

**SISTEM PENDUKUNG KEPUTUSAN
PERENCANAAN MATAKULIAH
BERBASIS WEB**

SKRIPSI



**Di susun Oleh :
Riyan Permadi
0818243**

**JURUSAN TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2013**

LEMBAR PERSETUJUAN

SISTEM PENDUKUNG KEPUTUSAN PERENCANAAN MATAKULIAH BERBASIS WEB

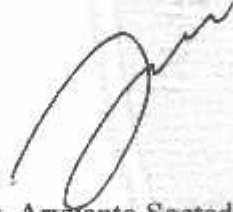
SKRIPSI

Disusun dan Diajukan untuk melengkapi dan memenuhi persyaratan guna mencapai gelar Sarjana Teknik

Di susun Oleh :
Riyan Permadi
0818243

Diperiksa dan Di setujui

Dosen Pembimbing I



Dr. Eng. Aryuanto Soetedjo, ST, MT

NIP. P. 1030800417

Dosen Pembimbing II



Mira Orisa, ST

NIP. P. 1031000435

Mengetahui,

Ketua Program Studi Teknik Informatika S-1



Josep Dedy Irawan, ST, MT

NIP. 19740416 200501 1 002

JURUSAN TEKNIK INFORMATIKA S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2013

SURAT PERNYATAAN ORISINALITAS

Yang bertanda tangan di bawah ini :

Nama : Riyan Permadi

Nim : 0818243

Program Studi : Teknik Informatika S-1

Dengan ini menyatakan bahwa skripsi yang saya buat adalah hasil karya sendiri, tidak merupakan plagiasi dari karya orang lain. Dalam skripsi ini tidak memuat karya orang lain, kecuali dicantumkan sumber sesuai dengan yang berlaku.

Dengan demikian surat pernyataan ini saya buat, dan apabila dikemudian hari ada pelanggaran atas surat ini saya bersedia menerima sanksinya.

Malang, Pebruari 2013
Yang membuat pernyataan



Riyan permadi
NIM 0818243

**SISTEM PENDUKUNG KEPUTUSAN
PERENCANAAN MATAKULIAH
BERBASIS WEB**

RIYAN PERMADI (0818243)
Jurusan Teknik Informatika S-1
Fakultas Teknologi Industri
Institut Teknologi Nasional Malang
Email : riyan123qwe@gmail.com

Pembimbing I Dr. Eng. Aryunto Soetedjo, ST, MT

Pembimbing II Mira Orisa, ST

ABSTRAK

Sistem pendukung keputusan perencanaan matakuliah sebagai modul untuk memprediksi hasil perencanaan matakuliah pada suatu institusi atau universitas dengan sistem pendidikan menggunakan sistem kredit semester. Permasalahan yang terjadi akibat dari kebingungan dalam menentukan pilihan matakuliah untuk rencana ke depan dalam proses pendidikan yang tidak terpaket. Prediksi dengan metode *edge-adjacency list* model akan memodelkan hirarki level orang tua dan anak pada syarat dan prasyarat matakuliah sehingga matakuliah dapat di tentukan untuk rencana yang akan dipilih. Sistem pendukung keputusan perencanaan matakuliah dibangun dengan menggunakan bahasa pemograman PHP dengan framework Yii pada *web server* apache dan aplikasi *database* MySQL. Hasil prediksi dengan menggunakan metode *edge-adjacency list* model memberikan lauran dalam bentuk prediksi perencanaan.

Kata Kunci : Sistem pendukung keputusan perencanaan matakuliah , sistem kredit semester , *edge-adjacency list* model

KATA PENGANTAR

Puji syukur atas kehadiran Allah SWT yang maha kuasa atas selesainya penyusunan laporan skripsi sistem pendukung perencanaan matakuliah berbasis web di Institut Teknologi Nasional Malang. Laporan ini merupakan Hasil penelitian yang dilakukan untuk membuat sistem pendukung keputusan perencanaan matakuliah sebagai prediksi perencanaan matakuliah pada setiap semester. terselesaikannya penyusunan buku Pedoman Pembimbingan dan Penulisan Skripsi, serta Teknik Publikasinya tidak terlepas dari bantuan semua pihak, oleh karena itu ucapan terima kasih disampaikan kepada:

1. Bapak Ir, Soeparno Djiwo, MT selaku sebagai Rektor Institut Teknologi Nasional Malang.
2. Bapak Joseph Dedy Irawan, ST, MT sebagai Ketua jurusan Teknik Informatika S-1 Institut Teknologi Nasional yang telah memberikan motivasi dan petunjuk disetiap persiapan skripsi
3. Bapak Dr. Eng. Aryuanto Soetedjo, St, MT sebagai pembimbing I yang telah membantu dalam penelitian dan penyusunan laporan
4. Bapak Sonny Prasetio, ST, MT sebagai pembimbing II yang telah membantu dalam penelitian dan penyusunan laporan

Akhirnya dengan segala kerendahan hati penulis berharap semoga laporan hasil penelitian sistem pendukung keputusan perencanaan matakuliah berbasis web bermanfaat untuk kepentingan dunia pendidikan.

Malang , Pebruari 2013

Penulis

Penulisan Skripsi

DAFTAR ISI

ABSTRAK.....	i
KATA PENGANTAR.....	ii
DAFTAR ISI.....	iii
DAFTAR GAMBAR.....	v
DAFTAR TABEL.....	vi
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Perumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	3
1.5 Manfaat.....	3
1.6 Metodologi.....	3
1.6.1 Metodologi Pengumpulan Data.....	3
1.6.2 Metodologi Pengembangan Sistem.....	4
1.7 Sistematika.....	4
BAB 2 LANDASAN TEORI.....	6
2.1 Pengertian Sistem Informasi.....	6
2.2 Sistem Kredit Semester.....	7
2.2.1 Beban Studi Mahasiswa.....	7
2.2.1.1 Beban Studi Untuk Penyelesaian Progran Studi.....	8
2.2.1.2 Perhitungan Indeks Prestasi.....	8
2.3 Decision Support System (DSS).....	9
2.3.1 Komponen-Komponen DSS.....	9
2.4 Edge-Adjacency List Model dari Pohon.....	11
2.4.1 Pohon Struktur Data.....	11
2.4.2 Daftar tepi.....	12
2.4.3 Daftar Adjacency.....	13
2.5 Bahasa Pemograman PHP.....	14
2.5.1 Penggunaan PHP.....	14
2.5.2 Sintak.....	15
2.5.3 Framework Yii.....	20
2.6 Apache Http Server.....	22
2.7 MySql.....	22
2.7.1 Mysql Workbench.....	23
BAB 3 ANALISA DAN PERANCANGAN.....	25
3.1 Analisa Struktur Pohon Kurikulum.....	25
3.2 Analisa Proses Rencana Studi.....	27
3.3 Analisis Model Base DSS.....	30
3.3.1 Analisa Edge-Adjacency List Model Dari Pohon Kurikulum.....	31
3.3.2 Analsis Perulangan Matakuliah Kurang Baik.....	39
3.3.3 Analisis Pemilihan Matakuliah Pilihan.....	39
3.4 Rancangan Desain Entitas Model Hubungan.....	41
3.5 Rancangan UML Diagram Sistem Informasi.....	42

3.6 Rancangan Desain Interface.....	44
BAB 4 IMPLEMENTASI DAN PENGUJIAN.....	47
4.1 Lingkungan Implementasi.....	47
4.1.1 Sistem Operasi ubuntu 12.04.....	47
4.1.2 Database MySql.....	48
4.1.3 PHP 5.....	51
4.2 Implementasi Modul.....	52
4.2.1 Modul Hirarki Matakuliah	52
4.2.2 Modul Perulangan Matakuliah.....	56
4.2.3 Modul Matakuliah Pilihan.....	58
4.3 Implementasi Antarmuka Sistem.....	60
4.3.1 Antarmuka User.....	60
4.4 Pelaksanaan Pengujian.....	62
BAB 5 KESIMPULAN DAN SARAN.....	65
5.1 Kesimpulan.....	65
5.2 Saran.....	65
DAFTAR PUSTAKA.....	66
LAMPIRAN.....	67

DAFTAR GAMBAR

Gambar 2.1: Komponen DSS[12].....	10
Gambar 2.2: Pohon Jenis Makan[9].....	11
Gambar 2.3: Pohon dengan Daftar Tepi[10].....	12
Gambar 2.4: Pohon dengan Daftar Adjency[10].....	13
Gambar 2.5: Model MVC Yii[13].....	21
Gambar 3.1: Pohon Kurikulum[2].....	26
Gambar 3.2: Diagram Use Case Perencanaan.....	28
Gambar 3.3: Diagram Aktivitas Analisis Proses Rencana Studi.....	29
Gambar 3.4: Diagram Squence Alur Berurutan	30
Gambar 3.5: Flowcart Model Base Alur.....	31
Gambar 3.6: Akar Dengan Dua Orang Tua.....	32
Gambar 3.7: Sturtuk Pohon Dengan Normalisasi.....	32
Gambar 3.8: Pohon 1 dengan Normalisai dan Nilai Tepi.....	33
Gambar 3.9: Pohon 2 Tanpa Normalisasi dengan Nilai Tepi.....	33
Gambar 3.10: Pohon 3 dengan Normalisasi dan Nilai Tepi.....	34
Gambar 3.11: Pohon 4 dengan Normalisasi dan Nilai Tepi.....	34
Gambar 3.12: Pohon 5 Tanpa dengan Normalisasi dan Nilai Tepi.....	35
Gambar 3.13: Pohon1 dengan Nilai Tepi dan Parent.....	36
Gambar 3.14: Pohon 2 dengan Nilai Tepi dan Parent.....	36
Gambar 3.15: Pohon 3 dengan Nilai Tepi dan Parent.....	37
Gambar 3.16: Pohon 4 dengan Nilai Tepi dan Parent.....	37
Gambar 3.17: Pohon 5 dengan Nilai Tepi dan Parent.....	38
Gambar 3.18: ERD Sistem	42
Gambar 3.19: UML Sistem Informasi.....	43
Gambar 3.20: Desain Antar Muka Sistem Matakuliah Input.....	44
Gambar 3.21: Desain Antar Muka Sistem Akademik View.....	45
Gambar 4.1: Urutan Pre-order Akar Matakuliah.....	53
Gambar 4.2: Post-order dengan Seleksi Matkauliah.....	54
Gambar 4.3: Post-order Seleksi Anak Paling Bungsu.....	55
Gambar 4.4: Hasil DSS Implementasi.....	56
Gambar 4.5: Hasil Perulangan Matakuliah Jelek.....	58
Gambar 4.6: Hasil Pembanding matakuliah pilihan.....	60
Gambar 4.7: Halaman Login Sistem.....	60
Gambar 4.8: Halaman Matakuliah Sistem.....	61
Gambar 4.9: Halaman Akdemik Sistem.....	62

DAFTAR TABEL

Tabel 2.1: Tabel Parameter Indeks Prestasi[2].....	8
Tabel 2.2 : Parameter Nilai[2].....	9
Tabel 2.3: Daftar Adjecency[10].....	14
Tabel 2.4: Operator Pembanding[5].....	17
Tabel 3.1: Data Edge-Adjacency List Model.....	38
Tabel 3.2: Desain Tabel Riwayat.....	40
Tabel 4.1: Syarat Ubuntu Hardware[14].....	47
Tabel 4.2: Struktur Tabel Matakuliah.....	48
Tabel 4.3: Struktur Tabel Akademik.....	49
Tabel 4.4: Struktur Tabel User.....	49
Tabel 4.5: Tabel Pakar Edge-Adjesensi List Model.....	50
Tabel 4.6: Tabel Struktur Matakuliah Pilihan.....	51
Tabel 4.7: Tabel Matakuliah Pilihan 2.....	59
Tabel 4.8: Hasil Persentasi Kuisisioner.....	63
Tabel 4.9: Hasil Uji Browser.....	64

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sistem Kredit Semester, atau SKS, dalam penyelenggaraan pendidikan menyatakan beban studi mahasiswa terhadap matakuliah yang diambilnya. Sistem Kredit Semester memungkinkan mahasiswa memilih sendiri matakuliah yang akan diambilnya dalam satu semester. Setiap matakuliah memiliki bobot kredit dengan nilai yang berbeda-beda. Besarnya kredit yang telah diambil menjadi pengakuan atas keberhasilan usaha belajar mahasiswa. Seorang mahasiswa dapat dinyatakan berhasil apabila telah menempuh 144 -160 SKS untuk program sarjana S1 dan 110 – 120 SKS untuk program diploma D3.

Dalam Sistem Kredit Semester, mahasiswa merencanakan setiap matakuliah yang dipilih. Setiap matakuliah yang dipilih membutuhkan konsultasi perencanaan matakuliah ke dosen pembimbing atau dosen wali. Dosen wali atau dosen pembimbing sebagai penasehat akan menentukan pilihan matakuliah perencanaan yang sesuai dengan kebutuhan. Mahasiswa akan terbantu oleh dosen wali dalam merencanakan matakuliah ke depan. Pelaksanaan konsultasi biasanya dilakukan sesuai jadwal yang telah di tentukan.

Terkadang terjadi proses kebingungan pada awal perencanaan matakuliah. Kebingungan di dalam awal perencanaan matakuliah dapat terjadi dari berbagai faktor. Faktor – faktor yang mempengaruhi diantaranya adalah kurang pemahaman tentang wawasan akan matakuliah yang akan direncanakan. Wawasan akan perencanaan matakuliah tersebut terdapat di pengalaman seseorang atau di dalam buku silabus akademik. Studi literatur mengenai "*A web-based decision support tool for academic advising*" oleh T. Feghali, I. Zbib, dan S. Hallal menyatakan pengetahuan tersebut dapat di implementasi dalam sistem^[8]

pendukung keputusan. Pengetahuan tersebut dapat dimasukkan ke dalam sistem pendukung keputusan dengan salah satu metode hirarki pohon. Hirarki pohon tersebut berisikan tentang syarat-syarat matakuliah yang menentukan jalur kebutuhan perencanaan matakuliah. Sistem pendukung kebutuhan pengetahuan atau hirarki pohon matakuliah akan di tanamkan pada Sistem Administrasi Informasi Akademik (SIKAD). Bagaimana penanaman sistem pendukung keputusan ke SIKAD untuk memberikan perencanaan awal yang sesuai dengan kebutuhan.

1.2 Perumusan Masalah

Dari faktor-faktor yang terjadi pada latar belakang maka penulis mengidentifikasi masalah sebagai berikut :

1. Bagaimana memilih rencana matakuliah yang baik untuk studi ke depan ?
2. Bagaimana membuat saran awal untuk menghasilkan sebuah rencana dari pengetahuan dan hirarki yang telah dibuat ?

1.3 Batasan Masalah

Batasan masalah dalam membangun sistem pendukung keputusan perencanaan matakuliah mencakup antara lain :

1. Membangun sistem pendukung keputusan berbasis *web*
 2. Membangun pengetahuan sistem dengan aturan yang berlaku pada silabus teknik informatika 2008 Institut Teknologi Nasional Malang
 3. Melihat syarat dan prasyarat pada matakuliah dalam memberikan prediksi perencanaan matakuliah
 4. Melihat index prestasi mahasiswa untuk memberikan prediksi perencanaan matakuliah
 5. Tidak melibatkan proses semester pendek
-

1.4 Tujuan

Tujuan penelitian dalam membangun sistem pengambilan keputusan perencanaan matakuliah antara lain :

1. Sebagai prediksi awal perencanaan matakuliah dalam proses perencanaan studi
2. Memaksimalkan batasan pengambilan sks untuk mendapatkan jumlah matakuliah yang tepat
3. Membantu kelulusan dengan tepat

1.5 Manfaat

Manfaat yang di terima pada penelitian dalam sistem pendukung keputusan perencanaan matakuliah diantara :

1. Sebagai alat bantu pendukung untuk perencanaan matakuliah pada proses rencana studi
2. Membantu menentukan prediksi matakuliah pilihan
3. Membantu perulangan matakuliah yang kurang baik agar nilai matakuliah tersebut menjadi baik

1.6 Metodologi

1.6.1 Metodologi Pengumpulan Data

Metode pengumpulan data yang digunakan untuk mendukung sistem pendukung keputusan, diantara :

1. Studi Pustaka

Pengambilan data dari buku-buku atau literatur tertulis lain yang terlibat dalam akademik. Data tersebut akan di gunakan sebagai acuan dalam meneliti sistem pendukung keputusan.

2. Observasi

Observasi merupakan metodologi langsung dengan melakukan pengamatan ke ruang lingkup penelitian. Observasi di arac kampus untuk mendapat informasi-informasi yang berkaitan dengan akademik dan berfungsi mendukung sistem pendukung keputusan.

1.6.2 Metodologi Pengembangan Sistem

Metodologi pengembangan sistem merupakan metode dalam sistem sebagai pengolah data yang telah ada sehingga menghasilkan sebuah hasil sesuai dengan keinginan. Metode pengembangan sistem bersumber dari buku-buku, jurnal, e-book, dan lain-lain. Sebagai contoh dari jurnal studi yang telah meneliti dari jurnal Paper 16-A *Prototype Student Advising Expert System Supported with an Object-Oriented database* oleh M. Ayman Al Ahmar Deputy Dean, College of Information Technology Ajman University of Science and Technology (AUST) United Arab Emirates (UAE) [7].

1.7 Sistematika

BAB 1. PENDAHULUAN

Bab pendahuluan materinya sebagian besar berupa latar belakang masalah, rumusan masalah, tujuan penelitian, batasan masalah, metode penelitian dan sistematika penulisan laporan penelitian.

BAB 2. LANDASAN TEORI

Bab Landasan Teori merupakan tinjauan pustaka, menguraikan teori-teori yang mendukung judul. Landasan teori dapat berupa definisi-definisi atau model yang langsung berkaitan dengan ilmu atau masalah yang diteliti. Pada bab ini juga dituliskan tentang komponen *tools/software* yang digunakan untuk pembuatan aplikasi atau untuk keperluan penelitian.

BAB 3. ANALISA KEBUTUHAN DAN PERANCANGAN

Bab ini berisi “analisa masalah”, yang akan menguraikan tentang analisa terhadap permasalahan yang terdapat di kasus yang sedang di teliti. Meliputi analisa terhadap masalah sistem yang sedang berjalan, prosedur pelaksanaan pekerjaan dari permasalahan, deskripsi kebutuhan informasi, dan pemodelan kebutuhan fungsional, sekaligus tentang tahap-tahap perancangan dan pembuatan sistem yang sedang diteliti sesuai dengan analisa kebutuhan dan pemodelannya meliputi perancangan perangkat lunak, arsitektur dan algoritma program, perancangan antar muka perangkat lunak, dan perancangan *database* meliputi diagram E-R, struktur tabel, dan relasi antar tabel.

BAB 4. IMPLEMENTASI DAN PENGUJIAN

Bab ini adalah implemantasi dan pengujian atas sistem yang telah selesai untuk memberikan hasil dari penelitian sebagai tolak ukur keberhasilan sistem.

BAB 5. KESIMPULAN DAN SARAN

Bab terkahir berisikan tentang kesimpulan dan saran dari penelitian yang terlaksana.

BAB II

LANDASAN TEORI

2.1 Pengertian Sistem Informasi

Menurut Jerry FithGerald, sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu. Informasi adalah data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat. Data merupakan material untuk suatu informasi.

Menurut Robert A. Leitch, sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Sistem informasi di rancang untuk mendukung manajemen, dan informasi pendukung keputusan.

Sistem informasi pendukung keputusan merupakan bagian dari sistem informasi yang digunakan untuk mendukung keputusan baik rencana atau pengambilan keputusan dalam organisasi. Menurut Moore and Chang, sistem pendukung keputusan dapat digambarkan sebagai sistem yang berkemampuan mendukung analisis *ad hoc* data, dan pemodelan keputusan, berorientasi keputusan, orientasi perencanaan masa depan, dan digunakan pada saat-saat yang tidak biasa. Menurut Keen dan Scoot Morton sistem Pendukung Keputusan merupakan penggabungan sumber-sumber kecerdasan individu dengan kemampuan komponen untuk memperbaiki kualitas keputusan. Sistem Pendukung Keputusan juga merupakan sistem informasi berbasis komputer untuk manajemen pengambilan keputusan yang menangani masalah- masalah semi struktur.

Sistem informasi pendukung keputusan dapat diterapkan pada sistem

administrasi akademik sebagai perencanaan. Perencanaan tersebut berkaitan dengan sistem pendidikan yang menggunakan sistem kredit semester. Sistem kredit semester (SKS) adalah suatu sistem penyelenggaraan pendidikan dimana beban studi mahasiswa, beban kerja tenaga pengajar, dan beban penyelenggaraan program Lembaga Pendidikan dinyatakan dalam kredit.

2.2 Sistem Kredit Semester

Sistem kredit semester adalah penyelenggara sistem pendidikan dengan satuan kredit. Satuan kredit tersebut digunakan untuk menyatakan besar beban studi mahasiswa sebagai pengakuan atas keberhasilan usaha kumulatif untuk program tertentu serta usaha untuk menyelenggarakan pendidikan oleh Perguruan Tinggi dan tenaga kerja. Takaran penilaian satuan kredit selama waktu pengalaman belajar di peroleh dari satu semester melalui kegiatan terjadwal per minggu sebanyak 1 jam perkuliahan atau 2 jam praktikum atau 4 jam kerja lapangan, yang masing-masing diiringi oleh kegiatan terstruktur dan mandiri selama 1-2 jam.

Semester adalah satuan waktu terkecil untuk menyatakan lamanya suatu program pendidikan dalam suatu jenjang pendidikan, dengan satuan lama 16-19 minggu dalam satu semester. Penilaian atas dasar sistem kredit tiap-tiap matakuliah dinamakan sebagai nilai kredit dan besar kredit yang memiliki perbedaan di masing-masing matakuliah.

2.2.1 Beban Studi Mahasiswa

Sistem Kredit Semester ditentukan oleh bebasnya jumlah sks yang dicapai dan indeks prestasi. Setiap matakuliah memiliki sks berbeda-beda dengan asumsi semakin besar nilai sks dalam matakuliah maka matakuliah tersebut memiliki kegiatan yang lebih banyak untuk diselesaikan. Pengambilan matakuliah oleh mahasiswa berbeda-beda dan bebas selama tidak melebihi beban maksimal yang telah ditentukan. Komposisi pengambilan matakuliah sesuai dengan kemampuan mahasiswa.

2.2.1.1 Beban Studi Untuk Penyelesaian Progran Studi

Dalam penyelesaian program studi maka mahasiswa wajib menyelesaikan sebagai berikut :

- (a) Untuk progran strata satu (S1) sekurang-kurannya 144 sks
- (b) Untuk program diploma tiga (D3) Sekurang-kurannya 110 sks
- (c) Besar beban studi untuk semester pertama di tentukan secara paket.
- (d) Besar beban studi yang dapat di ambil oleh seorang mahasiswa pada semester berikutnya ditentukan sebagai berikut :

Tabel 2.1: Tabel Parameter Indeks Prestasi^[2]

Indesk prestasi	Beban studi yang dapat di ambil
> 3	22 – 24 sks
2,50 – 2,99	19 – 21 sks
2,00 – 2,49	16 – 18 sks
1,50 – 1,99	12 – 15 sks
< 1,50	< 12 sks

2.2.1.2 Perhitungan Indeks Prestasi

Perhitungan indeks prestasi merupakan hal perlu di lakukan untuk mengetahui jumlah beban maksimal yang diambil sesuai tabel 2.1. Adapun kategori dan perhitungan indeks prestasi sebagai berikut :

- (a) Untuk menghitung indeks prestasi, nilai huruf diubah menjadi nilai bobot dengan ketentuan berikut :

Tabel 2.2 : Parameter Nilai^[2]

NILAI		
Angka	Huruf	Bobot
80 – 100	A	4,00
71 – 79	B+	3,50
65 – 70	B	3,00
61 – 64	C+	2,50
56 – 60	C	2,00
40 – 55	D	1,00
0 – 39	E	0,00

(b) Perhitungan Indeks Prestasi dilakukan sebagai berikut :

$$\text{Indes prestasi (IP)} = \frac{(\sum K . N)}{(\sum K)} \dots\dots\dots (2-1)$$

K = Jumlah sks (Satuan kredit semester) matakuliah yang diambilnya
 N = Nilai masing-masing matakuliah

2.3 *Decision Support System (DSS)*

Salah satu jenis aplikasi yang sangat menarik di kalangan manajemen perusahaan adalah *Decision Support System* atau disingkat DSS. DSS merupakan suatu sistem informasi yang diharapkan dapat membantu manajemen dalam proses pengambilan keputusan dan perencanaan. DSS dapat diterapkan di berbagai tempat termasuk ke dalam sistem akademik untuk perencanaan matakuliah dalam pengambilang SKS. Contoh persoalan yang dapat diterapkan pada sistem akademik untuk perencanaan matakuliah bisa menggunakan *decision tree*.

2.3.1 **Komponen-Komponen DSS**

Secara garis besar DSS terdiri dari tiga komponen yang mendukung proses dalam sistem pendukung keputusan atau DSS, diantara :

1. *Database*

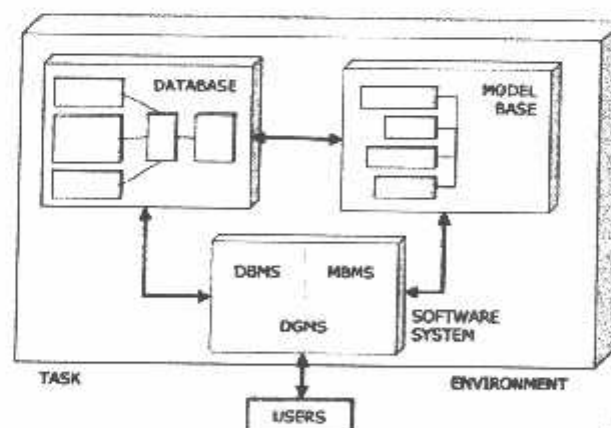
Sistem *database* berisi kumpulan dari semua data bisnis yang dimiliki perusahaan, baik yang berasal dari transaksi sehari-hari, maupun data dasar (master file). Untuk keperluan DSS, diperlukan data yang relevan dengan permasalahan yang hendak dipecahkan melalui simulasi.

2. *Model Base*

Komponen kedua adalah *Model Base* atau suatu model yang merepresentasikan permasalahan ke dalam format kuantitatif atau pohon kelas (hirarki pohon) sebagai dasar simulasi atau pengambilan keputusan, termasuk di dalamnya tujuan dari permasalahan (objektif), komponen-komponen terkait, batasan-batasan yang ada (*constraints*), dan hal-hal terkait lainnya.

3. *Interface*

Kedua komponen tersebut untuk selanjutnya disatukan dalam komponen ketiga (*software system*), setelah sebelumnya direpresentasikan dalam bentuk model yang “dimengerti” komputer. Contohnya adalah penggunaan teknik RDBMS (*Relational database Management System*), OODBMS (*Object Oriented database Management System*) untuk memodelkan struktur data. Sedangkan MBMS (*Model Base Management System*) dipergunakan untuk merepresentasikan masalah yang ingin dicari pemecahannya.



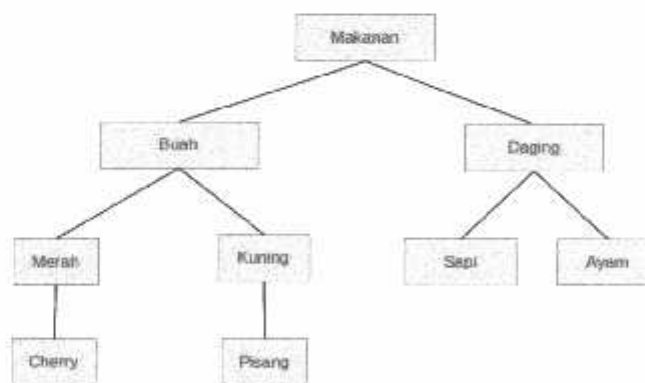
Gambar 2.1: Komponen DSS^[2]

2.4 Edge-Adjacency List Model dari Pohon

Pohon data merupakan *graph* tanpa siklus. *Graph* merupakan hubungan geometris yang menghubungkan berpasangan antar objek yang terdiri dari *node*, dan koleksi tepi. *node* dan tepi merupakan dua atau lebih *node* yang berdekatan. Sebuah pohon banyak digunakan struktur data yang mensimulasikan hirarki dengan satu *node* yang terikat. Sedangkan *edge-adjacency* merupakan representasi dari semua tepi dalam grafik sebagai daftar. Teknik *edge-adjacency list* model dari pohon merupakan gabungan dari representasi tepi dengan hirarki data untuk menunjukkan urutan dari dua *node* yang mengandung dua tepi yang sesuai. Apabila *node* diarahkan maka menunjukkan kemeunculan *node* lainnya sebagai *node* tujuan dari jalur yang sesuai yang tujuan.

2.4.1 Pohon Struktur Data

Pohon struktur data merupakan struktur hirarki data dengan satu set terkait *node*. Pohon struktur data dapat di definisikan kumpulan *node* yang memiliki *node root* sebagai *node* paling atas yang tidak memiliki orang tua atau induk, *node* paling bawah atau ujung yang hanya terdiri dari *node* memiliki induk atau orang tua saja, adapun *node* yang memiliki keduanya induk dan anak. Dapat kita lihat pada gambar 2.2 berikut :

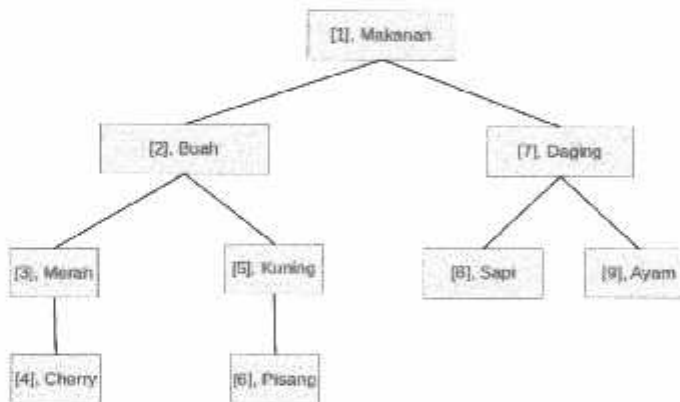


Gambar 2.2: Pohon Jenis Makan^[9]

Gambar 2.2 merupakan gambaran struktur data pohon. Selain Sebagai pohon struktur data memiliki metode *traversal*. Metode *traversal* merupakan langkah berjalan melalui *node* dari pohon dengan cara hubungan antara orang tua dan anak – anak dan juga tidak berjalan. Di dalam proses berjalan dengan melalui induk atau orang terlebih dahulu merupakan *pre-order*, sedangkan proses berjalan dengan melalui anak terlebih dahulu maka disebut juga *post-order*

2.4.2 Daftar tepi

Daftar tepi merupakan representasi sederhana dalam grafik. Representasi dalam berupa nama atau nilai *node* sebagai jalur yang akan di tempuh. Dalam daftar tepi setiap baris menentukan *node* dan induk *node*. Induk *node* paling atas tidak memiliki induk atau *parent*. Pemberian nilai pada setiap *node* akan mengarah ke kiri hingga ke dalam sampai paling ujung dan berbalik untuk ke *node* tetangga dengan jalur sebelahnya. Dapat dilihat representasi daftar tepi pada gambar 2.3 berikut :

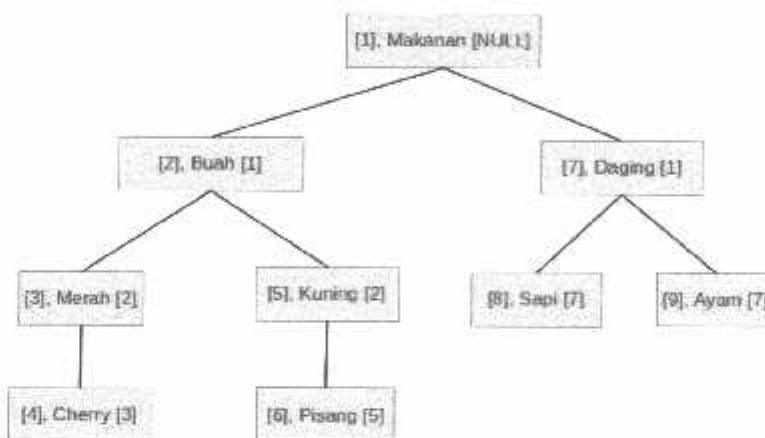


Gambar 2.3: Pohon dengan Daftar Tepi^[10]

Pada gambar 2.3 memiliki nilai di setiap tepi dengan setiap *node* hanya memiliki satu nilai. Berfungsi untuk mempermudah dalam pencarian data struktur data dan pengurutan sesuai kondisi hirarki yang berlaku.

2.4.3 Daftar Adjacency

Daftar *adjacency* merupakan proses lanjutan dari daftar tepi. Pada Daftar *adjacency* akan menampilkan nilai dari *node* induk atau orang tua pada *node* anak bersangkutan. Untuk *node root* atau yang paling teratas tidak memiliki nilai orang tua atau induk melainkan nilai NULL. Reperesentasi nilai tersebut dapat kita lihat pada gambar berikut :



Gambar 2.4: Pohon dengan Daftar *Adjacency*^[10]

Pada gambar 2.4 menjelaskan [Nomer tepi] , Namanode [Parent] dengan menyebutkan nilai *parent* akan membantu dalam proses *traversal pre-order*. Dapat kita desain ke dalam tabel yang nanti akan di terapkan pada *database* sebagai komponen dalam DSS. Dalam tabel tersebut memiliki relasi ke dalam diri sendiri *parent* sebagai id teratasnya sesuai dengan desain *tree* yang telah dibuat sebagai berikut :

Tabel 2.3: Daftar *Adjacency*^[10]

Nilai_Tepi	Nama_node	Parent
1	Makanan	NULL
2	Buah	1
3	Merah	2
4	Cherry	3
5	Kuning	2
6	Pisang	5
7	Daging	1
8	Sapi	7
9	Ayam	7

2.5 Bahasa Pemrograman PHP

PHP merupakan bahasa pemrograman yang di tanamkan ke dalam HTML yang dirancang untuk pengembangan *web* agar menghasilkan halaman *web* dinamis dengan bersifat *open source*. PHP juga sebagai *server-side script* diinterpretasikan oleh server *web* dengan module PHP untuk menghasilkan halaman *web* yang dihasilkan. *server-side script* merupakan teknik yang digunakan dalam desain website yang melibatkan *embedding script* dalam kode HTML yang menghasilkan permintaan penggunaan ke situs *web* server yang ditangani oleh *script* menjalankan *server-side* sebelum server merespon permintaan klien.

2.5.1 Penggunaan PHP

PHP sebagai bahasa pemrograman yang cocok untuk *server-side script* di dalam *web* server. Kode PHP akan mengeksekusi oleh PHP setiap waktu berjalan untuk membuat halaman *web* yang dinamis. PHP juga dapat digunakan dengan sistem manajemen *database* relasi (RDBMS). Sistem manajemen *database* relasi adalah sebuah sistem manajemen *database* (DBMS) yang didasarkan pada model relasi *database*. Sedangkan DBMS merupakan seperangkat program yang

memungkinkan untuk menyimpan, memodifikasi, dan mengekstrak informasi dari *database*, juga penyedia pengguna dengan alat untuk menambah, menghapus, mengakses, memodifikasi, dan menganalisa yang disimpan dalam satu lokasi. Selain hal tersebut PHP juga menyediakan sumber lengkap bagi pengguna untuk membangun, menyesuaikan dan memperpanjang sesuai kebutuhan. Pada awal pembuatan PHP dirancang sebagai pembuat halaman *web* yang dinamis namun banyak pengembang mengembang PHP ke pengembangan aplikasi cepat (RAD) sebagai contoh Yii, codeigniter, dll.

2.5.2 Sintak

Sintak PHP tertanam di dalam kode HTML dengan menggunakan *tag* untuk melampirkan bahasa PHP dalam file HTML. Sintak PHP dalam membuat dan mengedit halaman *web* memiliki cara yang sama membuat dan mengedit halaman HTML biasa. PHP server mendapatkan file untuk mengirimkan laporan HTML namun pernyataan PHP diproses oleh perangkat lunak PHP sebelum mereka dikirim ke pemohon. Contoh sintak PHP :

```
<?PHP echo "<p>Hello World"; ?>
```

Dalam pernyataan `<?PHP echo "<p>Hello World"; ?>` terdapat `<? PHP` sebagai *tag* awal untuk pembeda dan *tag* penutup `?>`. Intruksi `echo` merupakan sintak yang di pergunakan untuk memberi tahu PHP agar menampilkan teks sebagai plain HTML code dengan penutup `;` dan PHP akan memproses pernyataan PHP dengan hasil :

```
<p>Hello World
```

PHP dapat menyimpan beberapa tipe data. Tipe data yang pertama adalah integer dengan tipe data angka bulat misal -43, 0, 1, 50, dengan range jarak nilai antara -2 milyar hingga +2 milyar, yang kedua adalah float yang meliputi nilai desimal seperti 5,123 atau 12,09478573, yang ketiga adalah karakter string dengan tipe data huruf string seperti 'hello', yang keempat adalah data type boolean nilai dengan TRU dan FALSE.

Penggunaan variabel di dalam *script* PHP dimulai dengan tanda dolar (\$) yang memberi tahu bahwa itu adalah variabel nama. Nama variabel dapat berisi angka, huruf, dan garis bawah (_) yang mana variabel tidak dapat dimulai dengan angka. Ada *case sensitive* pada penamaan variabel misal \$a tidak sama dengan \$A. dapat kita lihat pada kode di bawah ini :

```
<?PHP
    $a = 1;
    $b = 1,5;
    $A = aku;
    echo "Nilai a=",$a;
    echo "Nilai b =",$b;
    echo "Nilai A =" ,$A;
?>
```

Dalam kode diatas ada tiga buah variabel \$a, \$b, dan \$A dengan nilai masing-masing. Maka hasilnya sebagai berikut :

```
Nilai a = 1
Nilai b = 1,5
Nilai A = Aku
```

Selain penamaan variabel PHP 5 juga menggunakan kondisi. Kondisi tersebut mengeksekusi blok pernyataan hanya saat kondisi benar. Ada dua bentuk kondisi yang pertama dengan menggunakan `if {} else {}` dan kedua dengan `switch () { case 1 break ; case 2 break;}`. Di dalam kondisi tersebut memiliki operator perbandingan. Operator perbandingan tersebut di gunakan untuk menentukan nilai benar dalam kondisi. Adapun macam operator sebagai berikut :

Tabel 2.4: Operator Pembandingan⁵¹

Operator	Arti operator
==	Apakah dua nilai sama nilainya?
===	Apakah dua nilai yang sama di kedua nilai dan tipe data?
>	Apakah nilai pertama lebih besar dari nilai kedua?
>=	Apakah nilai pertama lebih besar dari atau sama dengan nilai kedua?
<	Apakah nilai pertama lebih kecil dari nilai kedua?
<=	Apakah nilai pertama lebih kecil dari atau sama dengan nilai kedua?
!= , <>	Apakah kedua nilai tidak sama satu sama lain dalam nilai?
!==	Apakah Dua nilai tidak sama satu sama lain dalam nilai baik atau tipe data?

Dapat kita lihat pada kode berikut sebagai penggunaan pembandingan :

```

<?PHP
    $a = 1;
    $b = 2;
    if ($a <= $b)
    {
        echo "Nilai b Lebih besar dari pada nilai a";
    }
    else
    {
        echo "Nilai a Lebih besar dari pada nilai b";
    }
?>

```

Kode diatas merupakan kondisi dengan Jika nilai variabel a lebih kecil sama dengan dari variabel b maka tampilkan tesk Nilai b Lebih besar dari pada nilai a. Berbeda dengan menggunakan *case* kondisi, yang mana *case* kondisi disini sebagai pemilih *case* untuk pembandingan dengan kata lain kondisi dengan cara memilih sesuai dengan data input atau permintaan dapat kita lihat berikut ini :

```

<?PHP
$custState = 1;
switch ( $ custState ) {      {
    case 1 :
        $a = 0;
        echo "Hasil = ".$a;
        break;
    case 2 :
        $b = 1.0;
        echo "Hasil = ".$b;
        break;
    default:
        $c = 5;
        echo "Hasil = ".$c;
        break;
}
?>

```

Pada kondisi *case* ini kode PHP akan mengarahkan nilai variabel `$custState` ke pilihan dengan *case* sesuai. Dalam kasus ini adalah bernilai 1 dengan *case* 1 yang menampilkan Hasil = 0.

Selain kondisi PHP juga memiliki perulangan *for ... while* dan *do ... while*. Perulangan sering digunakan dalam kode untuk memberikan pernyataan blok yang diulangi. Perulangan dapat mengulangi jumlah tertentu beberapa kali sampai kondisi perulangan terpenuhi. Misal melakukan perulangan dari 1 – 10 dengan menggunakan *for*, *while*, dan *do while*

```

<?php
for ($i=1;$i<=10;$i++)
{
    echo $i." <br>";
}
?>

```

Kode perulangan *for* di atas memiliki variabel `$i`. Perulangan akan melakukan perulangan dari nilai 1 hingga lebih kecil dari sama dengan 10, yang penjumlahan perulangan + 1. Perulangan di atas memiliki kondisi mencapai dengan nilai 10 maka

perulangan akan berhenti dengan hasil :

```
1
2
3
4
5
6
7
8
9
10
```

Perulangan dengan *while* memiliki alur yang berbeda dengan *for*. Pembedanya adalah kondisi terdapat di atas dengan *increment* variabel ada di dalam blok dengan perulangan yang sama. Menginisialisalkan di awal bahwa variabel *\$i* memiliki nilai = 1 dengan kondisi ≤ 1 dan *increment* +1, sebagai berikut :

```
<?php
$i=1;
while($i<=10){
    echo $i++;
}
?>
```

Kode diatas menampilkan hasil sama perulangan dari 1 hingga 10. Adapun dengan menggunakan *do while*. Pembeda antara *while* dengan *do while* adalah letak pengujian kondisi pada *do while* berada di bawah. Dengan perulangan sama apabila kondisi telah benar akan melakukan perulangan hingga nilai salah. Dapat kita lihat pada kode berikut :

```

<?PHP
    $i = 0
    do{
        echo $i++;
    }while($i <=10)
?>

```

Kode di atas menjelaskan bahwa nilai variabel diinisialkan dengan pengujian kondisi di bagian bawah dan proses increment +1 berada dalam blok statement. Kode di atas menghasilkan output 1 2 3 4 5 6 7 8 9 10.

2.5.3 Framework Yii

Yii merupakan komponen berbasis PHP *web* kerangka aplikasi *web* dengan berorientasi objek dan bersifat *open source*. Proyek Yii bermula pada tanggal 1 januari 2008. Berawal untuk memperbaiki beberapa kekurangan dari framework PRADO karena lambatnya mengakses *halam web* yang kompleks dan akhirnya pada bulan Oktober 2008 reles versi alpha Yii, ketika tanggal 3 Desember 2008 Yii resmi dirilis. Yii memiliki keunggulan yang tertanam diantaranya :

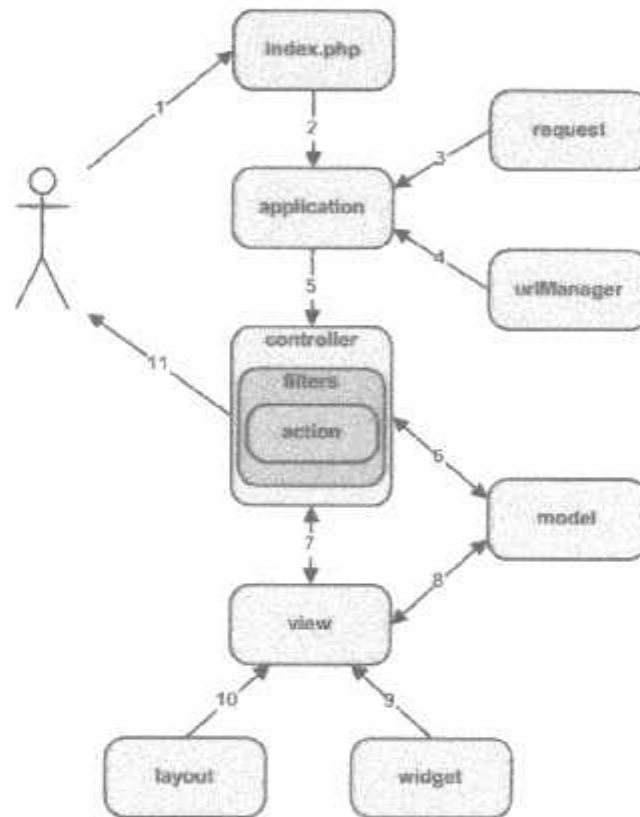
1. Route permintaan Yii atau *lifecycle*

Route permintaan atau *lifecycle* merupakan permintaan dari *browser* yang diterima oleh *route* untuk menentukan di mana aplikasi itu harus dikirim untuk di proses dan mengidentifikasi metode tertentu dalam kelas controller melewati permintaan sesuai tujuan. Proses rute yang pertama browser mengirimkan permintaan ke server hosting MVC, kemudian sebuah kontroler dipanggil untuk menangani permintaan, selanjutnya kontroler berinteraksi dengan model, selanjutnya kontroler memanggil *view*, dan *view* merender data (berupa HTML) dan mengembalikan ke browser untuk di tampilkan.

2. Yii menggunakan pola desain Model-View-Controller

MVC merupakan arsitektur dengan penyajian struktur *folder* untuk setiap

bagian dari model, *view*, dan *controller*. Model merupakan intruksi yang terdiri dari data aplikasi dan aturan bisnis. *View* merupakan intruksi berupa representasi output data, seperti grafik atau diagram. *Controller* bertugas untuk mengalirkan perintah permintaan dan bertanggung jawab untuk mengambil input pengguna, berinteraksi dengan model dan memerintahkan *view* untuk memperbaharui penampilan. Proses MVC pada Yii pada gambar 2.5 :



Gambar 2.5: Model MVC Yii^[13]

3. Objek-relasi pemetaan dan aktif *record*

Pada aplikasi *web* kita membangun *database* dalam relasi. Objek-relasi pemetaan menyediakan pemetaan tabel *database* objek kelas yang di lakukan berulang yang cukup membosankan. Di dalam Yii membantu dari perulangan dengan menyediakan objek-relasi pemetaan dalam *active record* (AR). *Active*

record adalah pola desain untuk akses *database* dengan cara orientasi objek. Disetiap tabel terdiri dari kelas objek dan kolom tabel menjadi atribut objek.

2.6 Apache Http Server

Apache Http server disebut juga sebagai apache. Apache menjadi software *web server* yang memiliki peranan penting dalam perkembangan *World Wide Web*. Apache merupakan software *open source* yang di pelihara dan dikembangkan bersama di bawah naungan apache software fondation. Apache tersedia di berbagai sistem oprasi , diantaranya Unix, FreeBSD, Linux, Solaris, Novell, NetWare, OS X, Microsoft Windows, OS/2, TPF, dan eComStation.

Apache memiliki beberapa fitur, banyak di implementasikan sebagai dikomplasi modul yang memperluas fungsional inti. Apache mendukung bahasa pemograman dalam *server-side* diantaranya Perl, Python, Tck, dan PHP. Selain hal tersebut Apache juga memiliki beberapa modul yang cukup populer diantaranya *mod_ssl* yang biasanya sebagai socket layer dan transport layer security, *mod_proxy* untuk proxy modul, *mod_rewrite* sebagai sebuah URL rewrite, *mod_log_config* untuk melihat log kostum, *mod_gzip* sebagai metode kompresi yang di peruntukan sebagai pengurangan berat dari halaman *web*, dan lain-lain. Apache juga memiliki virtual *hosting* yang berfungsi untuk pelayanan dengan banyak situs. Apache juga memiliki fitur pesan untuk kesalahan konfigurasi, DBMS berbasis *database* otentifikasi, otentifikasi password, sertifikat digital otentifikasi, dan apache mendukung otoritas HTML dan FTP.

2.7 MySql

MySql merupakan perangkat lunak sistem pengelolaan *database* atau DBMS yang dikembangkan oleh Oracle sebagai dari sistem *database* relasional yang distribusikan secara gratis di bawah Licensi GPL (Genegral Public Linsice). Mysql adalah turunan dari salah satu konsep utama dalam *database* ada telah ada

sebelumnya yaitu SQL. SQL merupakan konsep pengoprasian *database*, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoprasian data dikerjakan dengan mudah secara otomatis. MySQL memiliki GUI untuk mengelola data yang terdapat pada *database*. Antar muka MySQL disebut juga MySQL workbench.

2.7.1 MySQL Workbench

MySQL workbench adalah alat visual desain *database* yang terintegrasi SQL, administrasi, desain *database*, penciptaan, dan pemeliharaan yang menjadi satu lingkup pengembangan untuk MySQL. Pada sejarahnya MySQL workbench bermula dari DBDesigner4 yang merupakan sumber visual yang terbuka untuk desain *database* dan alat query untuk *database* MySQL pada tahun 2002/2003. DBDesigner4 memiliki fitur yang komprehensif termasuk *reverse engineering database* MySQL, model-to-sinkronisasi *database*, model *poster* cetak, kontrol versi dasar model skema dan editor query SQL. Pada akhir 2003 MySQLLab bergabung dengan perusahaan dan merubah antarmuka DBDesigner4 menjadi *bundle* GUI MySQL Tools. MySQL GUI *bundle* Tools merupakan *cross-platform* administrasi server *database* untuk membangun dan memanipulasi data dalam *database*. Dimana MySQL GUI *bundle* Tools mengalami penghentian pengembangan dan di gantikan oleh MySQL workbench hingga saat ini. MySQL workbench adalah gabungan dari MySQL GUI *bundle* dan DBDesigner4. Fitur yang terdapat pada MySQL workbench diantaranya :

1. Umum
 - a. *Database* koneksi dan manajemen *instance*
 - b. Sepenuhnya *scriptable* dengan python dan Lua
 - c. Dukungan untuk plugin kustom
 2. SQL editor
 - a. Skema objek menjelajah
 - b. SQL sintaks stabilo dan parser Pernyataan
 - c. Multiple hasil diedit set
 - d. SQL potongan koleksi
 - e. Koneksi SSH tunneling
 - f. *Unicode* dukungan
 3. Data pemodelan
-

- a. ER diagram
 - b. Drag'n'Drop visual yang pemodelan
 - c. *Reverse engineering* dari *script* SQL dan *database* hidup
 - d. Percetakan model
 - e. *Import* dari fabFORCE.net DBdesainer4
4. *Administrasi database*
- a. Mulai dan berhenti dari contoh *database*
 - b. Contoh konfigurasi
 - c. *Database account* pengelolaan
 - d. *Instance* variabel menjelajah
 - e. *Log file* browsing
 - f. Data dump *eksport / import*
-

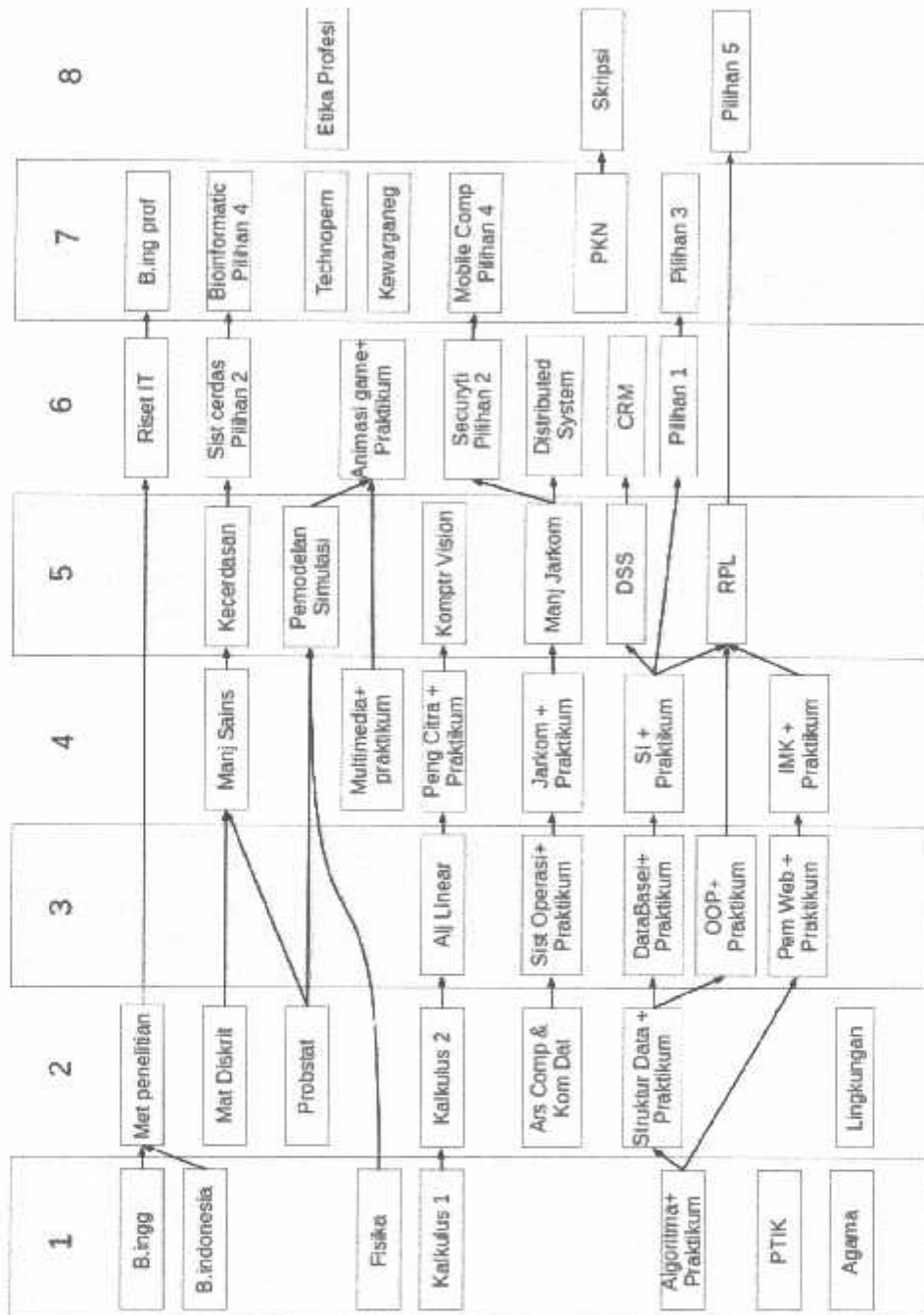
BAB III

ANALISA DAN PERANCANGAN

3.1 Analisa Struktur Pohon Kurikulum

Struktur pohon kurikulum merupakan gambaran tentang kurikulum pendidikan akan materi matakuliah yang berlaku di instansi pendidikan tersebut. Pohon kurikulum menjadi panutan dalam menempuh pendidikan. Pohon kurikulum menjadi sumber pengetahuan akademik bagi mahasiswa akan jenjang yang mereka tempuh. Pohon kurikulum berisikan akan hirarki matakuliah yang menunjukkan syarat dan prasyarat matakuliah yang berlaku disetiap matakuliah yang memiliki anak pada pohon kurikulum. Selain syarat dan prasyarat juga terdapat matakuliah yang berdiri sendiri tanpa memiliki anak pada pohon kurikulum.

Repersentasi pohon kurikulum tersebut biasanya di tanamkan pada sistem akademik untuk pemetaan matakuliah yang akan diambil pada proses pemograman rencana studi *On line*. Mahasiswa akan melakukan pemograman rencana studi sesuai dengan kurikulum dan hasil konsultasi bersama dosen pembimbing atau dosen wali. Dosen wali atau dosen pembimbing adalah seorang ahli yang membantu perencanaan matakuliah ke depan untu mahasiswa sesuai dengan kebutuhan dan tepat dalam kelulusan sesuai dengan pohon kurikulum. Dapat kita lihat pada gambar 3.1 berikut pohon kurikulum :



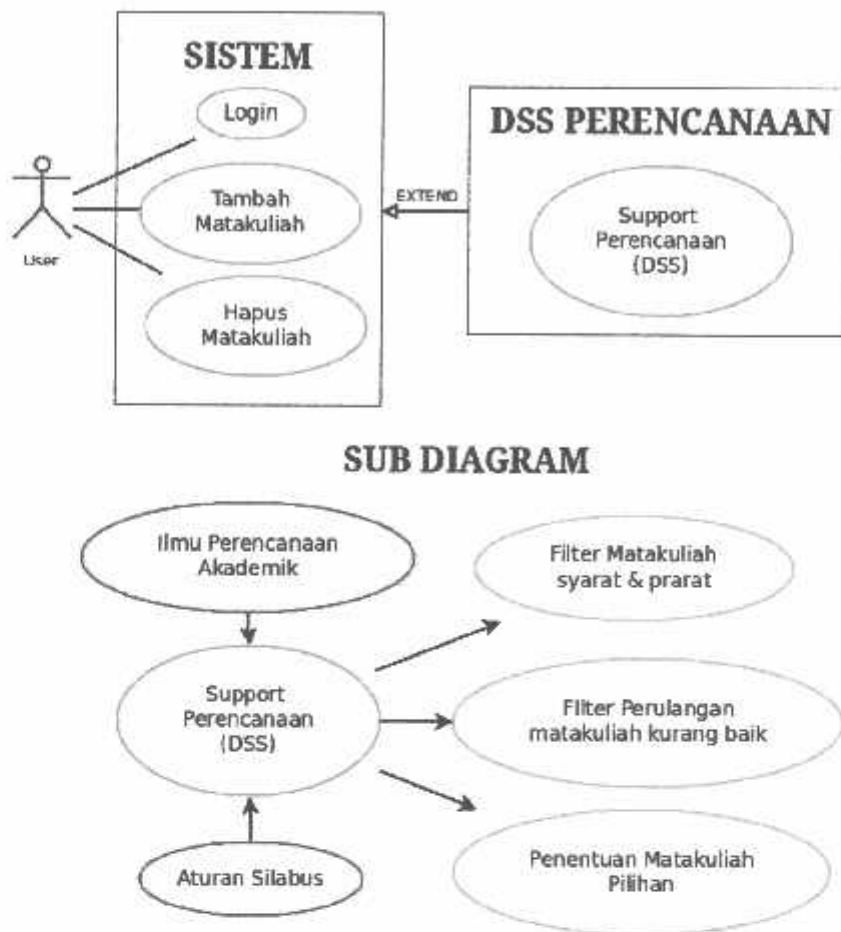
Gambar 3.1: Pohon Kurikulum^[2]

3.2 Analisa Proses Rencana Studi

Prosedur rencana studi adalah gambaran dalam proses perencanaan matakuliah oleh mahasiswa. Proses tersebut di gambar dalam *use case* diagram untuk memenuhi kebutuhan analisis perencanaan. Secara umum proses perencanaan berjalan sesuai dengan birokrasi yang berlaku. Masalah yang terjadi pada mahasiswa akan kebingungan dalam merencanakan perencanaan studi yang mereka pilih.

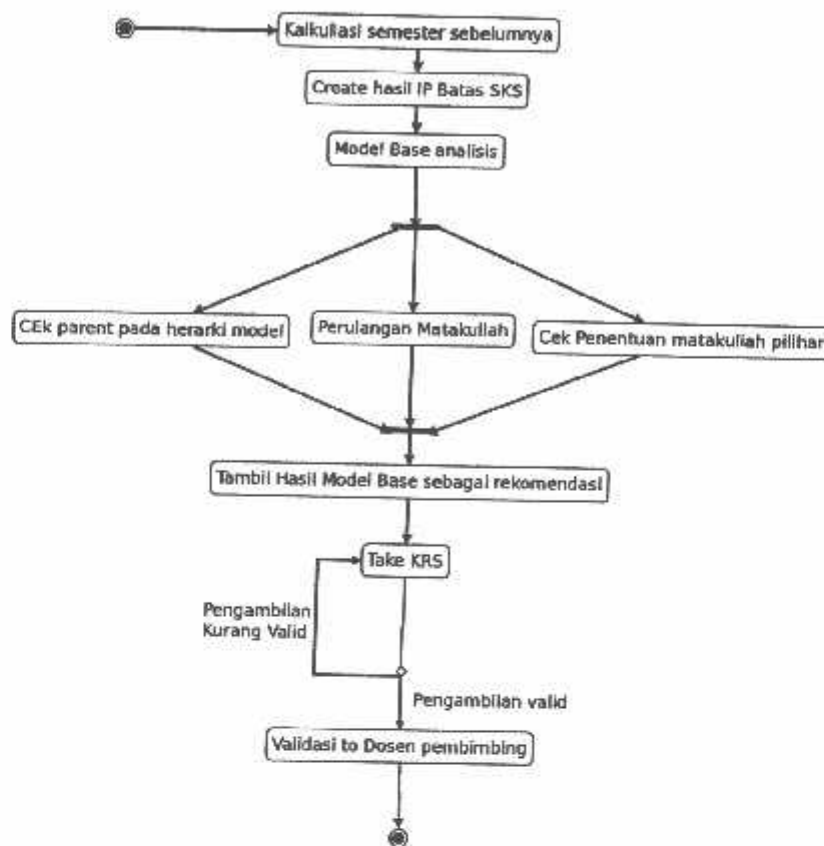
Oleh karena itu mahasiswa melakukan konsultasi akan rencana yang akan di tempuh sesuai ketentuan dalam silabus yang di bantu oleh dosen wali atau pembimbing agar lebih tepat dalam perencanaan matakuliah. Di dalam sistem akademik dapat di tanamkan DSS berupa pengetahuan silabus dan dosen wali. Pengetahuan silabus dan dosen wali pada DSS tersebut disimpan pada model *base* yang merupakan komponen DSS. Tujuannya untuk membantu memprediksi perencanaan matakuliah yang akan diambil. Prediksi perencanaan matakuliah tersebut sesuai dengan batas jumlah pengambilan sks matakuliah sesuai dengan indeks prestasi mahasiswa bersangkutan.

Model *base* pada komponen DSS yang berisikan pengetahuan silabus dan dosen wali sebagai representasi untuk membuat keputusan perencanaan. Letak Model *base* tersebut di tanamkan pada sistem bersama *database*. Di dalam *use case* diagram di sisipkan DSS perencanaan. Dengan kata lain mahasiswa akan mendapatkan beberapa prediksi perencanaan yang sekiranya sesuai dengan yang ada pada sistem. Sistem yang berlaku tersebut adalah sistem akademik atau SIAKAD yang menjadi sebagai proses rencana studi *online*. Dapat kita lihat pada gambar 3.2 *use case* diagram berikut :



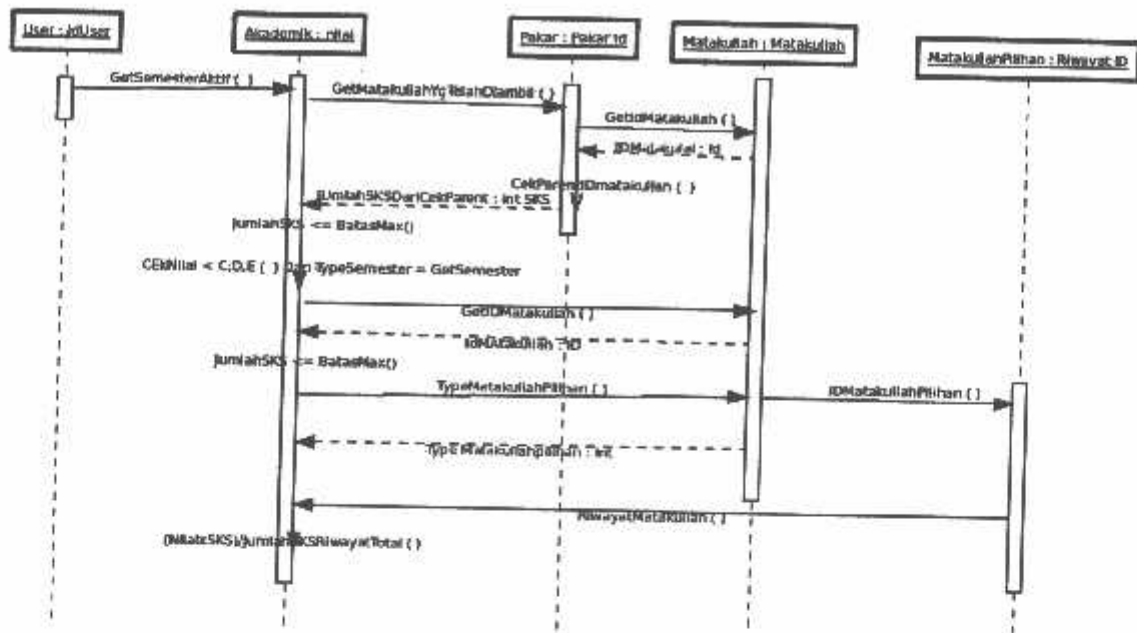
Gambar 3.2: Diagram *Use Case* Perencanaan

Ada pun diagram Aktivitas dan diagram *sequence* yang tercantum untuk memodelkan analisis rencana studi dengan tambahan model *base* sebagai penembahan yang berfungsi sebagai prediksi perencanaan matakuliah. Diagram aktifitas merupakan alur pada proses yang terjadi pada sistem perencanaan matakuliah. Diagram aktivitas akan memaparkan aktivitas hingga konsultasi ke dosen pembimbing yang akan dilihat kebutuhannya. Dapat kita lihat pada gambar 3.3 berikut ini :



Gambar 3.3: Diagram Aktivitas Analisis Proses Rencana Studi

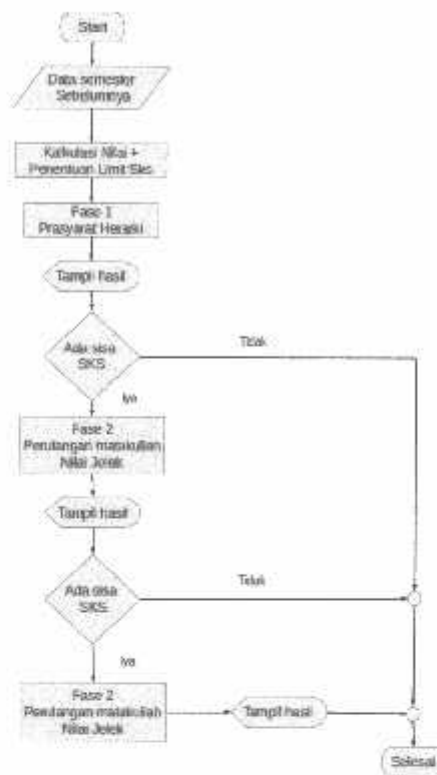
Selain diagram aktivitas yang mencatatkan akan analisa proses rencana studi juga memiliki diagram *sequence*. Diagram *sequence* merupakan diagram urut yang memaparkan alur proses berurutan. Berasal diagram aktivitas maka dapat terlihat terdapat 3 proses berurutan pada model *base*. Pada diagram *sequence* terdapat 5 instansi yang terlibat *User*, *Akademik*, *Pakar*, *Matakuliah*, dan *Matakuliah Pilihan*. 5 instansi tersebut memiliki urutan *user* sebagai data *idUser*, *Akademik* sebagai nilai akademik *user*, *pakar* sebagai model struktur hirarki, *matakuliah* sebagai utama *matakuliah*, dan *matakuliah pilihan* sebagai pembaca riwayat jenis *matakuliah pilihan*, berikut gambar diagram *sequence* 3.4 pada proses alur :



Gambar 3.4: Diagram *Sequence* Alur Berurutan

3.3 Analisis Model *Base* DSS

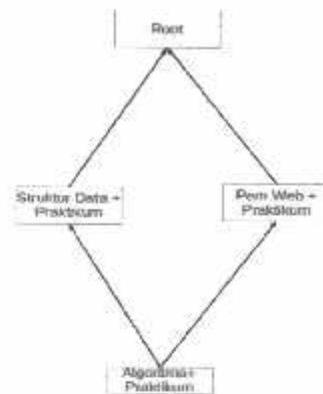
Dalam Analisis model *base* DSS perancangan terdapat tiga bagian yang menjadi dasar. Model *base* DSS sebagai komponen yang terlibat dalam sistem yang nanti akan mengolah data sebagai DSS. Tiga bagian dalam model *base* yaitu *Edge-Adjacency List* Model Dari Pohon dari struktur kurikulum, Perulangan matakuliah, dan Pembacaan riwayat nilai untuk membantu dalam memilih matakuliah pilihan sesuai dengan *index* kebutuhan tertinggi. Dalam Analisa Model *Base* terdapat tiga tingkatan untuk mempresentasikan hasil perencanaan. Dengan *Edge-Adjacency List* Model Dari Pohon dari struktur kurikulum sebagai tingkatan pertama yang berfungsi menyeleksi nilai matakuliah yang membutuhkan prasyarat untuk menampilkan matakuliah syarat. Kedua adalah Perulangan matakuliah, pada fase ini sistem akan memeriksa sisa limit dari batas sks yang telah didapat untuk menampilkan matakuliah yang perlu di ulang, dan fase ke tiga adalah fase matakuliah pilihan yang terjadi di akhir-akhir semester yaitu 6,7,8 yang membaca nilai riwayat anak matakuliah pada setiap matakuliah pilihan yang kemudian di bandingkan dan diambil nilai tertinggi sebagai pilihan. Ketiga fase tersebut dapat kita lihat pada gambar 3.5 flowchart yang menjelaskan model *base* alur data berikut :



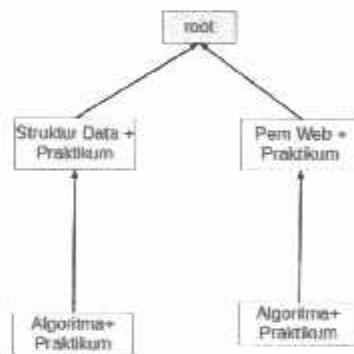
Gambar 3.5: Flowcart Model *Base Alur*

3.3.1 Analisa *Edge-Adjacency List* Model Dari Pohon Kurikulum

Analisa dapat dilakukan dengan melihat struktur pohon kurikulum. Pada struktur pohon kurikulum menjelaskan akan kelas hirarki mengenai turunan matakuliah yang menjadi prasyarat dan syarat dalam memilih rencana. Struktur pohon kurikulum memiliki asitektur mirip jalur *graph* yang belum terlihat jelas struktur hirarki yang baik. Oleh sebab itu analisa pertama adalah menormalisasikan struktur pohon kurikulum. Normalisasi merupakan langkah penyederhanaan struktur pohon menjadi lebih terstruktur. Normalisasi dilakukan dengan membuat akar baru apabila anak atau daun memiliki dua buah *parent*, sebagai contoh matakuliah Algoritma + praktikum yang memiliki dua *parent* atau orang tua yaitu Struktur Data dan Pemograman Web + Praktikum. Maka dapat kita lihat normalisasi pada gambar 3.6 dan 3.7:



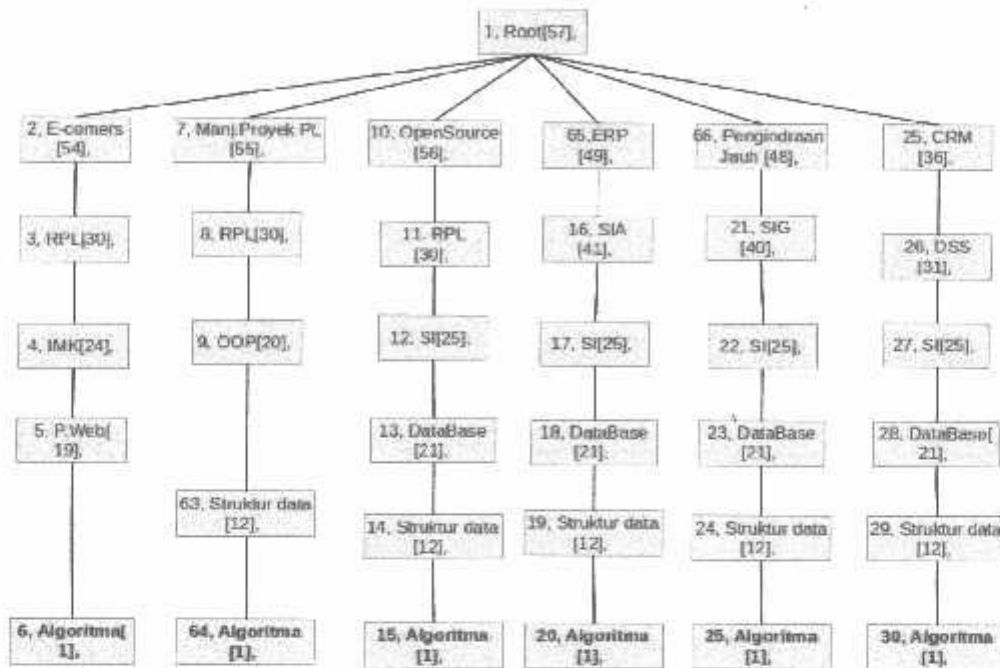
Gambar 3.6: Akar Dengan Dua Orang Tua



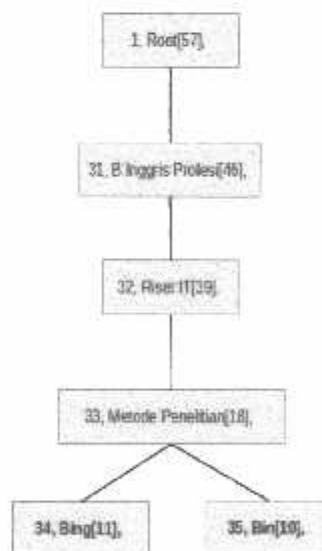
Gambar 3.7: Struktur Pohon Dengan Normalisasi

Pada normalisasi struktur pohon kurikulum akan memiliki lima macam *tree* yang berbeda secara garis besar yang menjadi pokok. Setiap pohon dari ke lima macam akan memiliki perbedaan. Perbedaan tersebut terletak pada anak dan orang tua pada setiap pohon. Dari kelima macam pohon dalam proses pemodelan ini tidak semua harus di normalisasikan ada beberapa struktur pohon tidak perlu di normalisasi karena telah memiliki struktur pohon yang cukup ideal. Adapaun gambar pemodelan tanpa normalisasi sesuai struktur pohon kurikulum. Pada proses normalisasi juga terdapat penamaan nilai tepi sesuai dengan metodologi *Edge-Adjacency List Model* pada kelima struktur pohon tersebut. Penaman bermula dari kiri menjorok paling bawah sampai *node* terakhir dan kembali ke kanan atas dan berulang hingga selesai.

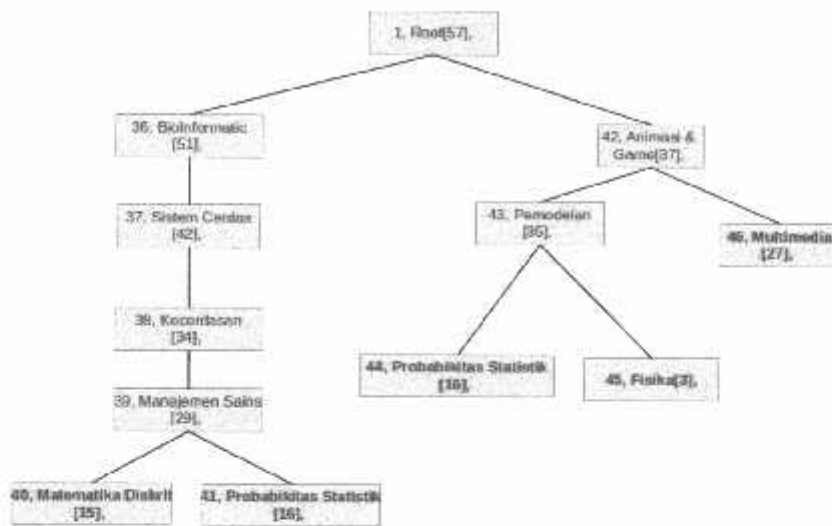
Dapat kita lihat pada gambar-gambar berikut untuk normalisasi dan penamaan nilai tepi :



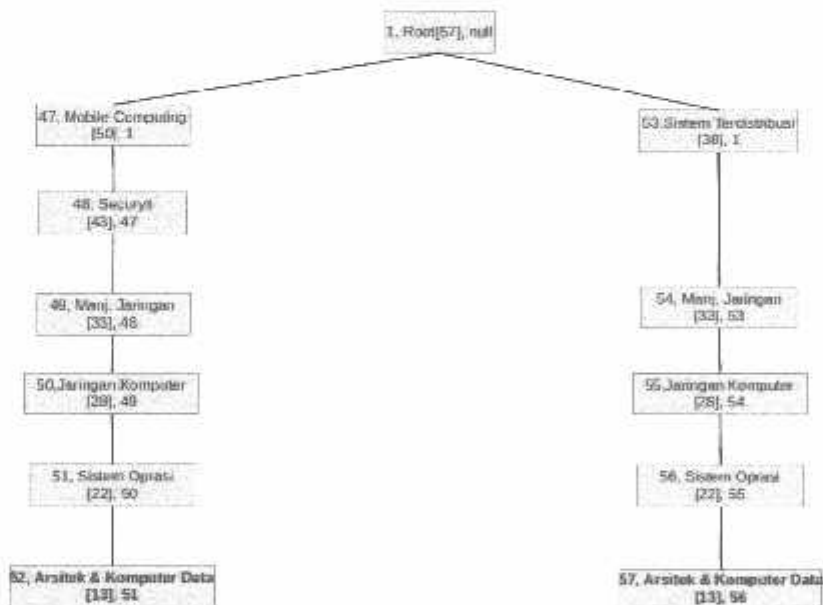
Gambar 3.8: Pohon 1 dengan Normalisasi dan Nilai Tepi



Gambar 3.9: Pohon 2 Tanpa Normalisasi dengan Nilai Tepi



Gambar 3.10: Pohon 3 dengan Normalisasi dan Nilai Tepi

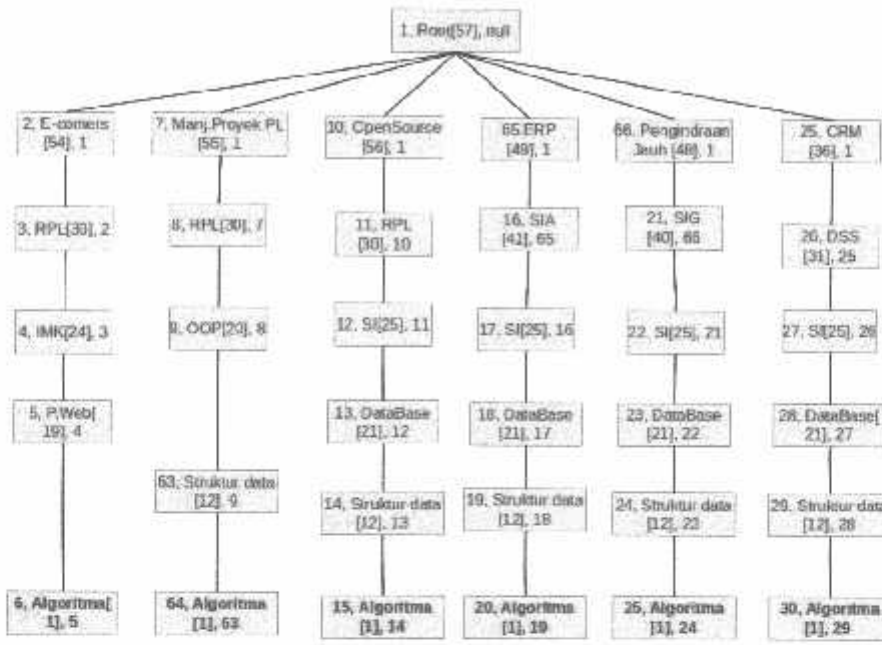


Gambar 3.11: Pohon 4 dengan Normalisasi dan Nilai Tepi

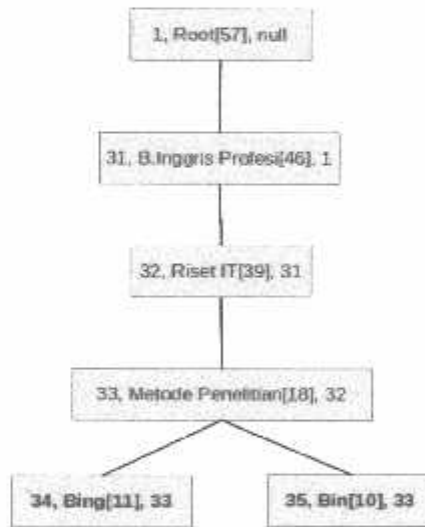


Gambar 3.12: Pohon 5 Tanpa dengan Normalisasi dan Nilai Tepi

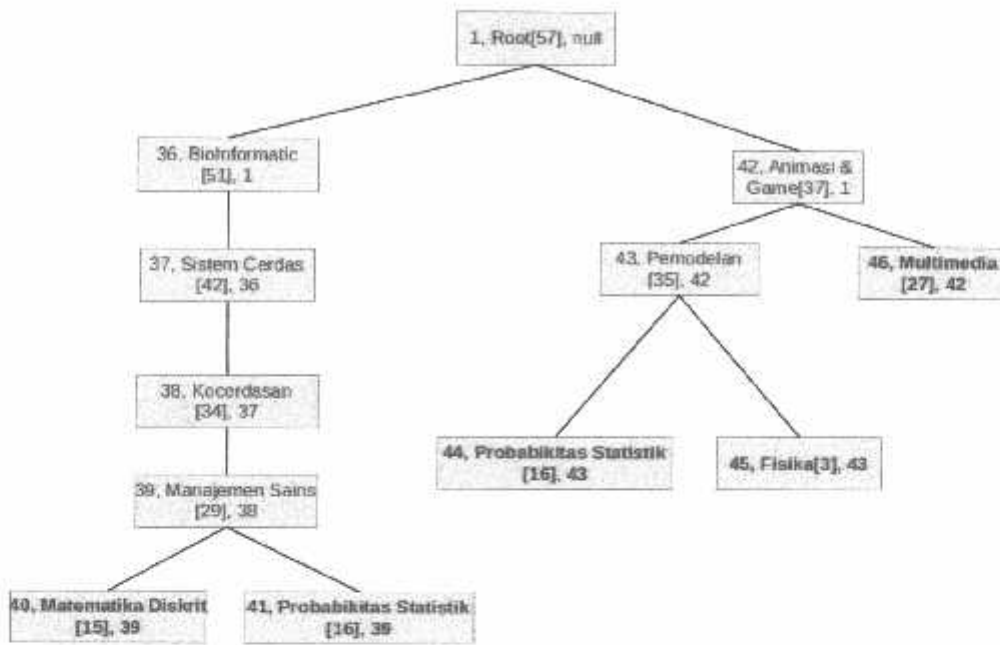
Dari gambar 3.8 hingga gambar 3.12 merupakan gambaran tentang pemodelan struktur pohon kurikulum yang memiliki struktur penting. Dalam penamaan tepi struktur pohon tersebut memiliki nilai tepi sebagai inisial pada masing-masing *node*. Analisa berikutnya adalah memberikan penamaan *parent* atau orang tua pada masing-masing *node*. Penamaan orang tua sesuai dengan metodologi diambil dari *node* atasnya maka nilai orang tua adalah nilai tepi pada *node* atasnya kecuali *root* karena *root* adalah teratas dari keseluruhan. Penamaan *node* orang tua terletak pada akhir setelah nama misal 34 bing[11] 33, artinya 34 merupakan nilai dari tepi pada *node* bahasa inggris yang memiliki nilai id 11 pada tabel dan merupakan *parent* pada *node* 33. *node* 33 merupakan penamaan *node* matakuliah metode penelitian, dan proses penamaan terus hingga *node root*, pengecualian pada *node root* merupakan *node* teratas dan tidak memiliki orang tua sehingga penamaan orang tua pada *node root* bernilai *Null*. Dapat kita lihat pada gambar-gambar berikut ini :



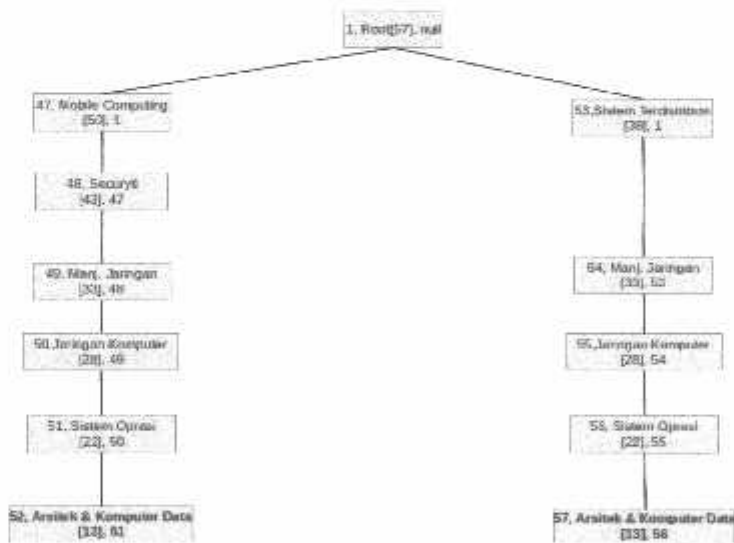
Gambar 3.13: Pohon1 dengan Nilai Tepi dan Parent



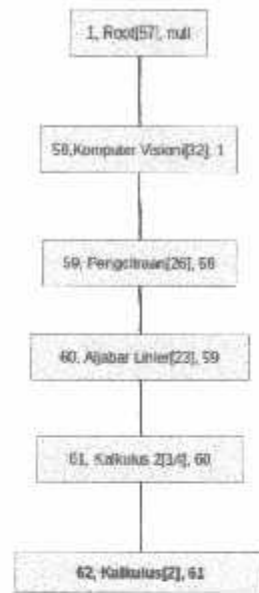
Gambar 3.14: Pohon 2 dengan Nilai Tepi dan Parent



Gambar 3.15: Pohon 3 dengan Nilai Tepi dan Parent



Gambar 3.16: Pohon 4 dengan Nilai Tepi dan Parent



Gambar 3.17: Pohon 5 dengan Nilai Tepi dan Parent

Pada hasil pembuatan struktur pohon dengan *Edge-Adjacency List Model* sesuai gambar 3.13 hingga gambar 3.17 dapat di konversi ke bentuk tabel untuk perancangan sistem dan DSS dengan *Edge-Adjacency List Model* sesuai analisa. Tabel tersebut terdiri dari tiga atribut id sebagai nilai tepi, id_matakuliah sebagai relasi tabel dalam menginisialkan matakuliah, dan orang tua sebagai penentu nilai tepi orang tua yang berelasi dengan id. Dapat kita lihat pada tabel berikut :

Tabel 3.1: Data *Edge-Adjacency List Model*

id	id_matakuliah	parent
1	57	null
2	54	1
3	30	2
4	24	3
5	19	4
6	1	5
7	55	1
8	30	7
9	20	8
10	56	1
34	11	33
35	10	33
36	51	1
37	42	36
38	34	37
39	29	38
40	15	39
41	16	39
42	37	1
43	35	42
44	16	43

11	30	10
12	25	11
13	21	12
14	12	13
15	1	14
16	41	65
17	25	16
18	21	17
19	12	18
20	1	19
21	40	66
22	25	21
23	21	22
24	12	23
25	36	1
26	31	25
27	25	26
28	21	27
29	12	28
30	1	28
31	46	1
32	39	31
33	18	32

45	16	43
46	27	43
47	50	1
48	43	47
49	33	48
50	28	49
51	22	50
52	13	51
53	38	1
54	33	53
55	28	54
56	22	55
57	13	56
58	32	1
59	26	58
60	23	59
61	14	60
62	2	61
63	12	9
64	1	63
65	49	1
66	48	1

3.3.2 Analisis Perulangan Matakuliah Kurang Baik

Analisis perulangan Matakuliah kurang baik merupakan kebutuhan sistem yang akan menampilkan matakuliah-matakuliah yang memiliki nilai yang tidak baik. Dalam hal ini untuk membuat suatu keputusan untuk menampilkan matakuliah kurang baik berdasarkan pada Rule yang telah dibuat. Rule tersebut di buat berdasarkan sisa *Limit* sks yang tersedia dan menyeleksi memilih sks yang mendekati maksimal, adapun *Rule* diantaranya:

1. Memilih matakuliah yang memiliki nilai C-E
2. Memilih matakuliah dengan sks yang mendekati batas maksimal
3. Memilih matakuliah sesuai dengan semester yang di tempuh

3.3.3 Analisis Pemilihan Matakuliah Pilihan

Analisis pemilihan matakuliah pilihan adalah kebutuhan mengenai jenis pilihan matakuliah. Matakuliah pilihan merupakan matakuliah yang dipilih salah satu

dari beberapa matakuliah yang ada. Dalam menentukan pilihan tersebut sistem akan memilih berdasarkan riwayat nilai dari anak-anak matakuliah pilihan. Riwayat nilai tersebut akan dijumlah dari setiap nilai dari anak-anak matakuliah pilihan yang bersangkutan dan dibagi sesuai dengan jumlah sks pada anak-anak matakuliah.

$$\text{Rata-rata nilai riwayat} = \frac{\sum (\text{Jumlah Nilai Setiap Anak Matakuliah})}{\sum (\text{Jumlah SKS Anak Matakuliah})} \dots (3-1)$$

Dari fungsi 3-1 di atas dapat diambil bahwa membandingkan nilai rata dari akademik sesuai riwayat. Dalam membedakan antara matakuliah pilihan tersebut dibuat suatu kategori pilihan sesuai dengan kebutuhan. Kategori tersebut di bagi menjadi lima yaitu pilihan 1, pilihan 2, pilihan 3, pilihan 4, dan pilihan 5. Dalam desain tabel tersebut untuk matakuliah pilihan akan terdapat atribut matakuliah pilihan sebagai pembantu untuk menentukan anak-anak matakuliah dan jenis matakuliah sebagai pembanding sesuai dengan kebutuhan semester pada saat bersangkutan.

Tabel 3.2: Desain Tabel Riwayat

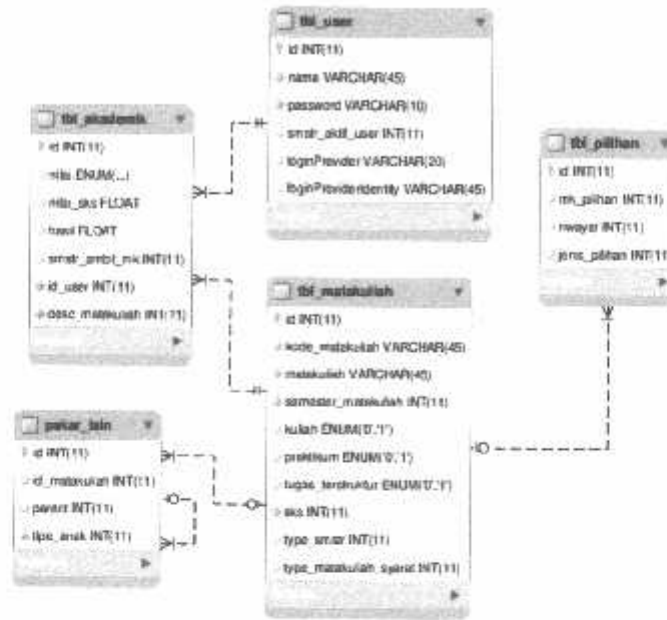
id	mk_pilihan	riwayat	jenis_pilihan
1	41	17	1
2	41	18	1
3	41	19	1
4	41	20	1
5	40	22	1
6	40	23	1
7	40	24	1
8	40	25	1
9	54	30	5
10	54	24	5
11	54	19	5
12	54	1	5
13	55	30	5
14	55	20	5
15	55	12	5
16	55	1	5
17	56	30	5
18	56	25	5
19	56	21	5
20	56	12	5
21	56	1	5
22	43	33	2
23	43	28	2
24	43	22	2

25	43	13	2
26	42	34	2
27	42	29	2
28	42	15	2
29	42	16	2

3.4 Rancangan Desain Entitas Model Hubungan

Desian ERD merupakan penggambaran model data mengenai relasi antar tabel dalam *database*. Model data merupakan alat yang digunakan dalam analisa untuk persyaratan data dan asumsi dalam sistem. ERD juga sebagai pondasi awal membangun sebuah aplikasi dengan menggunakan *database*. ERD memiliki tiga elemen umum sebagai dasar dalam model yaitu entitas sebagai hal-hal tentang apa yang kita cari, atribut sebagai data yang di kumpulkan tentang entitas, dan hubungan struktur yang di butuhkan untuk menarik informasi dari beberapa entitas.

Dalam membuat ERD dapat kita membuat entitas terlebih dahulu sesuai dengan analisa proses perencanaan matakuliah. Entitas yang terlibat matakuliah, akademik, *user*, pakar, dan matakuliah pilihan. Matakuliah entitas utama sebagai data utama. Akademik entitas yang di gunakan untuk menyimpan nilai dan kondisi pengambilan matakulia. *User* adalah entitas yang menjadi pelaku dalam sistem untuk melakukan perencanaan. Pakar merupakan entitas yang berisi *Edge-Adjacency List Model* yang berfungsi sebagai pemodelan data hirarki untuk tatanan makatkuliah. Matakuliah pilihan sebagai pengecualian matakuliah secara umum karena matakuliah pilihan menjadi matakuliah yang harus di pilih dari beberapa pilihan matakuliah. Relasi antara *user One to many* terhadap Akademik. Akedmik memiliki relasi *Many to One* ke Matakuliah. Matakuliah memiliki dua relasi yaitu *One to many* ke entiti pilihan dan pakar. Pakar memiliki relasi ke diri sendiri yang bersifat *One to many* untuk membuat rekursif dan penamaan hirarki, terdapat pada gambar 3.16 berikut :



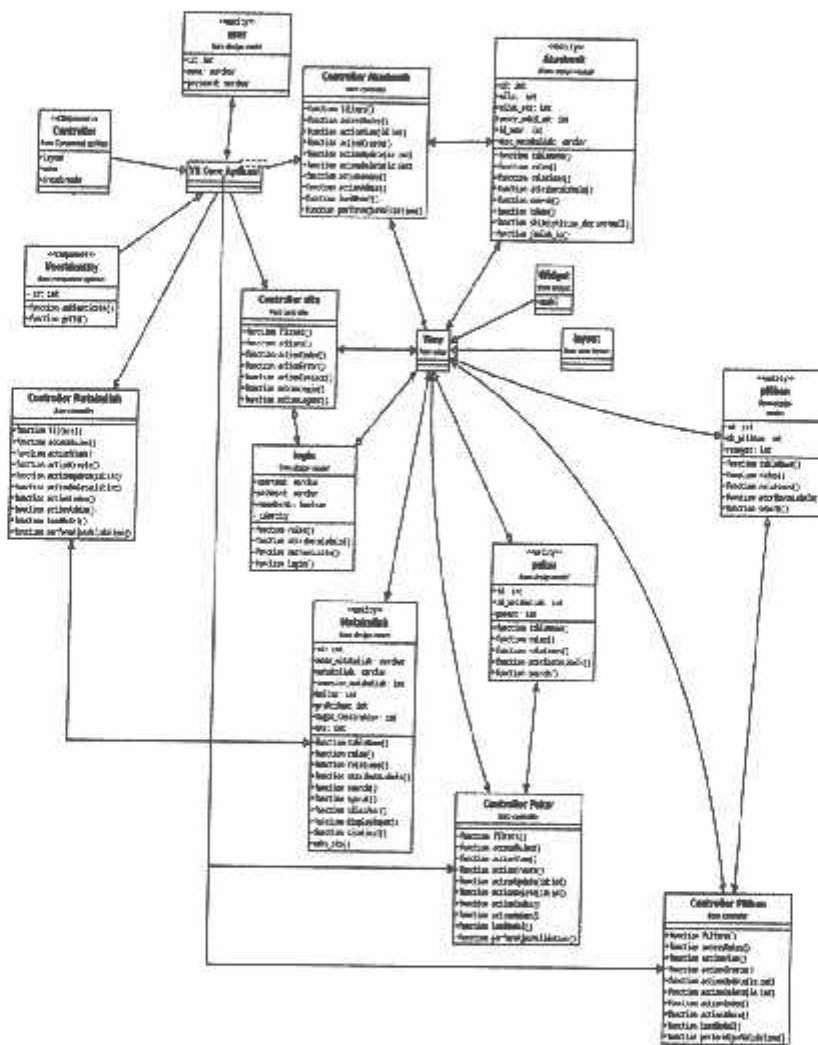
Gambar 3.18: ERD Sistem

3.5 Rancangan UML Diagram Sistem Informasi

Unified Modeling Language (UML) sebagai standar pemodelan di bidang orientasi objek rekayasa perangkat lunak. UML Menciptakan modeling visual dari perangkat lunak. Dalam UML memiliki konsep MVC (*Model View Controller*) sesuai dengan penggunaan framework Yii. Koscp *Model-View-Controller* yang sesuai dengan desain ERD. Dalam desain ERD memiliki 5 bagian yaitu Matakuliah, Akademik, *User*, Pakar pilihan, dan `tbl_pilihan` dengan masing-masing fungsi. Di setiap lima bagian memiliki 5 controller, 5 model, dan 5 view.

Dalam view tersimpan dalam *widget* untuk menampilkan nilai tabel pada masing controller. Aplikasi sebagai inti akan menentukan *controller* dan *action* dengan bantuan *UrlManager* sebagai contoh *controller* matakuliah maka akan mengarahkan pada `matakuliahControler` dan *actionIndex*. Aplikasi akan membuat instansi dari *controller* yang diminta untuk selanjutnya menagani permintaan pengguna. *Controller* akan menentukan *actionIndex* pada sebuah metode bernama *actionIndex* dalam kelas *controller* yang kemudian membuat dan menjalankan

function filter (Untuk akses atas controller dari permintaan). Apabila filter megijinkan Kemudian action membaca *post* model dari *database*. Action juga menyiapkan tampilan bernama *index* dengan model *post* yang mana *view* akan menampilkan atribut pada model *post* yang di eksekusi oleh beberapa *widget*. Hasil eksekusi akan di tanamkan ke dalam *layout* dan *action* akan mengakhiri pembuatan *view* dan menampilkan hasil kepada pengguna.

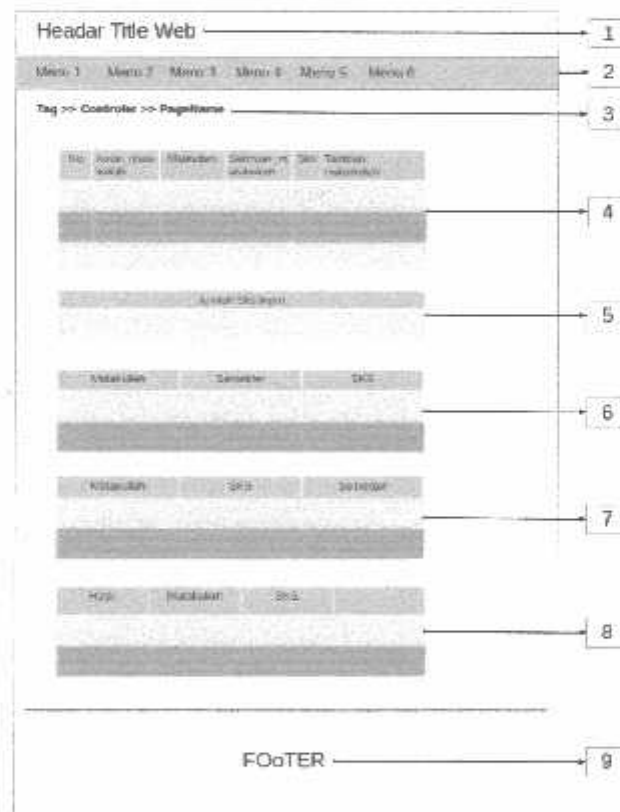


Gambar 3.19: UML Sistem Informasi

Pada proses pengiriman data pada setiap penghubung sesuai dengan gambar 2.5 Model MVC Yii. Terdapat Aplikasi sebagai letak sistem dan Core Yii sebagai pembantu dalam pengolahan. *Controller*, *view*, dan model saling berkaitan untuk mengalirkan data

3.6 Rancangan Desain *Interface*

Desain *interface* atau antar muka sistem dibuat agar membantu pengguna bisa mudah menggunakan. Dengan menggunakan bantuan alat LibreDraw akan mendesain antar muka sistem. Ada beberapa hal dasar dalam antar muka yaitu header, content, dan footer dapat kita lihat pada gambar 3.20 berikut :

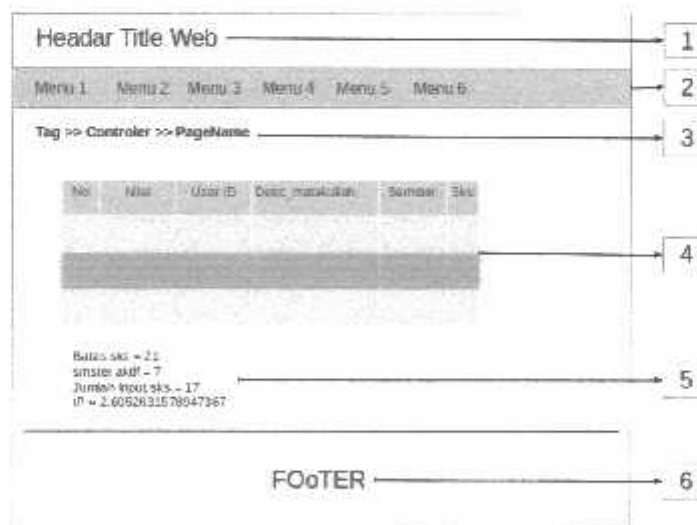


Gambar 3.20: Desain Antar Muka Sistem Matakuliah Input

Dari keterangan gambar 3.20 menyebutkan keterangan sesuai dengan angka di sampingnya. Keterangan sebagai berikut :

1. Bagian 1 merupakan *Header* Halaman *web* sebagai judul dalam sistem
2. Bagian 2 adalah macam-macam menu pada halaman *web*
3. Bagian 3 merupakan *tag* untuk mengetahui penamaan halaman
4. Bagian 4 adalah tabel berisikan tentang matakuliah-matakuliah yang siap diambilnya
5. Bagian 5 merupakan representasi jumlah input yang telah diambil
6. Bagian 6 merupakan tabel perencanaan matakuliah reguler
7. Bagian 7 merupakan tabel perencanaan matakuliah perulangan
8. Bagian 8 merupakan tabel perencanaan matakuliah pilihan
9. Bagian 9 adalah bagian *footer* pada halaman *web*

Selain gambar 3.20 juga terdapat desain antar muka untuk melihat hasil yang telah di inputkan. Sekligus dengan nilai-nilai dalam menyelesaikan matakuliah. Halaman tersebut terletak pada bagian akademik. Desain halaman Akademik pada gambar 3.21 berikut :



Gambar 3.21: Desain Antar Muka Sistem Akademik *View*

Pada gambar 3.21 adalah desain antar muka untuk akademik yang menampilkan nilai matakuliah dan lain-lain dengan keterangan sebagai berikut :

1. Bagian 1 merupakan Header Halaman *web* sebagai judul dalam sistem
 2. Bagian 2 adalah macam-macam menu pada halaman *web*
 3. Bagian 3 merupakan *tag* untuk mengetahui penamaan halaman
 4. Bagian 4 adalah tabel berisikan matakuliah yang telah diambil
 5. Keterangan Mengenai nilai IP, rencana semester, batas maks SKS
 6. Bagian 9 adalah bagian footer pada halaman *web*
-

BAB IV

IMPLEMENTASI DAN PENGUJIAN

4.1 Lingkungan Implementasi

Dalam ruang lingkup implementasi dalam membangun DSS untuk perencanaan dibutuhkan perangkat-perangkat lunak dan keras. Perangkat tersebut telah dijelaskan sebagian pada bab 2 dengan sub bab 2.5 , 2.6, dan 2.7 mengenai *web server* atau apache, MySQL *database*, dan PHP. Ruang Lingkup tersebut di pasang pada sistem oprasi linux dengan distro ubuntu versi 12.04 32-bit. Pada perangkat keras menggunakan prosesor cpu AMD Athlon(tm) X3 425 Processor × 3 dengan RAM 2.0 GiB dan hardisk 60GB. *Platform* tersebut tidak hanya berjalan pada sistem oprasi linux tetapi juga berjalan pada sistem oprasi lain.

4.1.1 Sistem Operasi ubuntu 12.04

Ubuntu merupakan salah satu distribusi linux yang berbasis debian dan distribusikan sebagai software bebas. Adapun fitur dalam ubuntu 12.04 dengan menggunakan vitur unity yang seblumnya gnome panel yang telah beralih semenjak versi 11.04. Hardware ubuntu mendukung batas minimal yang dianjurkan dalam penggunaan untuk ditanamkan diantara :

Tabel 4.1: Syarat Ubuntu *Hardware*⁽¹⁴⁾

Kebutuhan minimal	Desktop
Prosesor	700Mhz
Memory(RAM)	384Mb
Hard Driver	5 Gb
Monitor resolusi	1024x768

4.1.2 Database MySql

Pada keterangan Bab 2.7 menjelaskan *database* MySql menjadi perangkat lunak sistem manajemen basis data atau DBMS yang dikembangkan oleh Oracle dari sistem *database* relasional. Penerapan *database* pada DSS menjadi kebutuhan untuk menyimpan data analisa yang dibutuhkan. Proses analisi dengan gambar 3.17 ERD sistem adalah relasi antar tabel yang mana tabel matakuliah memiliki *Primary Key* (Pk) id atribut dengan struktur sebagai berikut :

Tabel 4.2: Struktur Tabel Matakuliah

Nama Colom	Type Data	Index
Id	Int(11)	Pk
kode_matakuliah	varchar(45)	
matakuliah	varchar(45)	
semester_matakuliah	int(11)	
kuliah	enum('0', '1')	
praktikum	enum('0', '1')	
tugas_terstruktur	enum('0', '1')	
sks	int(11)	
type_smstr	int(11)	
type_matakuliah_syarat	int(11)	

Dalam implementasi untuk membuat membutuhkan *script* SQL dalam MySql dengan *script* SQL sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `tbl_matakuliah` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `kode_matakuliah` varchar(45) NOT NULL,
  `matakuliah` varchar(45) NOT NULL,
  `semester_matakuliah` int(11) NOT NULL,
  `kuliah` enum('0','1') DEFAULT NULL,
  `praktikum` enum('0','1') DEFAULT NULL,
  `tugas_terstruktur` enum('0','1') DEFAULT NULL,
  `sks` int(11) NOT NULL,
  `type_smstr` int(11) DEFAULT NULL,
  `type_matakuliah_syarat` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`)
)
```

Tabel lain yang berkaitan dengan tabel matakuliah adalah tabel akademik. Tabel akademik berisikan tentang riwayat mahasiswa dalam mengambil dan menempuh matakuliah. Ada pun struktur tabel untuk akademik sebagai berikut :

Tabel 4.3: Struktur Tabel Akademik

Nama Colom	Tipe Data	Index	Link to
id	Int(11)	Pk	
nilai	enum('A', 'B', 'B+', 'C', 'C+', 'D', 'E')		
nilai_sks	float		
hasil	float		
smstr_ambil_mk	Int(11)		
id_user	Int(11)	fk(User)	tbl_user-> id
desc_matakuliah	Int(11)	fk(Mata kuliah)	tbl_matakuliah-> id

Dari struktur tabel 4.3 menjelaskan bahwa id sebagai *primary key* yang memiliki dua *foriegn key* terhadap *user* dan *matakuliah*. Adapun *script SQL* dalam pembuatanya :

```
CREATE TABLE IF NOT EXISTS `tbl_akademik` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nilai` enum('A','B','B+','C','C+','D','E') DEFAULT NULL,
  `nilai_sks` float DEFAULT NULL,
  `hasil` float DEFAULT NULL,
  `smstr_ambil_mk` int(11) DEFAULT NULL,
  `id_user` int(11) NOT NULL,
  `desc_matakuliah` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `fk_tbl_akademik_tbl_user1` (`id_user`),
  KEY `fk_tbl_akademik_tbl_matakuliah1` (`desc_matakuliah`),
)
```

Tabel lain yang terdapat pada desain ERD adalah tabel *user*. Tabel *user* adalah tabel untuk penamaan *user* siapa saja yang terlibat kedalam sistem. Struktur tabel *user* sebagai berikut :

Tabel 4.4: Struktur Tabel User

Nama Colom	Tipe Data	Index
Id	int(11)	Pk

nama	varchar(45)	
password	varchar(45)	
smstr_aktif_user	int(11)	

Pada tabel *user* terdapat *id* sebagai *primari key* pada tabel yang akan di alirkan ke tabel akademik. Adapun *script SQL* yang sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `tbl_user` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `nama` varchar(45) NOT NULL,
  `password` varchar(10) NOT NULL,
  `smstr_aktif_user` int(11) DEFAULT NULL,
  `loginProvider` varchar(20) DEFAULT NULL,
  `loginProviderIdentity` varchar(45) DEFAULT NULL,
  PRIMARY KEY (`id`) USING Btree
)
```

Adapun tabel lain sebagai hirarki yaitu tabel pakar. Tabel pakar berfungsi sebagai peimplementasian dari model *edge-adjacency list* untuk pemetaan prasyarat dan syarat matakuliah. Pada tabel pakar memiliki *Fk* terhadap dirinya sendiri karena memiliki model data yang memiliki *parent* atau orang tua terhadap *id* sebelumnya. Dapat kita liat struktur sebagai berikut :

Tabel 4.5: Tabel Pakar *Edge-Adjesensi List Model*

Nama Colom	Type Data	Index	Link to
<i>id</i>	int(11)	Pk	
<i>id_matakuliah</i>	int(11)	fk(Matakuliah)	tbl_matakuliah -> id
<i>parent</i>	int(11)	fk(pakar)	pakar -> id
<i>tipe_anak</i>	int(11)		

Tabel matakuliah pilihan merupakan berisikan tentang riwayat mengenai matakuliah yang memiliki struktur hirarki dengan matakuliah pilihan itu sendiri. Terdapat *id* sebagai *primary key* dan *foriegn key* yang tersambung ke tabel matakuliah. Adapun struktur dalam tabel berikut :

Tabel 4.6: Tabel Struktur Matakuliah Pilihan

Nama Colom	Tipe Data	Index	Link to
Id	int(11)	Pk	
mk_pilihan	int(11)	Fk(matakuliah)	tbl_matakuliah -> id
riwayat	int(11)		
jenis_pilihan	int(11)		

Dari tabel di atas ID memiliki PK sebagai index utama yang juga memiliki Fk ke tabel matakuliah. Fk tersebut untuk mengambail data matakuliah-matakuliah yang berkaitan dengan struktur pilihan. Dapat kita liat *script* SQL sebagai berikut :

```
CREATE TABLE IF NOT EXISTS `tbl_pilihan` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `mk_pilihan` int(11) DEFAULT NULL,
  `riwayat` int(11) DEFAULT NULL,
  `jenis_pilihan` int(11) DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `fk_tbl_pilihan_1_idx` (`mk_pilihan`)
)
```

4.1.3 PHP 5

PHP merupakan bahasa pemrograman yang di tanamkan ke dalam HTML yang dirancang untuk pengembangan *web* agar menghasilkan halaman *web* dinamis. PHP 5 merupakan versi PHP ke 5 yang dibutuhkan. Seperti yang di jelaskan pada bab 2.5 PHP juga sebagai *server-side script* diinterpretasikan oleh server *web* dengan module PHP untuk menghasilkan halaman *web* yang dihasilkan. PHP memiliki banyak module-module yang berfungsi untuk membantu proses sistem. Pada framework Yii yang dijelaskan pada bab 2.5.3 Yii merupakan komponen berbasis PHP *web* kerangka aplikasi *web* dengan berorientasi objek dan bersifat *open source*. Dalam penggunaan Yii ada beberapa modul yang perlu dipasang ke dalam PHP di antaranya :

1. PHP5-mcrypt

PHP5-mcrypt merupakan paket yang disediakan untuk fungsi mcrypt dalam skrip PHP sebagai enkripsi file dengan metode modern seperti AES pengganti dari crypt yang populer di unix.

2. PHP5-mysql

Paket PHP5-mysql merupakan paket yang menyediakan modul untuk mengkoneksikan *database* Mysql langsung dari PHP skrip termasuk juga generik mysql modul yang dapat terhubung ke semua versi Mysql .

3. PHP5-cli

Paket PHP5-cli menjadi paket yang menyediakan interpreter comand line perintah untuk pengujian skrip PHP dari shell atau melakukan tugas umum menggunakan skrip shell.

4. PHP5-memcached

paket PHP5-memcached ini berfungsi sebagai caching yang dirancang khusus untuk dinamis *web* aplikasi agar mengurangi beban *database* dengan menyimpan objek ke dalam memori.

5. PHP-apc

PHP-apc juga memiliki fungsi yang mirip dengan PHP5-memcached sebagai caching tetapi yang membedakan untuk area lokal apc berjalan lebih cepet dan dapat di gunakan secara bersama-sama.

4.2 Implementasi Modul

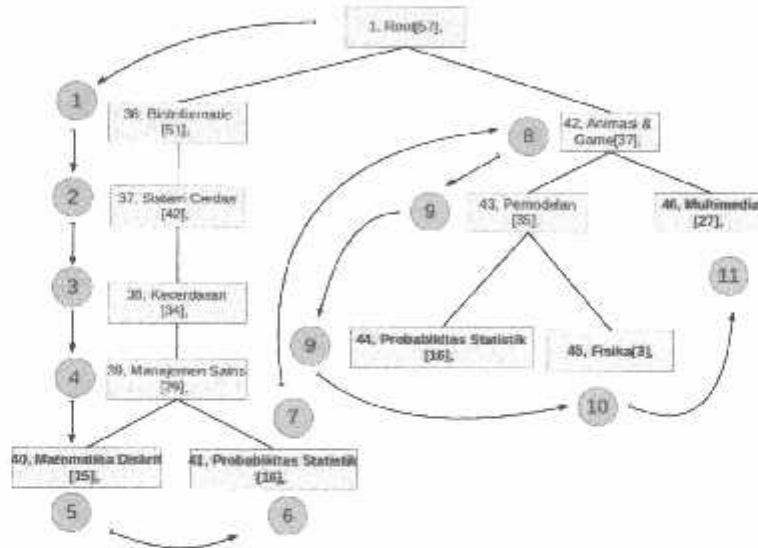
Implementasi modul adalah bagian dalam sistem informasi yang mendukung berjalannya proses bisnis. Di dalam DSS terdiri dari beberapa modul yaitu modul untuk hirarki penentu syarat prasyarat matakuliah, modul yang ke dua penentu matakuliah untuk perulangan, dan modul untuk menentukan matakuliah pilihan. Tiga modul tersebut adalah proses yang mendukung jalan DSS sesuai analisa model *base* DSS pada gambar 3.3

4.2.1 Modul Hirarki Matakuliah

Modul hirarki matakuliah berjalan pada model matakuliah. Modul yang terdapat pada model akan melakukan seleksi dengan aturan-aturan yang berlajan. Dalam aturan daiantara :

1. Pembacaan hirarki di mulai dari atas atau *pre-order* untuk menyeleksi secara keseluruhan disini menggunakan *pre-order tree* dengan nilai sesuai yang memilih dari kiri sampai paling bawah selanjutnya ke kanan ambil contoh implementasi pada gambar 3.9 maka akan muncul deretan matakuliah sesuai gambar 4.1 ilustrasi berikut :

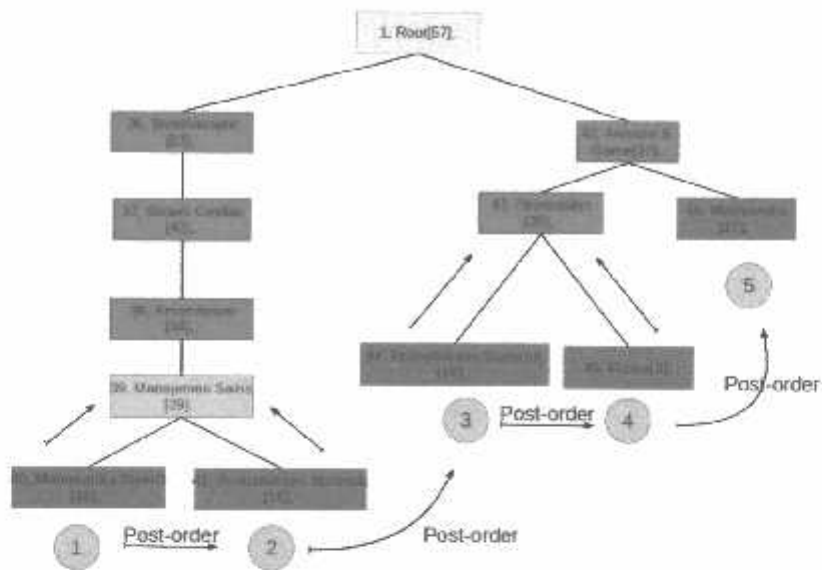
Dari hasil *pre-order* pada gambar 4.1 membarikan hasil *root* >> Bioinformatik >> Sistem cerdas >> Kecerdasan Buatan >> Manajemen Sains >> Matematika



Gambar 4.1: Urutan *Pre-order* Akar Matakuliah

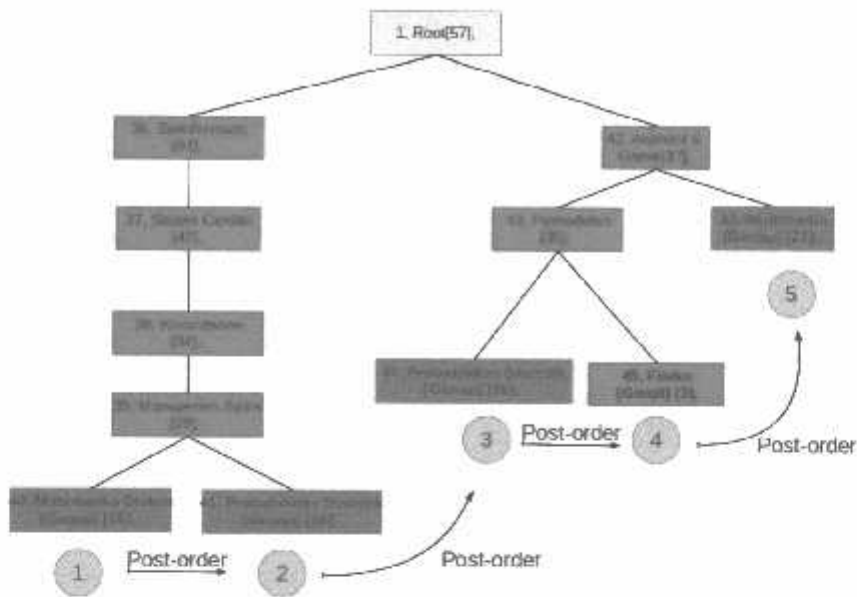
diskrit >> Probabilistas statistik >> Animasi dan *root*>>Game >> Pemodelan >>Probabilistas statistik >> Fisika>>Multi media.

2. Menyeleksi matakuliah bersyarat yang telah diambil untuk menampilkan bapak atau orang tua dalam heararki ini menggunakan *post-order* dari hasil nomor 1. *post-order* akan melihat matakuliah apa yang berada diatas matakuliah yang telah diambil misal telah diambil matakuliah Matematika diskrit, Probabilistas dan fisika maka hasilnya adalah manajemen sains pada gambar 4.2 berikut :



Gambar 4.2: *Post-order* dengan Seleksi Matakuliah

3. Menyeleksi matakuliah paling bungsu dengan menggunakan *post-order* sesuai semester dan tipe semester ganjil genap. Kasus *user A* sekarang adalah semester 2, maka matakuliah yang dapat diambil adalah semester 2 dan matakuliah semester genap dengan matakuliah yang terdapat dalam hirarki dengan posisi paling bungsu. Pertama menyeleksi matakuliah paling bungsu yaitu matematika diskrit, probabilistas, fisika, dan multimedia sesuai pada gambar akar 3.9 sebagai contoh. Untuk matakuliah Matematika diskrit, dan Probabilistas termasuk kedalam matakuliah semester 2 sedangkan multimedia merupakan matakuliah semester 4 paling bungsu yang ketiga matakuliah Matematika diskrit, probabilistas, dan multimedia termasuk kedalam semester genap sedangkan fisika termasuk semester ganjil yaitu semester 1 maka gambar *post-order* sebagai berikut :



Gambar 4.3: *Post-order* Seleksi Anak Paling Bungsu

Dalam Implementasikan pada program matakuliah terkait dengan nilai *user* dan daftar akademik matakuliah yang telah diambil. Dalam penentuan hirarki tersebut di jalan ke *script* SQL sebagai berikut :

```

SELECT * FROM (
SELECT a.* FROM
(SELECT d.* FROM pakar_lain a
inner join pakar_lain b on a.parent = b.id
inner join tbl_akademik c on a.id_matakuliah = c.desc_matakuliah
inner join tbl_matakuliah d on b.id_matakuliah = d.id

where c.id_user=".Yii::app()->user->id."
and c.smstr_ambil_mk <= ".Yii::app()->user-
>getState('smstr_aktif_user')."
and d.semester_matakuliah > ".Yii::app()->user-
>getState('smstr_aktif_user')."
and d.type_smstr = (".(Yii::app()->user->getState('smstr_aktif_user')
+))."%2)
GROUP BY b.id_matakuliah) a

UNION SELECT d.* FROM SKS.tbl_matakuliah d
WHERE EXISTS
(SELECT c.* FROM `pakar_lain` c
WHERE NOT EXISTS (SELECT b.* FROM `pakar_lain` b WHERE
c.`id`=b.`parent` )
AND c.`id_matakuliah`=d.`id` GROUP BY c.`id_matakuliah`)

```



```

AND d.`semester_matakuliah` > ".Yii::app()->user-
>getState('smstr_aktif_user')."
AND `type_smstr` = (".(Yii::app()->user->getState('smstr_aktif_user')
+1)."%2)) a
WHERE NOT EXISTS (SELECT b.* FROM tbl_pilihan b WHERE
b.mk_pilihan=a.id)
AND NOT EXISTS (SELECT c.* FROM tbl_akademik c WHERE
c.desc_matakuliah=a.id AND c.id_user=".Yii::app()->user->id.")

```

Kode tersebut dipasang pada *controller* matakuliah yang sebagai tampilan halaman kebutuhan. Pada desain antar muka bab 3.6 gambar 3.19 keterangan nomor 6. Menampilkan halaman hasil seleksi matakuliah dapat di lihat pada gambar 4.4 berikut :

MATAKULIAH	Semester matakuliah	sks.
Struktur Data + Praktikum	2	4
Kalkulus 2	2	4
Metode Penelitian	2	2
Arsitektur Komputer & Komunikasi Data	2	3
Matematika Diskrit	2	3
Probabilitas dan Statistik	2	2
Sistem Multimedia + Praktikum	4	4

Gambar 4.4: Hasil DSS Implementasi

4.2.2 Modul Perulangan Matakuliah

Perulangan matakuliah merupakan fase berikutnya setelah bab 4.2.1. Pada modul perulangan matakuliah terjadi proses mengecek matakuliah dengan nilai kecil sesuai dengan aturan pada bab 3.3.2. Perulangan matakuliah terjadi pada *action* akademik karena hanya menyeleksi pada bagian nilai-nilai pada matakuliah yang telah diambil. Untuk perulangan tidak mempergunakan pemodelan data hirarki melainkan aturan-aturan. Modul perulangan akan berlaku apabila sisa sks dari seleksi pada modul hirarki matakuliah. Oleh karena dibutuhkan fungsi untuk menyeleksi. Fungsi tersebut di beri nama *function sisaLimit()*. *Function sisaLimit()* ini akan

Fungsi tersebut di beri nama *function sisaLimit()*. *Function sisaLimit()* ini akan menyoeksi jumlah sks dari modul hirarki matakuliah yang tampil - batas limit, dapat kita lihat pada rumus berikut :

$$\text{Funtion sisaLimit() = } \sum \text{Sks.Matakuliah.Herarki} - \text{Batas Limit} \dots\dots\dots (4-1)$$

Untuk penentuan jumlah sks matakuliah hirarki terdapat pada variabel *Limit*. Yang mana akan menjumlah SKS pada matakuliah yang di tampilkan seperti fungsi berikut :

```
$limit = Yii::app()->db->createCommand("select sum(sks) a from (
        select sum(sks) a from (SELECT a.* FROM
(SELECT d.* FROM pakar_lain a
inner join pakar_lain b on a.parent = b.id
inner join tbl_akademik c on a.id_matakuliah = c.desc_matakuliah
inner join tbl_matakuliah d on b.id_matakuliah = d.id

where c.id_user='".Yii::app()->user->id.'"
and c.smstr_ambil_mk <= '".Yii::app()->user->getState('smstr_aktif_user')."
and d.semester_matakuliah > '".Yii::app()->user->getState('smstr_aktif_user')."
and d.type_smstr = ('.Yii::app()->user->getState('smstr_aktif_user')
+1)."%2)
GROUP BY b.id_matakuliah) a

UNION SELECT d.* FROM SKS.tbl_matakuliah d
WHERE EXISTS
(SELECT c.* FROM `pakar_lain` c
WHERE NOT EXISTS (SELECT b.* FROM `pakar_lain` b WHERE
c.`id`=b.`parent`)
AND c.`id_matakuliah`=d.`id` GROUP BY c.`id_matakuliah`)
AND d.`semester_matakuliah` > '".Yii::app()->user->getState('smstr_aktif_user')."
AND `type_smstr` = ('.Yii::app()->user->getState('smstr_aktif_user')
+1)."%2)
) a;")->queryAll();
```

Selain fungsi tersebut juga terdapat variabel *batasLimit*. Variabel tersebut berfungsi untuk menentukan batas sks yang diambil sesuai dengan nilai Indeks prestasi di setiap mahasiswa. Isi dari variebel *bataslimit* sebagai berikut :

```
$max=Yii::app()->db->createCommand("SELECT
        IF(SUM(a.hasil)/sum(b.sks)>=3, 24,
        IF(SUM(a.hasil)/sum(b.sks)>=2.5, 21,
        IF(SUM(a.hasil)/sum(b.sks)>=2.0, 15,
        IF(SUM(a.hasil)/sum(b.sks)>=1.5, 15, 10))))
```

```

FROM tbl_akademik a
inner join tbl_matakuliah b
on a.desc_matakuliah = b.id
where a.smstr_ambil_mk = '".Yii::app()->user-
>getState('smstr_aktif_user')."'
AND a.id_user='".Yii::app()->user->id."'")->queryAll();

```

Dari hasil pengurangan dari *Funtion sisaLimit()* maka akan menghasilkan sisa sks dari batas sks yang di tentukan. Sisa sks tersebut akan dipilih untuk mencari matakuliah yang mencukupi untuk memenuhi jumlah sisa sks. Dalam pencarian matakuliah kurang baik menggunakan aturan-aturan yang berlaku maka hal tersebut diimplementasikan ke dalam *builder* SQL dengan menggunakan sekenario Yii pada model seperti berikut :

```

$criteria->condition = "tblMatakuliah.type_smstr = '".
(Yii::app()->user->getState('smstr_aktif_user') (1) & 2)."'
and nilai_sks BETWEEN 0 AND 2";
$criteria->order="nilai asc";
$criteria->group "tblMatakuliah.matakuliah";

```

Dari hasil perhitungan tersebut itu akan menghasilkan matakuliah-matakuliah yang memiliki kekurangan yang mungkin harus di ulang untuk memperbaiki nilai. Adapaun hasilna sebagai berikut :

SKS Jelek

Displaying 1-3 of 3 results.

Matakuliah	Sks	Semester Matakuliah
Interaksi Manusia & Komputer	2	4
Sistem Informasi + Praktikum	4	4
Metode Penelitian	2	2

Gambar 4.5: Hasil Perulangan Matakuliah Jelek

4.2.3 Modul Matakuliah Pilihan

Dalam modul matakuliah pilihan merupakan modul yang berfungsi menentukan pilihan dari jenis matakuliah pilihan. Matakuliah pilihan terdiri dari 2 – 3 jenis macam. Pada analisa bab 3.3.3 menjelaskan bahwa penentuan pilihan matakuliah berdasarkan pada riwayat nilai mahasiswa. Riwayat nilai merupakan

proses menyimpan data nilai terhadap matakuliah-matakuliah yang telah disimpan pada tabel riwayat yang akan menghasilkan nilai perbandingan antara riwayat yang setara atau dalam semester yang sama dengan persamaan (3-1). Untuk contoh penentuan nilai perbandingan matakuliah pilihan 2. Matakuliah pilihan 2 terdiri dari 2 macam matakuliah yaitu sistem cerdas dan security. Riwayat kedua matakuliah ini terlihat pada tabel 4.7 berikut :

Tabel 4.7: Tabel Matakuliah Pilihan 2

Matakuliah pilihan	Riwayat	Jenis_pilihan	Nilai	SKS	Nilai x SKS
Sistem Cerdas	Kecerdasan buatan	2	3	3	9
	Manajemen Sains	2	2	2	4
	Matematika Diskrit	2	3	3	9
	Probabilistik dan Statistika	2	3	2	6
Security	Arsitektur komputer	2	1	3	3
	Sistem Operasi	2	2	4	8
	Jaringan Komputer	2	2	4	8
	Manajemen Jaringan Komputer	2	1	3	3

Dari tabel 4.7 terdapat nilai dari setiap riwayat dengan menggunakan persamaan (3-1) maka hasil dari persamaan berikut

$$\text{Rata-rata nilai riwayat security} = \frac{(3+8+8+3)}{(14)} = 1.57$$

$$\text{Rata-rata nilai riwayat sistem cerdas} = \frac{(9+4+9+6)}{(10)} = 2.8$$

Dari hasil persamaan maka akan menghasilkan nilai perbandingan dalam jenis pilihan 2 yaitu 1.57 dan 2.8 yang diambil nilai tertinggi sebagai kompetensi terbaik berdasarkan riwayat nilai-nilai yang terdapat pada tabel. Dapat kita lihat implementasi pada hasil gambar berikut :

Pilihan

Displaying 1-2 of 2 results.		
matakuliah	skrs	jumlah pilihan
SIA	3	1
Sistem Cerdas	3	2

Gambar 4.6: Hasil Pembandingan matakuliah pilihan

4.3 Implementasi Antarmuka Sistem

Dalam antarmuka sistem terdiri dari dua macam yang menjadi halaman pokok yaitu *user* dan *admin*. *User* disini adalah mahasiswa yang melakukan proses rencana studi dan *admin* bertugas sebagai manajemen sistem. DSS perencanaan berjalan pada *user* karena *user* adalah pengguna sistem.

4.3.1 Antarmuka User

Antarmuka *user* terdiri dari tiga macam secara umum yaitu halaman *login*, halaman Matakuliah, dan halaman Akademik. Halaman *Login* merupakan pintu masuk awal untuk melakukan *autentifikasi* data terhadap *user* atau mahasiswa yang akan masuk. Terdiri dari dua input yaitu nama dan password dari mahasiswa bersangkutan. Gambar tampilan halaman *login* sebagai berikut :

Gambar 4.7: Halaman *Login* Sistem

Setelah login mahasiswa dapat mengakses halaman matakuliah. Halaman matakuliah merupakan halaman yang menjadi untuk menambah matakuliah dan menampilkan hasil perencanaan dalam tabel. Dapat terlihat pada gambar 3.19 desain antarmuka menjadi 1 halaman. Menambah matakuliah adalah sebagai proses tambah matakuliah yang ingin direncanakan dalam proses KRS. Tabel di bawahnya menjadi implementasi hasil DSS untuk membantu mahasiswa untuk prediksi perencanaan. Antarmuka pada halaman matakuliah dapat kita lihat pada gambar 4.8 berikut :

DSS Perencanaan KRS

Home | Matakuliah | DSS Perencanaan KRS | Logout | Login | Register

Home > Matakuliah > Ambil Matakuliah

Ambil Matakuliah

Halaman: Perencanaan Matakuliah > DSS Perencanaan Matakuliah > DSS Perencanaan Matakuliah

Displaying 1-8 of 8 results

No	Kode Matakuliah	Nama Matakuliah	SKS	Prasyarat	Aksi
36	PS118	Customer Process Management	3	3	+
37	PS119	Account & Core & Produk	3	4	+
38	PS120	Green Terintegrasi	3	3	+
39	PS121	Praktikum Manajemen	3	3	+
40	PS122	GD	3	3	+
41	PS123	GD	3	3	+
42	PS124	Praktikum	3	3	+
43	PS125	Genetik	3	3	+

Ambil Matakuliah

All rights reserved.

Displaying 1-4 of 4 results

Kode Matakuliah	Nama Matakuliah	SKS	Prasyarat
Customer Process Management		3	3
Account & Core & Produk		3	4
Green Terintegrasi		3	3
Praktikum Manajemen		3	3

SKS Jelek

Displaying 1-2 of 2 results

No	Kode Matakuliah	Nama Matakuliah	SKS	Prasyarat
1	PS118	Customer Process Management	3	3
2	PS119	Account & Core & Produk	3	4

Pilihan

Displaying 1-2 of 2 results

No	Kode Matakuliah	Nama Matakuliah	SKS	Prasyarat
3	PS120	Green Terintegrasi	3	3
4	PS121	Praktikum Manajemen	3	3

Copyright © 2022 by M. Daryono
All Rights Reserved
Powered by UIN Ar-Raniry

Gambar 4.8: Halaman Matakuliah Sistem

Halaman yang dapat diakses mahasiswa adalah akademik. Halaman ini berisikan tentang tabel hasil perencanaan dan nilai selama studi hingga akhir. Adapun

fitur untuk menghapus matakuliah yang diinginkan. Sesuai dengan desain antarmuka pada gambar 3.20 menjelaskan tiap fungsi *form* maka dapat kita lihat gambar 4.9 berikut ini :



Gambar 4.9: Halaman Akademik Sistem

4.4 Pelaksanaan Pengujian

Pada pelaksanaan pengujian terhadap *user* yaitu mahasiswa teknik informatika Institut Teknologi Nasional Malang. Dengan cara, mahasiswa mencoba menggunakan aplikasi ini dan mempergunakan sistem pendukung keputusan. Setelah melakukan percobaan sistem oleh mahasiswa selanjutnya mengisi kuisiner dan komentar hasil dari percobaan aplikasi sistem pendukung keputusan. Berdasarkan pada Rumus perhitungan hasil kuisiner berikut :

$$\text{Perhitungan Kuisiner} = \frac{\sum \text{Jumlah tiap kategori}}{\sum \text{Total keseluruhan kategori}} * 100 \quad (4-1)$$

Dari hasil kuisiner mendapatkan hasil persentasi pada setiap pertanyaan sebagai berikut :

Pertanyaan 1. Apakah menurut anda sistem pendukung perencanaan ini baik

sebagai fungsinya ?

Table 4.1: Hasil Persentasi Dari Pertanyaan 1 Oleh Responden

	Frekuensi Reponden	Persentase
Sangat Baik	1	14.29%
Baik	6	85.71%
Cukup	0	0
Kurang Baik	0	0
Tidak Baik	0	0

Pertanyaan 2. Apakah sistem ini membantu proses pengambilan perencanaan matakuliah ?.

Table 4.2: Hasil Persentasi Dari Pertanyaan 2 Oleh Responden

	Frekuensi Reponden	Persentase
Sangat Baik	1	14.29%
Baik	4	57.14%
Cukup	2	28.57
Kurang Baik	0	0
Tidak Baik	0	0

Pertanyaan 3. Apakah sistem pendukung keputusan perencanaan ini memberikan lauran sesuai dengan harapan ?

Table 4.3: Hasil Persentasi Dari Pertanyaan 3 Oleh Responden

	Frekuensi Reponden	Persentase
Sangat Baik	0	0
Baik	4	57.14%
Cukup	3	42.86%
Kurang Baik	0	0
Tidak Baik	0	0

Dengan menyatukan hasil frekuensi responden dari setiap pertanyaan maka akan menghasilkan nilai dari sistem pendukung perencanaan matakuliah sebagai berikut :

Tabel 4.8: Hasil Persentasi Kuisisioner

	Frekuensi Reponden	Persentase
Sangat Baik	2	9.52%
Baik	14	66.67%
Cukup	5	23.81%
Kurang Baik	0	0
Tidak Baik	0	0

Pengujian Terhadap tampilan dengan menggunakan 5 macam *browser* yang

berbeda-beda. Dilihat pada aspek *form* pada hasil halaman *web* apakah baik, cukup, dan kurang baik. Diuji ke *browser* mozilla firefox, chromium, opera, safari dan IE, maka memberikan hasil :

Tabel 4.9: Hasil Uji *Browser*

Jenis Browser	Baik	Cukup	Kurang
FireFox Mozilla	√		
Chromium	√		
IE			√
Opera	√		
Safari	√		

Pada tabel 4.9 mendapatkan hasil bahwa uji tampilan yang di lakukan pada 4 macam *browser* yang berbeda yaitu firefox mozilla, chromium, Opera, dan Safari memiliki tampilan yang baik sedangkan pada *browser* IE kurang baik.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari analisa dan pengujian yang telah di lakukan, maka dapat diambil kesimpulan antara lain :

1. Penelitian ini untuk membangun aplikasi pendukung keputusan dalam merencanakan matakuliah
2. Aplikasi ini dibangun berdasarkan buku pedoman kurikulum dan silabus 2009-2014 Teknik Indormatika S-1 Institut Teknologi Nasional Malang
3. Aplikasi ini memberikan saran matakuliah yang perlu diambil dengan memaksimalkan jumlah SKS sehingga mahasiswa dapat lulus dengan tepat waktu

5.2 Saran

Adapun saran-saran yang dapat di sampaikan berkaitan dengan program sistem pendukung keputusan ini adalah sebagai berikut :

1. Sistem pendukung keputusan ini dapat dikembangkan menjadi lebih baik dengan penambahan fitur-fitur untuk pengukur berat jam kerja yang belum ada pada sistem ini
2. Bisa di tambahkan metode lain pada penentuan matakuliah pilihan seperti metode Uji-t sampel Independen

DAFTAR PUSTAKA

- [1] Kusrini, Konsep dan Aplikasi Sistem Pendukung Keputusan, Andi, Yogyakarta, 2007
- [2] Institut Teknologi Nasional, Buku panduan Silabus Akademik, Malang, 2008
- [3] Hariantono, Kristanto, Konsep dan Perancangan *database*, Andi, Yogyakarta, 2004
- [4] Chucher Clare, *Begining Databsc desain Novice to Profesional.*, Appres, New York, 2004
- [5] Janet Valade, *PHP and MySQL For Dummies*, 4th Edition, Appress, New York, 2009
- [6] Harrington, Jan L *Relation database desain and Implementasi*, USA:Morgan Kaufan, 2009
- [7] Al Ahmar M. Ayman, A Prototype Student Advising Expert System Supported with an Object-Oriented *database* , *The Computer Journal*, 100-1004
- [8] T. Feghali, I. Zbib, and S. Hallal, "A web-based decision support tool for academic advising", *Educational Technology & Society*, Vol. 14, No. 1, pp. 82–94, 2011.
- [9] Gijs Van Tulder, 2011, Storing Hierarchical Data in a *database* Article, <http://www.sitepoint.com/hierarchical-data-database/>. Tanggal akses 12 Desember 2012, pukul 20.00 WIB
- [10] Peter Brawley and Arthur Fuller, *Get It Done with MySQL 5&6*, Artful Software Developer ,2011, <http://www.artfulsoftware.com/mysqlbook/sampler/mysqled1ch20.html>. Tanggal akses 21 Desember 2012
- [11] Mike Hillyer, 2011, <http://web.archive.org/web/20100104011746/http://dev.mysql.com/tech-resources/articles/hierarchical-data.html>. Tanggal akses 30 Desember 2012, pukul 19.00 WIB
- [12] Wikipedia, http://en.wikipedia.org/wiki/Decision_Support_System. Tanggal akses 12 Desember 2012, pukul 20.00 WIB
- [13] Yii Framework, <http://yiiframework.com/doc/guide/1.1/en/basics.mvc>. Tanggal akses 12 Desember 2012, pukul 21.00 WIB
- [14] Doc.ubuntu.com, <https://help.ubuntu.com/community/Installation/SystemRequirements/FeistyFawn>. Tanggal akses 12 Desember 2012, pukul 21.15 WIB

LAMPIRAN



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
 FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
 PROGRAM PASCASARJANA MAGISTER TEKNIK

BNI (PERSERO) MALANG
 BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
 Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Nomor : ITN-135/T.INF/TA/2012

17 Oktober

Lampiran :
 Perihal :

Bimbingan Skripsi

Kepada : Yth. Sdr. Dr. Aryuanto Soetedjo, ST. MT.
 Dosen Pembimbing Program Studi Teknik Informatika S1
 Institut Teknologi Nasional
 M a l a n g

Dengan hormat

Sesuai dengan permohonan dan persetujuan dalam Proposal Skripsi untuk mahasiswa :

Nama : RIYAN PERMADI
 Nim : 0818243
 Prodi : Teknik Informatika S1
 Fakultas : Teknologi Industri

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya kepada Saudara/i selama masa waktu 6 (enam) bulan, terhitung mulai tanggal :

17 Oktober 2012 – 17 April 2013

Sebagai satu syarat untuk menempuh Ujian Sarjana Teknik, Program Studi Teknik Informatika S1.

Demikian agar maklum dan atas perhatian serta bantuannya kami sampaikan terima kasih.

Mengetahui
 Program Studi Teknik Informatika S1
 Ketua,

 Joseph Dedy Arawan, ST, MT
 NIP. 192304162005021002

Form S-4a



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Nomor : ITN-135/T.INF/TA/2012

17 Oktober

2012

Lampiran : -

Perihal : Bimbingan Skripsi

Kepada : Yth. Sdr. Mi Orisa, ST
Dosen Pembimbing Program Studi Teknik Informatika S1
Institut Teknologi Nasional
Malang

Dengan hormat

Sesuai dengan permohonan dan persetujuan dalam Proposal Skripsi untuk mahasiswa :

Nama : RIYAN PERMADI
Nim : 0818243
Prodi : Teknik Informatika S1
Fakultas : Teknologi Industri

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya kepada Saudara/i selama masa waktu 6 (enam) bulan, terhitung mulai tanggal ;

17 Oktober 2012 – 17 April 2013

Sebagai satu syarat untuk menempuh Ujian Sarjana Teknik, Program Studi Teknik Informatika S1.

Demikian agar maklum dan atas perhatian, serta bantuannya kami sampaikan terima kasih.

Mengetahui
Program Studi Teknik Informatika S1
Ketua,


Joseph Bedy Irawan, ST, MT
NIP. 197404162005021002

Form S-4a




**INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
JL. KARANGLO Km. 2 MALANG**

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : Riyan Permadi
NIM : 0818243
Jurusan : Teknik Informatika S-1
Judul : Sistem Pendukung Keputusan Perencanaan Matakuliah Berbasis Web

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) Pada :
Hari : Senin
Tanggal : 18 Pebruari 2012
Nilai : 86.15 (A)

**Panitia Ujian Skripsi
Ketua Majelis Penguji**


Joseph Dedy Irawan, ST, MT
NIP. 19740416 200501 1 002
Anggota Penguji

Penguji Pertama



Ali Mahmudi, BEng, PhD
NIP.P. 1031000429

Penguji Kedua



Yosep Agus Pranoto, ST
NIP.P. 1031000432



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
PROGRAM STUDI TEKNIK INFORMATIKA S-1
JL. KARANGLO Km. 2 MALANG

FORMULIR PERBAIKAN SKRIPSI

Nama : Riyan Permadi
NIM : 0818243
Jurusan : Teknik Informatika S-1
Judul : Sistem Pendukung Keputusan Perencanaan Matakuliah Berbasis Web

Tanggal	Peng uji	Uraian	Paraf
18 - 02- 2013	I	<ul style="list-style-type: none">• Hal 36, Mengenai Gambar 3.16 salah penomoran• Hal 40, Mengenai Gambar 3.17 tidak berisi• Menambahkan Pengujian CSS tampilan Web dengan Berbagai Browser• Penamaan Nomer Gambar Halaman 43 salah 3.6.1, dan 3.6.2• Perbaiki Kesimpulan dengan kata-kata yang bagus dan daftar pustaka	
18 - 02- 2013	II	<ul style="list-style-type: none">• Pendekatan UML dengan menambah 2 diagram activity dan diagram sequence• Gambar silabus di perbaiki• Perbaiki kata dan format penulisan	

Anggota Penguji

Penguji Pertama

Ali Mahmudi, BEng, PhD
NIP.P. 1031000429

Penguji Kedua

Yosep Agus Pranoto, ST
NIP.P. 1031000432

Mengetahui

Dosen Pembimbing I


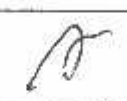

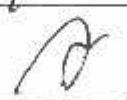

Dr. Eng. Aryuanto Soetedjo, ST, MT
NIP. P. 1030800417

Dosen Pembimbing II

Mira Orisa, ST
NIP. P. 1031000435

FORMULIR BIMBINGAN SKRIPSI

Nama : RIYAN PERMADI
NIM : 0818243
Masa Bimbingan : 17 Oktober 2012-17 April 2013
Judul Skripsi : SISTEM PENDUKUNG KEPUTUSAN PERENCANAAN
MATA KULIAH BERBASIS WEB

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1		Revisi Program Dss	
2		Program Selesai	
3		Revisi Bab I dan II	
4		Revisi Bab II, III dan IV	
5		Revisi Matalah	
6			
7			
8			
9			
10			


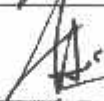

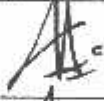

Malang, _____
Dosen Pembimbing,


Dr. Aryanto Soetedjo, ST. MT.
NIP. :

Form S-4b


FORMULIR BIMBINGAN SKRIPSI

Nama : RIYAN PERMADI
NIM : 0818243
Masa Bimbingan : 17 Oktober 2012-17 April 2013
Judul Skripsi : SISTEM PENDUKUNG KEPUTUSAN PERENCANAAN
MATA KULIAH BERBASIS WEB

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
-1		Revisi Program Dss	
2		Program Selesai	
3		Revisi Bab I dan II	
4		Revisi Bab III dan IV	
5		Revisi Makalah	
6			
7			
8			
9			
10			

Malang, _____

Dosen Pembimbing,


Mira Orisa, ST
NIP. : _____

Form S-4b

PENGUJIAN DSS PERENCANAAN MATAKULIAH

Nama : Sugiarto
Nim : 0918001
Angkatan : 2009
Tanggal : 15 Januari 2013

- Menurut anda, apakah sistem pendukung perencanaan ini baik ?
 - Sangat baik
 - Baik
 - Cukup
 - Kurang baik
 - Buruk
- Menurut anda, apakah sistem ini membantu proses pengambilan perencanaan matakuliah ?
 - Sangat membantu
 - Membantu
 - Cukup membantu
 - Kurang membantu
 - Tidak membantu
- Menurut anda, apakah sistem pendukung keputusan perencanaan ini sesuai harapan ?
 - Sangat sesuai
 - Sesuai
 - Cukup sesuai
 - Kurang sesuai
 - Tidak sesuai

Komentar / Saran / Kritik Hasil Pengujian :

ada baiknya untuk bisa diambil di semester 7 bila semua syarat telah terpenuhi

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

PENGUJIAN DSS PERENCANAAN MATAKULIAH

Nama : Mega Putri Darliana

Nim : 1018134

Angkatan : 2010

Tanggal : 16 Januari 2013

1. Menurut anda, apakah sistem pendukung perencanaan ini baik ?
 - a. Sangat baik
 - b. Baik
 - c. Cukup
 - d. Kurang baik
 - e. Buruk
2. Menurut anda, apakah sistem ini membantu proses pengambilan perencanaan matakuliah ?
 - a. Sangat membantu
 - b. Membantu
 - c. Cukup membantu
 - d. Kurang membantu
 - e. Tidak membantu
3. Menurut anda, apakah sistem pendukung keputusan perencanaan ini sesuai harapan ?
 - a. Sangat sesuai
 - b. Sesuai
 - c. Cukup sesuai
 - d. Kurang sesuai
 - e. Tidak sesuai

Komentar / Saran / Kritik Hasil Pengujian :

Dari hasil yang saya coba dari sistem perencanaan Matakuliah ini
membantu proses pengambilan perencanaan matakuliah.
Mungkin masih harus diperbaiki di bagian matakuliah pilihan.

PENGUJIAN DSS PERENCANAAN MATAKULIAH

Nama : Deny Arif Kurniawan
Nim : 108135
Angkatan : 2010
Tanggal : 10 Desember 2012

1. Menurut anda, apakah sistem pendukung perencanaan ini baik ?
 - a. Sangat baik
 - b. Baik
 - c. Cukup
 - d. Kurang baik
 - e. Buruk
2. Menurut anda, apakah sistem ini membantu proses pengambilan perencanaan matakuliah ?
 - a. Sangat membantu
 - b. Membantu
 - c. Cukup membantu
 - d. Kurang membantu
 - e. Tidak membantu
3. Menurut anda, apakah sistem pendukung keputusan perencanaan ini sesuai harapan ?
 - a. Sangat sesuai
 - b. Sesuai
 - c. Cukup sesuai
 - d. Kurang sesuai
 - e. Tidak sesuai

Komentar / Saran / Kritik Hasil Pengujian :

hasil sistem sesuai dg yg dianjurkan @ silabus Universitas,
akan lebih baik jika DSS Perencanaan Matakuliah ini
@integrasikan dg sistem informasi akademik yang sudah
ada @ Universitas.