *backward chaining* yang digunakan secara terpisah. *Forward Chaining* digunakan jika pengguna tidak mengetahui jurusan/program studi yang sesuai atau yang diminatinya. Alasan penggunaan metode *forward chaining* ini karena sistem pakar ini akan mencari kesimpulan dari hasil sebuah sesi konsultasi tanya jawab secara menyeluruh. Kesimpulan ini akan berupa rekomendasi (saran) program studi/jurusan yang sebaiknya diambil. Karena itu, penggunaan metode *forward chaining* pada *inference engine* lebih sesuai dan dapat meningkatkan efisiensi kerja. Sebaliknya, *backward chaining* digunakan jika pengguna sudah mengetahui jurusan/program studi yang sesuai atau yang diminatinya. Dalam hal ini, pengguna hanya ingin mengetahui seberapa besar tingkat kecocokan dirinya dengan jurusan yang diminatinya. Karena itu, penggunaan metode *backward chaining* pada *inference engine* lebih sesuai dan dapat meningkatkan efisiensi kerja.

3.2.6.1. *Forward Chaining*

Secara umum, proses kerja *Forward Chaining* pada *inference engine* dapat digambarkan oleh diagram alir gambar 3.7. berikut ini :



Gambar 3.7 Proses kerja *inference engine*

3.2.6.1.1. Inisialisasi tabel kerja

Inisialisasi tabel kerja dilakukan setiap memulai program atau bila akan memulai sesi konsultasi baru. Tabel kerja (*workplace*) utama yang digunakan adalah tabel TempRule dan tabel TempPremis. Tabel TempRule digunakan untuk menyimpan status dari suatu *rule* pada saat proses pengambilan keputusan, apakah berstatus *active*, *discarded* atau *triggered*. Sedangkan tabel TempPremis digunakan untuk menyimpan status dari suatu premis, apakah berstatus *free*, *true*, *false*, *ask* dan *not ask*. Proses inisialisasi tabel kerja adalah dengan melakukan :

- Pengosongan isi *field* UserValue pada tabel TempPremis.
- Pengosongan isi *field* RCF pada tabel TempRule.
- Pengosongan isi *field* PCF pada tabel TempPremis.
- Pengosongan isi *field* TCF pada tabel Tjurusan.
- Melakukan perubahan isi *field* PStatus pada tabel TempPremis menjadi *free* pada tiap *record*.
- Melakukan perubahan isi *field* StatusTanya pada tabel TempPremis menjadi *Ask* pada tiap *record*.
- Melakukan perubahan isi *field* RStatus pada tabel TempRule menjadi *active* pada tiap *record*.

Setelah inisialisasi dilakukan, *inference engine* siap dijalankan.

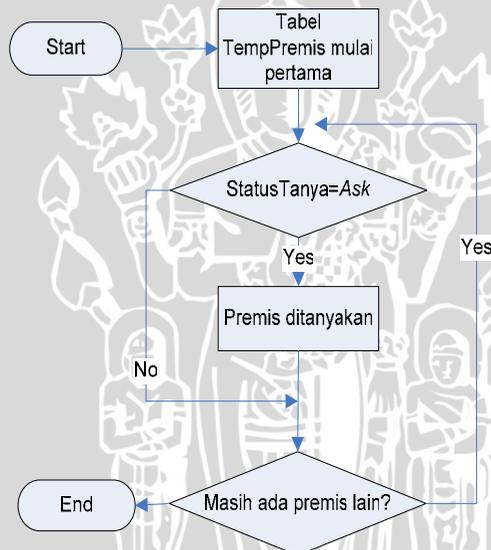
3.2.6.1.2. Proses Pengambilan Keputusan

Proses dalam pengambilan keputusan terbagi dalam beberapa tahap, yaitu tahap awal, cek status premis, cek aturan dan hitung RCF, serta tahap akhir atau penampilan rekomendasi.

- Tahap awal

Proses pengambilan keputusan pada *inference engine* dimulai dengan menerima *value* yang diberikan *user* setelah melalui rangkaian pertanyaan. Pertanyaan diambil dari tabel premis mulai *record* pertama. Pertanyaan pertama tentang tipe jurusan yang akan diambil, apakah eksakta atau non eksakta. Apabila user memilih eksakta, maka *rule* yang memiliki tipe jurusan non eksakta akan di eliminasi (RStatus = *discarded*). Selain itu, StatusTanya pada premis yang mempunyai kode awal Non eksakta (N.....) di ubah menjadi *not*

ask, begitu juga sebaliknya. Kemudian premis selanjutnya akan dicek StatusTanya nya, apakah *ask* atau tidak, jika *ask* maka premis tersebut ditanyakan. Ketika premis berada pada awal pertanyaan jurusan (misal nomor premis ES001), jika user menjawab “yes” maka premis-premis selanjutnya yang bertipe sama (ES....) akan ditanyakan, jika tidak StatusTanya premis-premis tersebut diubah menjadi *not ask* dan *rule* yang mempunyai tipe bidang keilmuan sains akan di eliminasi. Pertanyaan akan terus keluar sampai tidak ada premis lagi yang harus ditanyakan (Gambar 3.8.Menampilkan pertanyaan). *Value* atau jawaban dari *user* tersebut disimpan dalam *UserValue* pada tabel TempPremis sesuai dengan nomor dari pertanyaan premis asal jawaban tersebut.

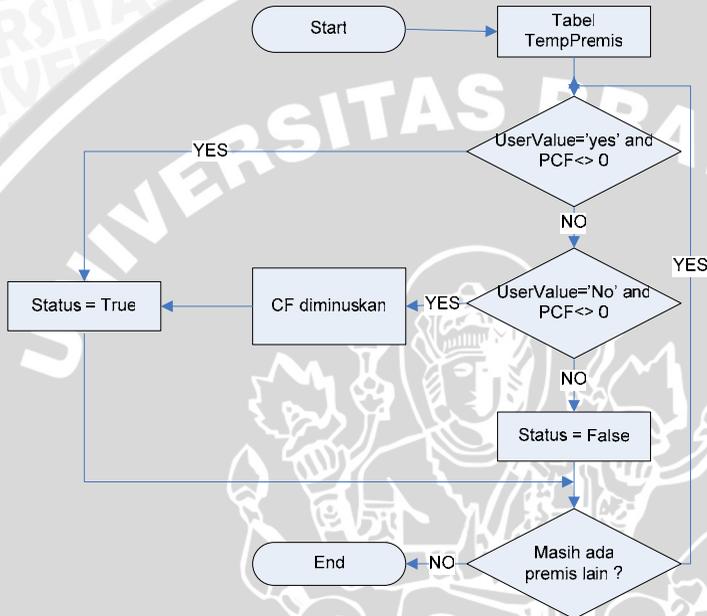


Gambar 3.8 Menampilkan pertanyaan

- Cek status premis

Setelah tahap awal, dilakukan proses pengecekan premis, dimana *value* tiap premis yang digunakan akan ditentukan statusnya apakah benar (*true*) atau salah (*false*). Jika suatu premis dianggap benar (*uservalue* = 'yes' dan $PCF < > 0$) maka *field* PStatus pada tabel TempPremis yang semula *free* diganti dengan *true*. Selain itu, suatu premis juga dianggap benar (*true*) apabila CF yang dimasukkan *user*

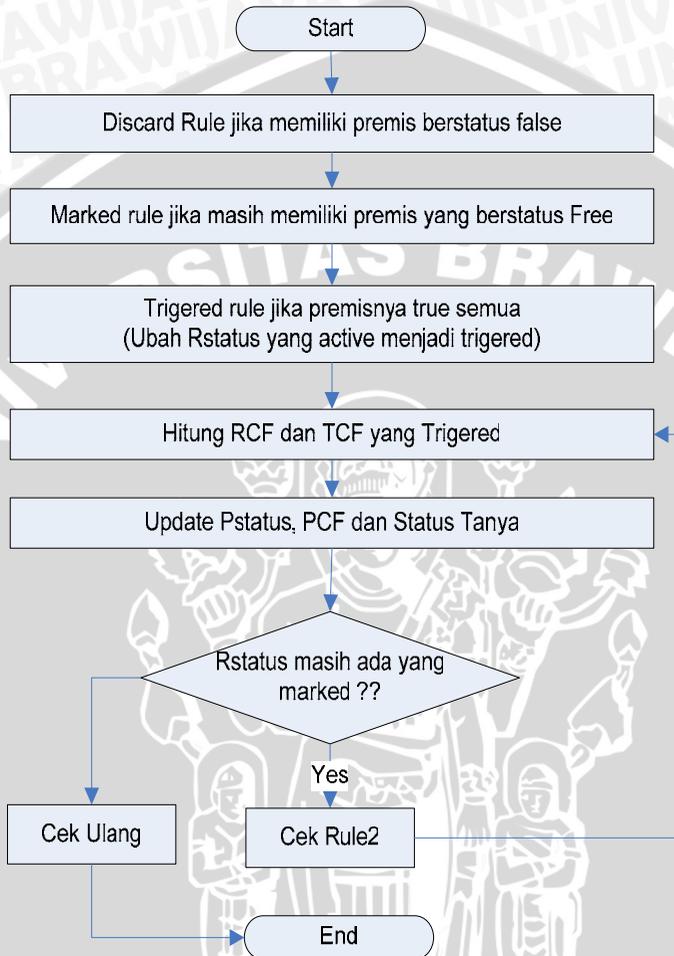
tidak bernilai 0 (walaupun UserValue tidak menjawab ‘yes’. Apabila hal itu terjadi, menurut Kusri (2006) maka nilai CF Premis yang tidak sesuai akan dikurangkan. Hal ini untuk mengantisipasi jika sebenarnya tidak ada *rule* yang benar-benar terpenuhi semua premisnya (Gambar 3.9).



Gambar 3.9 Proses Pengecekan Premis

- Cek Rule

Proses kemudian dilanjutkan dengan pengecekan aturan (Gambar 3.10). Setiap aturan yang masih bersifat *active* akan diperiksa, jika ada premis dari aturan tersebut yang belum diketahui kebenarannya, (masih ada pstatus yang *free*) maka Rstatus = *Marked*. Pengecekan dilakukan sampai aturan yang berstatus *active* tidak ada. Setelah itu aturan yang berstatus *marked* dicek sekali lagi sampai yang berstatus *marked* tidak ada. Jika semua premis dari aturan tersebut bernilai *true*, maka kesimpulan akhir aturan tersebut dianggap benar (Rstatus = *triggered*).

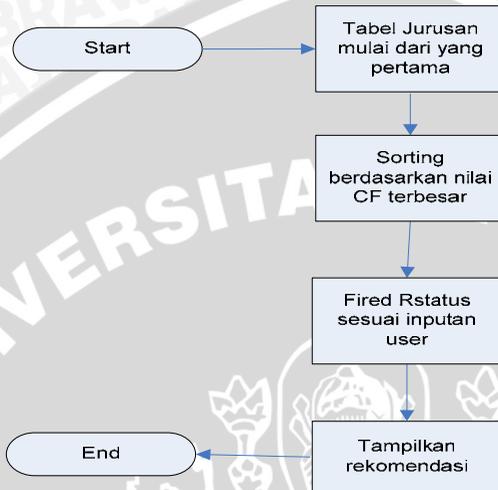


Gambar 3.10 Proses Pengecekan Rule

- Tahap akhir

Proses terakhir adalah proses penampilan kesimpulan (Gambar 3.11). Aturan difilter berdasarkan Status *rule* dan kode jawaban dari aturan tersebut. Sebelumnya, pada akhir proses tanya jawab, pengguna diminta untuk memasukkan berapa batas CF yang akan ditampilkan. Aturan yang diterima adalah aturan yang termasuk dalam nilai CF yang di inputkan pengguna. Untuk melihat

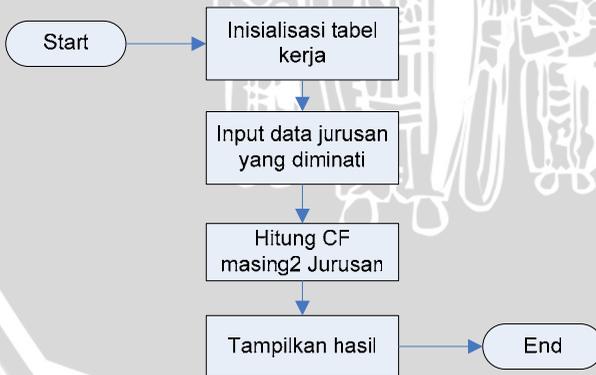
keterangan tentang kesimpulan *inference engine*, pengguna dapat melihat melalui detail jurusan.



Gambar 3.11 Proses Penampilan keputusan

3.2.6.2. Backward Chaining

Secara umum, proses kerja *Backward Chaining* pada *inference engine* dapat digambarkan oleh diagram alir gambar 3.12. berikut ini:



Gambar 3.12 Proses kerja *inference engine*

3.2.6.2.1. Inisialisasi tabel kerja

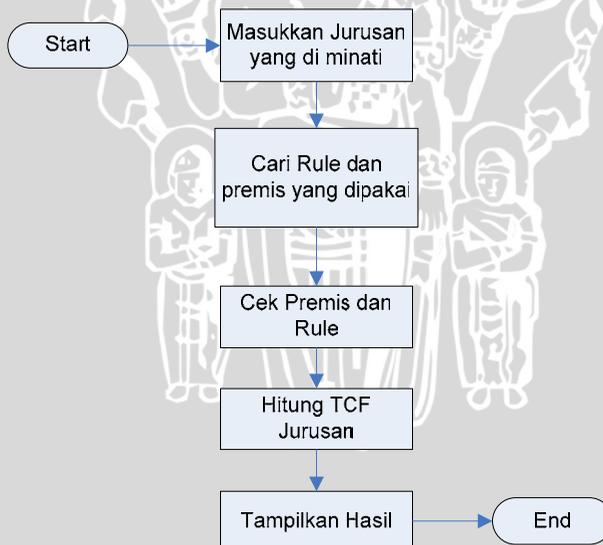
Proses inisialisasi tabel kerja adalah dengan melakukan :

- Pengosongan isi *field* UserValue pada tabel TempPremis.
- Pengosongan isi *field* RCF pada tabel TempRule.
- Pengosongan isi *field* PCF pada tabel TempPremis.
- Melakukan perubahan isi *field* PStatus pada tabel TempPremis menjadi *free* pada tiap *record*.
- Melakukan perubahan isi *field* StatusTanya pada tabel TempPremis menjadi *Not Ask* pada tiap *record*.
- Melakukan perubahan isi *field* RStatus pada tabel TempRule menjadi *active* pada tiap *record*.

Setelah inisialisasi dilakukan, *inference engine* siap dijalankan.

3.2.6.2.2. Proses Penampilan Hasil

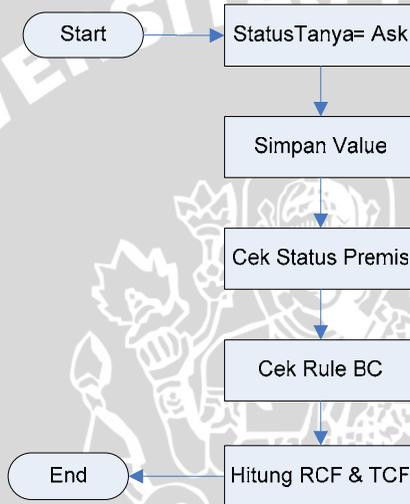
Garis besar proses penampilan hasil dapat dilihat pada gambar 3.13 berikut ini :



Gambar 3.13 Proses Penampilan Hasil

3.2.6.2.3. Cek Rule dan Premis

Pada proses cek *rule* dan Premis ini, aturan yang tidak dipakai akan *discard*, sedangkan yang dipakai akan *ditrigger*. Premis dari aturan yang *ditrigger* akan ditanyakan dan aturan tersebut akan dihitung CF nya. Aturan yang diterima adalah aturan yang nilai CF nya memenuhi batas CF. Proses pengecekan *rule* dan premis dapat dilihat pada gambar 3.14 berikut



Gambar 3.14 Cek Rule dan Premis BC

3.2.7. Contoh Cara menghitung CF

Misalkan jika diketahui ada beberapa aturan berikut :

- R1 IF [PU002]Suka membaca koran, buku, majalah
AND [PU001]Suka menonton berita
THEN [PU004]Wawasan luas, CF = 0,75
- R2 IF [PU002]Suka membaca koran, buku, majalah
AND [PU003]Suka *browsing* internet
THEN [PU004]Wawasan luas, CF = 0,70

- R3 IF [PU005]Selalu tepat waktu
AND [PU006]Tidak suka menunda pekerjaan
AND [PU007]Menepati peraturan
THEN [PU018]Disiplin tinggi, CF = 0,85
- R4 IF [ES001]Tertarik pada bidang sains
AND [PU004]Wawasan luas
AND [PU009]Teliti dan sabar
AND [PU010]Mampu berpikir kritis dan logis
THEN [ES003]Bidang Keilmuan = Sains, CF = 0,70
- R5 IF [ES001]Tertarik pada bidang sains
AND [PU08]Disiplin tinggi
AND [PU011]memiliki daya analisis yang kuat
THEN [ES003]Bidang Keilmuan = Sains, CF = 0,65
- R6 IF [ET001]Tertarik pada bidang Teknik
AND [PU013]Mampu bekerjasama dalam tim
AND [PU012]Suka Menghitung
THEN [ET003]Bidang Keilmuan = Teknik, CF = 0,75
- R7 IF [ET001]Tertarik pada bidang Teknik
AND [PU021]Suka kegiatan lapangan
AND [PU009]Teliti dan sabar
THEN [ET003]Bidang Keilmuan = Teknik, CF = 0,65
- R24 IF [ES003]Bidang keilmuan Sains
AND [E002]Nilai rata-rata fisika ≥ 7
AND [E010]Suka bekerja di laboratorium
THEN Jurusan = [J001]FISIKA, CF = 0,70
- R25 IF [ES002]Ingin mempelajari ilmu sains
AND [ES004]Ingin belajar tentang gejala alam,
elektronika dan instrumentasi, alat optik dll
THEN jurusan = [J001]FISIKA, CF = 0,80
- R40 IF [ET002]Ingin mempelajari ilmu tehnik
AND [ET004]Ingin belajar seni desain bangunan dan
struktur bangunan terpadu

THEN jurusan = [J010]TEHNIK ARSITEKTUR, CF = 0,80

R41 IF [ET003]Bidang keilmuan tehnik
AND [PU004]Nilai rata-rata matematika >7
AND [E013]Suka menggambar
AND [PU019]Kreatif dan imajinasi tinggi
THEN jurusan = [J010]TEHNIK ARSITEKTUR, CF = 0.75

Dari hasil tanya jawab dengan sistem, diperoleh Nilai CF dari pengguna sebagai berikut :

[PU001] YES	CF = 0,8	[E002] NO	CF = -0,4
[PU002] YES	CF = 0,8	[E004] YES	CF = 0,8
[PU003] NO	CF = -0,6	[E010] NO	CF = -0,6
[PU005] NO	CF = -0,6	[E013] NO	CF = -0,4
[PU006] YES	CF = 0,8	[ES001] YES	CF = 0,6
[PU007] YES	CF = 0,6	[ES002] YES	CF = 0,8
[PU009] YES	CF = 0,8	[ES004] YES	CF = 0,6
[PU010] YES	CF = 0,6	[ET001] YES	CF = 0,6
[PU011] YES	CF = 0,6	[ET002] YES	CF = 0,8
[PU012] YES	CF = 0,8	[ET004] YES	CF = 0,8
[PU013] YES	CF = 0,6		
[PU019] NO	CF = -0,4		
[PU021] NO	CF = -0,6		

Dari data diatas, akan diperoleh perhitungan CF sebagai berikut :

$$R1 \quad CF[PU004] = \min (CF[PU002], CF[PU001]) \\ = 0,8$$

$$CF(H,e) = CF(H,E)*CF(E,e) = 0,75 * 0,8 = 0,6$$

$$R2 \quad CF[PU004] = \min (CF[PU002], CF[PU003]) \\ = -0,6$$

$$CF(H,e) = CF(H,E)*CF(E,e) = 0,70 * -0,6 = -0,42$$

$$[PU004] = Cfcombine (R1, R2) = [0,6 + (-0,42)] / (1-0,42) = 0,31$$

$$R3 \quad CF[PU008] = \min (CF[PU005], CF[PU006], CF[PU007])$$

$$= -0,6$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,85 * -0,6 = -0,51$$

$$R4 \quad CF[ES003] = \min(CF[ES001], CF[PU004], CF[PU009], CF[PU010]) = 0,31$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,70 * 0,31 = 0,22$$

$$R5 \quad CF[ES003] = \min(CF[ES001], CF[PU008], CF[PU004]) = -0,51$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,65 * -0,51 = -0,33$$

$$[ES002] = Cfcombine (R4,R5) = [0,22+(-0,33)]/(1-0,22) = -0,14$$

$$R6 \quad CF[ET003] = \min(CF[ET001], CF[PU013], CF[PU012]) = 0,6$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,75 * 0,6 = 0,45$$

$$R7 \quad CF[ET003] = \min(CF[ET001], CF[PU021], CF[PU009]) = 0,6$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,65 * -0,6 = -0,39$$

$$[ET002] = Cfcombine (R6,R7) = [0,45+(-0,39)]/(1-0,39) = 0,10$$

$$R24 \quad CF[J001] = \min(CF[E002], CF[ES003], CF[E010]) = -0,6$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,7 * -0,6 = -0,42$$

$$R25 \quad CF[J001] = \min(CF[ES002], CF[ES004]) = 0,6$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,8 * 0,6 = 0,48$$

$$*[J001] = Cfcombine (R24,R25) = [0,48+(-0,42)]/(1-0,42) = 0,10$$

$$CF \text{ jurusan Fisika} = 0,10$$

$$R40 \quad CF[J010] = \min(CF[ET002], CF[ET004]) = 0,8$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,8 * 0,8 = 0,64$$

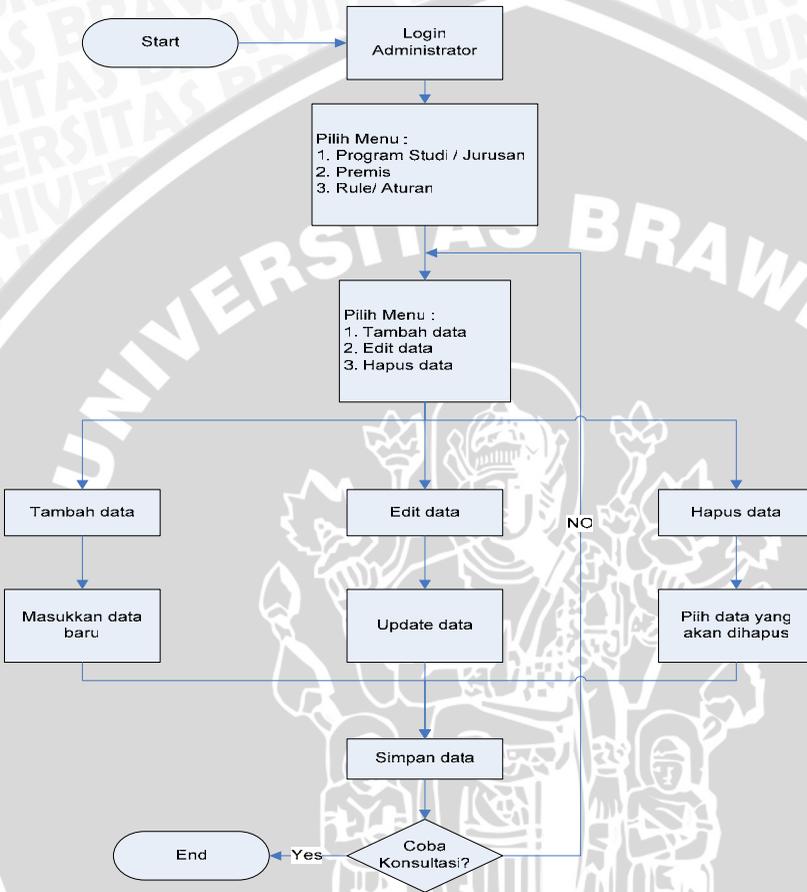
$$R41 \quad CF[J010] = \min(CF[ET003], CF[PU004], CF[PU019], CF[E013]) = -0,4$$

$$CF(H,e) = CF(H,E) * CF(E,e) = 0,75 * -0,4 = -0,30$$

$$*[J010] = Cfcombine (R40,R41) = [0,64+(-0,30)]/(1-0,30) = 0,49$$

$$CF \text{ Jurusan Teknik arsitektur} = 0,49$$

3.2.8. Administrator

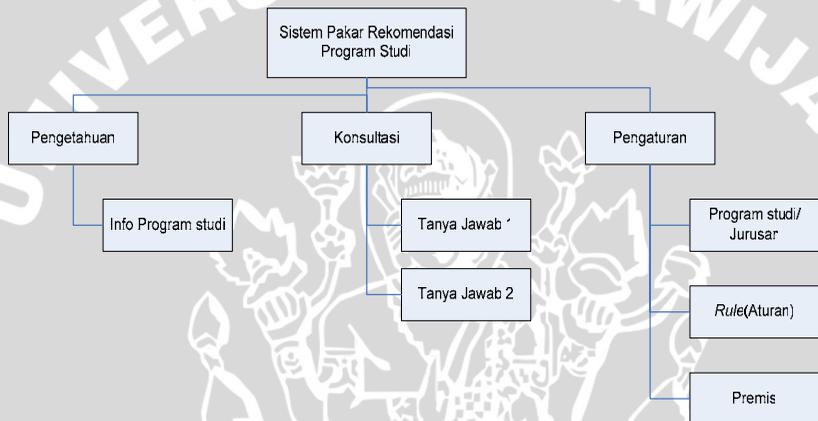


Gambar 3.15. Administrasi Sistem

Administrator memiliki hak penuh untuk mengubah sistem, baik untuk menambah jurusan, aturan, premis atau hanya sekedar meng-*update* data. Jika ingin menambah data jurusan, sistem akan mengecek apakah jurusan tersebut ada dalam basis data. Jika ada, berarti akan dilakukan edit data jurusan, jika tidak, berarti menambah jurusan baru. Ketika sistem menambah jurusan baru, maka akan diikuti dengan pembuatan aturan baru. Aturan baru ini harus dibuat dengan persetujuan dari ahlinya.

3.2.9. Perancangan *user interface*

Dalam merancang *user interface*, aspek yang harus diperhatikan antara lain aspek kenyamanan dan kemudahan bagi pengguna (*user*). Kenyamanan berarti pengguna harus merasa nyaman dalam membaca tampilan *interface*. Kemudahan berarti pengguna harus dapat menggunakan *user interface* dengan mudah. Untuk itu *user interface* akan dirancang sedemikian rupa agar kedua aspek tersebut dapat dipenuhi.



Gambar 3.16. Menu Utama

3.2.10. Kuisisioner

Setelah sistem pakar diujicobakan pada calon mahasiswa, akan dilakukan *pooling* terhadap calon mahasiswa yang telah mencoba aplikasi sistem pakar ini. Kuisisioner berisi tentang beberapa pertanyaan, antara lain :

1. Menurut anda, apakah sistem ini lebih membantu calon mahasiswa dalam memilih program studi/jurusan di Perguruan Tinggi?
 - A. Tidak Membantu
 - B. Kurang Membantu
 - C. Cukup Membantu
 - D. Sangat Membantu
2. Melalui media ini apakah Anda merasa lebih mudah dalam memilih Jurusan di Perguruan Tinggi ?
 - A. Tidak
 - B. Kurang

- C Cukup D. Ya
3. Apakah sistem ini memberi banyak informasi tentang studi/jurusan di Perguruan Tinggi?
- A. Tidak B. Kurang
C Cukup D. Ya
4. Secara keseluruhan, bagaimana penilaian anda terhadap sistem ini?
- A. Tidak Bagus B. Kurang Bagus
C Cukup Bagus D. Sangat Bagus
5. Menurut Anda, apakah rekomendasi (informasi) yang diperoleh tentang Jurusan di Perguruan Tinggi sudah sesuai dengan kemampuan yang Anda miliki ?
- A. Belum Sesuai B. Kurang Sesuai
C Cukup Sesuai D. Sudah Sesuai
6. Apakah calon mahasiswa akan mengikuti saran yang di rekomendasikan oleh sistem?
- A. Tidak B. Ragu-Ragu
C. Mempertimbangkan D. Ya
7. Apakah Anda menyarankan teman Anda yang lain untuk menggunakan sistem ini dalam memilih jurusan di Perguruan Tinggi ?
- A. Tidak B. Tidak tahu
C. Mungkin D. Ya
8. Apakah bagi Anda tampilan sistem ini cukup menarik ?
- A. Belum Menarik B. Kurang Menarik
C. Cukup Menarik D. Sangat Menarik

BAB IV PEMBAHASAN

4.1. Lingkungan implementasi

Lingkungan implementasi yang akan dijelaskan dalam sub bab ini adalah lingkungan implementasi perangkat keras dan perangkat lunak

4.1.1. Lingkungan Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan sistem pakar ini adalah :

1. Prosesor Intel Pentium 4 2,8 Ghz
2. Memori 512 Mb
3. Harddisk 80 Gb
4. Monitor 15'
5. Keyboard
6. Mouse

4.1.2. Lingkungan perangkat Lunak

Perangkat lunak yang digunakan dalam pengembangan sistem pakar ini adalah :

1. Sistem operasi Windows XP SP2
2. Borland Delphi7
3. Microsoft SQLServer 2000

4.2. Implementasi Program

4.2.1. *Forward Chaining*

4.2.1.1. Inisialisasi tabel kerja

Inisialisasi tabel kerja dilakukan bila akan memulai sesi konsultasi baru. Tabel kerja (*workplace*) utama yang digunakan adalah tabel TempRule dan tabel TempPremis. Setelah inisialisasi dilakukan, *inference engine* siap dijalankan. Berikut adalah *Store procedure* yang digunakan untuk inisialisasi tabel kerja *forward chaining*

```
create proc init_tabel_kerja
as
  update TempPremis set UserValue = ", PCF = 0, PStatus = 'free',
  StatusTanya = 'a';
  update TempRule set RCF = 0, RStatus = 'active';
  update tjurusan set tcf=0
go
```

4.2.1.2. Proses Pengambilan Keputusan

Proses dalam pengambilan keputusan terbagi dalam beberapa tahap, yaitu tahap awal, cek status premis, cek aturan dan hitung RCF, serta tahap akhir atau penampilan rekomendasi.

- Tahap awal

Proses pengambilan keputusan pada *inference engine* dimulai dengan menerima *value* yang diberikan *user* setelah melalui rangkaian pertanyaan. Berikut adalah *store prosedur* yang digunakan untuk menampilkan pertanyaan.

```

create proc ambil_pertanyaan
  @@nopremis varchar(5) output,
  @@pertanyaan varchar(100) output
as
declare premis_cursor cursor for select tempremis.NoPremis, tpremis.premis
from (TPremis inner join Tempremis on TPreMis.NoPremis =
TempPremis.NoPremis)
where TempPremis.statustanya = 'a' order by tempremis.nopremis asc
open premis_cursor
fetch next from premis_cursor into @@nopremis, @@pertanyaan
while ((@@fetch_status=0) and (exists (select nopremis from tmetarule where
nopremis = @@nopremis)))
begin
  fetch next from premis_cursor into @@nopremis, @@pertanyaan
end
if @@fetch_status = -1
begin
  set @@nopremis = "
  set @@pertanyaan = "
end
close premis_cursor
deallocate premis_cursor
go

```

Apabila user memilih eksakta, maka *rule* non eksakta akan di eliminasi (RStatus = *discarded*). Selain itu, StatusTanya pada premis yang mempunyai kode awal Non Eksakta (N.....) di ubah menjadi *not ask*, begitu juga sebaliknya. Setiap premis akan dicek StatusTanya nya, jika *ask* maka premis tersebut ditanyakan. Ketika premis berada pada awal pertanyaan jurusan (misal nomor premis ES001), jika user menjawab “yes” maka premis-premis selanjutnya yang bertipe sama (ES....) akan ditanyakan, jika tidak StatusTanya premis-premis tersebut diubah menjadi *not ask* dan *rule* yang mempunyai tipe bidang keilmuan sains akan di eliminasi. Pada penggunaan metode ini, sistem akan memeriksa dan menelusuri semua kemungkinan bidang ilmu yang sesuai untuk mengetahui minat/keinginan, bakat, dan kemampuan user baik yang disadari maupun yang tidak disadari (misal bakat terpendam) agar dapat menentukan bidang studi terbaik untuk direkomendasikan. Berikut adalah *store prosedur* yang digunakan.

```

create proc jawab @nopremis varchar(5), @pcf float, @uservalue
varchar(3)
as
update temppremis set uservalue = @uservalue, pcf = @pcf, statustanya
= 'na' where nopremis=@nopremis
set @nopremis = rtrim(@nopremis)
--- jika no premis 4 digit dan merupakan pertanyaan awal 001
if (len(@nopremis)=4) and (right(@nopremis,3)='001')
begin
if @uservalue = 'no'
begin
--- jika dijawab 'no', yang punya kode awal sama tidak ditanya
update temppremis set statustanya = 'na', uservalue= "
where left(nopremis,1)=left(@nopremis,1) and left(nopremis,2)<>'PU'
end
end
if @uservalue = 'yes'
begin
--- jika dijawab 'yes', yang punya kode awal tidak sama tidak ditanya
update temppremis set statustanya = 'na'
where left(nopremis,1)<>left(@nopremis,1) and left(nopremis,2)<>'PU'
end
if (len(@nopremis)=5) and (right(@nopremis,3)='001')
begin
if @uservalue = 'no'
begin
--- jika dijawab no,yang punya kode awal jurusan sama tidak ditanya
update temppremis set statustanya = 'na'
where left(nopremis,2)=left(@nopremis,2) and left(nopremis,2)<>'PU'
end
end
end
go

```

Cek status premis

Setelah tahap awal, dilakukan proses pengecekan premis, dimana *value* tiap premis yang digunakan akan ditentukan statusnya apakah benar (*true*) atau salah (*false*). Berikut adalah *store prosedur* yang digunakan

```

create proc cek_premis
as
  update tempremis set pstatus = 'true' where uservalue = 'yes'
  and pcf <> 0
  update tempremis set pstatus = 'true', pcf = pcf*-1
  where uservalue = 'no' and pcf <> 0
  update tempremis set pstatus = 'false'
  where uservalue = "" and pcf = 0 and not exists
  (select * from tmetarule where nopremis = tempremis.nopremis )
go

```

- Cek Rule

Proses kemudian dilanjutkan dengan pengecekan aturan. Setiap aturan yang masih bersifat *active* akan diperiksa, jika ada premis dari aturan tersebut yang belum diketahui kebenarannya (masih ada yang *free*), maka statusnya dimarked. Pengecekan dilakukan sampai aturan yang berstatus *active* tidak ada. Setelah itu aturan yang berstatus *marked* di cek ulang sampai tidak ada aturan yang bersatus *marked* Jika semua premis dari aturan tersebut bernilai *true*, maka kesimpulan akhir aturan tersebut dianggap benar (*Rstatus = triggered*) dan dihitung RCF dan TCF nya. Berikut adalah beberapa *store procedure* yang digunakan



```
create proc cek_rule
as
```

```
--- yang didiscard yang pstatus = false
```

```
update temprule set rstatus = 'discarded' where norule in
(select trelasi.norule from trelasi inner join temppremis on trelasi.nopremis =
temppremis.nopremis where temppremis.pstatus= 'false' )
```

```
--- yang di discard yang punya kode bidangilmu sama yang awal pertanyaan
jurusan dijawab 'no'
```

```
update temprule set rstatus = 'discarded' where norule in
(select trule.norule from temppremis inner join trule on
left(temppremis.nopremis,2)= trule.bidilmu
where(right(nopremis,3)= '001' and uservalu='no')and(trule.bidilmu<>'PU'))
```

```
--- yang di discard yang punya tipe sama dg yang awal pertanyaan tipe
```

```
update temprule set rstatus = 'discarded' where norule in(select trule.norule
from trule inner join tilmu on trule.BidIlmu=TIllmu.Kode
inner join temppremis on left(temppremis.nopremis,1) = tilmu.tipe
where (right(nopremis,3) = '001' and uservalu='')and (tilmu.tipe <> 'A'))
```

```
--- yang di marked yang rstatusnya 'active' and pstatus premis2nya masih
ada yang 'free'(metarule)
```

```
update temprule set rstatus = 'marked' where norule in
(select trelasi.norule from ((trelasi inner join temprule on trelasi.nopremis =
trelasi.nopremis)
inner join temppremis on trelasi.nopremis = temppremis.nopremis)
where temprule.rstatus <> 'discarded' and trelasi.norule = temprule.norule
and temprule.rstatus = 'active' and temppremis.pstatus = 'free')
```

```
--- sisanya yang ga ke discard n ga di marked di triger (artinya, Pstatusnya
True semua)
```

```
update temprule set rstatus = 'triggered' where rstatus = 'active'
```

```
go
```

```

create proc marked
as
  declare @n integer
  set @n = 0
  while @n<=2
  begin
    delete from tcf
    execute hitungcf1 ---> (Cari nilai minimum tiap aturan dan hitung RCF)
    execute hitungcf2 --- > (Hitung TCF)
    execute update_temp ---> (Update Pstatus, StatusTanya dan PCF)
    execute cek_rule2 --- > (Ubah status dari yang marked ke triger)
    --berhenti jika yang berstatus marked sudah tdk ada + 1 kali cek lagi
    --(ketika marked udah ga ada, dia tetap menghitung cf n cek rule yg tersisa)
    if not exists(select * from temprule where rstatus = 'marked')
    begin
      set @n = @n + 1
    end
  end
end
go

```

Store Procedure untuk hitung CF1, Hitung CF2, update_temp dan cek rule2 dapat dilihat pada lampiran

- Tahap akhir
Proses terakhir adalah proses penampilan kesimpulan. Jumlah kesimpulan yang ditampilkan tergantung input 'rank' yang diminta pengguna

```

create proc forwardchaining @rank integer
as
  execute cek_premis
  execute cek_rule
  execute marked
  exec ('select top ' + @rank + ' kdjurusan,jurusan,tcf from tjurusan where tcf
  <> 0 order by tcf desc')
go

```

4.2.2. *Backward Chaining*

4.2.2.1. Inisialisasi tabel kerja

Inisialisasi tabel kerja dilakukan bila akan memulai sesi konsultasi baru. Seperti pada *Forward Chaining*, Tabel kerja (*workplace*) utama yang digunakan adalah tabel TempRule dan tabel TempPremis. Setelah inisialisasi dilakukan, *inference engine* siap dijalankan. Terdapat perbedaan status tanya pada inisialisasi *backward chaining*. Berikut adalah *Store procedure* yang digunakan untuk inisialisasi tabel kerja *backward chaining*

```
create proc init_tabel_kerja_bc
as
delete from temp2 --- > berisi daftar kode kesimpulan yang akan dicari(goal)
update TempPremis set UserValue = ", PCF = 0, PStatus = 'free',
StatusTanya = 'na';
update TempRule set RCF = 0, RStatus = 'active';
update tjurusan set tcf=0
go
```

4.2.2.2. Proses *Backward Chaining*

Backward Chaining dimulai dengan menentukan jurusan yang akan menjadi *goal* (jurusan yang akan dicek). Kemudian dicari semua aturan yang dipakai oleh jurusan tersebut beserta premis yang digunakan. Premis yang digunakan, status tanyanya akan di '*ask*' dan *rule* yang tidak dipakai akan di *discard*.

Berikut adalah *store procedure* yang digunakan

```

create proc cari_rule_premis_bc
as
declare @i integer
declare @y integer
create table #temp
( norule varchar(5),
  premis varchar(5))
set @i = 1
while (1=1)--- while(true)
begin
  set @y = @i - 1
  --- isi #temp dengan premis2 dari rule yang terpake
  insert into #temp select trule.norule,trelasi.nopremis from
    (trule inner join trelasi on trule.norule = trelasi.norule)
    where trule.kdkesimp in (select kdkesimp from temp2 where status=@y)
    and trelasi.nopremis not in (select nopremis from tmetarule)

  --- update isi temp2 dengan kode kesimp rule yang mau dicari premisnya
  lagi (cari yang metarule)
  insert temp2(kdkesimp,status) select kdkesimp,@i from trule
  where kdkesimp in (select distinct trelasi.nopremis
    from (trule inner join trelasi on trule.norule = trelasi.norule)
    where trule.kdkesimp in (select kdkesimp from temp2 where status=@y)
    and trelasi.nopremis in (select nopremis from tmetarule))

  --- jika ternyata tidak ada yg metarule lagi
  if @@rowcount = 0
    break
  select @i = @i + 1
end

--- premis yang dipakai di 'ask'
update tempremis set statustanya = 'a' where nopremis in (select premis
from #temp)

--- rule yang ga dipakai di discard
update temprule set rstatus = 'discarded' where norule not in (select norule
from #temp)
drop table #temp
go

```

- Cek Rule

Proses kemudian dilanjutkan dengan pengecekan aturan. Setiap aturan yang masih bersifat *active* akan diperiksa, jika ada premis dari aturan tersebut yang belum diketahui kebenarannya, maka statusnya *dimarked* dan beralih pada aturan selanjutnya sampai aturan yang berstatus *active* tidak ada. Setelah itu aturan yang berstatus *marked* di cek ulang sampai tidak ada aturan yang bersatus *marked*. Jika semua premis dari aturan tersebut bernilai *true*, maka kesimpulan akhir aturan tersebut dianggap benar ($Rstatus = triggered$) dan dihitung RCF dan TCF nya. Berikut adalah beberapa *store procedure* yang digunakan.

```
create proc cek_rule_bc
as
--- update rstatus=marked jika rstatus = active n salah satu dari premis masih
ada yg free alias metarule
update temprule set rstatus = 'marked' where norule in
(select trelasi.norule from ((trelasi inner join temprule on trelasi.nopremis =
trelasi.nopremis)
inner join temppremis on trelasi.nopremis = temppremis.nopremis)
where temprule.rstatus <> 'discarded' and trelasi.norule = temprule.norule
and temprule.rstatus = 'active' and temppremis.pstatus = 'free')
update temprule set rstatus = 'triggered' where rstatus = 'active'
go
```

- Tahap akhir

Proses terakhir adalah proses penampilan hasil. Hasil yang ditampilkan adalah nilai total CF (TCF) dari jurusan yang menjadi *'goal'* (jurusan yang dicek).

```
create proc backwardchaining2
as
execute cek_premis
execute cek_rule_bc
execute marked
select kdjurusan,jurusan,tcf from tjurusan where tcf <> 0 order by tcf desc
go
```

4.3. Implementasi Sistem

1. User Login

Form ini (Gambar 4.1) mengawali sistem pakar rekomendasi program studi ini. User diminta memasukkan nama dan menekan tombol ‘ masuk ’ untuk memulai program



Gambar 4.1 Form User Login

2. Home

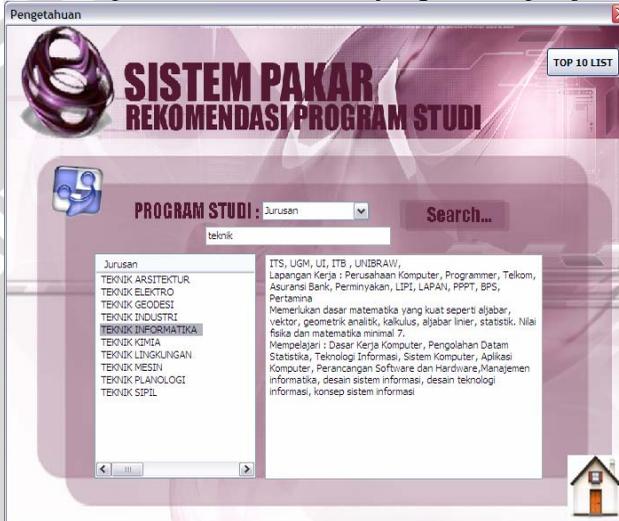
Pada Sistem Pakar ini, terdapat 3 Menu Utama pada Form *Home* (Gambar 4.2) Ketiga menu utama tersebut adalah Menu Pengetahuan, Konsultasi dan Pengaturan



Gambar 4.2 Form Home

3. Pengetahuan

Menu pengetahuan (Gambar 4.3) digunakan untuk mencari informasi tentang jurusan-jurusan. Untuk sementara, hanya tersedia 48 pilihan jurusan dari berbagai Universitas Negeri di Indonesia. Informasi juga mencakup lapangan kerja, hal-hal yang diperhatikan dan perkiraan mata kuliah yang akan dipelajari.



Gambar 4.3 Form Pengetahuan

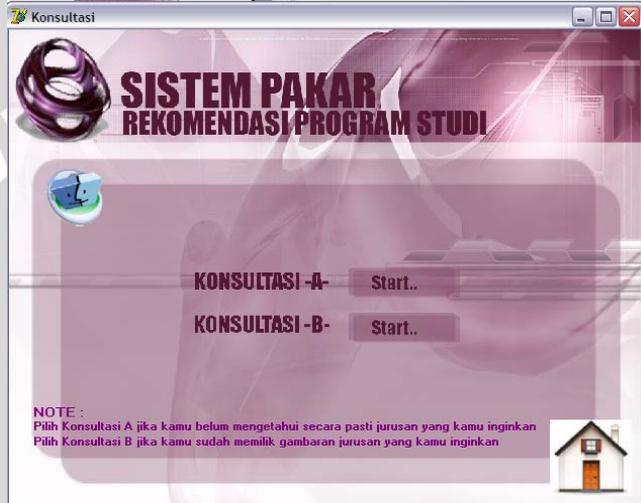
Kolom ke dua pada form diatas menampilkan penjelasan dari jurusan yang terpilih di kolom jurusan. Pada Form pengetahuan ini, terdapat tombol 'Top 10 List' yang menampilkan daftar 10 jurusan yang paling banyak masuk daftar pencarian.



Gambar 4.4 Top 10 List

4. Konsultasi

Menu konsultasi (gambar 4.5) dibagi menjadi 2 sub menu, yaitu konsultasi A dan konsultasi B. Konsultasi A digunakan jika calon mahasiswa tidak memiliki gambaran tentang program studi/jurusan yang diminati, sedangkan konsultasi B digunakan jika calon mahasiswa tersebut sudah memiliki gambaran program studi yang ingin diambil.



Gambar 4.5 Form Konsultasi

➤ *Forward Chaining (Konsultasi A)*

Berikut adalah beberapa contoh pertanyaan yang diajukan pada konsultasi A (Gambar 4.6 dan Gambar 4.7).

Pada form ini, *user* diminta memilih jawaban 'Ya' (*yes*) atau 'Tidak' (*no*). Selain itu, *user* juga diminta untuk menentukan tingkat kepastian dengan memilih salah satu *radiobutton* yang tersedia (Ragu-ragu, mungkin, kemungkinan besar, pasti/yakin, sangat pasti/sangat yakin). Jawaban yang diinputkan *user* akan ditampilkan pada form tanya saat itu juga. Setelah *user* yakin dengan jawaban yang diinputkan, *user* dapat menekan tombol *next* untuk menampilkan pertanyaan selanjutnya. Tombol Stop digunakan jika *user* ingin memaksa sistem menghentikan pertanyaan dan langsung memulai proses inferensi sebelum semua pertanyaan terjawab dengan hanya sedikit informasi yang telah diberikan

The screenshot shows a window titled "Tanya" with a "Stop" button in the top right corner. The main content area has a purple background with a keyboard and a hand. At the top left is a logo of a purple ring. The title "SISTEM PAKAR REKOMENDASI PROGRAM STUDI" is displayed in bold. The question "E001 Apakah kamu Berminat pada jurusan Eksakta ?" is shown in purple. Below it, the text "yes 0.6" is displayed. There are two buttons: "YES" and "NO". Below these are three radio button options: "Ragu-ragu", "Mungkin", and "Kemungkinan Besar", with the third one selected. At the bottom, there are two more radio button options: "Pasti / Yakin" and "Sangat Pasti / Sangat Yakin". A "NEXT" button is located at the bottom right.

Gambar 4.6 Form Tanya(1)

The screenshot shows a window titled "Tanya" with a "Stop" button in the top right corner. The main content area has a purple background with a keyboard and a hand. At the top left is a logo of a purple ring. The title "SISTEM PAKAR REKOMENDASI PROGRAM STUDI" is displayed in bold. The question "ES002 Apakah kamu Ingin mempelajari ilmu sains ?" is shown in purple. Below it, the text "yes 0.6" is displayed. There are two buttons: "YES" and "NO". Below these are three radio button options: "Ragu-ragu", "Mungkin", and "Kemungkinan Besar", with the third one selected. At the bottom, there are two more radio button options: "Pasti / Yakin" and "Sangat Pasti / Sangat Yakin". A "NEXT" button is located at the bottom right.

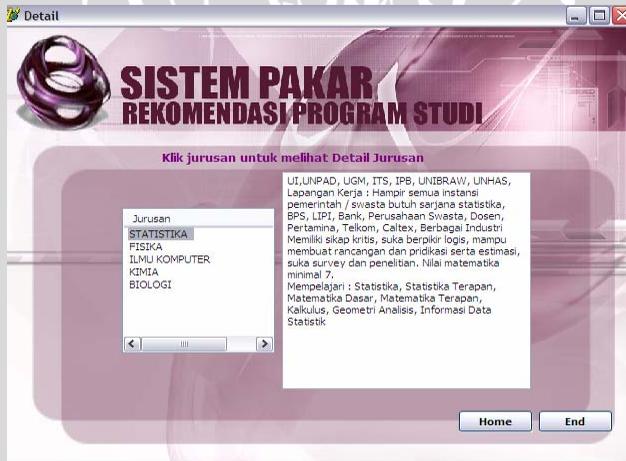
Gambar 4.7 Form Tanya(2)

Berikut adalah hasil dari sesi konsultasi A setelah melakukan tanya jawab terhadap salah satu calon mahasiswa. Dari sesi tanya jawab tersebut diperoleh 5 pilihan jurusan yang memiliki CF tertinggi dengan jurusan Statistika dan Fisika pada ranking pertama dan kedua



Gambar 4.8 Hasil Konsultasi A

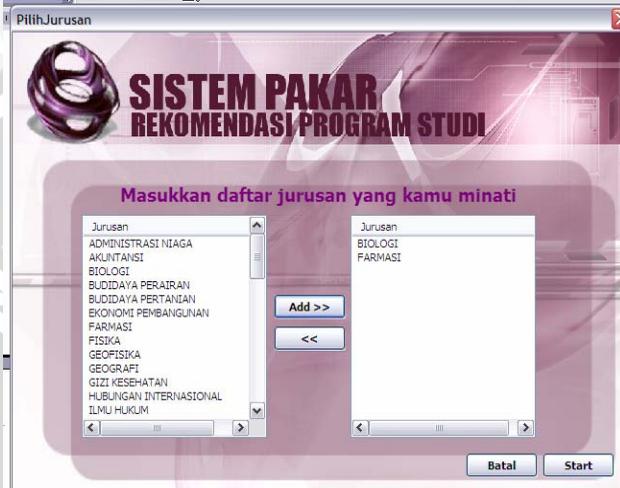
Untuk melihat informasi jurusan dari ke lima rekomendasi program studi, dapat diklik tombol detail di pojok kanan bawah form hasil konsultasi. Pada form Detail, untuk melihat tampilan informasi masing-masing jurusan, klik nama jurusan pada box sebelah kiri.



Gambar 4.9 Form Detail Konsultasi

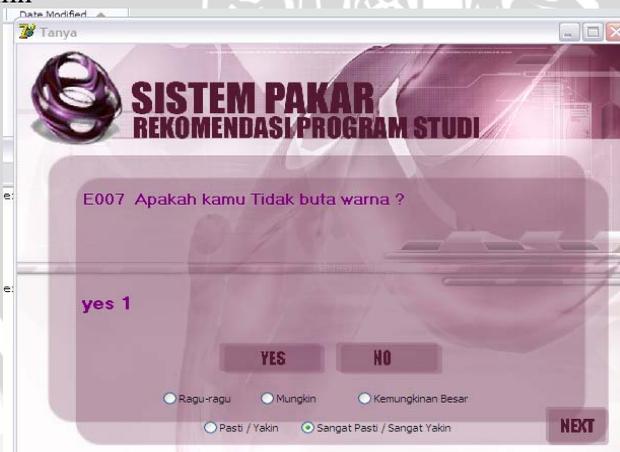
➤ **Backward Chaining (Konsultasi B)**

Konsultasi B dimulai dengan memasukkan daftar jurusan yang diminati sebagai ‘Goal’ dari *Backward Chaining*.



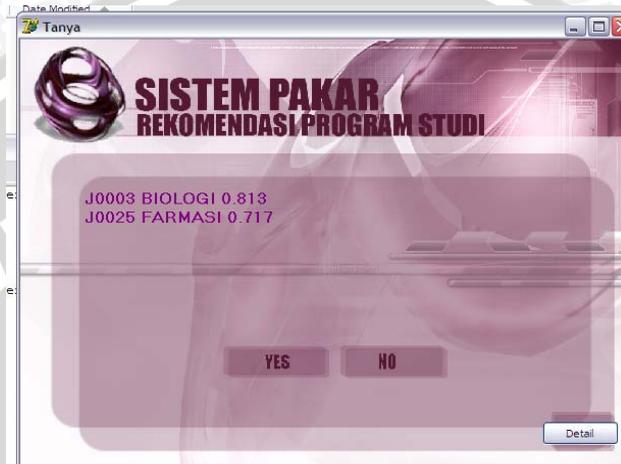
Gambar 4.10 Form Pilih Jurusan

Tekan tombol Start untuk memulai sesi tanya jawab konsultasi B. Sama seperti pada *forward Chaining*, akan muncul beberapa pertanyaan yang berkaitan dengan jurusan yang dipilih



Gambar 4.11 Form Tanya(3)

Setelah sesi tanya jawab berlangsung, akan diketahui hasil dari sesi konsultasi seperti contoh dibawah ini



Gambar 4.12 Hasil Konsultasi B

Dari hasil konsultasi, dapat diketahui bahwa calon mahasiswa ini lebih cocok di program studi biologi. Detail jurusan dapat dilihat pada form detail dengan menekan tombol detail pada form.

5. Pengaturan

Menu pengaturan hanya digunakan untuk administrator. Oleh karena itu, harus login terlebih dahulu sebagai administrator sistem. Terdapat 3 sub menu, yaitu pengaturan Jurusan, Premis dan Aturan (*Rule*). Pada menu-menu inilah administrator dapat mengubah, menghapus, dan menambah baik jurusan, premis maupun aturan. Perubahan apapun yang dilakukan administrator harus dengan sepengetahuan dan seijin Ahli / Pakar dalam pembuatan sistem pakar ini. Form dapat dilihat pada gambar 4.13.



Gambar 4.13 Form Pengaturan

- Login Admin
Untuk dapat mengubah sistem, administrator harus *login* terlebih dahulu



Gambar 4.14 Form Login Admin

- Atur Jurusan/Program studi
Pada form ini, administrator dapat menambah, mengedit, dan menghapus data jurusan yang tersimpan pada database

Atur1

SISTEM PAKAR REKOMENDASI PROGRAM STUDI

Jurusan : Cari

Hasukkan nama jurusan secara lengkap

Kode Jurusan :

Jurusan :

Bidang Ilmu :

Universitas :

Detail

LapanganKerja : Lembaga Biologi Nasional, UPT, Pertamina, Industri Farmasi, Industri Peternakan, Kesehatan, Kehutanan, Dosen, Industri Makanan dan Minuman, Ahli Ekologi, Konsultan Produk

Tidak perlu berbekal jenius, diperlukan kerajinan, mempunyai kesukaan terhadap matematika. Nilai matematika minimal 7.

Mempelajari : Analisis, Aljabar, Matematika Terapan, Geometri Analis, Kalkulus, Finite Elemen Method, Permodelan Sistem

Buttons: Back, Tambah, Simpan, Hapus, Batal, Premis, Aturan

Gambar 4.15 Form Atur Jurusan

➤ Atur Premis

Pada form ini, administrator dapat menambah, mengedit, dan menghapus data premis yang tersimpan pada database

AturPremis

SISTEM PAKAR REKOMENDASI PROGRAM STUDI

Buttons: Back, Next

No Premis	Premis
EK001	Tertarik pada bidang Kesehatan
EK002	Ingin mempelajari ilmu kesehatan
EK003	Bidang Keilmuan Kesehatan
EK004	Ingin belajar ilmu untuk merawat orang dan profesi keperawatan
EK005	Ingin belajar mengobati penyakit manusia
EK006	Ingin belajar mengobati penyakit terutama gigi dan mulut
EK007	Ingin belajar mengobati penyakit hewan
EK008	Ingin belajar psikologi seseorang, psikologi industri, psikologi pendidikan, dll
EK009	Ingin belajar meracik obat, kimia farmasi, farmasi klinik
EK010	Ingin belajar kandungan gizi makanan dan menu makanan sehat

No Premis:

Premis:

Buttons: Tambah, Edit, Hapus, Simpan, Bidang Ilmu

Gambar 4.16 Form Atur Premis

- **Atur Bidang Ilmu**
Pada form ini, administrator dapat menambah, mengedit, dan menghapus data bidang ilmu yang tersimpan pada database

Kode	Bidang Ilmu	Tipe
EH	Peternakan	Eksakta
EJ	Perikanan	Eksakta
EK	Kesehatan	Eksakta
EP	Pertanian	Eksakta
ES	Sains	Eksakta
ET	Teknik	Eksakta
NB	Bahasa/Sastra	Non Eksakta

Gambar 4.17 Form Atur Bidang Ilmu

- **Atur Aturan (Rule)**
Pada form ini, administrator dapat menambah, mengedit, dan menghapus Aturan yang tersimpan pada database

No Rule	Kode Kesimp	Kesimpulan	CF	Bidang Ilmu
R0001	FU004	Wawasan Luas	0.76	Umum
R0002	FU004	Wawasan Luas	0.72	Umum
R0003	FU008	Disiplin tinggi	0.81	Umum
R0004	ES003	Bidang Keilmuan Sains	0.76	Sains
R0005	ES003	Bidang Keilmuan Sains	0.71	Sains
R0006	ET003	Bidang Keilmuan Teknik	0.77	Teknik
R0007	ET003	Bidang Keilmuan Teknik	0.83	Teknik
R0008	EK003	Bidang Keilmuan Kesehatan	0.75	Kesehatan
R0009	EK003	Bidang Keilmuan Kesehatan	0.81	Kesehatan
R0010	EP003	Bidang Keilmuan Pertanian	0.8	Pertanian
R0011	EP003	Bidang Keilmuan Pertanian	0.72	Pertanian

Gambar 4.18 Form Atur Rule

Pada form atur rule, terdapat pilihan tambah, edit, hapus dan detail aturan. Untuk menambah atau mengedit suatu aturan, terdapat form berikut

Add/Edit

SISTEM PAKAR REKOMENDASI PROGRAM STUDI

NoRule: R0003 Bidang Ilmu: Umum

CF: 0.81

Kode Kesimpulan: Premis Jurusan PU008

No Premis	Premis
PU001	Suka menonton berita
PU002	Suka membaca Koran, buku atau Maj...
PU003	Suka Browsing Internet
PU004	Wawasan Luas
PU005	Selalu tepat waktu
PU006	Tidak suka menunda pekerjaan
PU007	Menepati peraturan
PU008	Disiplin tinggi
PU009	Teliti & Sabar

Buttons: Simpan, Batal

Gambar 4.19 Form Add/Edit Rule

Sedangkan untuk menambah, mengedit atau menghapus premis yang digunakan dalam suatu aturan (relasi), tersedia form detail rule berikut

RuleDetail

SISTEM PAKAR REKOMENDASI PROGRAM STUDI

NoRule: R0004 Bidang Ilmu: Eksakta

Kesimpulan: ES003 Bidang Keilmuan Sains

Premis: Premis :

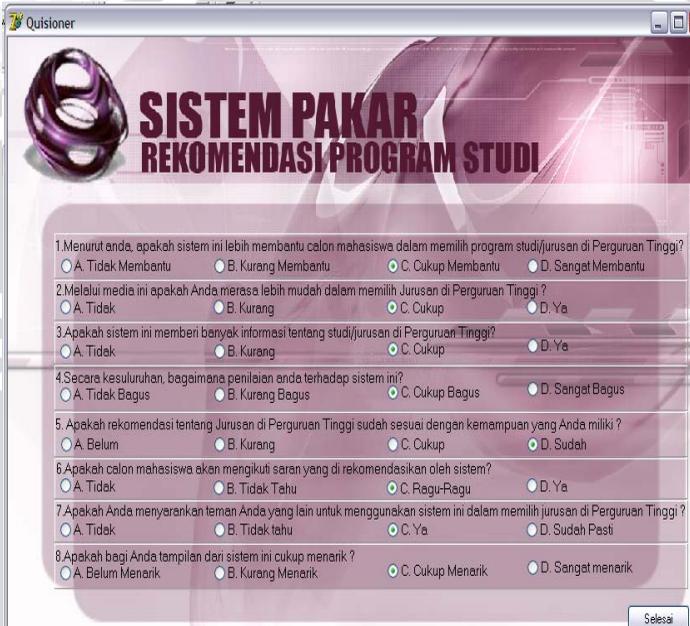
NoPremis	Premis
E001	Berminat pada jurusan Eksakta
E002	Mempunyai nilai rata-rata fisika > 7
E003	Mempunyai nilai rata-rata kimia > 7
E004	Mempunyai nilai rata-rata matematika > 7
E005	Mempunyai nilai rata-rata biologi > 7
E006	Peduli pada alam dan lingkungan sekitar
E007	Tidak bura warna
E008	Tidak mudah jijk terhadap sesuatu
E009	Suka dan tidak takut binatang
E010	Suka kegiatan di laboratorium
E011	Suka Menggambar
E012	Mempunyai fisik dan daya tahan yang baik
EH001	Tertarik pada bidang Peternakan
EH002	Ingin mempelajari ilmu Peternakan
EH003	Bidang Keilmuan Peternakan
EH004	Ingin belajar tentang pemeliharaan hewan ternak...
EH005	Ingin belajar mengenai pakan hewan ternak, teh...
FH006	Ingin belajar manajemen dan pemasaran us...

Buttons: Simpan, Hapus, Batal

Gambar 4.20 Form Rule Detail

4.4. Kuisisioner

Setelah melakukan konsultasi, para calon mahasiswa diminta mengisi kuisisioner untuk mengetahui tingkat keberhasilan sistem ini



The image shows a screenshot of a questionnaire window titled "SISTEM PAKAR REKOMENDASI PROGRAM STUDI". The window contains eight questions, each with four radio button options (A, B, C, D). The questions are:

1. Menurut anda, apakah sistem ini lebih membantu calon mahasiswa dalam memilih program studi/jurusan di Perguruan Tinggi?
 A. Tidak Membantu B. Kurang Membantu C. Cukup Membantu D. Sangat Membantu
2. Melalui media ini apakah Anda merasa lebih mudah dalam memilih Jurusan di Perguruan Tinggi ?
 A. Tidak B. Kurang C. Cukup D. Ya
3. Apakah sistem ini memberi banyak informasi tentang studi/jurusan di Perguruan Tinggi?
 A. Tidak B. Kurang C. Cukup D. Ya
4. Secara keseluruhan, bagaimana penilaian anda terhadap sistem ini?
 A. Tidak Bagus B. Kurang Bagus C. Cukup Bagus D. Sangat Bagus
5. Apakah rekomendasi tentang Jurusan di Perguruan Tinggi sudah sesuai dengan kemampuan yang Anda miliki ?
 A. Belum B. Kurang C. Cukup D. Sudah
6. Apakah calon mahasiswa akan mengikuti saran yang di rekomendasikan oleh sistem?
 A. Tidak B. Tidak Tahu C. Ragu-Ragu D. Ya
7. Apakah Anda menyarankan teman Anda yang lain untuk menggunakan sistem ini dalam memilih jurusan di Perguruan Tinggi ?
 A. Tidak B. Tidak tahu C. Ya D. Sudah Pasti
8. Apakah bagi Anda tampilan dari sistem ini cukup menarik ?
 A. Belum Menarik B. Kurang Menarik C. Cukup Menarik D. Sangat menarik

A "Selesai" button is located at the bottom right of the questionnaire form.

Gambar 4.21 Kuisisioner

4.5. Pengujian Sistem

4.5.1. Studi Kasus

Studi kasus yang diambil adalah siswa kelas 3 SMAN 2 Kota Pasuruan angkatan 2007/2008. Jumlah keseluruhan populasi adalah 234 siswa, sehingga diperlukan sampel sebanyak :

$$n = 234 / (1 + 234(0,1)^2) = 71 \text{ siswa}$$

Pengujian dilakukan baik yang mengalami kesulitan menentukan bidang studi maupun tidak. Untuk yang mengalami kesulitan, diharapkan dapat membantu menentukan program studi yang sesuai,

dan untuk yang tidak mengalami kesulitan dapat dijadikan pertimbangan

4.5.2. Hasil Quisioner

Dari kuisisioner awal, diperoleh data dari 71 responden, 66 siswa (92,96%) diantaranya ingin melanjutkan studi Sarjana/ Strata1, sedangkan sisanya 5 orang (7,04%) tidak ingin melanjutkan studi lebih lanjut. Dari 66 siswa yang ingin melanjutkan studi, 47 orang diantaranya (71,27%) mengaku masih bingung dalam menentukan jurusan/ program studi yang ingin diambil. Walaupun begitu, banyak diantaranya yang sudah memiliki sedikit gambaran tentang jurusan yang diinginkan, tetapi masih memerlukan pertimbangan lebih lanjut

Kuisisioner akhir yang diberikan kepada siswa memiliki empat pilihan jawaban yaitu A, B, C, dan D dengan jumlah pertanyaan sebanyak 8. Pertanyaan selengkapnya dapat dilihat pada bab 3 halaman 47.

Jawaban A : mempresentasikan bahwa sistem ini tidak membantu pemilih dalam mendapatkan informasi pemilihan jurusan.

Jawaban B : mempresentasikan bahwa sistem ini kurang membantu dalam memilih jurusan di perguruan tinggi.

Jawaban C : mempresentasikan bahwa sistem ini sudah cukup mewakili dan cukup membantu dalam memilih jurusan di perguruan tinggi.

Jawaban D : mempresentasikan bahwa sistem ini sangat membantu calon mahasiswa dalam mendapatkan informasi pemilihan jurusan serta memilih jurusan di perguruan tinggi berdasarkan bidang minat dan kemampuan yang dimiliki.

Dari hasil jawaban kuisisioner tersebut diperoleh bahwa:

9. 59 siswa (89,39%) menyatakan sistem ini cukup membantu calon mahasiswa dalam memilih jurusan di Perguruan Tinggi (memilih jawaban C) dan 7 siswa (10,61%) menyatakan sangat membantu calon mahasiswa dalam memilih jurusan di Perguruan Tinggi (memilih jawaban D).
10. 62 siswa (93,4%) menyatakan sistem ini mempermudah calon mahasiswa dalam pemilihan jurusan di perguruan tinggi karena

tidak harus bertemu langsung dengan ahlinya. (memilih jawaban C dan D)

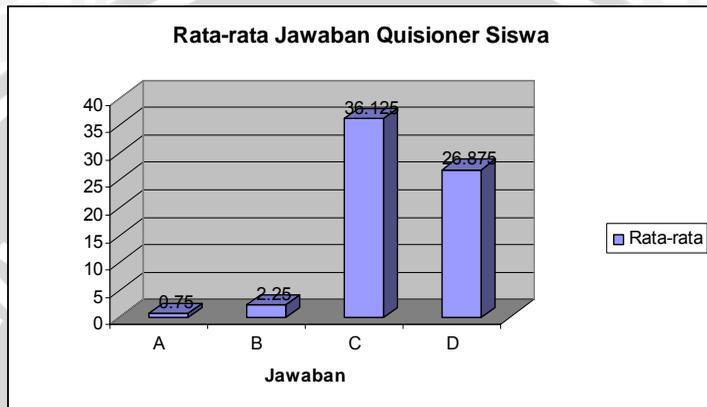
11. 11 siswa (16,67%) menyatakan sistem ini memberi cukup informasi jurusan di Perguruan Tinggi (memilih jawaban C) dan 55 siswa (83,33%) menyatakan sistem telah banyak memberikan informasi jurusan di Perguruan Tinggi (memilih jawaban D)
12. 27 siswa (40,9%) menilai sistem ini secara keseluruhan cukup bagus dan 39 siswa lainnya (59,1%) menilai bahwa sistem ini secara keseluruhan sangat bagus
13. 50 siswa (75,76%) menyatakan rekomendasi/ saran yang diberikan sistem ini sudah sesuai dengan kemampuan yang mereka miliki
14. Sebagian besar pengguna (56 siswa; 84,85 %) akan mempertimbangkan saran yang diberikan sistem (memilih jawaban C) dan 9 siswa (13,64%) menyatakan akan mengikuti rekomendasi sistem (memilih jawaban D). Sedangkan sisanya (1 siswa; 1,5 %) masih ragu dengan rekomendasi yang diberikan oleh sistem ini.
15. 34 siswa (51,52%) menyatakan akan menyarankan kepada teman-teman yang lain untuk mencoba sistem pakar ini, 29 siswa (43,94%) menyatakan mungkin akan menyarankan sistem pakar ini kepada teman-temannya, sedangkan sisanya tidak tahu (3 siswa, 4,54%)
16. 41 siswa (62,12%) menilai bahwa tampilan sistem ini cukup menarik dan 25 siswa lainnya (37,88%) menilai bahwa tampilan sistem ini sangat menarik

Tabel berikut memperlihatkan hasil kuisioner selengkapnya

Tabel 4.1. Tabel Jawaban Kuisioner

Pertanyaan	A	B	C	D
1	0	0	59	7
2	0	4	32	30
3	0	0	11	55
4	0	0	27	39
5	6	10	34	16
6	0	1	56	9
7	0	3	29	34
8	0	0	41	25

Jumlah	6	18	289	215
Rata-rata	0.75	2.25	36.125	26.875
%	1,14%	3,41%	54,73%	40,72%



Gambar 4.22. Rata-rata jawaban Quisioner Siswa

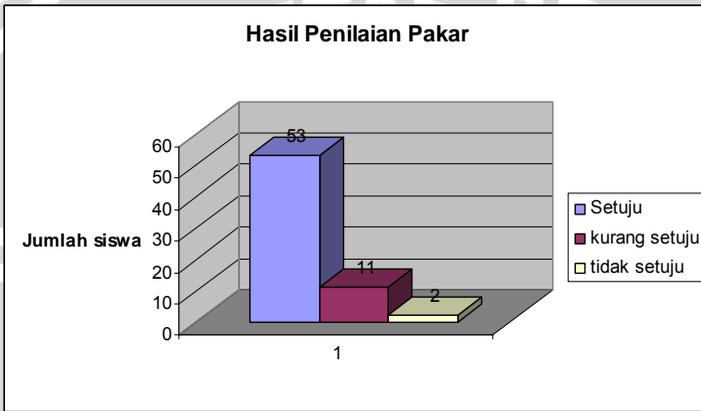
Dari tabel 4.1 dan gambar 4.22 diatas dapat dilihat bahwa mayoritas siswa memilih jawaban C, bahwa sistem pakar ini sudah cukup mewaili dan cukup membantu dalam memilih jurusan / program studi di perguruan tinggi (54,73%, rata-rata 36,125), disusul dengan jawaban D bahwa sistem pakar ini sangat membantu calon mahasiswa dalam memilih jurusan / program studi (40,72%, rata-rata 26,875)

Untuk menguji sistem pakar ini, juga diminta kesediaan Guru Bimbingan dan Konseling sebagai pakar di bidang pendidikan terutama rekomendasi bidang studi untuk memberi penilaian.

Dari Penilaian tersebut diperoleh bahwa :

1. Setuju dengan rekomendasi yang diberikan Sistem Pakar = 53 siswa (80,30 %)
2. Kurang Setuju dengan rekomendasi yang diberikan Sistem Pakar = 11 siswa (16,67 %). Alasan yang dikemukakan diantaranya adalah, ada beberapa rekomendasi yang dianggap kurang sesuai dengan siswa, selain itu sistem juga mengeluarkan rekomendasi dengan CF jurusan sangat rendah (dibawah 0,01), Guru BK menyarankan, tidak menampilkan rekomendasi yang nilai CF nya sangat rendah.

3. Tidak Setuju dengan rekomendasi Sistem Pakar = 2 siswa, (3,03%). Alasan yang dikemukakan, Sistem hanya merekomendasikan 1 jurusan dan jurusan tersebut tidak sesuai dengan penjurusan yang diambil oleh siswa di SMA nya sekarang. Misalnya, Siswa Frisca Dyah P, Kelas IPS, sistem hanya merekomendasikan Teknik arsitektur dengan $CF = 0,279$



Gambar 4.23. Hasil Penilaian pakar terhadap rekomendasi sistem



Gambar 4.24. Persentase tingkat persetujuan pakar terhadap rekomendasi sistem

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Kesimpulan yang dapat diambil dari penulisan Tugas Akhir ini adalah sebagai berikut :

1. Sistem pakar berbasis aturan dapat digunakan untuk rekomendasi program studi/ jurusan.
2. Dalam menentukan rekomendasi program studi yang sesuai berdasarkan fakta-fakta yang ada, dapat digunakan Kaidah Produksi sebagai representasi pengetahuan dengan bantuan metode inferensi *forward chaining* dan *backward chaining*.
3. Mayoritas siswa (dari hasil kuisisioner) menyatakan bahwa sistem pakar rekomendasi program studi ini cukup membantu dan mempermudah siswa untuk memilih program studi jika melanjutkan studi ke jenjang Sarjana / Strata 1.

5.2. Saran

Saran yang dapat diberikan untuk pengembangan sistem pakar lebih lanjut adalah :

1. Menambahkan fasilitas penjelasan dan perbaikan pengetahuan pada sistem pakar berikutnya.
2. Menambahkan suatu prosedur atau fungsi agar tidak menampilkan terlalu banyak pertanyaan.
3. Lebih memperdetail basis pengetahuan dengan informasi-informasi lain, baik jumlah program studi maupun keterangan-keterangan lainnya termasuk aturan-aturan yang digunakan.

DAFTAR PUSTAKA

- Adrianto, Roy, 2003, *Perancangan Program Aplikasi Sistem pakar untuk diagnosa awal kanker kandungan*, Surabaya: UK Petra.
- Arhami, Muhammad, 2005, *Konsep Dasar Sistem Pakar*, Yogyakarta: ANDI OFFSET.
- Dinarga, M.Sinar dkk, 2006, *Jurusan Apa Buat kamu (non eksakta)* Yogyakarta: ANDI OFFSET.
- Dinarga, M.Sinar dkk, 2006, *Jurusan Apa Buat kamu (eksakta)* Yogyakarta: ANDI OFFSET.
- D.G.Dogolite, 1993, *Developing Knowledge - based Systems using VP-Expert*. New York : Macmillan Publishing Company.
- Kusrini, 2006, *Sistem Pakar, Teori dan Aplikasi*, Yogyakarta: ANDI OFFSET.
- Kusumadewi, Sri, 2003, *Artificial Intelligence (Teknik dan Aplikasinya)*, Yogyakarta: PT. GRAHA ILMU
- Martin, J & Oxman, S, 1988, *Building Expert System a tutorial* , New Jersey :Prentice Hall Inc
- McLeod, Raymond.Jr, 1995, *Management Information System, 6 ed.* New Jersey :Prentice Hall Inc
- Pribadi, Riska P., 2005, *Perancangan dan Pembuatan Rule Based System untuk informasi pemilihan jurusan*, Surabaya : PENS-ITS

Primagama, 2006, *Panduan memilih jurusan* , Yogyakarta :
Primagama

Schnupp, Peter H, 1989, *Building Expert System in Prolog*,
Munich
: Amzy!Inc

Setiawan, Andi, 1993, *ARTIFICIAL INTELLIGENCE*,
Yogyakarta: ANDI OFFSET.

Tempo, 2007, *Panduan memilih jurusan* , Jakarta: Penerbit Tempo

Turban, E, 1995, *Decission Support System and Expert
Systems*, USA: Prentice Hall International Inc.



LAMPIRAN 1
ATURAN YANG DIPAKAI

			R9	IF [EK001] AND [E008] AND [PU009] AND [PU011] THEN [EK003], CF =0,81
R1	IF [PU001] AND [PU002] THEN [PU004], CF = 0,76			
R2	IF [PU002] AND [PU003] THEN [PU004], CF = 0,72		R10	IF [EP001] AND [EP004] AND [PU014] THEN [EP003], CF = 0,80
R3	IF [PU005] AND [PU006] AND [PU007] THEN [PU008], CF = 0,81		R11	IF [EP001] AND [E006] THEN [EP003], CF = 0,72
R4	IF [ES001] AND [PU004] AND [PU009] AND [PU010] THEN [ES003], CF = 0,76		R12	IF [EI001] AND [PU014] AND [E009] THEN [EI003], CF = 0,75
R5	IF [ES001] AND [PU008] AND [PU011] THEN [ES003], CF = 0,71		R13	IF [EI001] AND [E006] AND [E008] THEN [EI003], CF = 0,70
R6	IF [ET001] AND [PU013] AND [PU012] THEN [ET003], CF = 0,77		R14	IF [EH001] AND [PU014] THEN [EH003], CF = 0,78
R7	IF [ET001] AND [PU014] AND [PU009] AND [E007] THEN [ET003], CF =0,83		R15	IF [EH001] AND [E006] AND [E008] THEN [EH003], CF = 0,72
R8	IF [EK001] AND [E007] AND [PU008] AND [PU015] THEN [EK003], CF =0,75		R16	IF [NE001] AND [PU019] AND [PU011] THEN [NE003], CF = 0,78
			R17	IF [NE001] AND [PU012] AND [N006] THEN [NE003], CF = 0,76

R18 IF [NH001]
AND [N006]
AND [PU010]
THEN [NH003], CF = 0,8

R19 IF [NH001]
AND [PU016]
AND [NH004]
THEN [NH003], CF = 0,75

R20 IF [NS001]
AND [N008]
AND [PU011]
THEN [NS003], CF = 0,81

R21 IF [NS001]
AND [PU009]
AND [N009]
THEN [NS003], CF = 0,75

R22 IF [NB001]
AND [NB004]
AND [PU015]
THEN [NB003], CF = 0,82

R23 IF [NB001]
AND [PU018]
AND [N010]
THEN [NB003], CF = 0,78

R24 IF [ES002]
AND [ES004]
THEN [J0001], CF = 0,80

R25 IF [ES003]
AND [E002]
AND [E010]
THEN [J0001], CF = 0,76

R26 IF [ES002]
AND [E003]
AND [ES005]
THEN [J0002], CF = 0,82

R27 IF [ES003]
AND [PU018]
AND [E010]
THEN [J0002], CF = 0,77

R28 IF [ES002]
AND [E005]
AND [ES006]
THEN [J0003], CF = 0,81

R29 IF [ES003]
AND [PU018]
AND [E010]
THEN [J0003], CF = 0,78

R30 IF [ES002]
AND [PU012]
AND [ES007]
THEN [J0004], CF = 0,82

R31 IF [ES003]
AND [PU018]
AND [E004]
THEN [J0004], CF = 0,75

R32 IF [ES002]
AND [E004]
AND [ES009]
THEN [J0005], CF = 0,80

R33 IF [ES003]
AND [PU011]
AND [PU012]
THEN [J0005], CF = 0,78

R34 IF [ES002]
AND [PU014]
AND [ES010]
THEN [J0006], CF = 0,80

R35 IF [ES003]
AND [E012]
THEN [J0006], CF = 0,75

R36 IF [ES002]
AND [ES011]
THEN [J0007], CF = 0,78

R37 IF [ES003]
AND [E002]
AND [E012]
THEN [J0007], CF = 0,78

R38 IF [ES002]
AND [ES008]
THEN [J0008], CF = 0,78

R39 IF [ES003]
AND [PU011]
AND [PU012]
THEN [J0008], CF = 0,78

R40 IF [ET002]
AND [ET005]
AND [PU009]
THEN [J0009], CF = 0,85

R41 IF [ET003]
AND [E011]
AND [E004]
THEN [J0009], CF = 0,78

R42 IF [ET002]
AND [ET004]
AND [PU009]
THEN [J0010], CF = 0,85

R43 IF [ET003]
AND [E011]
AND [E004]
AND [PU019]
THEN [J0010], CF = 0,80

R44 IF [ET002]
AND [ET006]
AND [PU011]
AND [E012]
THEN [J0011], CF = 0,85

R45 IF [ET003]
AND [PU013]
AND [E003]
AND [E005]
THEN [J0011], CF = 0,78

R46 IF [ET002]
AND [ET008]
AND [E012]
THEN [J0012], CF = 0,82

R47 IF [ET003]
AND [PU011]
AND [E002]
THEN [J0012], CF = 0,71

R48 IF [ET002]
AND [ET007]
AND [PU009]
THEN [J0013], CF = 0,82

R49 IF [ET003]
AND [E011]
AND [PU019]
THEN [J0013], CF = 0,75

R50 IF [ET002]
AND [ET009]
AND [PU018]
THEN [J0014], CF = 0,84

R51 IF [ET003]
AND [PU008]
AND [PU011]
AND [E011]
THEN [J0014], CF = 0,78

R52 IF [ET002]
AND [ET010]
AND [PU009]
AND [PU008]
THEN [J0015], CF = 0,85

R53 IF [ET003]
AND [E002]
AND [E004]
AND [PU010]
THEN [J0015], CF = 0,77

R54 IF [ET002]
AND [ET011]
AND [PU009]
THEN [J0016], CF = 0,81

R55 IF [ET003]
AND [PU008]
AND [E004]
AND [E011]
THEN [J0016], CF = 0,78

R56 IF [ET002]
AND [ET012]
AND [PU008]
THEN [J0017], CF = 0,80

R57 IF [ET003]
AND [PU009]
AND [E003]
AND [E010]
THEN [J0017], CF = 0,78

R58 IF [ET002]
AND [ET013]
AND [PU008]
THEN [J0018], CF = 0,82

R59 IF [ET003]
AND [PU009]
AND [E004]
AND [E011]
THEN [J0018], CF = 0,77

R60 IF [EK002]
AND [EK004]
AND [PU013]
THEN [J0019], CF = 0,85

R61 IF [EK003]
AND [PU018]
AND [E005]
AND [E012]
THEN [J0019], CF = 0,78

R62 IF [EK002]
AND [EK005]
AND [PU018]
AND [E010]
THEN [J0020], CF = 0,85

R63 IF [EK003]
AND [PU010]
AND [PU016]
AND [E005]
THEN [J0020], CF = 0,78

R64 IF [EK002]
AND [EK006]
AND [PU018]
THEN [J0021], CF = 0,85

R65 IF [EK003]
AND [E010]
THEN [J0021], CF = 0,73

R66 IF [EK002]
AND [EK007]
AND [PU018]
THEN [J0022], CF = 0,84

R67 IF [EK003]
AND [E005]
AND [E009]
THEN [J0022], CF = 0,78

R68 IF [EK002]
AND [EK010]
AND [PU018]
THEN [J0023], CF = 0,84

R69 IF [EK003]
AND [E005]

	AND [E010]	R78	IF [EP002]
	THEN [J0023], CF = 0,75		AND [EP007]
R70	IF [EK002]		AND [PU009]
	AND [EK008]		AND [E008]
	AND [PU018]		THEN [J0028], CF = 0,80
	THEN [J0024], CF = 0,81	R79	IF [EP003]
R71	IF [EK003]		AND [E005]
	AND [PU010]		AND [PU011]
	AND [PU017]		THEN [J0028], CF = 0,76
	THEN [J0024], CF = 0,75	R80	IF [EP002]
R72	IF [EK002]		AND [EP009]
	AND [EK009]		AND [PU004]
	AND [PU018]		AND [PU013]
	THEN [J0025], CF = 0,81		THEN [J0029], CF = 0,80
R73	IF [EK003]	R81	IF [EP003]
	AND [E003]		AND [E003]
	AND [E010]		AND [PU019]
	THEN [J0025], CF = 0,75		THEN [J0029], CF = 0,76
R74	IF [EP002]	R82	IF [EH002]
	AND [EP005]		AND [EH004]
	AND [PU009]		AND [E012]
	AND [E008]		THEN [J0030], CF = 0,80
	THEN [J0026], CF = 0,80	R83	IF [EH003]
R75	IF [EP003]		AND [E005]
	AND [E005]		AND [PU018]
	AND [EP008]		THEN [J0030], CF = 0,75
	THEN [J0026], CF = 0,76	R84	IF [EH002]
R76	IF [EP002]		AND [EH007]
	AND [EP006]		AND [PU004]
	AND [PU004]		THEN [J0031], CF = 0,80
	AND [PU017]	R85	IF [EH003]
	THEN [J0027], CF = 0,80		AND [E005]
R77	IF [EP003]		AND [PU013]
	AND [E012]		THEN [J0031], CF = 0,75
	AND [PU011]	R86	IF [EH002]
	THEN [J0027], CF = 0,75		AND [EH006]

AND [PU017]
THEN [J0032], CF = 0,80

R87 IF [EH003]
AND [PU011]
AND [PU013]
THEN [J0032], CF = 0,76

R88 IF [EH002]
AND [EH005]
AND [E010]
THEN [J0033], CF = 0,80

R89 IF [EH003]
AND [E003]
AND [PU011]
THEN [J0033], CF = 0,75

R90 IF [EI002]
AND [EI004]
AND [PU013]
THEN [J0034], CF = 0,80

R91 IF [EI003]
AND [E005]
AND [E012]
AND [PU011]
THEN [J0034], CF = 0,75

R92 IF [EI002]
AND [EI005]
AND [PU017]
AND [PU004]
THEN [J0035], CF = 0,80

R93 IF [EI003]
AND [E005]
AND [PU013]
THEN [J0035], CF = 0,76

R94 IF [EI002]
AND [EI006]
AND [PU016]
THEN [J0036], CF = 0,81

R95 IF [EI003]
AND [E005]
AND [PU011]
AND [PU017]
THEN [J0036], CF = 0,76

R96 IF [NE002]
AND [NE004]
AND [N009]
THEN [J0037], CF = 0,80

R97 IF [NE003]
AND [N002]
AND [PU004]
AND [PU011]
THEN [J0037], CF = 0,76

R98 IF [NE002]
AND [NE005]
AND [PU018]
THEN [J0038], CF = 0,81

R99 IF [NE003]
AND [N003]
AND [PU008]
THEN [J0038], CF = 0,75

R100 IF [NE002]
AND [NE006]
AND [PU010]
AND [PU016]
THEN [J0039], CF = 0,80

R101 IF [NE003]
AND [N002]
AND [PU004]
AND [PU011]
THEN [J0039], CF = 0,78

R102 IF [NH002]
AND [NH005]
AND [PU011]
THEN [J0040], CF = 0,80

R103 IF [NH003] AND [PU004] AND [PU009] AND [PU013] THEN [J0040], CF = 0,77

R104 IF [NH002] AND [NH006] AND [N009] THEN [J0041], CF = 0,80

R105 IF [NH003] AND [PU004] AND [PU011] AND [PU013] THEN [J0041], CF = 0,75

R106 IF [NH002] AND [NH007] AND [N008] THEN [J0042], CF = 0,80

R107 IF [NH003] AND [PU004] AND [PU017] AND [N010] THEN [J0042], CF = 0,77

R108 IF [NS002] AND [NS006] AND [PU013] THEN [J0043], CF = 0,81

R109 IF [NS003] AND [PU004] AND [N009] THEN [J0043], CF = 0,75

R110 IF [NS002] AND [NS005] AND [N008] THEN [J0044], CF = 0,80

R111 IF [NS003] AND [PU004] AND [N007] THEN [J0044], CF = 0,75

R112 IF [NS002] AND [NS004] AND [PU013] THEN [J0045], CF = 0,81

R113 IF [NS003] AND [PU004] AND [N010] THEN [J0045], CF = 0,75

R114 IF [NB002] AND [NB007] AND [PU009] THEN [J0046], CF = 0,81

R115 IF [NB003] AND [N004] AND [PU002] THEN [J0046], CF = 0,77

R116 IF [NB002] AND [NB006] AND [PU009] THEN [J0047], CF = 0,82

R117 IF [NB003] AND [N005] AND [PU002] THEN [J0047], CF = 0,75

R118 IF [NB002] AND [NB005] AND [PU009] THEN [J0048], CF = 0,80

R119 IF [NB003] AND [PU002] THEN [J0048], CF = 0,70

<i>Kode</i>	<i>Jurusan</i>	<i>Kode</i>	<i>Jurusan</i>
J0001	Fisika	J0025	Farmasi
J0002	Kimia	J0026	Budidaya Pertanian
J0003	Biologi	J0027	Sosial Ekonomi Pertanian
J0004	Matematika	J0028	Ilmu tanah
J0005	Statistika	J0029	Tehnologi Hasil Pertanian
J0006	Geografi	J0030	Produksi Ternak
J0007	Geofisika	J0031	Tehnologi Hasil Ternak
J0008	Ilmu Komputer	J0032	Sosial Ekonomi peternakan
J0009	Teknik Sipil	J0033	Nutrisi Makanan Ternak
J0100	Teknik arsitektur	J0034	Budidaya Perairan
J0011	Teknik Lingkungan	J0035	Tehnologi Hasil perikanan
J0012	Teknik Geodesi	J0036	Manajemen Sumberdaya Perairan
J0013	Teknik Planologi	J0037	Ekonomi Pembangunan
J0014	Teknik Industri	J0038	Akuntansi
J0015	Teknik Elektro	J0039	Manajemen
J0016	Teknik Mesin	J0040	Ilmu Hukum
J0017	Teknik Kimia	J0041	Ilmu Politik
J0018	Teknik Informatika	J0042	Hubungan Internasional
J0019	Ilmu Keperawatan	J0043	Ilmu Pemerintahan
J0020	Kedokteran Umum	J0044	Administrasi Niaga
J0021	Kedokteran Gigi	J0045	Ilmu Komunikasi
J0022	Kedokteran hewan	J0046	Sastra Indonesia
J0023	Gizi Kesehatan	J0047	Sastra Inggris
J0024	Psikologi	J0048	Sastra Jepang

UNIVERSITAS BRAWIJAYA



LAMPIRAN 3

Tabel Premis

KODE	PREMIS
E001	Berminat pada jurusan Eksakta
E002	Mempunyai nilai rata-rata fisika > 7
E003	Mempunyai nilai rata-rata kimia > 7
E004	Mempunyai nilai rata-rata matematika > 7
E005	Mempunyai nilai rata-rata biologi > 7
E006	Peduli pada alam dan lingkungan sekitar
E007	Tidak buta warna
E008	Tidak mudah jijik terhadap sesuatu
E009	Suka dan tidak takut binatang
E010	Suka kegiatan di laboratorium
E011	Suka menggambar
E012	Mempunyai fisik dan daya tahan yang baik
-N001	Berminat pada jurusan Non Eksakta
-N002	Mempunyai nilai rata-rata ekonomi > 7
-N003	Mempunyai nilai rata-rata akuntansi > 7
-N004	Mempunyai nilai rata-rata bahasa Indonesia > 7
-N005	Mempunyai nilai rata-rata bahasa Inggris > 7
-N006	Jujur dan dapat dipercaya
-N007	Tertarik dan peka terhadap masalah ekonomi
-N008	Suka berorganisasi
-N009	Suka mengamati dan menganalisa masalah sosial
-N010	Menguasai bahasa asing dengan baik
PU001	Suka menonton berita
PU002	Suka membaca Koran, buku atau Majalah
PU003	Suka Browsing Internet
*PU004	Wawasan Luas
PU005	Selalu tepat waktu
PU006	Tidak suka menunda pekerjaan
PU007	Menepati peraturan
*PU008	Disiplin tinggi
PU009	Teliti & Sabar
PU010	Mampu berpikir logis
PU011	Mempunyai daya analisis / prediksi kuat
PU012	Suka Menghitung

PU013	Mampu bekerja sama dalam tim
PU014	Suka kegiatan lapangan
PU015	Suka menghafal & daya ingat baik
PU016	Berjiwa pemimpin
PU017	Suka bergaul & supel
PU018	Rajin & tekun
PU019	Kreatif dan imajinasi tinggi
ES001	Tertarik pada bidang Sains
ES002	Ingin mempelajari ilmu sains
*ES003	Bidang Keilmuan Sains
ES004	Ingin belajar tentang gejala alam, elektronika dan instrumentasi, alat optik dll
ES005	Ingin belajar tentang bahan kimia, biokimia, kimia analitik, kimia karbon,dll
ES006	Ingin belajar ilmu hayat, ekologi, mikrobiologi, dan genetika
ES007	Ingin belajar aljabar dan geometri, matematika komputasi dan terapan
ES008	Ingin belajar membuat program, sistem informasi dan multimedia
ES009	Ingin belajar tentang survey, analisis data dan riset operasi
ES010	Ingin belajar ilmu bumi, sistem informasi geografi, kartografi dan topografi
ES011	Ingin belajar fisika bumi, geofisika lingkungan, gejala alam
ET001	Tertarik pada bidang Tehnik
ET002	Ingin mempelajari ilmu tehnik
*ET003	Bidang Keilmuan Tehnik
ET004	Ingin belajar seni desain bangunan dan struktur bangunan terpadu
ET005	Ingin belajar rancang bangun, teknologi konstruksi, transportasi dan hidrologi
ET006	Ingin belajar pengelolaan lingkungan, AMDAL, limbah dan pencemaran
ET007	Ingin belajar perencanaan pembangunan sarana dan prasarana kota
ET008	Ingin belajar ilmu perpetaan, pencitraan dan penentuan

	posisi
ET009	Ingin belajar tentang manajemen pabrik, teknologi dan sistem industri
ET010	Ingin belajar elektronika, listrik, kontrol dan elektro komunikasi
ET011	Ingin belajar seluk beluk permesinan, optimasi energi dan otomatisasi
ET012	Ingin mempelajari kimia terapan, perancangan alat industri kimia
ET013	Ingin mempelajari teknologi informasi, jaringan dan perangkat lunak
EK001	Tertarik pada bidang Kesehatan
EK002	Ingin mempelajari ilmu kesehatan
*EK003	Bidang Keilmuan Kesehatan
EK004	Ingin belajar ilmu untuk merawat orang dan profesi keperawatan
EK005	Ingin belajar mengobati penyakit manusia
EK006	Ingin belajar mengobati penyakit terutama gigi dan mulut
EK007	Ingin belajar mengobati penyakit hewan
EK008	Ingin belajar psikologi seseorang, psikologi industri, psikologi pendidikan,dll
EK009	Ingin belajar meracik obat, kimia farmasi, farmasi klinik
EK010	Ingin belajar kandungan gizi makanan dan menu makanan sehat
EP001	Tertarik pada bidang Pertanian
EP002	Ingin mempelajari ilmu pertanian dan tanaman
*EP003	Bidang Keilmuan Pertanian
EP004	Menyukai hal-hal yang berhub. dengan tanaman
EP005	Ingin belajar membudidayakan tanaman pertanian
EP006	Ingin belajar manajemen agrobisnis agroindustri, tata niaga pertanian
EP007	Ingin belajar konservasi tanah, pemanfaatan lahan dan klasifikasi tanah
EP008	Suka bercocok tanam/ berkebun / merawat tanaman
EP009	Ingin belajar teknologi pengolahan pasca panen

EH001	Tertarik pada bidang Peternakan
EH002	Ingin mempelajari ilmu Peternakan
*EH003	Bidang Keilmuan Peternakan
EH004	Ingin belajar tatacara pemeliharaan hewan ternak dan reproduksi ternak
EH005	Ingin belajar mengenai pakan hewan ternak, tehnik pemberian pakan
EH006	Ingin belajar manajemen dan pengembangan usaha peternakan
EH007	Ingin belajar teknologi pengolahan hasil peternakan
EI001	Tertarik pada bidang Perikanan
EI002	Ingin mempelajari ilmu Perikanan
*EI003	Bidang Keilmuan Perikanan
EI004	Ingin belajar budidaya perairan tawar dan payau
EI005	Ingin belajar teknologi dan tata niaga hasil perikanan
EI006	Ingin belajar konservasi dan manajemen sumberdaya perairan
NE001	Tertarik pada bidang Ekonomi
NE002	Ingin mempelajari ilmu Ekonomi
*NE003	Bidang Keilmuan Ekonomi
NE004	Ingin belajar micro macro ekonomi dan pertumbuhan ekonomi
NE005	Ingin belajar pengolahan data keuangan dan penyusunan laporan keuangan
NE006	Ingin belajar manajerial perusahaan dan manajerial umum
NH001	Tertarik pada bidang Hukum
NH002	Ingin mempelajari ilmu Hukum
*NH003	Bidang Keilmuan Hukum
NH004	Berjiwa adil dan bijaksana
NH005	Ingin belajar ilmu hukum dan tatanegara, hukum perdata dan pidana
NH006	Ingin belajar analisa masalah politik dan kekuatan politik
NH007	Ingin belajar diplomasi, politik internasional, administrasi internasional

NS001	Tertarik pada bidang Sosial
NS002	Ingin mempelajari ilmu Sosial
*NS003	Bidang Keilmuan Sosial
NS004	Ingin belajar jurnalistik, public relation dan komunikasi sosial
NS005	Ingin belajar pelaksanaan tata niaga perdagangan barang dan jasa
NS006	Ingin belajar kebijaksanaan umum(public policy) dan analisa politik
NB001	Tertarik pada bidang Sastra
NB002	Ingin mempelajari ilmu Sastra
*NB003	Bidang Keilmuan Sastra
NB004	Indra pendengaran baik
NB005	Ingin belajar bahasa,sastra dan karya seni dalam bahasa Jepang
NB006	Ingin belajar bahasa dan kesusastraan inggris
NB007	Ingin belajar bahasa dan kesusastraan indonesia



LAMPIRAN 4

- Store Procedure Hitung CF1 (mencari nilai minimum dari masing-masing aturan / *rule* dan mengalikannya dengan nilai CF *Rule* untuk mendapatkan RCF)

```

create proc hitungcf1
as
declare @no_rule varchar(5)
declare @mincf float
declare @xcf float

--- cari nilai minimum dari premis di aturan tersebut (hanya yang triger aja)
declare cf_cursor cursor for select trelasi.norule, min(temppremis.pcf)
from ((temppremis inner join trelasi on temppremis.nopremis =
trelasi.nopremis)
inner join temprule on trelasi.norule = temprule.norule) where
temprule.rstatus = 'triggered'
group by trelasi.norule

open cf_cursor
fetch next from cf_cursor into @no_rule,@mincf
while (@@fetch_status=0)
begin
--- kalikan nilai min dengan Cf rule dari pakar untuk menghitung RCF
select @xcf = cf from trule where norule = @no_rule
update temprule set RCF = @xcf * @mincf where norule = @no_rule
fetch next from cf_cursor into @no_rule,@mincf
end
close cf_cursor
deallocate cf_cursor
go
    
```

- Hitung CF2 digunakan untuk menghitung Nilai CF dari aturan yang memiliki kode kesimpulan yang sama

$$\text{CF}_{\text{combine}} \begin{cases} \frac{\text{CF}_1 + \text{CF}_2(1 - \text{CF}_1)}{\text{CF}_1 + \text{CF}_2} & \text{kedua duanya} > 0 \\ 1 - \text{Min}(|\text{CF}_1|, |\text{CF}_2|) & \text{salah satu} < 0 \\ \frac{\text{CF}_1 + \text{CF}_2(1 + \text{CF}_1)}{\text{CF}_1 + \text{CF}_2(1 + \text{CF}_1)} & \text{kedua duanya} < 0 \end{cases}$$

```

create proc hitungcf2
as
declare @kdkesimp varchar(5)          declare @rcf2asli float
declare @kdkesimp2 varchar(5)        declare @rcf12 float
declare @rcf float                    declare @minrcf float
declare @rcf2 float                   declare @i integer

---cari semua kode kesimpulan dan rcf yang status rule-nya trigger
declare kdkesimp_cursor cursor for select trule.kdkesimp, temprule.rcf
from trule inner join temprule on trule.norule = temprule.norule
where kdkesimp in
(select trule.kdkesimp from temprule inner join trule on
temprule.norule = trule.norule
where temprule.rstatus = 'triggered')order by trule.kdkesimp

open kdkesimp_cursor
fetch next from kdkesimp_cursor into @kdkesimp,@rcf
fetch next from kdkesimp_cursor into @kdkesimp2,@rcf2
set @i=0
while (@@fetch_status=0 or @@fetch_status = -1)
begin
if @kdkesimp <> @kdkesimp2
begin
set @rcf2asli = @rcf2
set @rcf2 = 0
end

if @@fetch_status = -1
begin
set @rcf2 = 0
end

if @rcf >= 0 and @rcf2 >= 0
begin
select @rcf12 = @rcf + @rcf2*(1-@rcf)
end
else if @rcf<0 and @rcf2<0
begin
select @rcf12 = @rcf + @rcf2*(1+@rcf)
end
else

```

begin

```
• select @minrcf = abs(@rcf)
  if(abs(@rcf2)<@minrcf)
  begin
    select @minrcf = abs(@rcf2)
  end
  select @rcf12 = (@rcf + @rcf2)/(1-@minrcf)
end
```

```
if left(@kdkesimp,1) = 'J'
```

```
begin
  update tjurusan set tcf = @rcf12 where kdjurusan = @kdkesimp
end
```

```
if not exists(select * from tcf where kdkesimp = @kdkesimp)
```

```
begin
  insert into tcf values(@kdkesimp, @rcf12)
end
```

```
else
```

```
begin
  update tcf set tcf = @rcf12 where kdkesimp = @kdkesimp
end
```

```
if(@@fetch_status = -1)
```

```
break;
```

```
if @kdkesimp = @kdkesimp2
```

```
begin
```

```
  set @rcf = @rcf12
```

```
end
```

```
else
```

```
begin
```

```
  set @rcf = @rcf2asli
```

```
end
```

```
set @kdkesimp = @kdkesimp2
```

```
fetch next from kdkesimp_cursor into @kdkesimp2,@rcf2
```

```
end
```

```
close kdkesimp_cursor
```

```
deallocate kdkesimp_cursor
```

```
GO
```

Store Procedure Cek Rule2 (Setelah yang premis metarule dicek, rule dicek ulang)

```
create proc cek_rule2
as
--- ubah status dari marked ke triger tapi yang premisnya udah true
smua (ga ada yg masih free)
update temprule set rstatus = 'triggered' where
rstatus='marked' and norule not in
(select temprule.norule from (temprule inner join trelasi on
temprule.norule = trelasi.norule)
inner join temppremis on trelasi.nopremis = temppremis.nopremis
where temprule.rstatus = 'marked' and temppremis.pstatus = 'free')
go
```

Store Procedure Update Temp (Update premis metarule)

```
create proc update_temp
as
declare @kdkesimp varchar(5)
declare @tcf float
declare tcf_cursor cursor for select kdkesimp,tcf from tcf
open tcf_cursor
fetch next from tcf_cursor into @kdkesimp,@tcf
while(@@fetch_status=0)
begin
--- update premis yang metarule
update temppremis set pcf = @tcf, pstatus = 'true', statustanya='na'
where nopremis = @kdkesimp
fetch next from tcf_cursor into @kdkesimp,@tcf
end
close tcf_cursor
deallocate tcf_cursor
update temprule set rstatus = 'triggered'
where norule in (select norule from trule where kdkesimp in (select
kdkesimp from tcf))
go
```

LAMPIRAN 5**HASIL REKOMENDASI SISTEM PAKAR
KELAS : BAHASA**

NO	NAMA	JURUSAN
1	Yanu Eko P	Sastra Inggris (0,721) Sastra Jepang (0,698) Hub. Internasional (0.69) Ilmu Komunikasi (0,591) Ilmu Pemerintahan (0,527)
2	Evi Rosdiana Dewi	Sastra Indonesia (0,532) Sastra Inggris (0,53) Sastra Jepang (0,51) Ilmu Komunikasi (0,343) Hub. Internasional (0.036)
3	Lailatul Muzdalifah	Sastra Indonesia (0,775) Sastra Inggris (0,77) Sastra Jepang (0,418) Ilmu Komunikasi (0,38)
4	Binnazir Husin	Sastra Indonesia (0,869) Sastra Jepang (0,846) Sastra Inggris (0,841) Ilmu Komunikasi (0,63) Psikologi(0,228)
5	Ro' diatul Azizah Efendi	Ilmu Pemerintahan(0.94) Sastra Inggris (0,862) Sastra Jepang (0,842) Psikologi (0,837) Biologi (0,687)
6	Eka Nur Susanti	Sastra Inggris (0,862) Sastra Jepang (0,846) Ilmu Komunikasi (0,717)
7	Aidah MJ	Sastra Inggris (0,77) Sastra Jepang (0,746)
8	Hanif Syamsiah	Ilmu politik (0.71) Ilmu Komunikasi (0,701) Ilmu Pemerintahan (0,701)

		Sastra Indonesia (0,621) Sastra Jepang (0,592)
9	Andarini Sukmawati	Ilmu Keperawatan (0,299)
10	Fariza Riyadini	Ilmu Komunikasi (0,47) Hub.Internasional (0,47) Sastra Jepang (0,46)
11	Yasmin Pristi Wahyuli	Ilmu Hukum (0,529) Ilmu Politik (0,524) Ilmu Komunikasi (0,425)
12	Rini Ardilah	Sastra Inggris (0,721) Sastra Jepang (0,698) Ilmu Komunikasi (0,591)
13	M. As'adul Munir	Sastra Indonesia (0,775) Sastra Jepang (0,746) Ilmu Komunikasi (0,64)
14	Linda Monica	Sastra Indonesia (0,775) Sastra Jepang (0,746) Ilmu Pemerintahan (0,527)
15	Feni Andi	Ilmu Politik (0,421) Sastra Jepang (0,404) Ilmu komunikasi (0,173)

LAMPIRAN 6**HASIL REKOMENDASI SISTEM PAKAR
KELAS : IPA**

NO	NAMA	JURUSAN
1	Afita farahma	Tek. Hasil Pertanian (0,644) Geofisika (0,634) Farmasi (0,47) Teknik Lingkungan(0,361)
2	Alamanda Widyaswara P	Sastra Indonesia (0,855) Gizi Kesehatan (0,55) Ilmu Keperawatan (0,54)
3	Alfia Triana	Biologi (0,705) Kimia (0,705) Ilmu Keperawatan (0,696) Gizi Kesehatan (0,688) Psikologi(0,655)
4	Danang Sulistyio	Teknik Sipil (0,73) Manajemen (0,719) Teknik Industri (0,597) Psokologi (0,49)
5	Dewi Setyawati	Budidaya Pertanian (0,527) Ilmu Komunikasi (0,526) Geofisika(0,508) Gizi Kesehatan (0,389) Psikologi (0,378) Biologi (0,375)
6	Nuris Dian Syah	Statistika (0,723) Farmasi (0,717) Teknik Kimia (0,675) Teknik Arsitektur (0,667)
7	Citra Asmarani A.P	Ilmu Komunikasi (0,511) Statistika (0,401) Biologi (0,253)
8	Indry Lianty N	Ilmu Keperawatan (0,70) Gizi Kesehatan (0,389) Psikologi (0,378) Kedokteran Umum (0,358)
9	Erna Yulita	Ilmu Keperawatan (0,739)

		Psikologi (0,717) Farmasi (0,425)
10	Meri Latifah	Biologi (0,517) Hub.Internasional (0,419) Kedokteran Umum (0,383) Farmasi (0,359) Ilmu Keperawatan (0,170)
11	Dinda Tri Astuti R	Psikologi (0,64) Kedokteran Umum (0,586) Ilmu Hukum (0,499) Hub. Internasional (0,419)
12	Benidiktus Anindito	Ilmu Komputer (0,517)
13	M.Amirudin	Ilmu Komputer (0,823) Matematika (0,811) Teknik Mesin (0,797) Statistika (0,756) Teknik Informatika(0,605)
14	Zulvia Devi Rara	Sastra Indonesia (0,922) Sosial Ekonomi Pertanian (0,793) Tehnik Informatika (o,727) Ilmu Komputer (0,676) Matematika (0,644) Tek.Hasil Pertanian (0,638) Ilmu Tanah (0,63) Budidaya Pertanian (0,63)
15	Raden Ahm. A	Ilmu Komputer (0,652) Farmasi (0,64) Teknik Industri (0,629) Kimia (0,557) Teknik Kimia (0,49) Teknik Sipil (0,419)
16	Andri Wardhani	Teknik Arsitektur (0,745) Ilmu Komputer (0,627) Teknik Mesin (0,409)
17	Risty Ayu Fauziah	Ilmu Keperawatan (0,546) Gizi Kesehatan (0,535) Psikologi (0,527)
18	Rendra Prasetya	Kedokteran Hewan (0,747) Kedokteran Umum (0,739)

		Budidaya Pertanian (0,733) Biologi (0,727) Tek. Hasil Perikanan (0,717) Ilmu Komputer (0,717) Teknik Informatika (0,64)
19	Choirunnisak	Gizi Kesehatan (0,598) Kedokteran Umum (0,546) Psikologi (0,527) Kimia (0,481) Biologi (0,48)
20	Miftah Huda P	Ilmu Keperawatan (0,822) Fisika (0,804) Ilmu Komputer (0,8) Kedokteran Umum (0,728) Biologi (0,727) Gizi Kesehatan (0,716) Psikologi (0,706)
21	Nur Faizah	Budidaya Pertanian (0,415) Farmasi (0,3) Ilmu Keperawatan (0,218)
22	Moch. Alfian	Matematika (0,638) Teknologi Hasil Ternak(0,58) Farmasi (0,527) Biologi (0,527) Teknik Mesin (0,409) Teknik Elektro (0,403)
23	Gia Rachmat Sulung	Ilmu Komputer (0,752) Teknik Lingkungan (0,739) Fisika (0,632) Biologi (0,421)
24	Sofia Ulfa	Kedokteran Hewan (0,581) Gizi Kesehatan (0,578) Fisika (0,43) Ilmu Komputer (0,419)
25	Ida Fitria	Ilmu keperawatan (0,832) Gizi Kesehatan (0,822) Biologi (0,808) Psikologi (0,806) Farmasi (0,806)

LAMPIRAN 7

HASIL REKOMENDASI SISTEM PAKAR KELAS : IPS

NO	NAMA	JURUSAN
1	M.Umar Jainuri	Ilmu Komunikasi (0,269) Ilmu Pembangunan (0,269) Hub,Internasional (0,1)
2	Rofik Atul Adawiyah	Akuntansi (0,723) Eko. Pembangunan (0,559) Ilmu Komputer (0,419) Manajemen (0,291)
3	Ali Ahmad Al Jufri	Ilmu hukum (0,59) Ilmu politik (0,333)
4	Cahyono Arif Wibowo	Manajemen (0,831) Teknik Industri (0,827) Adm.niaga (0,791)
5	Yenny	Manajemen (0,465) Akuntansi (0,264)
6	Yanthie	Akuntansi (0,925) Psikologi (0,923) Manajemen (0,834) Ilmu Keperawatan (0,782)
7	Mega Purnama	Ilmu Hukum (0,855) Manajemen (0,847) Sastra Indonesia (0,794)
8	A. Wienata.P	Ilmu Politik (0,827) Ilmu Hukum (0,751) Hubungan Internasional (0,312) Ilmu Komunikasi (0,266)
9	Mahardika Puspa Dewi	Adm.Niaga (0,724) Akuntansi (0,654) Ilmu Pemerintahan (0,296)
10	Frisca Dyah P	Teknik Arsitektur (0,279)
11	Wakhrudin	Adm. Niaga (0,718) Ilmu Pembangunan (0,717) Teknik Industri (0,621)

		Manajemen (0,489) Ekonomi Pembangunan (0,489) Ilmu Komunikasi (0,266)
12	Harmoko	Manajemen (0,806) Akuntansi (0,806) Psikologi (0,763)
13	Ferry Rizki N	Ilmu hukum (0,604) Ilmu Politik (0,601) Psikologi (0,383)
14	Resti Kartika M	Manajemen (0,723) Ekonomi Pembangunan (0,63) Psikologi (0,572) Akuntansi (0,488)
15	M.Wisnu Yogirandi	Ilmu komunikasi (0,010) Ilmu pembangunan (0.010) Ekonomi pembangunan (0.009) Manajemen (0,005)
16	Ahmi Wihansyah	Ilmu hukum (0,72) Akuntansi (0,065) Manajemen (0,023)
17	P.H.Rahma Putri	Manajemen (0,53) Hub. Internasional (0,416) Sastra Inggris (0,32) Sastra Indonesia (0,326) Ekonomi Pembangunan (0,023)
18	Ridlo Wahab	Manajemen (0,384) Ilmu Hukum (0,221) Ilmu politik (0,012) Hub. Internasional (0.07)
19	Yunita Setyorini	Akuntansi (0,902) Gizi Kesehatan (0,77) Ilmu Keperawatan (0,546)
20	Dian Asti Risnasari	Psikologi (0,201) Ilmu politik (0,191)
21	Sulistianingsih	Akuntansi (0,395) Psikologi (0,305) Ekonomi pembangunan (0,198)
22	Dian Ayu Damayanti	Gizi kesehatan (0,467) Akuntansi (0,266)

		Ekonomi Pembangunan (0,233)
23	Danis Catur R	Ilmu Hukum (0,824) Ilmu Politik (0,819) Adm. Niaga (0,76) Hub. Internasional (0,746) Ilmu Komunikasi (0,717) Manajemen (0,642)
24	Bachtiar Rahmad S	Psikologi (0,713) Ekonomi Pembangunan (0,679) Manajemen (0,637) Akuntansi (0,634) Hub. Internasional (0,584)
25	Ikhsan Firmansyah	Manajemen (0,673) Ilmu komunikasi (0,12)
26	Siti Nurjannah	Ilmu komunikasi (0,55) Adm. Niaga (0,42) Ilmu pemerintahan (0,288)

