

**Model Penjadwalan Matakuliah Secara Otomatis
Berbasis Algoritma Depth-First Search (DFS)**

SKRIPSI

Oleh:

MOCHAMMAD HABIB

NIM. 07650039



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

**Model Penjadwalan Matakuliah Secara Otomatis
Berbasis Algoritma Depth-First Search (DFS)**

SKRIPSI

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

Oleh :
MOCHAMMAD HABIB
NIM. 07650039

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

Model Penjadwalan Matakuliah Secara Otomatis Berbasis Algoritma Depth-First Search (DFS)

SKRIPSI

Oleh:

MOCHAMMAD HABIB

NIM. 07650039

Telah Diperiksa dan Disetujui untuk Diuji:

Tanggal : 04 Juli 2014

Dosen Pembimbing I

Dosen Pembimbing II

Fatchurrochman, M.Kom
NIP. 19700731 200501 1 002

A' La Syaugi, M.Kom
NIP. 19771201 200801 1 007

Mengetahui,
Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdiان
NIP. 19740424 200901 1 008

Model Penjadwalan Matakuliah Secara Otomatis Berbasis Algoritma Depth-First Search (DFS)

SKRIPSI

Oleh:

Mochammad Habib
NIM. 07650039

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)

Tanggal, 18 Juli 2014

Susunan Dewan Penguji :	Tanda Tangan
1. Penguji Utama	: Dr. Cahyo Crysdian NIP. 19740424 200901 1 008 ()
2. Ketua Penguji	: M. Amin Hariyadi, M.T NIP. 19670118 200501 1 001 ()
3. Sekretaris Penguji	: Fatchurrochman, M.Kom NIP. 19700731 200501 1 002 ()
4. Anggota Penguji	: A'la Syauqi, M.Kom NIP. 19771201 200801 1 007 ()

Mengesahkan,
Ketua Jurusan Teknik Informatika

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

**SURAT PERNYATAAN
ORISINALITAS PENELITIAN**

Saya yang bertanda tangan di bawah ini:

Nama : Mochammad Habib

NIM : 07650039

Fakultas/ Jurusan : Sains dan Teknologi/ Teknik Informatika

Judul Penelitian : Model Penjadwalan Matakuliah Secara Otomatis Berbasis
Algoritma Depth-First Search (DFS)

Menyatakan dengan sebenar-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur plagiat karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur plagiat, maka saya bersedia untuk mempertanggung jawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 07 Juli 2014

Yang Membuat Pernyataan,

Mochammad Habib

NIM. 07650039

Persembahan

Skripsi ini Saya persembahkan untuk yang selalu menemani setiap langkah, hela nafas dan sujudku. Engkau telah membuatku mengerti tentang makna kehidupan dan indahnya keikhlasan. Engkau Tuhanku Allah SWT tiada satu nikmatpun yang dapat kudustakan. Shalawat serta salam semoga tetap terhaturkan kepada Nabi Muhammad SAW yang telah membawa berita gembira kepada seluruh umat di penjuru dunia.

Buat insan yang selalu menemani setiap langkah kehidupan dan memberikan senyum semangat serta setia dalam segala lara duka perjuangan hingga rasa syukur terucap tiada henti.

Terima kasih buat orang-orang tercinta Bapak & Emak, ini bukti cinta kasih dan doa sepanjang malam dariMu

Sepanjang malam yang kau rangkai menjadi untaian harapan dan tonggak masa depan yang menyejukkan kalbuku.

Buat kakak-kakak (Cak Fin, Cak Aman, Cak Asan, Mbak Im, Cak Nul, Cak Syur, Cak Hanif dan Cak Hadi) terimakasih banyak atas semua motivasi dalam bentuk spirituil, moril, maupun materi serta dukungan untuk mewujudkan cita-cita untuk

mencapai ridha Allah

Tak lupa untuk sahabat terbaik juga teman-teman seperjuangan TI angkatan '07 yang tidak bisa saya sebutkan satu-persatu, terimakasih atas semua kerjasamanya dalam bahu mambahu mencari ilmu.

Para kader militan IMM Revivalis, Pelopor dan Reformer semoga tetap istiqomah dan tegar seperti karang di lautan dalam berjuang.

MOTTO

“Jangan percaya dengan kegagalan, anggaplah semua hanya ketidakberuntungan yang setiap saat & waktu bisa kamu rubah dengan gagasan disertai tindakan untuk menjadi sebuah kesuksesan. Dan apa yang ingin kalian tahu serta ingin capai yaitu mimpi & harapan jangan pernah dikalahkan oleh rasa minder.”



KATA PENGANTAR

Bismillahirrahmanirrahim.....

Assalamualaikum Wr. Wb.

Segala puji bagi Tuhan Semesta Alam Allah Azza Wajalla karena atas rahmat, taufik dan Hidayah-Nya, sehingga penulis dapat menyelesaikan penulisan tugas akhir ini dengan judul “Mesin Penjawab Otomatis Al Hadits Dengan Menggunakan Bayesian”. Shalawat serta salam semoga tetap tercurahkan kepada junjungan kita Nabi Muhammad SAW beserta keluarga dan sahabatnya sampai hari akhir nanti.

Penulis menyadari bahwa banyak pihak yang telah membantu dalam menyelesaikan penulisan tugas akhir ini. Untuk itu, iringan doa’ dan ucapan terima kasih penulis sampaikan kepada:

1. Fatchurrochman, M.Kom selaku dosen pembimbing utama & A’la syauqi, M.Kom selaku pembimbing ke dua. karena atas bimbingan, pengarahan dan kesabaran beliau, penulisan tugas akhir ini dapat terselesaikan
2. Prof. Dr. Mudjia Raharjo, M,Si selaku Rektor Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
3. Dr. drh. Bayyinatul Muchtaromah, M.Si. selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
4. Dr. Cahyo Crysdiyan, selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
5. Segenap Dosen Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang.
6. Teman-teman TI ’07, Laboran dan Staff administrasi Jurusan Teknik Informatika yang banyak membantu selama penelitian.

Semoga Allah memberikan balasan atas jerih payah keringat yang telah diberikan kepada penulis. Akhir kata, penulis berharap skripsi ini bermanfaat dan dapat menjadi inspirasi bagi peneliti lain serta menambah khasanah ilmu pengetahuan. Dengan kerendahan hati penulis mohon maaf apabila ada kesalahan dalam penulisan skripsi ini.

Malang, 25 Juni 2014

Penulis



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
LEMBAR PERSEMBAHAN	iv
HALAMAN PERNYATAAN	v
KATA PENGANTAR	vi
DAFTAR ISI	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR	xii
ABSTRAK	xiv
BAB I PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian.....	5
1.6 Metode Penelitian	5
1.7 Sistematika Penulisan	6
BAB II TINJAUAN PUSTAKA	
2.1 Penjadwalan	9
2.2 <i>Depth-first Search</i>	15
2.2.1 <i>Metode DFS</i>	17
2.3 Cara Kerja DFS.....	19
BAB III ANALISIS DAN PERANCANGAN	
3.1 Tahap Penelitian	23
3.1.1 Identifikasi Masalah	23
3.1.2 Studi Literatur	23
3.1.3 Analisis Data.....	24

3.1.4	Spesifikasi <i>Software</i>	28
3.1.5	Spesifikasi <i>Hardware</i>	28
3.2	Tahap Implementasi.....	29
3.2.1	Pemodelan Data.....	30
3.2.1.1	<i>Depth-first Search</i>	30
3.3	Tahap Perancangan.....	32
3.3.1	Desain Sistem	32
3.3.2	Desain <i>Input</i>	34
3.3.3	Desain <i>Output</i>	36
3.3.4	Desain Proses.....	37
3.3.5	Pemodelan UML	38
3.3.6	Activity Diagram Alir Data.....	38
3.3.7	Desain <i>Database</i>	42
3.3.8	Desain entity relationship diagram.....	46
BAB IV IMPLEMENTASI DAN UJI COBA		
4.1	Deskripsi Program	47
4.2	Implementasi Program.....	57
4.2.1	<i>Depth-first search</i>	57
4.3	Uji Coba Aplikasi	61
4.3.1	<i>Depth-first search</i>	61
4.4	Kajian Penelitian dalam Alquran	65
BAB V PENUTUP		
5.1	Kesimpulan	69
5.2	Saran	69

DAFTAR PUSTAKA	69
LAMPIRAN	70

DAFTAR TABEL

Tabel 2.1 Contoh Jadwal Tabel Matakuliah	12
Tabel 2.2 Tabel Keterangan Nilai Preferensi	14
Tabel 2.3 Tabel Kelas	21
Tabel 2.4 Tabel Reprerentasi Graf dengan DFS	31
Tabel 2.5 Tabel Jadwal Kuliah Dari Proses Graf DFS	31
Tabel 2.6 Tabel Desain Input	34
Tabel 2.7 Tabel Desain Out Put	36
Tabel 2.8 Tabel Desain Proses	37
Tabel 3.1 Tabel Dosen	42
Tabel 3.2 Tabel Matakuliah	42
Tabel 3.3 Tabel Ruang Kuliah	42
Tabel 3.4 Tabel Data Jurusan	43
Tabel 3.5 Tabel Data Pengampu	43
Tabel 3.6 Tabel Plot Mengajar	47
Tabel 3.7 Tabel Hasil Depth-first Search	51
Tabel 3.8 Tabel Bentrok Dosen	55
Tabel 3.9 Tabel Bentrok Ruang Kuliah	57

DAFTAR GAMBAR

Gambar 2.3 Pohon Ruang Status Pembangkitan Permutasi A, B, C	17
Gambar 2.4 Tahapan Pembentukan Pohon DFS	17
Gambar 2.5 Pembentukan Pohon Ruang Status Persoalan Pembangkitan Permutasi A, B, C dengan Algoritma DFS	18
Gambar 2.6 Pencarian Terpendek	20
Gambar 2.7 Alur Graf Pencarian	21
Gambar 3.1 Tahap Implementasi	29
Gambar 3.2 Ilustrasi Graf dengan Depth-first Search	31
Gambar 3.3 Desain Sistem	32
Gambar 3.4 Use Case Diagram	38
Gambar 3.5 Activity Diagram Input Data Dosen	39
Gambar 3.6 Activity Diagram Input Data Matakuliah	39
Gambar 3.7 Activity Diagram Input Data Ruang	40
Gambar 3.8 Activity Diagram Input Data Pengampuan	40
Gambar 3.9 Activity Diagram Input Data Preferensi	41
Gambar 4.0 Activity Diagram Generate	41
Gambar 4.1 Interface Halaman Penjadwalan Matakuliah	45
Gambar 4.2 Interface Form Input Plot Mengajar Dosen	47
Gambar 4.3 Source Code Depth-first Search Penjadwalan	48
Gambar 4.4 Source Code DFS	49
Gambar 4.5 Source Code Bentrok Matakuliah	50
Gambar 4.6 Contoh Data Pengampuan	50
Gambar 4.7 Contoh Data Ruang	51

ABSTRACT

Habib, Mochammad. 2013. Course Scheduling Model Based Algorithm Automatically Depth-First Search (DFS). Informatics Engineering Department, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University of Malang.

Advisors : (I) Fatchurrochman, M.Kom. (II) A'La Syauqi, M.Kom.

Keywords : Scheduling, Course, *Depth-first Search (DFS)* .

Scheduling is a decision-making process is often done in the industry engaged in the service or services (Penedo, 2012). Scheduling issues related to the allocation of resources to the tasks and specific functions. The purpose of scheduling is to menoptimalkan one or several objectives. But during this scheduling is still done manually by accommodating these limitations, but have not considered efisiensi either side of the space and the use of existing facilities and in terms of convenience for faculty and students. It also needs to be considered efficient use of space, the need for proper allocation of space for a course based on space and load capacity of the utility space. In this study, breadth-first search method will be applicable to the selection sesuai parameters for optimal results. And after going through trials, then the resulting class schedule proved the hard constraints are not violated. Although this application only uses sample data on any of the majors in the faculty of science and technology. However, in testing this application there are subjects that have not been scheduled by using the depth-first search. Thus, by tracing the course of the scheduled time slot can be used. Therefore the use of breadth-first search is deemed still not quite optimal.

ABSTRAK

Habib, Mochammad. 2014. **Model Penjadwalan Matakuliah Secara Otomatis Berbasis Algoritma Depth-First Search (DFS)**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing: (I) Fatchurrochman, M.Kom. (II) A'La Syauqi, M.Kom

Kata Kunci: Penjadwalan, Matakuliah, *Depth-first Search (DFS)*.

Penjadwalan merupakan sebuah proses pengambilan keputusan yang sering dilakukan di industri yang bergerak di bidang pelayanan atau jasa (Penedo, 2012). Persoalan penjadwalan berkaitan dengan pengalokasian sumber daya ke dalam tugas-tugas dan fungsi-fungsi tertentu. Tujuan penjadwalan adalah untuk mengoptimalkan satu atau beberapa tujuan. Tapi selama ini penjadwalan yang dilakukan masih bersifat manual, dengan mengakomodasi keterbatasan-keterbatasan tersebut, namun belum mempertimbangkan sisi efisiensi baik dari sisi penggunaan ruang dan fasilitas yang ada maupun dari sisi kenyamanan bagi dosen maupun mahasiswa. Selain itu juga perlu diperhatikan efisiensi penggunaan ruang, yaitu perlunya pengalokasian ruang yang tepat bagi suatu matakuliah berdasarkan kapasitas ruang dan beban utilitas ruang tersebut. Pada penelitian ini metode breadth-first search akan diterapkan pada pemilihan parameter yang sesuai untuk hasil yang optimal. Dan setelah melalui uji coba, maka jadwal kuliah yang dihasilkan terbukti tidak melanggar *hard constraints*. Walaupun aplikasi ini hanya menggunakan sampel data pada satu jurusan saja di fakultas sains & teknologi. Namun, dalam uji coba aplikasi ini masih terdapat matakuliah yang belum terjadwal dengan menggunakan depth-first search. Maka, matakuliah tersebut dijadwalkan dengan menelusuri slot waktu yang bisa digunakan. Oleh karena itu penggunaan algoritma breadth-first search ini dirasa masih belum cukup optimal.

BAB I

PENDAHULUAN

1.1. Latar Belakang

Penjadwalan merupakan sebuah proses pengambilan keputusan yang sering dilakukan di industri yang bergerak di bidang pelayanan atau jasa (Penedo, 2012). Persoalan penjadwalan berkaitan dengan pengalokasian sumber daya ke dalam tugas-tugas atau fungsi-fungsi tertentu. Tujuan penjadwalan adalah untuk mengoptimalkan satu atau beberapa tujuan.

Pembuatan jadwal tidak lepas dengan mengatur waktu. Manajemen waktu yang tepat akan berakibat positif terhadap kegiatan yang akan dilaksanakan. Pembuatan jadwal dengan memanfaatkan waktu yang sudah terjadwal dengan baik maka tidak akan ada kegiatan yang sia-sia. Sesuai dengan firman Allah SWT dalam surat al-‘Ashr ayat 1 – 3, yaitu:

وَالْعَصْرِ ﴿١﴾
 إِنَّ الْإِنْسَانَ لِرَبِّهِ لَكَنُفٍ ﴿٢﴾
 إِلَّا الَّذِينَ ءَامَنُوا وَعَمِلُوا الصَّالِحَاتِ وَتَوَّصَوْا بِالْحَقِّ ﴿٣﴾

وَتَوَّصَوْا بِالصَّبْرِ ﴿٤﴾

“Demi masa. Sesungguhnya manusia itu benar-benar berada dalam kerugian. Kecuali orang-orang yang beriman dan mengerjakan amal sholeh dan nasehat-menasehati supaya mentaati kebenaran dan nasehat-menasehati supaya menepati kesabaran”. (Qs. Al-‘Ashr/103 : 1-3).

Ayat-ayat di atas menunjukkan betapa pentingnya waktu dalam kehidupan manusia ini, karena Allah tidak bersumpah terhadap sesuatu di dalam Al Qur'an kecuali untuk menunjukkan kelebihan yang dimilikinya.

Bahkan dalam ayat lain Allah menegaskan bahwa dengan menggunakan waktu tersebut seorang hamba bisa mengambil pelajaran dan bersyukur, sebagaimana yang tersebut dalam firman Allah SWT :

وَهُوَ الَّذِي جَعَلَ اللَّيْلَ وَالنَّهَارَ خِلْفَةً لِّمَنْ أَرَادَ أَنْ يَذَّكَّرَ أَوْ أَرَادَ شُكُورًا

Dan Dia (pula) yang menjadikan malam dan siang silih berganti bagi orang yang ingin mengambil pelajaran atau orang yang ingin bersyukur. (Al Furqan : 62)

يَأْكُلُ مِمَّا الْأَرْضِ نَبَاتُ بِهِءَ فَأَحْتَلَطَ السَّمَاءِ مِنْ أَنْزَلْتَهُ كَمَا الدُّنْيَا الْحَيَوَةَ مَثَلُ إِنَّمَا

عَلَيْهَا قَدِيرُونَ أَنَّهُمْ أَهْلُهَا وَظَنَ وَأَزَيَّتْ زُخْرُفَهَا الْأَرْضُ أَخَذَتْ إِذَا حَتَّى وَالْأَنْعَامُ النَّاسُ

الْآيَاتِ نَفْصِلُ كَذَلِكَ بِالْأَمْسِ تَغْنَبَ لَمْ كَانَ حَصِيدًا فَجَعَلْنَاهَا نَهَارًا أَوْ لَيْلًا أَمْرُنَا أَتَدَهَا

يَتَفَكَّرُونَ لِقَوْمٍ ﴿٦٤﴾

“Sesungguhnya perumpamaan kehidupan duniawi itu, adalah seperti air (hujan) yang kami turunkan dan langit. Lalu tumbuhlah dengan suburnya karena air itu tanaman-tanaman bumi. Diantaranya ada yang dimakan manusia dan binatang ternak. Hingga apabila bumi itu telah sempurna keindahannya, dan memakai

(pula) perhiasannya, dan pemilik-pemiliknya mengira bahwa mereka pasti menguasainya, tiba-tiba datanglah kepadanya azab kami diwaktu malam atau siang, lalu kami jadikan (tanaman-tanamannya) laksana tanam-tanaman yang sudah disabit, seakan-akan belum pernah tumbuh kemarin. Demikianlah kami menjelaskan tanda-tanda kekuasaan (Kami) kepada orang-orang berfikir". (Qs. Yunus/10: 24).

Berdasarkan Alquran Surat Yunus/10 ayat 24 tersebut, manusia dianjurkan untuk terus berpikir mengenai pemanfaatan apa-apa yang telah Allah ciptakan di bumi. Ayat tersebut juga mengajurkan kepada hamba Allah untuk terus menggali dan memperhatikan apa-apa yang ada di alam semesta ini. berdasarkan ayat tersebut dengan memanfaatkan teknologi informasi yang semakin berkembang akhir-akhir ini, seperti halnya penjadwalan otomatis di Universitas Islam Negeri Maulana Malik Ibrahim Malang dirancang. Semoga aplikasi yang akan dibangun kali ini dapat bermanfaat bagi pihak perguruan tinggi dan bagi pembaca.

Maka dari itu penjadwalan matakuliah telah menjadi salah satu bagian terpenting dalam proses belajar mengajar sebuah universitas. Dalam pembuatan jadwal matakuliah, sering dijumpai berbagai proses persoalan seperti keterbatasan ruang, jumlah ruang yang dapat digunakan untuk menyelenggarakan perkuliahan, keterbatasan kapasitas ruang, dan keterbatasan jam mengajar dosen. Semakin bertambahnya jumlah matakuliah yang akan dijadwalkan, semakin kompleks persoalan penjadwalan yang dihadapi.

Selama ini, penjadwalan yang dilakukan masih bersifat manual, dengan mengakomodasi keterbatasan-keterbatasan tersebut, namun belum memperimbangkan sisi efisiensi baik dari sisi penggunaan ruang dan fasilitas yang ada maupun dari sisi kenyamanan bagi dosen maupun mahasiswa. Selain itu juga perlu diperhatikan efisiensi penggunaan ruang, yaitu perlunya pengalokasian ruang yang tepat bagi suatu matakuliah berdasarkan kapasitas ruang dan beban utilitas ruang tersebut.

Untuk mengatasi masalah penjadwalan matakuliah, sebagai bahan rujukan bahwa beberapa peneliti telah melakukan riset penjadwalan matakuliah (Jamnezhad, M.E, Javidan, R, Dezfouli, M.A., 2011) telah menerapkan algoritma genetic dalam menyusun jadwal kuliah mingguan. (Aldast, M.M., Saheb, M., Najjar, Tamimi, M.H., Takruri, T.O., 2005) telah melakukan penjadwalan matakuliah Palestine Polytechnic University (PPU) menggunakan teknik pemrograman oarallel untuk menangani *multi-objective optimization with soft and hard constraint*.

Penelitian ini didesain sebuah model penjadwalan matakuliah otomatis berbasis algoritma Depth-first search. Depth-first search adalah algoritma yang melakukan pencarian secara menurun yang mengunjungi simpul secara preorder yaitu mengunjungi suatu simpul kemudian mengunjungi semua simpul yang bertetangga dengan simpul-simpul yang tadi dikunjungi, demikian seharusnya dan algoritma DFS menggunakan graf sebagai media representasi persoalan.

Kelebihan dari Depth-first search pada prosesnya tidak akan menemui jalan buntu, menjamin ditemukannya solusi. Jika ada satu solusi, makan breadtg-first

search akan menemukannya, jika ada lebih dari satu solusi, maka solusi minimum akan ditemukan untuk mendapatkan hasil yang komplit dan optimal. Tapi di lain sisi juga memiliki kelemahan yaitu membutuhkan memori yang banyak, karena harus menyimpan semua simpul yang pernah dibangkitkan. Hal ini harus dilakukan agar DFS dapat melakukan penelusuran simpul-simpul sampai di level bawah membutuhkan waktu yang cukup lama. Pada penelitian ini algoritma depth-first search diterapkan pada pemilihan parameter yang sesuai dan hasil yang optimal.

1.2. Rumusan Masalah

Berdasarkan tentang apa yang ditulis pada latar belakang, maka dalam penelitian ini akan membahas mengenai apakah *depth-first search* dapat digunakan untuk menghasilkan jadwal kuliah yang optimal & valid?

1.3. Tujuan

Adapun tujuan dari penelitian ini adalah:

- a. Membuat aplikasi penjadwalan matakuliah otomatis dengan *depth-first search*
- b. Membuat jadwal matakuliah yang tidak melanggar batasan-batasan yang sudah ditentukan.

1.4. Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah:

- a. Data yang digunakan adalah data penjadwalan pada Fakultas Sains & Teknologi semester genap di jurusan Teknik Informatika 2014/2015.

- b. Pada aplikasi penjadwalan ini ini hanya memakai data matakuliah yang memiliki 2 SKS.
- c. Hard-constraints dan soft-constraints sudah ditentukan sebelumnya.

1.5. Manfaat

Manfaat penelitian ini diharapkan dapat memberikan kontribusi dan acuan serta pertimbangan bagi pengelola sistem penjadwalan matakuliah yang optimal.

1.6. Metode Penelitian

Untuk mencapai tujuan yang telah dirumuskan sebelumnya, maka metodologi yang digunakan dalam penulisan skripsi ini adalah:

- a. Pada tahap ini yaitu lokasi penelitian adalah Fakultas Sains & Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
- b. Selanjutnya pengumpulan data seperti database matakuliah.
- c. Study Literatur, analisa ini dilakukan dengan tujuan mencari referensi baik dari sumber buku bacaan atau internet yang berkaitan dengan penelitian ini. Adapun topic yang akan dikaji meliputi Penjadwalan, Breadth-first Search, dan tafsir teknologi dari pandangan Al Qur'an.
- d. Merancang pembuatan aplikasi Model Penjadwalan Matakuliah Secara Otomatis Berbasis Algoritma Breadth-First Search.
- e. Mengimplementasikan rancangan sistem dengan cara membangun serta membuat perangkat lunak aplikasi Model Penjadwalan Matakuliah Secara Otomatis Berbasis Algoritma Breadth-First Search.
- f. Menguji perangkat lunak yang sudah dibuat kemudian menganalisa hasil output apakah sudah sesuai tujuan yang sudah dirumuskan & optimal.

- g. Pada tahap ini yaitu menulis laporan penelitian sebagai dokumentasi dari penelitian yang meliputi penulisan teori penunjang, perancangan-perancangan perangkat lunak, pembuatan perangkat lunak, uji coba perangkat lunak, serta analisis dan kesimpulan.

1.7. Sistematika Penyusunan

Secara garis besar sistematika skripsi ini adalah sebagai berikut:

Bab I Pendahuluan

Bab pertama dari skripsi yang memuat berbagai alasan tentang yang melatar belakangi dilakukan penelitian ini, untuk apa penelitian tersebut dan mengapa penelitian harus dilakukan. Oleh sebab itu, bab ini terdiri atas : latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, metode penelitian, dan sistematika penyusunan skripsi.

Bab II Tinjauan Pustaka

Bab dua berisi argumentasi ilmiah yang digunakan sebagai referensi. Bahan pustaka yang digunakan, diperoleh dari berbagai sumber seperti : jurnal penelitian, laporan penelitian, buku teks, diskusi ilmiah, maupun temuan-temuan hasil pencarian di internet. Berikutnya mengkaji hasil temuan pustaka yang berhubungan dengan konsep-konsep yang dipermasalahkan dan dipakai dalam analisis, termasuk mengkaji tentang teori-teori keislaman yang berkaitan dengan penelitian ini.

Bab III Perancangan Sistem

Bab tiga, berisi tentang analisa dan desain sistem secara terstruktur, yang menjelaskan tentang perancangan sistem yang dilakukan dalam pembuatan sistem,

yang meliputi alur proses jalannya sistem yang dijelaskan dalam bentuk diagram, *use case*, *flowchart*, perancangan antar muka, perancangan proses.

Bab IV Hasil dan Pembahasan

Bab empat membahas tentang implementasi dari aplikasi yang dibuat secara keseluruhan. Serta melakukan pengujian terhadap aplikasi yang dibuat untuk mengetahui apakah aplikasi tersebut telah berjalan sesuai dengan yang diinginkan.

Bab V Penutup

Penutup berisi kesimpulan dan saran dari hasil penelitian. Kesimpulan merupakan pernyataan singkat yang dijabarkan dari hasil penelitian dan pembahasan, untuk membuktikan kebenaran dari hasil temuan pustaka yang diperoleh sekaligus menjawab tujuan penelitian. Sedangkan saran adalah rekomendasi untuk penelitian selanjutnya, yang didasarkan atas pengalaman dan pertimbangan dari hasil penelitian yang dilakukan.

BAB II

KAJIAN PUSTAKA

2.1. Penjadwalan

Penjadwalan merupakan bagian strategis dari proses perencanaan dan pengendalian produksi dan juga merupakan rencana pengaturan urutan kerja serta pengalokasian sumber baik waktu maupun fasilitas untuk setiap operasi yang harus diselesaikan. Menurut Thomas E Morton dan David W Pentico (2001:p12), “Penjadwalan merupakan proses pengorganisasian, pemilihan dan penentuan waktu penggunaan sumber daya yang ada untuk menghasilkan *output* seperti yang diharapkan dalam waktu yang diharapkan pula”.

Menurut kamus besar bahasa Indonesia penjadwalan berasal dari kata ‘jadwal’ yang artinya pembagian waktu berdasarkan rencana pengaturan urutan kerja, daftar atau tabel kegiatan atau rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci. Sedangkan penjadwalan itu sendiri adalah proses, cara dan perbuatan untuk menjadwalkan atau memasukkan ke dalam jadwal.

Proses penjadwalan tidak semudah yang dipikirkan, dalam pembuatan jadwal banyak faktor yang harus dipertimbangkan agar jadwal yang dibuat optimal dan tidak ada bentrok antara jadwal satu dengan jadwal yang lainnya. Masalah yang harus dihadapi dalam pembuatan jadwal adalah menghindari adanya bentrok antar jadwal, dengan data yang digunakan sangat besar dan waktu yang digunakan untuk menyelesaikan jadwal sangat lama apalagi jika pembuatan jika pembuatan jadwal masih belum terkomputerisasi. Hal ini tentunya sangat

tidak menguntungkan bagi petugas pembuat jadwal. Petugas harus mempunyai ketelitian tinggi, pikiran tenang dan tenaga yang besar dalam pembuatan jadwal.

Pengertian penjadwalan dapat berbeda-beda tergantung dari konteks di mana kata tersebut dibicarakan, baik dari segi industri, bisnis, manajemen atau pendidikan. Penjadwalan dalam bidang pendidikan memiliki pengertian secara khusus sebagai durasi waktu dari waktu kerja yang dibutuhkan untuk melakukan serangkaian aktivitas kerja dalam kegiatan belajar mengajar. Penjadwalan juga merupakan proses penyusunan daftar perkuliahan atau daftar kegiatan belajar mengajar yang akan dilakukan untuk mencapai atau mewujudkan suatu tujuan tertentu yang juga menampilkan hasil akhir berupa laporan. Penjadwalan matakuliah dibahas tentang bagaimana mengatur pembagian durasi waktu perkuliahan sesuai dengan aturan yang berlaku di perguruan tinggi, baik pembagian jadwal untuk tiap mahasiswa pada matakuliah tertentu maupun dosen yang mengajar di perguruan tinggi tersebut.

Perguruan tinggi dalam hal ini UIN MALIKI Malang juga bergantung pada jadwal untuk melaksanakan semua kegiatan yang ada. Salah satunya adalah kegiatan perkuliahan. Proses perkuliahan membutuhkan jadwal agar berjalan dengan lancar. Proses pembuatan jadwal perkuliahan harus mempertimbangkan *constraints* yang cukup kompleks agar jadwal tersebut bisa digunakan. Proses pembuatan jadwal juga pasti mempunyai beberapa kendala yang bisa saja menyulitkan petugas untuk membuat jadwal perkuliahan. Permasalahan penjadwalan ini pada umumnya disebut sebagai *University Course Timetabling Problem*.

University Course Timetabling Problem merupakan proses penjadwalan matakuliah pada sebuah perguruan tinggi yang hasilnya tidak melanggar *constraints* yang sudah ditentukan. Batasan-batasan yang harus dipenuhi dalam penjadwalan matakuliah pada setiap perguruan tinggi berbeda-beda. Tidak hanya itu, jumlah matakuliah, dosen, hari dan ruangan juga berbeda, hal ini juga berpengaruh pada proses pembuatan jadwal matakuliah.

Penjadwalan memegang peran yang sangat penting bagi industri manufaktur maupun industri jasa (Pinedo, 2012). Permasalahan penjadwalan merupakan permasalahan kombinatorial yang rumit karena memiliki daerah alternative solusi yang luas dan banyak dijumpai lokak optimal (Yu, 2006). Permasalahan tersebut menjadi salah satu permasalahan kombinatorial yang mendapatkan banyak perhatian dari para peneliti (Amirthagadeswarab, K. S., Arunachalam, V. P., 2006). Beberapa diantaranya membuktikan bahwa permasalahan tersebut bertipe *NP-hard (non deterministic polynomial –time hard)* atau tipe permasalahan yang sulit untuk diselesaikan untuk ukuran yang besar. (Xia, W., Wu, Z., 2006).

Permasalahan mata kuliah menjadi salah satu masalah kompleks yang selalu dihadapi oleh setiap universitas. Penjadwalan mata kuliah merupakan penyusunan dan pengaturan jadwal mata kuliah – mata kuliah pada slot waktu yang tersedia selama satu minggu beserta pembagian ruang kelasnya seperti ditunjukkan pada table di bawah ini.

Tabel 2.1 Contoh jadwal matakuliah

	Kelas A	Kelas B	Kelas C	Kelas D
Senin	MK1		MK 4	MK6
Selasa		MK2	MK3	
Rabu	MK 5			
Kamis	MK8	MK7	MK 10	
Jumat	MK 9	MK 11		MK13

Penjadwalan tersebut harus memperhatikan beberapa hal seperti kapasitas ruang, dosen, mahasiswa, dan jumlah SKS setiap mata kuliah. Masalah penjadwalan meliputi optimasi beberapa kriteria termasuk betasan-batasan seperti kebijakan kurikulum, pemilihan ruang kelas yang sesuai, dan ketersediaan dosen pengajar (Oner,A., Ozcan,S., Dengi., 2011).

Dalam penjadwalan mata kuliah, terdapat dua kategori batasan yaitu batasan yang bersifat harus atau disebut batasan mutlak (*hard constraint*) dan batasan yang bersifat preferensi atau disebut lunak (*soft constraint*) (Oner,A., Ozcan,S., Dengi, D., 2011). Batasan yang bersifat mutlak yang wajib dipenuhi, yaitu:

1. Seorang dosen tidak dapat mengajar lebih dari satu mata kuliah dalam waktu yang sama.
2. Sekelompok mahasiswa (dengan tingkatan/semester yang sama) tidak dapat dijadwalkan mengikuti kuliah lebih dari satu dalam waktu yang sama.

3. Hanya satu mata kuliah yang dapat diselenggarakan di suatu ruang kuliah pada slot waktu yang ada.
4. Jumlah alokasi slot waktu setiap mata kuliah harus sesuai dengan beban SKS mata kuliah.
5. Kapasitas ruang kuliah harus mampu menampung sejumlah mahasiswa yang akan mengambil suatu mata kuliah.
6. Beberapa mata kuliah harus dijadwalkan pada ruang kuliah tertentu seperti laboratorium komputer.

Sedangkan batasan lunak yang bersifat “Preferensi” merupakan batasan yang boleh dilanggar jika perlu, yaitu:

1. Preferensi dosen pengajar mata kuliah dan preferensi mahasiswa seharusnya dipertimbangkan. Preferensi ini dapat dinyatakan dengan bilangan numerik yang merupakan tingkat ketidakpuasan mahasiswa atau dosen.
2. Mata kuliah dengan beban 4 SKS dapat dilaksanakan dalam satu sesi atau dalam dua sesi masing-masing 2 SKS, tergantung preferensi dosen.
3. Dosen sebaiknya memiliki hari libur mengajar.

Jamnez, et al. (2011) menambahkan batasan lunak sebagai berikut:

1. Waktu jeda antar mata kuliah dalam satu hari bagi mahasiswa seharusnya diminimalkan.
2. Ruang kelas untuk mahasiswa semester yang sama sebaiknya menggunakan ruang yang sama.

Berbagai tujuan penyusunan mata kuliah seperti dihasilkannya jadwal yang efisien, jumlah kelas yang berurutan sedikit dari seorang dosen, dan sebagainya menyebabkan persoalan penjadwalan mata kuliah sebagai persoalan optimasi multi-objektif (Aldast, M.M., Saheb, M., Najjar, Tamimi, M.H., Takruri, T.O., 2005). Dalam penyusunan jadwal kuliah, preferensi dosen juga dapat dijadikan salah satu tujuan seperti dicontohkan pada penelitian (Oner, A., Ozcan, S., Dengi, D., 2011) yang menggunakan tabel preferensi dosen seperti ditunjukkan pada tabel di bawah ini. Angka $[0,10]$ pada tabel preferensi dosen seperti ditunjukkan pada tabel ketidakpuasan, artinya semakin tinggi nilai, semakin tidak puas.

Tabel 2.2 Keterangan nilai preferensi

Jam	Ke-	Senin	Selasa	Rabu	Kamis	Jumat
1		10	10	10	10	9
2		9	4	2	8	9
3		2	0	0	0	9
4		2	0	0	0	9
5		1	2	1	1	10
6		1	2	1	1	10
7		7	5	6	3	10
8		7	5	6	3	10

0 : tersedia waktu mengajar, semakin kecil semakin disukai oleh dosen.

10 : tidak tersedia waktu mengajar.

Pada penelitian ini, didesain sebuah model penjadwalan mata kuliah otomatis berbasis algoritma breadth-first search. Breadth-first search adalah algoritma yang melakukan pencarian secara melebar yang mengunjungi simpul secara preorder yaitu mengunjungi suatu simpul kemudian mengunjungi semua simpul yang bertetangga dengan simpul-simpul yang tadi dikunjungi, demikian seharusnya dan algoritma depth-first search menggunakan graf sebagai media representasi persoalan.

Kelebihan dari depth-first search pada prosesnya tidak akan menemui jalan buntu, menjamin ditemukannya solusi. Jika ada satu solusi, maka breadth-first search akan menemukannya, jika ada lebih dari satu solusi, maka solusi minimum akan ditemukan untuk mendapatkan hasil yang komplit dan optimal. Tapi di lain sisi juga memiliki kelemahan yaitu membutuhkan memori yang banyak, karena harus menyimpan semua simpul yang pernah dikunjungi. Hal ini harus dilakukan agar breadth-first dapat melakukan penelusuran simpul-simpul sampai di level bawah membutuhkan waktu yang cukup lama.

Pada penelitian ini metode breadth-first search akan diterapkan pada pemilihan parameter yang sesuai dan untuk hasil yang optimal.

3.1 Depth-First Search

Depth-first Search adalah algoritma yang melakukan pencarian secara menurun yang mengunjungi simpul secara preorder yaitu mengunjungi suatu simpul kemudian mengunjungi semua simpul yang bertetangga dengan simpul

tersebut terlebih dahulu. Selanjutnya, simpul yang belum dikunjungi dan bertetangga dengan simpul-simpul yang tadi dikunjungi, demikian seterusnya. Algoritma depth-first search menggunakan graf sebagai media representasi persoalan. Jika graf berbentuk pohon berakar, Berikut adalah penjelasan mengenai strategi secara umum dari Depth-First Search: maka semua simpul pada aras d dikunjungi lebih dahulu sebelum simpul-simpul pada aras $d+1$.

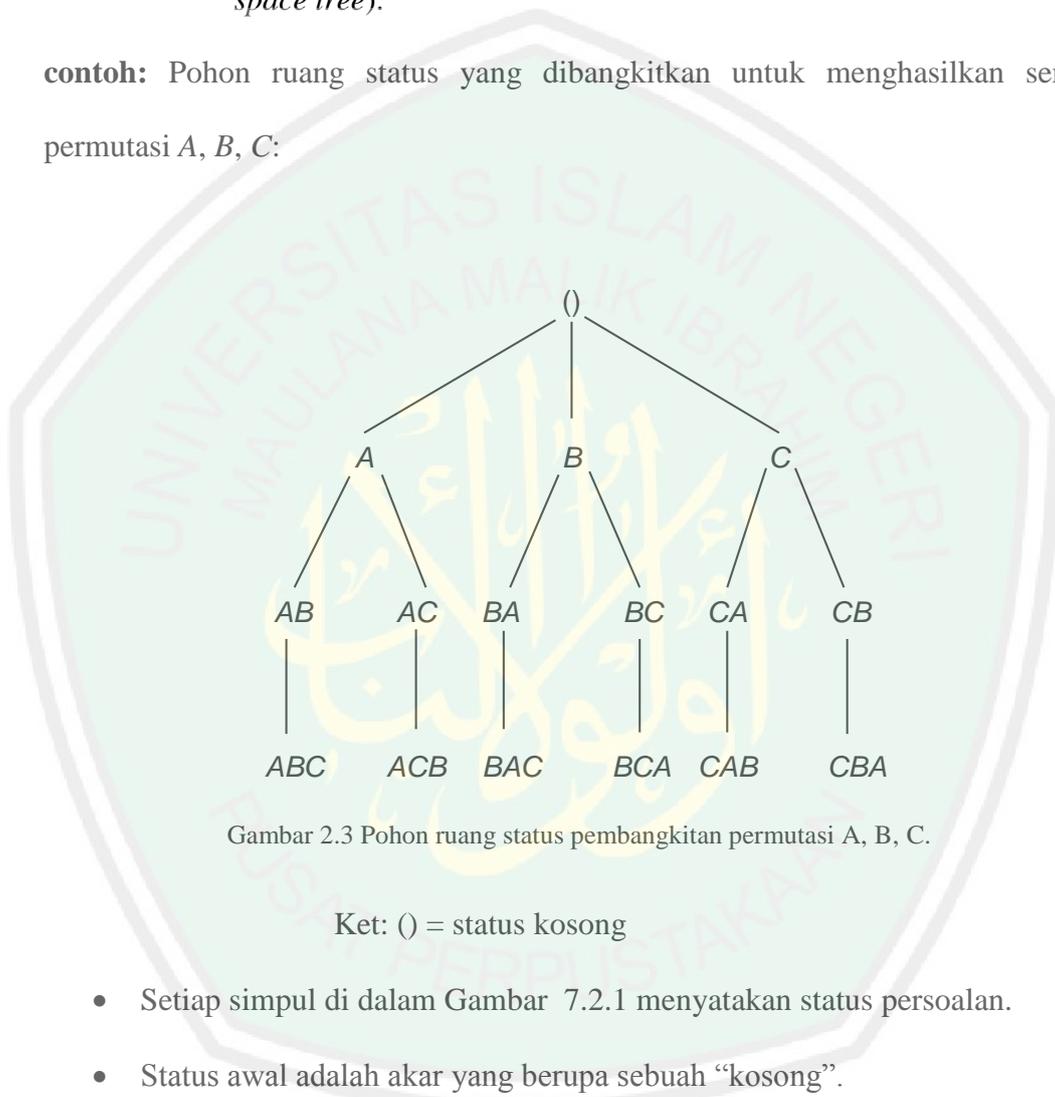
Algoritma ini memerlukan sebuah antrian q untuk menyimpan simpul yang telah dikunjungi. Simpul-simpul ini diperlukan sebagai acuan untuk mengunjungi simpul-simpul yang bertetangga dengannya. Tiap simpul yang telah dikunjungi masuk ke dalam antrian hanya satu kali. Algoritma ini juga membutuhkan table Boolean untuk menyimpan simpul yang telah dikunjungi sehingga tidak ada simpul yang dikunjungi lebih dari satu kali.

Pohon dinamis menyatakan status-status persoalan pada saat pencarian solusi berlangsung.

- **Status persoalan** (*problem state*): simpul-simpul di dalam pohon dinamis yang memenuhi kendala (*constraints*).
- **Status solusi** (*solution state*): satu atau lebih status yang menyatakan solusi persoalan.
- **Status tujuan** (*goal state*): status solusi yang merupakan simpul daun
- **Ruang solusi** (*solution space*): himpunan semua status solusi.

- **Ruang status** (*state space*): Seluruh simpul di dalam pohon dinamis dan pohonnya dinamakan juga **pohon ruang status** (*state space tree*).

contoh: Pohon ruang status yang dibangkitkan untuk menghasilkan semua permutasi A, B, C :

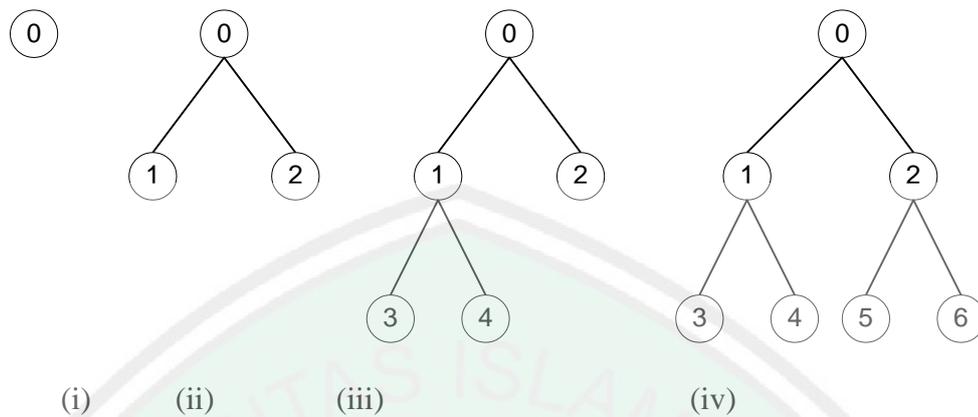


Gambar 2.3 Pohon ruang status pembangkitan permutasi A, B, C .

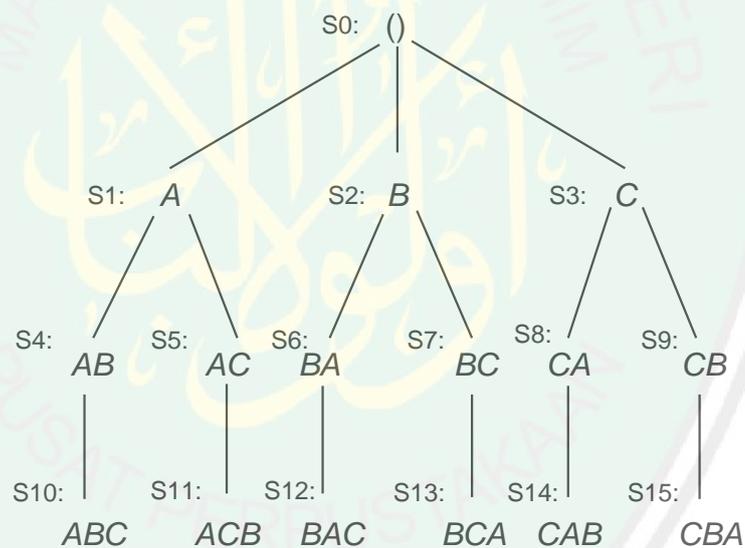
Ket: $()$ = status kosong

- Setiap simpul di dalam Gambar 7.2.1 menyatakan status persoalan.
- Status awal adalah akar yang berupa sebuah “kosong”.
- Setiap daun pada pohon tersebut ($ABC, ACB, BAC, BCA, CAB, CBA$) menyatakan status solusi, dan semua daun adalah ruang solusi.

Metode Pencarian Melebar (BFS)



Gambar 2.4 Tahap pembentukan pohon depth-first search



Gambar 2.5 Pembentukan pohon ruang status persoalan permutasi A, B, C dengan algoritma depth-first search

Pohon ruang status yang dibentuk selama pencarian solusi dengan algoritma DFS

- Dengan mengikuti lintasan dari simpul akar (S0) ke simpul solusi(S10), kita memperoleh konfigurasi urutan perpindahan blok dari status awal sampai ke status akhir.

- Dengan algoritma depth-first search, jika terdapat sebuah solusi, maka depth-first search menjamin dapat menemukannya, dan jika terdapat lebih dari satu buah solusi, depth-first search selalu menemukan solusi pertama pada aras pohon yang paling rendah.

Kompleksitas waktu algoritma depth-first search:

Asumsi: setiap simpul dapat membangkitkan b buah simpul baru.

$$1 + b + b^2 + b^3 + \dots + b^d = (b^{d+1} - 1)/(b - 1) = O(b^d).$$

- Kompleksitas ruang algoritma depth-first search = sama dengan kompleksitas waktunya, karena semua simpul daun dari pohon harus disimpan di dalam memori selama proses pencarian.

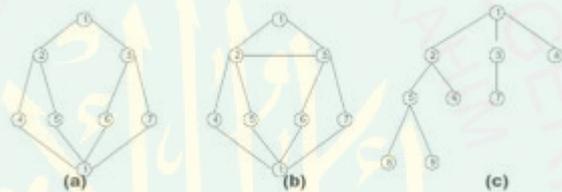
4.1 Cara Kerja Algoritma Depth-First Search.

Dalam algoritma depth-first search, simpul anak yang telah dikunjungi disimpan dalam suatu antrian. Antrian ini digunakan untuk mengacu simpul-simpul yang bertetangga dengannya yang akan dikunjungi kemudian sesuai urutan pengantrian.

Untuk memperjelas cara kerja algoritma depth-first beserta antrian yang digunakannya, berikut langkah-langkah algoritma depth-first:

1. Masukkan simpul ujung (akar) ke dalam antrian.
2. Ambil simpul dari antrian, lalu cek apakah simpul merupakan solusi.
3. Jika simpul merupakan solusi, pencarian selesai dan hasil dikembalikan.

4. Jika simpul bukan solusi, masukkan seluruh simpul yang bertetangga dengan simpul tersebut (simpul anak) ke dalam antrian.
5. Jika antrian kosong dan setiap simpul sudah dicek, pencarian selesai dan mengembalikan hasil solusi tidak ditemukan.
6. Ulangi pencarian dari langkah kedua.



Gambar 2.6 Pencarian lintasan terpendek

Maka penyelesaiannya adalah:

Gambar (a) DFS (1): 1, 2, 3, 4, 5, 6, 7, 1.

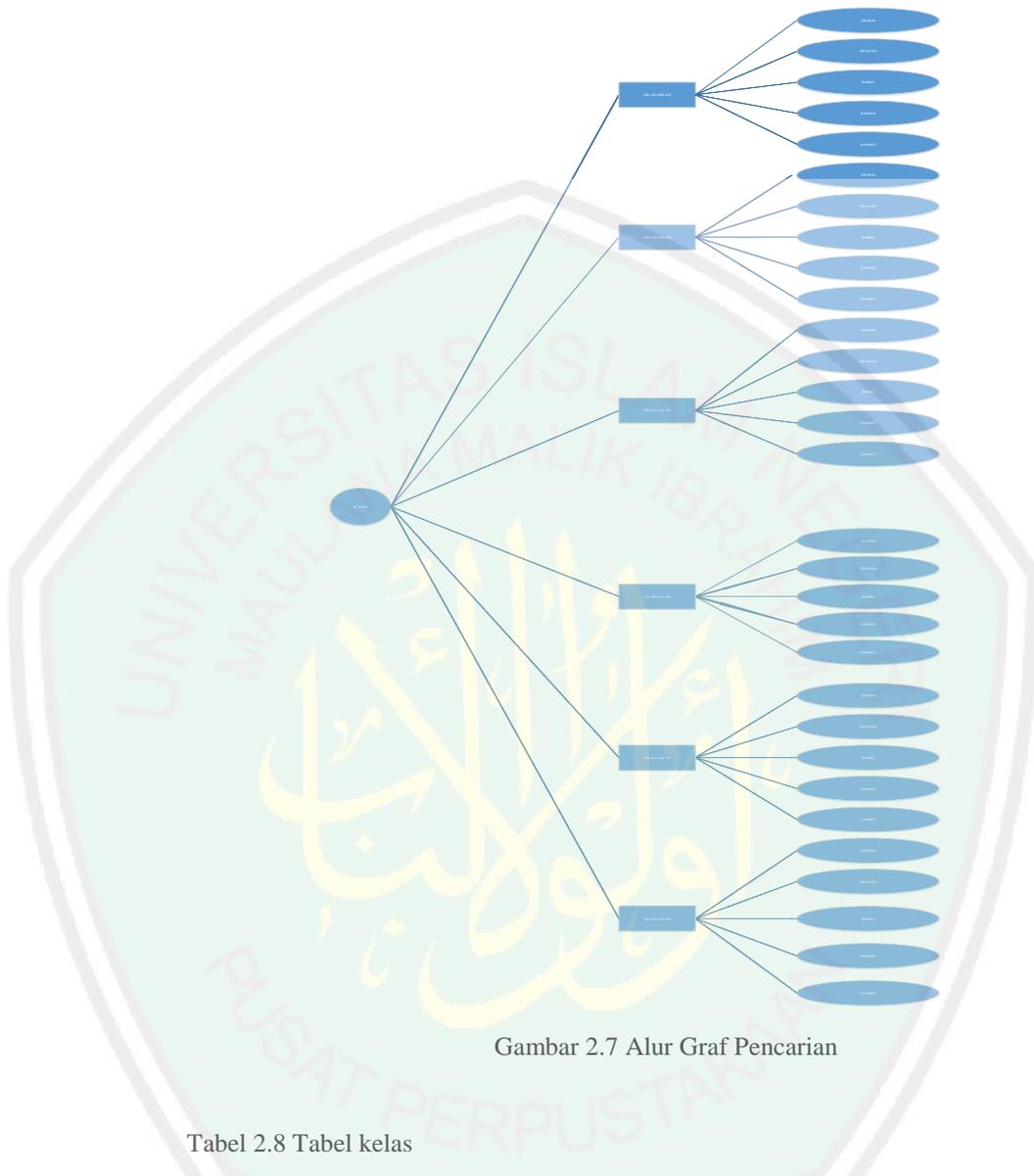
Gambar (b) DFS (1): 1, 2, 3, 4, 5, 6, 7, 1

Gambar (c) DFS (1): 1, 2, 3, 4, 5, 6, 7, 9

Pencarian Lintasan Terpendek Dengan Depth-First Search:

Adapun contoh cara untuk mencari lintasan terpendek dengan menggunakan algoritma depth-first search adalah sebagai berikut:

Diketahui sebuah penjadwalan matakuliah, dengan memiliki inisial seperti yang ditunjukkan dibawah ini yang dibentuk dengan sebuah graf.



Gambar 2.7 Alur Graf Pencarian

Tabel 2.8 Tabel kelas

	Senin	Selasa	Rabu	Kamis	jumat
Jam	06.30	09.30	07.20	11.10	06.30
Jam	07.20	10.20	08.10	13.00	07.10
Jam	08.10	11.10	08.20	14.20	08.20

Contoh pertanyaan: Sebutkan lintasan daun yang akan ditempuh untuk penjadwalan yang tepat dan efisien pada satu matakuliah di ruang B-306, jika matakuliah memiliki sks (2). Ruang dan jam yang tersedia ada pada hari senin di jam 08.10? Titik awal perjalanan adalah RUANG B-306. Gunakan algoritma depth-first search.

Maka dengan menggunakan algoritma depth-first search, lintasan tercepat yang didapat adalah sebagai berikut:

Ruang B-306_Senin_Selasa_Rabu_Kamis_Jumat_06.30_07.20_08.10. Rute tersebut didapat dari pencarian secara melebar. Hal tersebut dapat dijabarkan sebagai berikut:

- Pertama, pointer menunjuk pada daun yang ada di sebelah kanan, yaitu Senin (Ruang B.306 – Senin).
- Setelah proses pertama, kemudian dilanjutkan pada tetangga senin yaitu selasa dan selanjutnya mengarah pada tetangga terdekat, yaitu rabu.
- Pointer mencari tetangga rabu yaitu kamis kemudian tetangga terdekatnya yaitu jumat, namun karena tidak ada, maka pointer kembali ke senin dan masuk ke daun berikutnya, yakni jam 06.30.
- Proses akan diulang hingga pointer menunjuk pada ruang B-306 hari senin dan di jam 08.10.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Tahap Penelitian

3.1.1 Identifikasi Masalah

Penjadwalan merupakan sebuah proses pengambilan keputusan yang sering dilakukan di industri yang bergerak di bidang pelayanan atau jasa (Penedo, 2012). Persoalan penjadwalan berkaitan dengan pengalokasian sumber daya ke dalam tugas-tugas atau fungsi-fungsi tertentu. Tujuan penjadwalan adalah untuk mengoptimalkan satu atau beberapa tujuan.

Pembuatan jadwal yang cukup rumit ditambah dengan data yang harus dibuat jadwalnya sangat besar apalagi jika pembuatan jadwal masih menggunakan cara manual menjadi permasalahan yang dibicarakan pada penelitian kali ini. pembuatan jadwal matakuliah jika dilakukan secara manual bisa saja jadwal yang terbentuk melanggar batasan-batasan yang seharusnya tidak boleh dilanggar, mengingat dalam pembuatannya dibutuhkan ketelitian tinggi dan tenaga yang cukup besar. Oleh karena itu, masalah yang dibicarakan pada penelitian ini adalah bagaimana membuat aplikasi penjadwalan matakuliah otomatis. Jadwal yang terbentuk nantinya akan memenuhi batasan-batasan dalam pembuatan jadwal matakuliah.

3.1.2 Studi Literatur

Pembuatan aplikasi penjadwalan matakuliah otomatis kali ini juga membutuhkan informasi-informasi yang mendukung dalam proses pembuatannya. Informasi-informasi tersebut bisa diambil dari berbagai sumber seperti buku, jurnal, karya ilmiah, internet, narasumber dan lain-lain. Informasi-informasi tersebut seperti proses penjadwalan matakuliah, teori tentang *algoritma breadth-first search & depth-first search*, pemodelan data matakuliah dalam graf, implementasi algoritma, dan informasi-informasi lain yang dianggap perlu dalam proses pembuatan aplikasi ini. pencarian informasi tersebut dilakukan tidak hanya pada awal pembuatan aplikasi ini melainkan juga di tengah-tengah proses pembuatan juga masih dilakukan pencarian informasi mengenai implementasi dari *algoritma depth-first search* untuk pembuatan jadwal matakuliah.

3.1.3 Analisis Data

Pada saat pembuatan jadwal matakuliah terdapat batasan atau *constraint* yang harus dipenuhi atau tidak boleh dilanggar, agar jadwal yang terbentuk dapat digunakan. Batasan-batasan jadwal masing-masing perguruan tinggi pasti berbeda. Apalagi di UIN MALIKI Malang, pada saat pembuatan jadwal terdapat cukup banyak batasan yang harus dipenuhi. Sehingga pada saat proses pembuatannya harus hati-hati dan teliti agar tidak terjadi pelanggaran batasan tersebut. Batasan-batasan tersebut selanjutnya dibagi menjadi dua yaitu *hard constraint* dan *soft constraint*.

Perkuliah di UIN MALIKI Malang dilaksanakan selama 5 hari yaitu hari senin sampai dengan hari jum'at. Pelaksanaan perkuliahan dilaksanakan

mulai pukul 06.30 – 17.20 WIB. Namun, jarak waktu tersebut tidak semua bisa digunakan karena ada batasan-batasan yang harus dipenuhi. Misalkan, untuk matakuliah semester I dan II harus dimulai pada pukul 08.10 WIB dan jarak waktu kuliah pukul 06.30 – 08.10 tidak bisa digunakan karena pada waktu tersebut mahasiswa semester I dan II masih melangsungkan kegiatan Ta'lim di masing-masing Mabna.

Selain itu, untuk mahasiswa semester I dan II juga terdapat perkuliahan PPBA (Program Perkuliahan Bahasa Arab) yang dilaksanakan mulai dari pukul 14.00 – 20.00 WIB. Sehingga jarak waktu antara 14.00 – 17.20 WIB tidak bisa digunakan oleh mahasiswa semester I dan II. Perkuliahan yang dilaksanakan pada hari jum'at juga mempunyai batasan sendiri yaitu perkuliahan tidak boleh dilaksanakan di atas pukul 10.40 WIB dikarenakan terdapat pelaksanaan sholat jum'at. Perkuliahan pada hari jum'at akan dimulai kembali mulai pada pukul 13.10 WIB. Selain aturan mengenai jarak waktu kuliah yang bisa digunakan, aturan-aturan lain yang juga harus dipenuhi adalah tidak ada dosen yang mengajar lebih dari satu kelas pada hari dan jam yang sama. Jika terdapat jadwal yang seperti itu maka terjadi bentrok waktu mengajar dosen maka jadwal tersebut dikatakan tidak valid.

Selain bentrok dosen, bentrok pemakaian ruangan juga harus dihindari. Jadwal yang terbentuk tidak boleh ada 1 ruangan digunakan lebih dari 1 kelas pada saat bersamaan. Jika terjadi hal seperti itu maka jadwal tersebut dikatakan tidak valid. Batasan-batasan dalam pembuatan jadwal matakuliah di atas selanjutnya disebut *hard constraint*. *Hard constraint* adalah batasan-batasan

jadwal matakuliah yang harus dipenuhi, jika tidak terpenuhi atau terjadi pelanggaran maka jadwal tersebut dikatakan tidak valid.

Selain *hard constraint*, batasan lain yang juga perlu untuk dipertimbangkan yaitu *soft constraint*. Batasan ini bersifat tidak wajib untuk dipenuhi. Jadi, jika terdapat pelanggaran *soft constraint* maka matakuliah tersebut masih bisa dikatakan valid. Beberapa contoh dari *soft constraint* seperti tidak ada matakuliah pada saat istirahat sholat dhuhur berjamaah, yaitu pukul 11.30-12.20 WIB. Mahasiswa semester III dan IV diusahakan tidak ada kuliah pada pukul 15.40 WIB dikarenakan pada jam tersebut dilaksanakan PPBI (Program Perkuliahan Bahasa Inggris). Batasan lain adalah preferensi hari dosen. Preferensi hari dosen ini dimaksudkan dosen bisa memilih hari dimana dosen tersebut tidak bisa mengajar. Jadi, jadwal tersebut akan menghindari hari yang ada pada preferensi dosen. Selain itu, jadwal terbentuk diusahakan menghindari dari jadwal mengajar dosen yang berturut-turut tanpa jeda waktu istirahat dalam satu hari sebanyak 9 SKS.

Perkuliahan di UIN MALIKI Malang dilaksanakan selama 5 hari, yaitu hari senin sampai dengan hari jum'at yang dimulai pada pukul 06.30-17.20 WIB. Jarak waktu perkuliahan menggunakan bobot SKS, dimana bobot tersebut akan menentukan lama waktu perkuliahan. Lama waktu perkuliahan untuk 1 SKS adalah 50 menit.

Proses pembuatan jadwal matakuliah diawali dengan proses pemasaran matakuliah. Pada proses ini akan ditentukan matakuliah mana saja yang akan digunakan. Setelah proses pemasaran, selanjutnya adalah pengampunan.

Pada saat proses pengampuan akan ditentukan dosen dan matakuliah yang diampu atau yang akan diajarkan. Misalkan, dosennya adalah Fatchurrochman, M.Kom dan matakuliah yang diampu adalah Pemrograman Berorientasi Obyek. Selain informasi matakuliah dan dosen, pengampuan juga akan menampung informasi semester distribusi dari matakuliah dan jumlah kelas dari matakuliah tersebut.

Adapun *hard constraint* pada pembuatan aplikasi ini adalah

- Tidak ada dosen yang mengajar lebih dari satu matakuliah yang pada hari dan jam yang sama.
- Tidak ada ruangan yang digunakan oleh lebih dari satu matakuliah pada hari dan jam yang sama.
- Matakuliah untuk mahasiswa semester I dan II dimulai pada pukul $\geq 08.10WIB$.
- Matakuliah untuk mahasiswa semester I dan II tidak boleh dilaksanakan pada pukul $> 14.00 WIB$.
- Matakuliah yang dilaksanakan pada hari jum'at tidak boleh dilaksanakan pada rentang waktu $> 10.40 WIB$ dan $< 13.10 WIB$.

Adapun *soft constraint* pada pembuatan aplikasi ini adalah

- Tidak ada matakuliah yang dilaksanakan pada saat istirahat sholat dhuhur yaitu antara 11.30 – 12.20 WIB.
- Matakuliah untuk mahasiswa semester III dan IV tidak boleh terlaksana pada pukul $> 15.40 WIB$.
- Preferensi dosen

- Tidak ada dosen yang mengajar 9 SKS berturut-turut tanpa jeda istirahat selama satu hari.

3.1.4 Spesifikasi Software

Pada saat pembuatan aplikasi penjadwalan matakuliah otomatis kali ini membutuhkan beberapa *software* yang mendukung pembuatan aplikasi ini. Aplikasi ini dibuat dengan menggunakan bahasa pemrograman Java dan Microsoft Access sebagai database penyimpanannya. Adapun *software-software* yang mendukung tersebut sebagai berikut:

- NetBeans IDE 7.1
- Gel
- Microsoft Office 2007

3.1.5 Spesifikasi Hardware

Proses pembuatan dan pengujian aplikasi ini membutuhkan sebuah laptop dengan spesifikasi yang saya gunakan sebagai berikut:

- Processor core Intel i3.
- RAM 2 Gb.
- HDD 500 Gb.
- VGA Intel HD Graphics
- Mouse

3.2 Tahap Implementasi

Pada tahap implementasi berisi tahapan atau cara yang digunakan untuk mengimplementasikan algoritma yang digunakan dalam proses pembuatan aplikasi penjadwalan matakuliah. Adapun tahapan-tahapan tersebut sebagai berikut:



Gambar3.1 TahapImplementasi

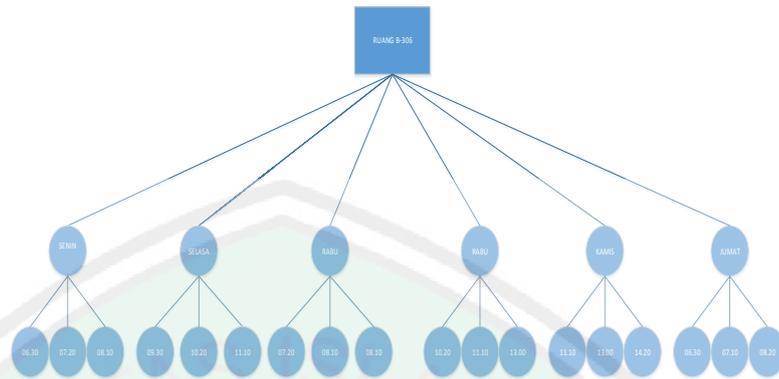
Pembuatan aplikasi penjadwalan matakuliah otomatis ini akan menggunakan *algoritma depth-first search*. Pada saat implementasinya masing-masing algoritma tersebut mempunyai tugas tersendiri pada saat pembuatan jadwal matakuliah. Tetapi, kedua algoritma tersebut saling mendukung untuk pembuatan jadwal kuliah.

Depth-first search akan difokuskan untuk menyelesaikan pemenuhan *hard constraints*. Metode ini akan memodelkan data penjadwalan ke dalam bentuk graf. Graf tersebut akan berisi simpul dengan sisi yang menghubungkan sepasang simpul dan warna dari tiap simpul. Pewarnaan tiap simpul akan melihat sifat ketetanggaan tiap-tiap simpul. Jika sepasang simpul tersebut bertetangga maka kedua simpul tersebut harus mempunyai warna yang berbeda.

3.2.1 Pemodelan Data

3.2.1.1 *Depth-first Search*

Dalam bentuk *depth-first search* kita buat graf yang terdiri dari simpul dan sisi, dimana sepasang simpul dikatakan bertetangga jika terdapat sisi yang menghubungkan simpul tersebut. Pada pengimplementasian *depth-first search* dalam sebuah graf untuk penjadwalan matakuliah adalah dengan memodelkan data penjadwalan ke dalam simpul dan sisi.

Gambar 3.2 Ilustrasi dengan graf *depth-first search*Tabel 3.3 Representasi dengan graf *depth-first search*

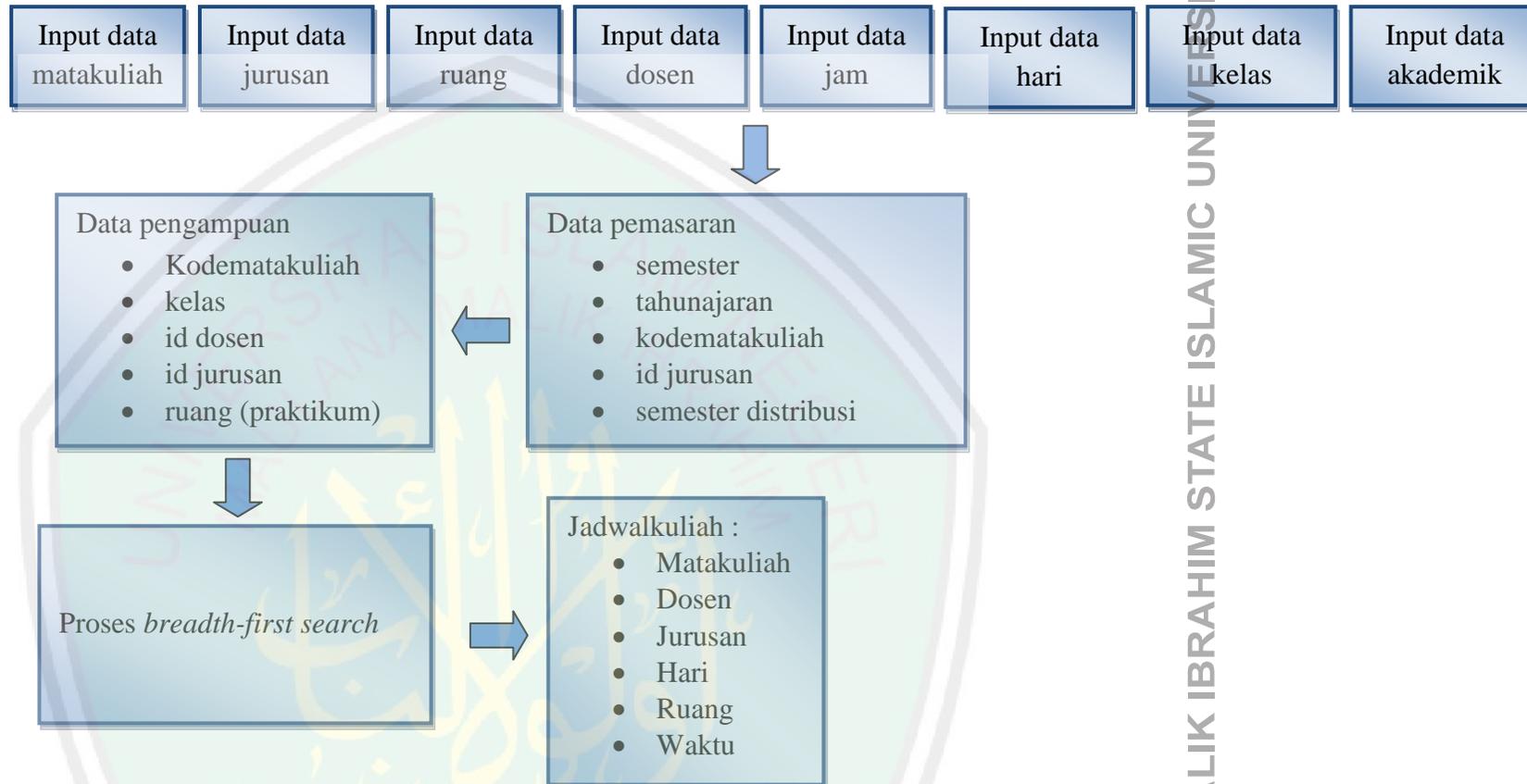
simpul	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12
v1	0	1	1	1	1	0	0	0	0	0	0	0
v2	1	0	0	0	1	1	0	0	0	0	0	0
v3	1	0	0	1	0	0	1	1	0	0	0	0
v4	1	0	1	0	1	0	0	1	0	0	0	0
v5	1	1	0	1	0	1	0	1	1	1	0	0
v6	0	1	0	0	1	0	0	0	0	1	0	0
v7	0	0	1	0	0	0	0	1	0	0	1	0
v8	0	0	1	1	1	0	1	0	1	0	1	0
v9	0	0	0	0	1	0	0	1	0	1	1	1
v10	0	0	0	0	1	1	0	0	1	0	0	0
v11	0	0	0	0	0	0	1	1	1	0	0	0
v12	0	0	0	0	0	0	0	0	1	0	0	0

Tabel 3.4 Jadwal kuliah dari proses graf *depth-first search*

id	KodeMatak <ul style="list-style-type: none">uliah	Id <ul style="list-style-type: none"> Dosen	Kelas	hari	Ruang	Waktu
1	0765211	65101	A	Rabu	B306	06.30 – 08.10
2	0765211	65101	B	Jum'at	B306	06.30 – 08.10
3	0765211	65101	C	Kamis	B306	06.30 – 08.10
4	0765211	65101	D	Senin	B306	06.30 – 08.10
5	0765211	65101	E	Selasa	B306	06.30 – 08.10
6	0765308	65139	A	Rabu	B306	08.10 – 10.40
7	0765308	65139	B	Kamis	B306	08.10 – 10.40
8	0765308	65139	C	Senin	B306	08.10 – 10.40
9	0765308	65139	D	Selasa	B306	08.10 – 10.40

3.3 Tahap Perancangan

3.3.1 Desain Sistem



Gambar 3.5 Desain sistem

Data yang digunakan dalam proses penjadwalan adalah data pengampuan. Data pengampuan berisi kode dari data master, antara lain kode mata kuliah, kode dosen, kode jurusan, kode kategori mata kuliah, kelas. Selain data pengampuan, data hari, data ruang dan data waktu juga digunakan dalam proses penjadwalan yang diambil kodenya saja.

Setelah data penjadwalan tersebut dimasukkan ke dalam *database*, maka proses selanjutnya yaitu *generate*. Langkah awal proses ini adalah membuat jadwal dengan *depth-first search*. Jadwal yang dihasilkandari proses ini adalah jadwal kuliah yang tidak melanggar *hard-constraints*. Pembentukan solusi baru dibentuk dengan modifikasi solusi awal dengan menelusuri slot yang kosong. Solusi baru akan terpilih jika bentroknya lebih kecil atau sama dengan bentrok dari solusi awal/saat ini, jika solusi baru mempunyai bentrok lebih besar dari solusi awal, solusi baru bisa terpilih menjadi solusi terbaik (*best so far*) dengan perhitungan probabilitas. Perhitungan probabilitas ini tergantung temperature pada saat itu. Hasil akhir dari aplikasi ini yaitu jadwal kuliah yang dapat dilihat dengan memilih masing-masing jurusan. Tampilan awal *output* dari aplikasi ini berbentuk *table* yang berisi jadwauliah.

3.3.2 Desain Input

Masukkan atau input merupakan awal dimulainya proses informasi. Bahan mentah dari informasi adalah data yang terjadi dari transaksi yang dilakukan oleh Organisasi, Dinas atau Instansi.

Tabel 3.6 Desain input

Nama input	Alat untuk meng-entry data	Bentuk input	Yang menyediakan data	Yang mengentry data	Periode input	Deskripsi input	Data informasi yang di-entrykan
Input dosen	Keyboard, mouse	Data	Jurusan	Administrator	Fleksibel	Berisikan data dosen	Id_dosen, NIP, Nama, Alamat, No.Telpon, id jurusan
Input matakuliah	Keyboard, mouse	Data	Jurusan	Administrator	Fleksibel	Berisikan data matakuliah	Kode_matakuliah, nama, sks, id jurusan, id kategorimatakuliah
Input jam	Keyboard, mouse	Data	Jurusan	Administrator	Fleksibel	Berisikan rentang waktu kuliah	Id_waktu, jam mulai, jam akhir
Input ruang	Keyboard, mouse	Data	Jurusan	Administrator	Fleksibel	Berisikan data ruang kuliah	Id ruang, nama ruang, id jurusan
Input hari	Keyboard, mouse	Data	Jurusan	Administrator	Fleksibel	Berisikan data hari kuliah	Id hari, namahari

Input pengampuan	Keyboard, mouse	Data	Jurusan	Administrator	Fleksibel	Berisikan data pengampuan	Kodematakuliah, id dosen, semester distribusi, id semester, kelas, id jurusan, tahunajaran
Input preferensidosen	Keyboard, mouse	Data	Dosen	Administrator	Fleksibel	Berisikan data preferensiharidosen	Id dosen, id hari, id jurusan
Input kelas	Keyboard, mouse	Data	Jurusan	Administrator	Fleksibel	Berisikan data namakelas	Id kelas, namakelas
Input pemasaran	Keyboard, mouse	Data	Jurusan	Administrator	Fleksibel	Berisikan data matakuliah yang akandipasarkan	Kodematakuliah, id semester, tahunajaran, semester distribusi

3.3.3 Desain Output

Desain output dimaksudkan untuk menentukan kebutuhan output dari sistem yang dirancang dari proses pembentukannya.

Tabel 3.7 Desain output

Nama laporan	Bentuk laporan	Periodela poran	Alat untuk menampilkan laporan	Pembuat laporan	Penerima laporan	Data/inform asi yang ditampilkan	Deskripsi laporan
Jadwal matakuliah	Table	6 bulansekali	Monitor	Administrator	Administrator, jurusan	Jadwal matakuliah tiap semester	Berisikan data jadwalmatakuliah
Print out	Print out	Fleksibel	Kertas	Administrator	Administrator, jurusan	Jadwal matakuliah tiap semester	Berisikan data jadwalmatakuliah

3.3.4 Desain Proses

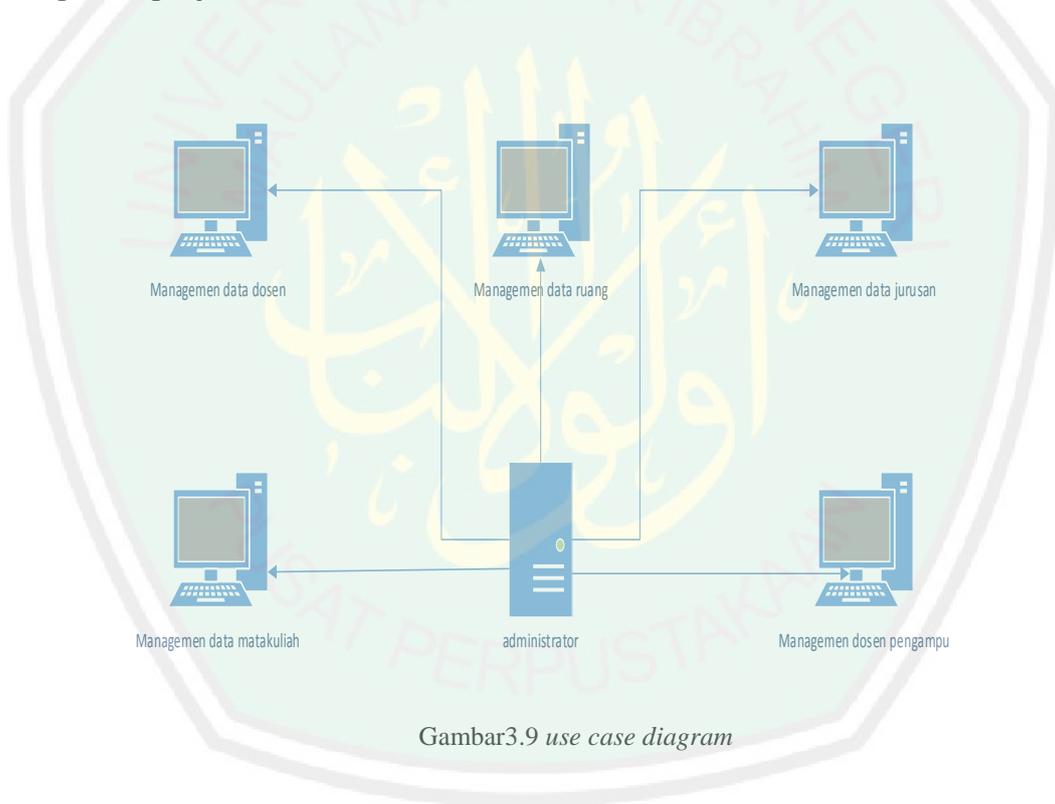
Tabel 3.8 Desain proses

Nama proses	Deskripsi proses	Proses	Output proses
Input	Memasukkan data keperluanpenjadwalan	Memasukkan data	Database
View	Menampilkan data keperluanpenjadwalan	Menampilkan data	Table yang berisi data yang akanditampilkan
Update	Mengubah data keperluanpenjadwalan	Mengubah data	Perubahan data pada database
Search	Mencari data keperluanpenjadwalan	Mencaria data	Pencarian data
Cetak	Mencetakjadwalkuliah	Mencetakjadwulkuliah	Print out jadwalkuliah
Penjadwalan	Proses penjadwalanmatakuliah	Membuatjadwalotomatis (generate)	Jadwalmatakuliah

3.3.5 Pemodelan UML

1. Use case diagram

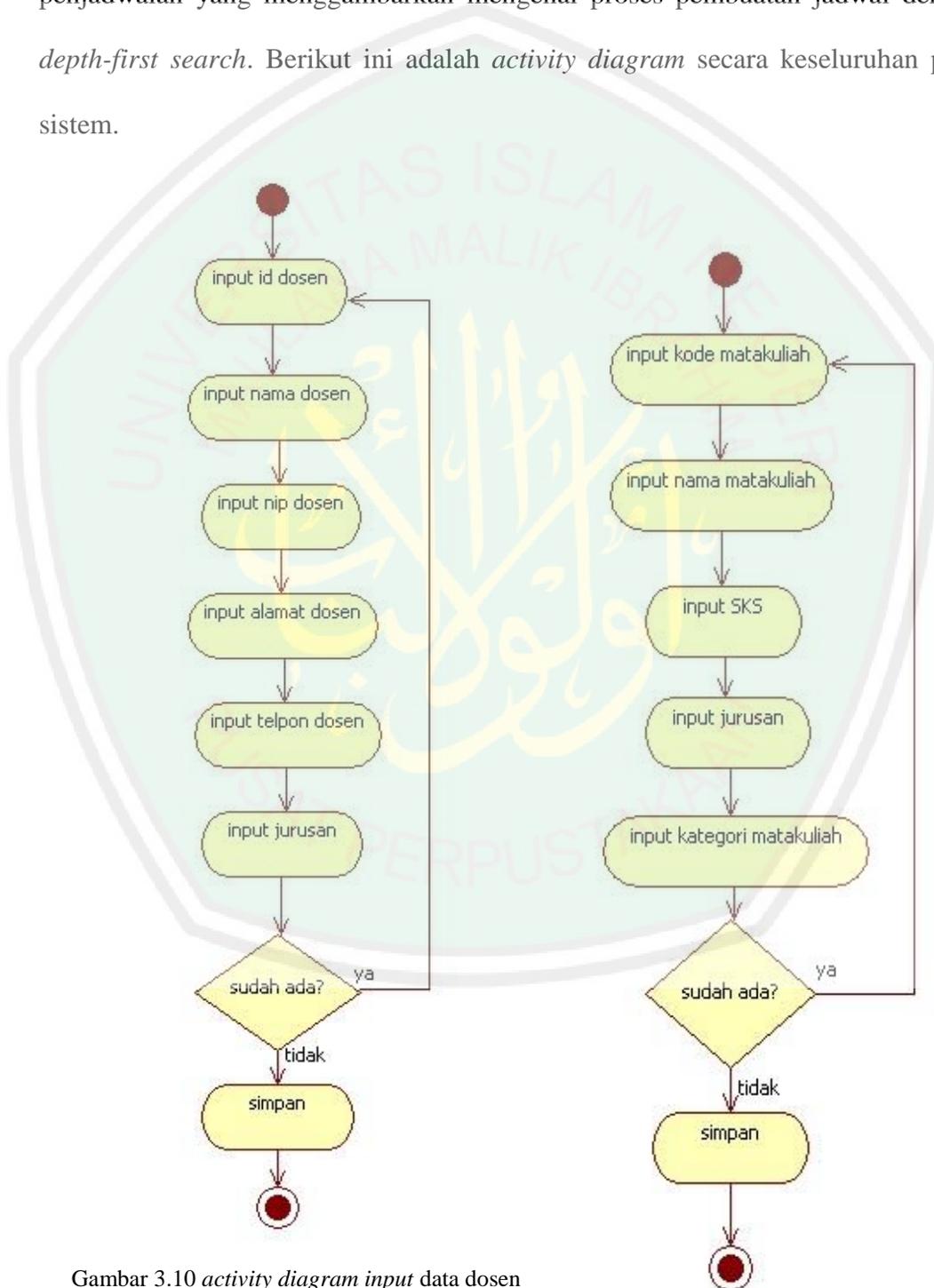
Pemodelan sistem menggunakan use case diagram untuk mengetahui gambaran mengenai fungsi / proses yang terjadi pada sistem. Pada aplikasi ini, *actor* dalam sistem adalah administrator, administrator akan memegang kendali penuh pada aplikasi ini. Berikut ini adalah gambaran umum dari *use case diagram* aplikasi penjadwalan.



2. Activity diagram

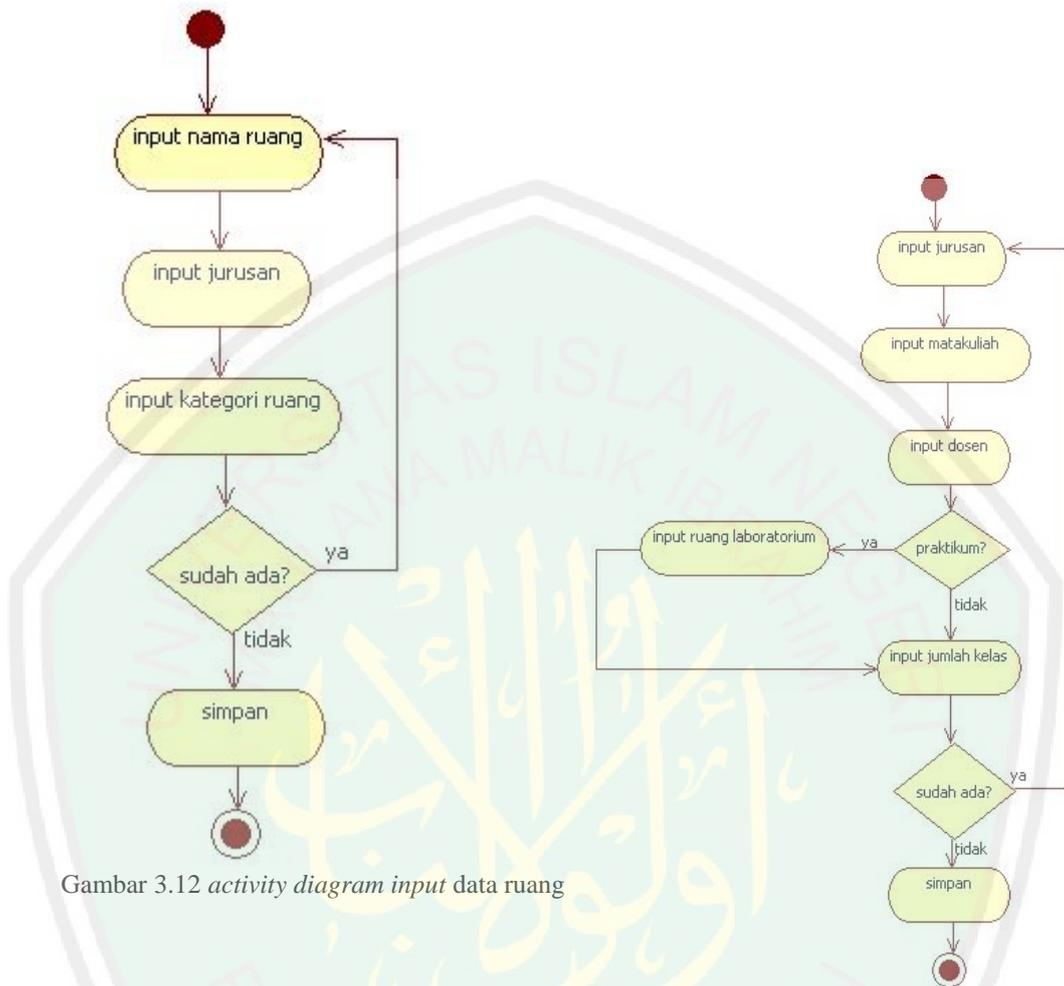
Penggambaran / pemodelan alur kerja dari sebuah proses sistem dan sebuah urutan aktifitas kerja dari sebuah proses akan digambarkan dari sebuah *activity diagram*. Aktifitas yang digambarkan mulai dari input data master seperti data matakuliah, dosen, hari, ruang, waktu, preferensi, data akademik. Kemudian

dilanjutkan dengan proses pengisian data penjadwalan seperti data pemasaran dan data pengampunan. Setelah semua selesai terdapat aktifitas *generate* atau proses penjadwalan yang menggambarkan mengenai proses pembuatan jadwal dengan *depth-first search*. Berikut ini adalah *activity diagram* secara keseluruhan pada sistem.



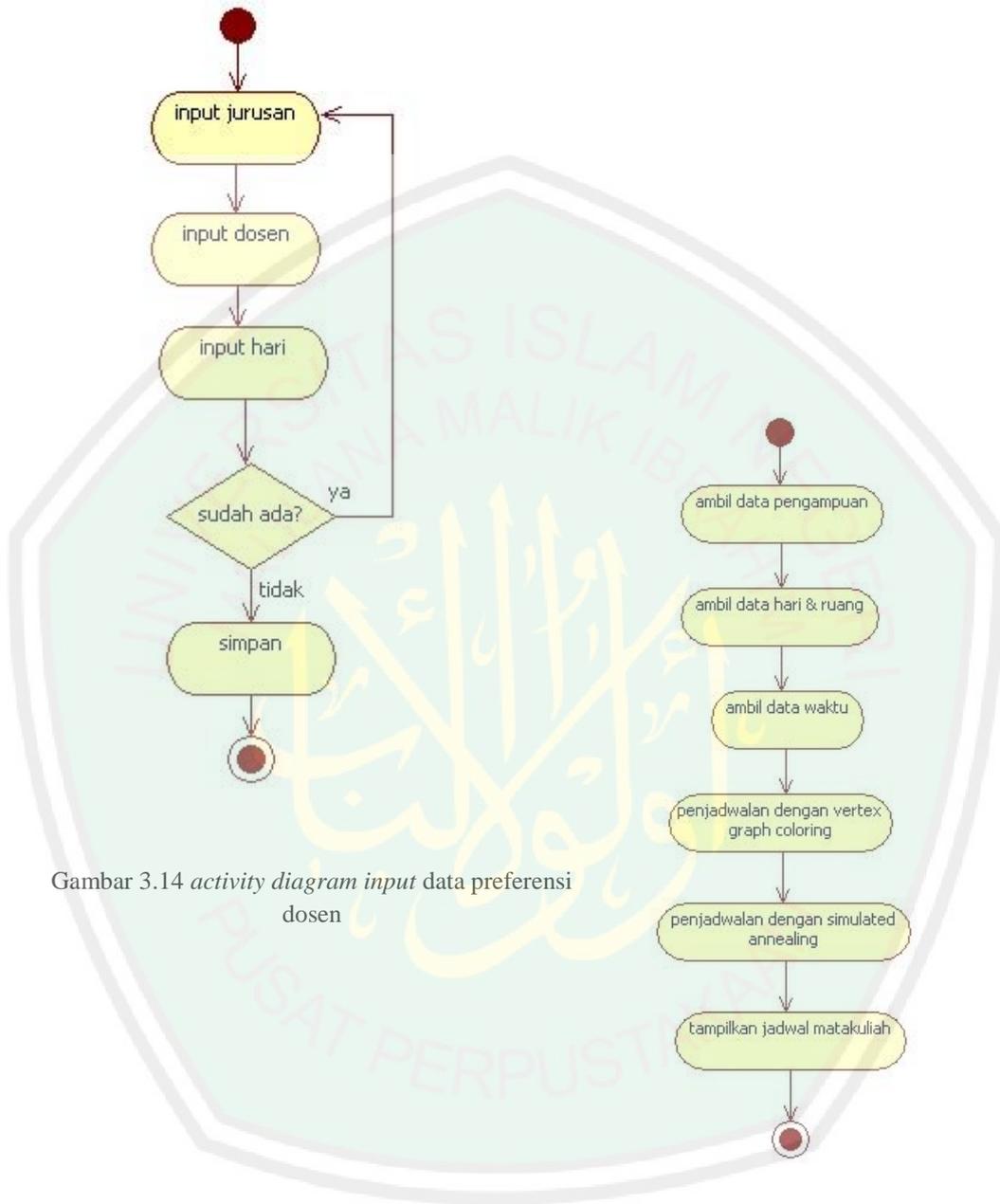
Gambar 3.10 activity diagram input data dosen

Gambar 3.11 activity diagram input data matakuliah



Gambar 3.12 *activity diagram input data ruang*

Gambar 3.13 *activity diagram input data pengampuan*



Gambar 3.14 *activity diagram* input data preferensi dosen

Gambar 3.15 *activity diagram* generate jadwal kuliah

3.3.7 Desain Database

Pada tahap ini akan menggambarkan sebuah *Entity Relationship Diagram* (ERD) yang menggambarkan hubungan atau relasi antar *table*. ERD ditampilkan dalam bentuk *physical model*. Selain itu, bagian akhir tahap ini terdapat *desaintable* dari database yang digunakan. Terdapat beberapa *table* yang digunakan yang ditunjukkan beserta tipe data masing-masing *field*. Selain itu, juga terdapat *primary key* dan *foreign key* pada masing-masing *table*.

1. Desaintable database

Tabel 3.16 Table dosen

No	Namakolom	Tipe data	Keterangan
1	Id_dosen	Char (30)	Primary key
2	NIP_dosen	Char (30)	
3	Nama_dosen	Varchar (100)	
4	Alamat_dosen	Varchar (100)	
5	No_telpon_dosen	Char (30)	
6	Id_jurusan	Integer	Foreign key

Tabel 3.17 Table mata kuliah

No	Namakolom	Tipe data	Keterangan
1	Kode_matakuliah	Char (30)	Primary key
2	Nama_matakuliah	Varchar (100)	
3	SKS	Integer	
4	Kategori_matakuliah	Integer	
5	Id_jurusan	Integer	Foreign key

Tabel 3.18 Table ruang kuliah

No	Namakolom	Tipe data	Keterangan
1	Id_ruang	Integer	Primary key
2	Nama_ruang	Varchar (100)	
3	Id_jurusan	Integer	Foreign key

Tabel 3.19 Tabel hari kuliah

No	Namakolom	Tipe data	Keterangan
1	Id_hari	Integer	Primary key
2	Nama_hari	Varchar (100)	

Tabel 3.20 Tabel data jurusan

No	Namakolom	Tipe data	Keterangan
1	Id_jurusan	Integer	Primary key
2	Nama_jurusan	Varchar (50)	

Tabel 3.21 Tabel data pengampu

No	Namakolom	Tipe data	Keterangan
1	Id	Integer	Primary key
2	Id_semester	Integer	Foreign key
3	Tahun_ajaran	Char (30)	
4	Kode_matakuliah	Char (30)	Foreign key
5	Id_jurusan	Integer	Foreign key
6	Id_dosen	Char (30)	Foreign key
7	Kelas	Char (10)	
8	Semester_distribusi	Char (10)	

Tabel 3.22 Tabel waktu kuliah

No	Namakolom	Tipe data	Keterangan
1	Id_waktu	Integer	Primary key
2	Waktu_mulai	Char (25)	
3	Waktu_akhir	Char (25)	

Tabel 3.23 Tabel kelas kuliah

No	Namakolom	Tipe data	Keterangan
1	Id_kelas	Integer	Primary key
2	Nama_kelas	Char (25)	

Tabel 3.24 Tabel data semester (gasal/genap)

No	Namakolom	Tipe data	Keterangan
1	Id_semester	Integer	Primary key
2	Semester	Varchar (50)	

Tabel 3.25 Tabel transfer dosen

No	Namakolom	Tipe data	Keterangan
1	Id_dosen	Char (30)	Foreign key
2	Id_jurusan	Integer	Foreign key

Tabel 3.26 Tabel transfer matakuliah

No	Namakolom	Tipe data	Keterangan
1	Id_matakuliah	Char (30)	Foreign key
2	Id_jurusan	Integer	Foreign key

Tabel 3.27 Tabel pemasaran

No	Namakolom	Tipe data	Keterangan
1	Id_semester	Integer	Foreign key
2	Tahun_ajaran	Char (30)	
3	Kode_matakuliah	Char (30)	Foreign key
4	Id_jurusan	Integer	Foreign key

Tabel 3.28 Tabel Jadwal Waktu kuliah

No	Nama kolom	Tipe data	Keterangan
1	Id_pengampuan	Integer	Foreign key
2	Id_waktu	Integer	Foreign key

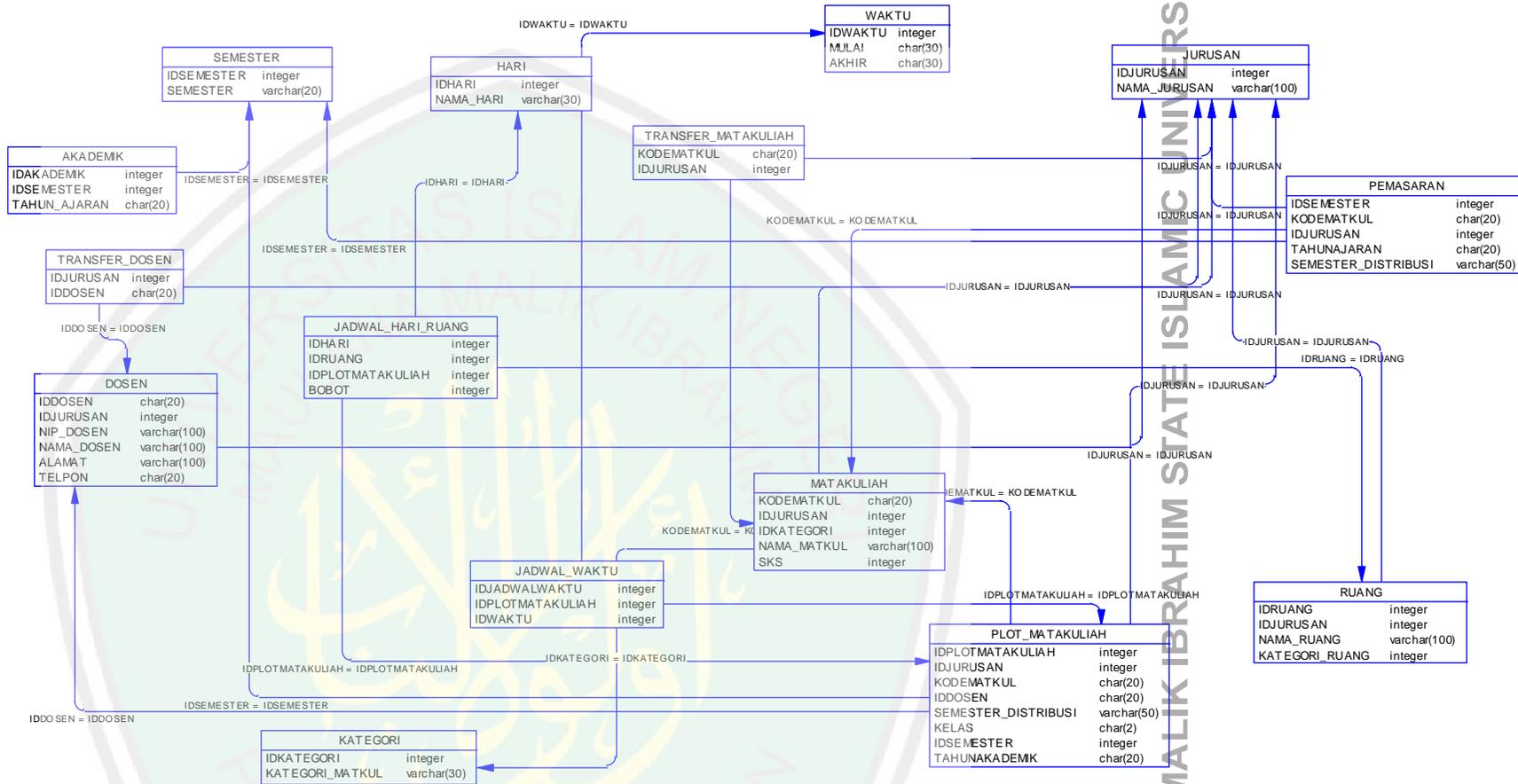
Tabel 3.29 Tabel Jadwal Hari & Ruang

No	Nama kolom	Tipe data	Keterangan
1	Id_pengampuan	Integer	Foreign key
2	Id_hari	Integer	Foreign key
3	Id_ruang	Integer	Foreign key
4	Penalty	Char (10)	

Tabel 3.30 Tabel Akademik

No	Namakolom	Tipe data	Keterangan
1	Id	Integer	Primary key
2	Id semester	Integer	Foreign key
3	Tahun_ajaran	Varchar (20)	
4	Aktif	Integer	

2. Desain entity relationship diagram



Gambar3.21 Desain Entity Relationship Diagram

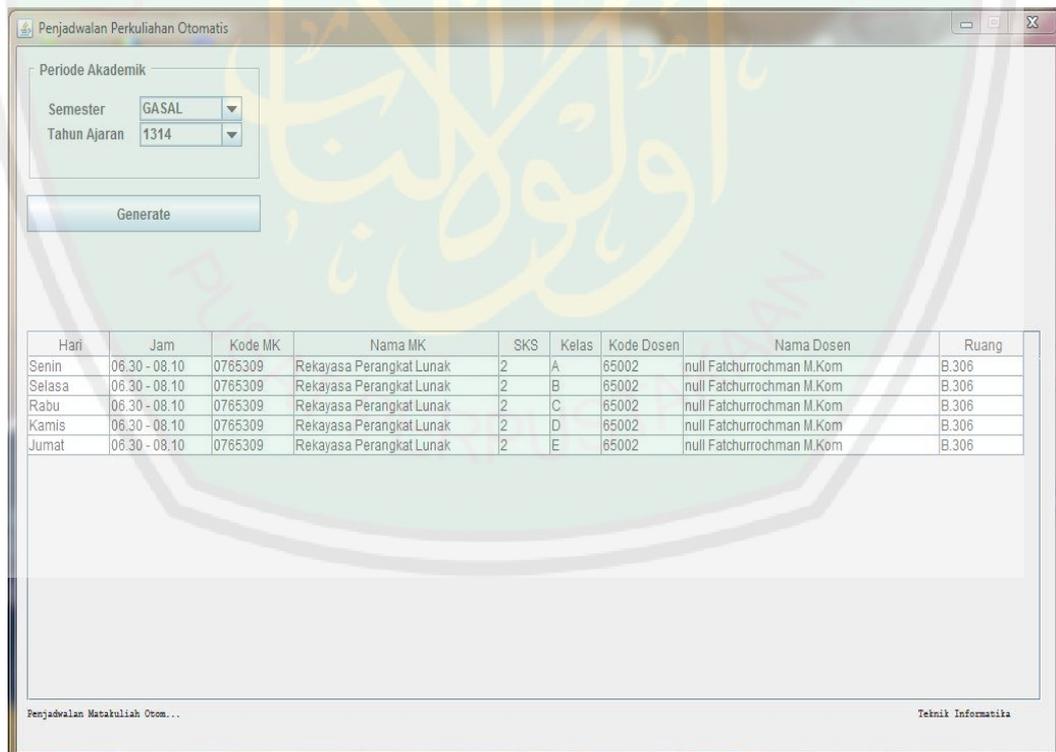


BAB IV

IMPLEMENTASI DAN UJI COBA

4.1 Deskripsi Program

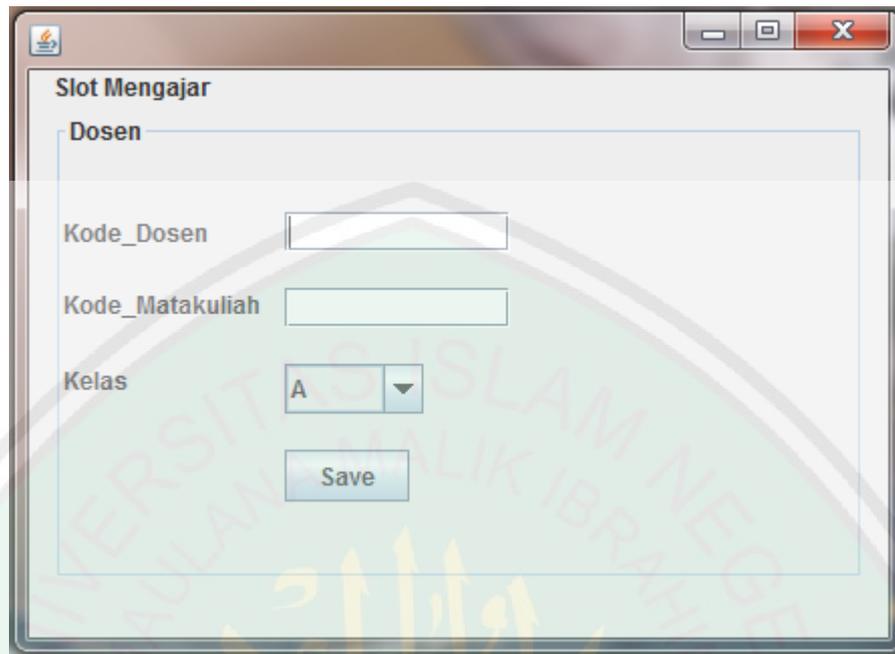
Aplikasi penjadwalan otomatis kali ini dibangun dengan menggunakan bahasa pemrograman Java. Pembuatan aplikasi ini memanfaatkan NetBeans IDE 7.1 sebagai *editor* untuk penulisan kode program dan Microsoft Access sebagai database. Untuk memudahkan pengguna dalam mengoperasikan aplikasi ini maka dibuat sebuah tampilan *interface* berbasis GUI (*Graphic User Interface*). Berikut ini adalah tampilan utama dari aplikasi ini.



Gambar 4.1 Tampilan *interface* halaman penjadwalan matakuliah

Pada aplikasi ini juga terdapat beberapa fitur pengisian data. Data-data tersebut adalah data yang mendukung dalam proses pembuatan jadwal matakuliah. Data akademik, data dosen, data matakuliah, data ruang, data hari, data waktu kuliah, data kelas termasuk data master pada aplikasi ini. Data akademik harus diisi terlebih dahulu dan datanya diaktifkan terlebih dahulu agar dapat menjalankan proses pemasaran dan pengampuan.

Sebelum proses pengampuan terdapat proses pemasaran matakuliah. Setelah proses pemasaran matakuliah dilakukan, selanjutnya adalah proses pengampuan matakuliah. Pada proses pengampuan ini jika terdapat matakuliah praktikum dan membutuhkan ruang laboratorium tertentu maka penempatan ruang tersebut dilakukan pada saat pengisian data pengampuan. Data dari proses pengampuan ini yang akan digunakan pada aplikasi kali ini untuk pembuatan jadwal matakuliah. Selain itu, juga terdapat fitur pengisian referensi dosen. Berikut ini adalah tampilan *interface* dari fitur-fitur aplikasi yang sudah dijelaskan di atas.



The image shows a software window titled "Slot Mengajar". Inside the window, there is a form with the following fields:

- Dosen**: A text input field.
- Kode_Dosen**: A text input field.
- Kode_Matakuliah**: A text input field.
- Kelas**: A dropdown menu currently showing "A".
- Save**: A button located below the "Kelas" dropdown.

Gambar 4.2 Tampilan *interface form input* Plot Mengajar Dosen

Selanjutnya pada form plot mengajar digunakan sebagai dasar dari penyusunan jadwal perkuliahan otomatis ini. Cara kerjanya adalah dengan mengambil satu baris data dari tabel Plot Mengajar (misalnya dari Kode Matakuliah 0765309 Kelas A) kemudian dilakukan pencarian ruang dan slot waktu yang masih kosong.

Perhitungan Manual

Berikut adalah contoh semisal perhitungan manual dari penjadwalan dengan komponen:

- Mata kuliah = Statistika A, Statistika B, Statistika C, Statistika D,
Rangkaian Digital A, Rangkaian Digital B, Rangkaian Digital C, Rangkaian Digital D, Studi Fiqih A, Studi Fiqih B, Studi Fiqih C, Studi Fiqih D
- Hari = Senin, Selasa, Rabu, Kamis, Jum'at, Sabtu
- Jam = 06.30-07.20, 07.20-08.10, 08.10-09.00, 09.00-09.50,
09.50-10.40, 10.40-11.30, 11.30-12.20, 12.20-13.10,
13.10-14.00, 14.00-14.50, 14.50-15.40, 15.40-16.30,
16.30-17.20
- Ruang = B105, B106, B107, B108, B304,
- Keadaan awal(suhu) =10
- Reduksi keadaan= -1

Dan *constraint*-nya:

- Tidak boleh ada matakuliah yang bentrok.(point 3(hard constrain))
- Waktu dhuhur.(point 1(soft constrain))

Jadwal awal(inisial state) dengan keadaan awal=10

Iterasi 1:

Statistika A, Rabu, 12.20-13.10, B105

Statistika B, Sabtu, 07.20-08.10, B108

Statistika C, Jum'at, 07.20-08.10, B108

Statistika D, Selasa, 13.10-14.00, B107

Rangkaian Digital A, Jum'at, 14.00-14.50, B105

Rangkaian Digital B, Sabtu, 15.40-16.30, B107

Rangkaian Digital C, Rabu, 15.40-16.30, B106

Rangkaian Digital D, Jum'at, 10.40-11.30, B108

Studi Fiqih A, Rabu, 15.40-16.30, B106

Studi Fiqih B, Selasa, 10.40-11.30, B107

Studi Fiqih C, Senin, 12.20-13.10, B108

Studi Fiqih D, Kamis, 14.00-14.50, B105

Bobot bentrokan(fungsi biaya)= $(0 \times 1) + (2 \times 3) = 6$

Jadwal baru(new state) dengan keadaan baru=9

Iterasi 2:

Statistika A, Sabtu, 16.30-17.20, B106

Statistika B, Jum'at, 10.40-11.30, B107

Statistika C, Selasa, 09.00-09.50, B106

Statistika D, Kamis, 11.30-12.20, B108

Rangkaian Digital A, Rabu, 07.20-08.10, B106

Rangkain Digital B, Kamis, 11.30-12.20, B106

Rangkaian Digital C, Jum'at, 07.20-08.10, B106

Rangkaian Digital D, Kamis, 14.50-15.40, B105

Studi Fiqih A, Sabtu, 08.10-09.00, B108

Studi Fiqih B, Rabu, 07.20-08.10, B106

Studi Fiqih C, Senin, 08.10-09.00, B105

Studi Fiqih D, Jum'at, 08.10-09.00, B106

Bobot bentrokan(fungsi biaya)= $(2 \times 1)+(2 \times 3)=8$

.....dst

Jadwal baru(new state) dengan keadaan baru=3

Iterasi 8:

Statistika A, Rabu, 12.20-13.10, B105

Statistika B, Sabtu, 07.20-08.10, B108

Statistika C, Jum'at, 07.20-08.10, B108

Statistika D, Selasa, 11.30-12.20, B107

Rangkaian Digital A, Jum'at, 14.00-14.50, B105

Rangkaian Digital B, Sabtu, 15.40-16.30, B107

Rangkaian Digital C, Rabu, 08.10-09.00, B106

Rangkaian Digital D, Jum'at, 10.40-11.30, B108

Studi Fiqih A, Rabu, 15.40-16.30, B106

Studi Fiqih B, Selasa, 10.40-11.30, B107

Studi Fiqih C, Senin, 12.20-13.10, B108

Studi Fiqih D, Kamis, 14.00-14.50, B105

Bobot bentrokan(fungsi biaya)= $(1 \times 1)+(0 \times 3)=1$

Jadwal baru(new state) dengan keadaan baru=2

Iterasi 9:

Statistika A, Rabu, 12.20-13.10, B105

Statistika B, Sabtu, 07.20-08.10, B108

Statistika C, Jum'at, 07.20-08.10, B108

Statistika D, Selasa, 13.10-14.00, B107

Rangkaian Digital A, Jum'at, 14.00-14.50, B105

Rangkaian Digital B, Sabtu, 15.40-16.30, B107

Rangkaian Digital C, Rabu, 15.40-16.30, B106

Rangkaian Digital D, Jum'at, 10.40-11.30, B108

Studi Fiqih A, Rabu, 15.40-16.30, B108

Studi Fiqih B, Selasa, 10.40-11.30, B107

Studi Fiqih C, Senin, 11.30-12.20, B108

Studi Fiqih D, Kamis, 14.00-14.50, B105

Bobot bentrokan(fungsi biaya)= $(1 \times 1)+(0 \times 3)=1$

Jadwal baru(new state) dengan keadaan baru=1

Iterasi 10:

Statistika A,Sabtu,07.20-08.10,B106

Statistika B,Senin,14.00-14.50,B107

Statistika C,Senin,13.10-14.00,B106

Statistika D,Senin,07.20-08.10,B106

Rangkaian Digital A,Senin,12.20-13.10,B304

Rangkaian Digital B,Jum'at,11.30-12.20,B304

Rangkaian Digital C,Kamis,12.20-13.10,B106

Rangkaian Digital D,Senin,15.40-16.30,B107

Studi Fiqih A,Selasa,09.50-10.40,B108

Studi Fiqih B,Selasa,10.40-11.30,B106

Studi Fiqih C,Rabu,12.20-13.10,B105

Studi Fiqih D,Kamis,12.20-13.10,B105

Bobot bentrokan(fungsi biaya) = $(0 \times 1) + (0 \times 3) = 0$

Jadwal akhir :

Statistika A,Sabtu,07.20-08.10,B106

Statistika B,Senin,14.00-14.50,B107

Statistika C,Senin,13.10-14.00,B106

Statistika D,Senin,07.20-08.10,B106

Rangkaian Digital A,Senin,12.20-13.10,B304

Rangkaian Digital B,Jum'at,11.30-12.20,B304

Rangkaian Digital C,Kamis,12.20-13.10,B106

Rangkaian Digital D,Senin,15.40-16.30,B107

Studi Fiqih A,Selasa,09.50-10.40,B108

Studi Fiqih B,Selasa,10.40-11.30,B106

Studi Fiqih C,Rabu,12.20-13.10,B105

Studi Fiqih D,Kamis,12.20-13.10,B105

Jadwal yang mungkin bisa :

Statistika A, Rabu, 12.20-13.10, B105

Statistika B, Sabtu, 07.20-08.10, B108

Statistika C, Jum'at, 07.20-08.10, B108

Statistika D, Selasa, 13.10-14.00, B107

Rangkaian Digital A, Jum'at, 14.00-14.50, B105

Rangkaian Digital B, Sabtu, 15.40-16.30, B107

Rangkaian Digital C, Rabu, 15.40-16.30, B106

Rangkaian Digital D, Jum'at, 10.40-11.30, B108

Studi Fiqih A, Rabu, 15.40-16.30, B108

Studi Fiqih B, Selasa, 10.40-11.30, B107

Studi Fiqih C, Senin, 11.30-12.20, B108

Studi Fiqih D, Kamis, 14.00-14.50, B105

Perhitungan :

Algoritma :

IF (SolusiBaru lebih baik) (kondisi 1)

SolusiSementara = SolusiBaru

NilaiEvaluasiSementara = NilaiEvaluasiBaru

ELSE (kondisi 2)

Delta = SolusiBaru - SolusiSementara

IF $\exp(-\text{Delta}/T) > \text{Random}(0 \dots 1)$:

SolusiSementara = SolusiBaru

NilaiEvaluasiSementara = NilaiEvaluasiBaru

IF (jadwal baru lebih baik) (kondisi 1)

Jadwal awal = Jadwal baru

Nilai bentrok awal = Nilai bnetrok baru

ELSE (kondisi 2)

Delta = nilai bentrok baru – nilai bentrok awal

IF $\exp(-\text{Delta}/T) > \text{Random}(0 \dots 1)$:

Jadwal awal = jadwal Baru

Nilai bentrok Sementara = Nilai bentrok Baru

1. Iterasi 1 dan 2 tidak memenuhi kondisi 1 dikarenakan solusi lebih buruk, maka lakukan kondisi 2 :

Delta = nilai bentrok₂ - nilai bentrok₃ = 6 - 12 = 6

- Jika $\exp(-\text{delta}/T) > \text{random acak}(0 \dots 1)$

Jika probabilitas lebih besar maka ganti jadwal awal dengan jadwal baru, jika lebih kecil tetap gunakan jadwal awal

2. Jika jadwal baru lebih baik maka gantikan jadwal awal dengan jadwal baru (kondisi 1).
3. Iterasi 9 dan 10 memenuhi kondisi 1 maka iterasi dihentikan dan iterasi 10 dipilih menjadi jadwal karena dianggap sudah optimum, nilai bentrok = 0.
4. Kemungkinan lain yang bisa terjadi untuk menghentikan iterasi apabila delta bernilai 0 hal ini bisa terjadi dikarenakan dari beberapa kali iterasi menghasilkan nilai bentrok tetap, sehingga jadwal itu ditetapkan sebagai hasil akhir dan iterasi dihentikan seperti iterasi 8 dan 9 (berlaku soft constrain).

Delta = nilai bentrok₈ - nilai bentrok₉ = 1 - 1 = 0

4.2 Implementasi Program

4.2.1 *Depth-first search*

Setelah proses pengampunan selesai dilakukan, data dari proses pengampunan ini yang nantinya akan digunakan untuk proses penjadwalan. Proses penjadwalan yang pertama dilakukan dengan menggunakan *depth-first search*. Algoritma ini bertugas untuk memenuhi *hard constraints*. Algoritma ini akan mencari solusi terbaik dalam masalah penjadwalan otomatis saat dibutuhkan.

```

public final class DFS {
    // variable
    private final int MAX_VERTS = 20;
    private Vertex vertexList[];
    private int adjMat[][];
    private int nVerts;
    private Stack stack;
    private TJadwals jadwalList;
    public BFS()
    {
        //inisialisasi variable
        vertexList = new Vertex[MAX_VERTS];
        jadwalList = new TJadwals();
        adjMat = new int[MAX_VERTS][MAX_VERTS];
        nVerts = 0;
        for(int y=0; y<MAX_VERTS; y++)
        for(int x=0; x<MAX_VERTS; x++)
            adjMat[x][y] = 0;
        stack = new Stack();
    }
    //menambah vertex
    public void addVertex(String lab)
    {
        vertexList[nVerts++] = new Vertex(lab);
    }
    //menambah edge
    public void addEdge(int start, int end)
    {
        adjMat[start][end] = 1;
        adjMat[end][start] = 1;
    }
    //manampilkan vertex ke console
    public void displayVertex(int v)
    {
        System.out.print(" Current Vertex: " + vertexList[v].label+"\n");
    }
    //proses penelusuran dfs
    public StringBuilder dfs(int mode)
    {
        //init start node
        vertexList[0].wasVisited = true;
        displayVertex(0);
        stack.push(0);
        StringBuilder sb = new StringBuilder();
        //cek posisi stack
        while( !stack.isEmpty() )
        {
            int v = getAdjUnvisitedVertex( stack.peek() );
            if(v == -1)
                stack.pop();
            else

```

```

int v = getAdjUnvisitedVertex( stack.peek() );
if(v == -1)
stack.pop();
else
    {
        //status node jika sudah dikunjungi
vertexList[v].wasVisited = true;
        // displayVertex(v);
        //mode penelusuran
System.out.println("Path : " +vertexList[v].label );
sb.append("\nLabels: ").append(vertexList[v].label);
        String hasil="";
for (int i=0;i<jadwalList.TDataJadwal.length;i++ ){
if (jadwalList.TDataJadwal[i]!=null){
System.out.println(jadwalList.TDataJadwal[i].ID);
}
else{
break;
}
}
sb.append("\n Hasil :").append(hasil+"\n");

stack.push(v);
    }
    //reset status node
for(int j=0; j<nVerts; j++)
vertexList[j].wasVisited = false;
return sb;
}
//mendapatkan relasi antar node
public int getAdjUnvisitedVertex(int v)
{
for(int j=0; j<nVerts; j++)
if(adjMat[v][j]==1 && vertexList[j].wasVisited==false)
return j;
return -1;
}
}

```

```

import java.sql.*;
import java.util.Date;
import java.util.Calendar;
import java.text.DateFormat;
import java.text.SimpleDateFormat;
class Aturan
{
    private String pola="H.mm";
    SimpleDateFormat sdf=new SimpleDateFormat(pola);
    private long ljm=0;
    private long lja=0;
    private Date jammu;
    private Date jamak;

    public boolean constr_ruang(Connection
connection,Statement statement,String semester,String
tahunajaran,String kelas,String hari,String ruang,String
kapasitas,Date jammu,Date jamak)
    {
        boolean test=false;
        String h="";
        String jm="";
        String ja="";
        String r="";
        String t="";
        String s="";

        try
        {
            statement = connection.createStatement();
            String sql="select
Hari, JamMulai, JamAkhir, Ruang, TahunAjaran, Semester from
JadwalKuliah";
            ResultSet rs=statement.executeQuery(sql);

            while(rs.next())
            {
                h=rs.getString(1);
                jm=rs.getString(2);
                ja=rs.getString(3);
                r=rs.getString(4);
                t=rs.getString(5);
                s=rs.getString(6);

                ljm=sdf.parse(jm).getTime();
                lja=sdf.parse(ja).getTime();

                if(hari.equals(h) && jammu.getTime()>=ljm &&
jammu.getTime()<lja && ruang.equals(r) &&
tahunajaran.equals(t) && semester.equals(s))
                {
                    test=true;
                    break;
                }
            }
        }
    }
}

```

```

}
    else if(hari.equals(h) && jammu.getTime()<lja
&& jamak.getTime()>ljm && ruang.equals(r) &&
tahunajaran.equals(t) && semester.equals(s))
    {
        test=true;
        break;
    }
    else if(hari.equals(h) && ljm>=jammu.getTime()
&& lja<=jamak.getTime() && ruang.equals(r) &&
tahunajaran.equals(t) && semester.equals(s))
    {
        test=true;
        break;
    }
    else
    {
    }
}
statement.close();
}
catch(Exception e)
{
    System.out.println("Error :"+e);
}
return test;
}
}

```

Gambar 4.4 Source code bentrok ruang & waktu

Penjelasan kode program:

- a. Mulai perulangan bentrok dosen
- b. Ketika pemasaran matakuliah sama, maka langsung ke perulangan berikutnya
- c. Ketika hari, jam, dan dosennya sama, penalti ditambah 1

- d. Ketika lebih dari 1 sks, hari dan dosen sama, jam kedua sama dengan jam pertama matakuliah yang lain, penalti ditambah 1
- e. Ketika lebih dari 2 sks, hari dan dosen sama, jam ketiga sama dengan jam pertama matakuliah yang lain, penalti ditambah 1

4.3 Uji Coba Aplikasi

4.3.1 *Depth-First Search (DFS)*

Depth-first search digunakan untuk memenuhi *hard constraints*.

Bentrok jadwal dosen tidak terjadi karena sifat dari *depth-first search* yang akan mencari solusi terbaik dari setiap masalah penjadwalan matakuliah.. Selain itu jika terdapat dosen yang mengajar dua matakuliah pada hari yang sama, maka pemberian slot waktu akan memperhatikan waktu yang sudah terjadwal terlebih dahulu pada hari itu. pengisian slot waktu yang menghubungkan matakuliah yang mempunyai hari dan ruang sama akan menghilangkan slot waktu yang sudah digunakan oleh hari dan ruang yang sama, sehingga tidak terjadi bentrok ruang kuliah.

Bentrok jadwal matakuliah yang didistribusikan untuk mahasiswa semester I atau II tidak akan terjadi karena secara otomatis slot waktu yang memungkinkan untuk melanggar bentrok tersebut akan diset 0.

Penjadwalan matakuliah otomatis yang menggunakan data penjadwalan 1 fakultas sebagai datanya, terdapat beberapa hal yang harus diperhatikan. Hal-hal tersebut antara lain, adanya dosen yang mengajar pada lebih dari 1 jurusan. Jika terdapat kasus yang seperti itu jadwal mengajar dari dosen

tersebut harus berbeda baik hari maupun waktunya, sehingga harus memeriksa dosen sama yang sudah terjadwal pada hari yang sama agar tidak terjadi bentrok waktu mengajar dosen. Selain itu, juga terdapat penggunaan ruang yang terdapat dalam gedung SAINSTEK/BJ Habibie yang di beberapa jurusan berbeda penggunaannya. Pada beberapa jurusan ruang yang ada di gedung SAINSTEK/BJ Habibie tidak digunakan untuk matakuliah praktikum, sehingga penggunaan ruang juga berbeda dengan ruang laboratorium.

Hasil akhir dari penjadwalan menggunakan *depth-first search* ini masih terdapat matakuliah yang belum terjadwal. Matakuliah yang belum terjadwal tersebut akan dibuat jadwalnya dengan menelusuri slot waktu yang kosong yang bisa digunakan. Beberapa faktor yang mempengaruhi adanya matakuliah yang tidak terjadwal adalah slot waktu yang digunakan pada hari dan ruang yang sama sudah digunakan oleh matakuliah lain. Selain itu, pengisian slot waktu juga mempertimbangkan jadwal dosen dengan hari yang sama yang sudah terjadwal terlebih dahulu. Sehingga slot waktu akan dikurangi dengan waktu jadwal dosen tersebut.

Beberapa faktor lain yang juga mempengaruhi hasil dari penjadwalan menggunakan *depth-first search* ini adalah pembentukan graf. Jumlah matakuliah yang membentuk graf akan mempengaruhi pendistribusian hari dan ruang. Jika graf terbentuk hanya dengan 2 matakuliah maka pendistribusian hari tidak akan maksimal karena hanya memakai 2 hari untuk penjadwalannya. Berbeda jika graf terbentuk dengan 5 matakuliah, maka pendistribusian hari dan ruang akan lebih

merata. Berikut ini adalah hasil uji coba dengan menggunakan data penjadwalan jurusan teknik informatika di fakultas Sains dan Teknologi.

Jumlah data	:	74
Jumlah pelanggaran <i>hard constraints</i>	:	0
Jumlah pelanggaran <i>soft constraints</i>	:	2

Berikut ini adalah hasil uji coba pada 1 jurusan di fakultas Sains dan Teknologi.

1. Jurusan Teknik Informatika

Jumlah hari	:	5
Slot waktu	:	06.30 – 17.20 WIB
Jumlah dosen	:	32
Jumlah matakuliah	:	33
Jumlah ruang kuliah	:	13
Jumlah matakuliah pengampuan	:	138
Jumlah matakuliah yang belum terjadwal	:	5
Jumlah pelanggaran <i>hard constraints</i>	:	0

pada jurusan Teknik Informatika terdapat 5 matakuliah yang belum terjadwal, jumlah ini sangat wajar. Faktor-faktor yang mempengaruhi antara lain karena slot waktu pada hari dan ruang yang dieksekusi sudah digunakan oleh matakuliah sebelumnya. Selain itu, jika terdapat dosen yang sama dan mengajar

pada hari yang sama maka slot waktu yang digunakan akan dikurangi sesuai waktu mengajar dosen yang sudah terjadwal sebelumnya.

Jadwal kuliah untuk jurusan Teknik Informatika tidak ada yang melanggar *hard constraints*. Untuk mengetahui perbandingan dengan jadwal yang sudah dikeluarkan oleh jurusan Teknik Informatika berikut ini adalah jadwal kuliah jurusan Teknik Informatika hasil dari aplikasi penjadwalan dan jadwal kuliah hasil dari keluaran admin jurusan Teknik Informatika.

a. Bentrok dosen

Berikut ini adalah contoh jadwal kuliah hasil dari *depth-first search*.

Matakuliah	Dosen	Kelas	Hari	Waktu
REKAYASA PERANGKAT LUNAK	FATCHURROCHMAN, M.Kom	A	Senin	06.30 - 08.10
REKAYASA PERANGKAT LUNAK	FATCHURROCHMAN, M.Kom	B	Selasa	06.30 - 08.10
REKAYASA PERANGKAT LUNAK	FATCHURROCHMAN, M.Kom	C	Rabu	06.30 - 08.10
REKAYASA PERANGKAT LUNAK	FATCHURROCHMAN, M.Kom	D	Kamis	06.30 - 08.10
REKAYASA PERANGKAT LUNAK	FATCHURROCHMAN, M.Kom	E	Jumat	06.30 - 08.10

Berikut ini adalah contoh jadwal kuliah dari jurusan Teknik Informatika.

Matakuliah	Dosen	Kelas	Hari	Waktu
MATEMATIKA LANJUT	Dr. SUHARTONO, M.Kom	A	Rabu	08.10 - 10.40
MATEMATIKA LANJUT	Dr. SUHARTONO, M.Kom	B	Kamis	09.50 - 12.20
MATEMATIKA LANJUT	Dr. SUHARTONO, M.Kom	C	Jumat	09.00 - 11.30
MATEMATIKA DISKRIT	HANI NURHAYATI, M.T	A	Jumat	08.10 - 10.40
MATEMATIKA DISKRIT	HANI NURHAYATI, M.T	B	Rabu	09.50 - 12.20

b. Bentrok ruang kuliah

Berikut ini adalah contoh jadwal kuliah hasil dari *depth-first search*.

Matakuliah	Dosen	Ruang	Hari	Waktu
ANALISIS & PERANCANGAN SISTEM	MUHAMMAD AINUL YAQIN, M.Kom	B306	senin	06.30 - 08.10
MATEMATIKA DISKRIT	RIRIEN KUSUMAWATI, S.Si, M.Kom	B306	senin	08.10 - 10.40
KEWIRAUSAHAAN	FACHRUL KURNIAWAN, ST., M. MT	B306	senin	12.20 - 14.00

4.4 Kajian Penelitian dalam Al-Qur'an

Penjadwalan matakuliah merupakan proses menempatkan data penjadwalan ke dalam slot waktu. Penjadwalan matakuliah juga mempertimbangkan hari kuliah dan penggunaan ruang sebagai sumber daya dalam proses belajar mengajar. Proses penjadwalan tidak seenaknya menempatkan matakuliah pada slot waktu yang kosong. Tetapi, dalam penjadwalan terdapat batasan atau criteria kapan seharusnya matakuliah itu terlaksana dan tidak boleh dilaksanakan. Penempatan matakuliah pada slot waktu dengan mempertimbangkan berbagai parameter yang sangat berpengaruh pada pelaksanaan belajar mengajar. Parameter tersebut bisa saja dari distribusi semester dan juga hari yang digunakan oleh matakuliah tersebut.

Pelaksanaan matakuliah yang mempunyai kriteria-kriteria tertentu dalam pelaksanaannya ini sama halnya dengan pelaksanaan sholat fardhu. Sholat fardhu dalam islam juga mempunyai criteria waktu tersendiri dalam pelaksanaannya. Pelaksanaan sholat fardhu tidak seenaknya sendiri tanpa mempertimbangkan criteria waktu dari sholat fardhu tersebut. Misalkan, waktu sholat maghrib dilaksanakan pada saat terbenamnya matahari, sehingga ketika

matahari belum terbenam itu berarti tidak boleh melaksanakan sholat maghrib.

Allah swt berfirman dalam surat An-Nisa' ayat 103.

وَاذْكُرُوا الصَّلَاةَ قَضَيْتُمْ فَاِذَا

مَوْقُوتًا كَتَبَ الْمُؤْمِنِينَ عَلَى كَانَتْ الصَّلَاةَ

“Maka apabila kamu telah menyelesaikan shalat(mu), ingatlah Allah di waktu berdiri, di waktu duduk dan di waktu berbaring. kemudian apabila kamu telah merasa aman, Maka dirikanlah shalat itu (sebagaimana biasa). Sesungguhnya shalat itu adalah fardhu yang ditentukan waktunya atas orang-orang yang beriman” (Qs. An-Nisa’/4: 95).

Penentuan waktu sholat dapat menggunakan bermacam-macam cara, yang pastinya juga selalu berkembang seiring perkembangan zaman. Pada zaman dahulu masih menggunakan sumber daya alam seperti matahari untuk penentuan waktu sholat. Tetapi, seiring perkembangan zaman, penentuan sholat juga dilakukan dengan menggunakan jam atau media teknologi yang lain. Sama halnya dengan penentuan waktu sholat, penjadwalan matakuliah juga menggunakan beberapa cara yang berkembang seiring berkembangnya teknologi.

Pada saat sebelum perkembangan teknologi saat ini, kebanyakan penjadwalan dilakukan dengan cara manual. Sekarang, seiring perkembangan

teknologi informasi, penjadwalan matakuliah dapat dilakukan dengan mudah dan cepat serta menghasilkan jadwal yang tidak melanggar batasan tertentu. Pada penelitian ini memanfaatkan perkembangan teknologi informasi yang semakin berkembang. Namun, perkembangan teknologi juga tidak lepas dari Alquran yang memang isinya adalah apa-apa yang terjadi di dunia ini. Allah swt berfirman dalam surat Yunus ayat 10.

﴿يُؤْمِنُونَ لَّا قَوْمَ عَنَّا وَالنُّذُرَ الْآيَاتِ تُغْنِي وَمَا وَاللَّأَرْضِ السَّمَوَاتِ فِي مَا ذَا أَنْظُرُوا قَلِيلٍ﴾

“Perhatikanlah apa yang ada di langit dan di bumi. Tidaklah bermanfaat tanda kekuasaan Allah dan rasul-rasul yang memberi peringatan bagi orang-orang yang tidak beriman” (Qs. Yunus/10: 101).

Ayat di atas memberitahu bahwa semua yang Allah ciptakan di dunia ini tidak ada yang sia-sia. Semua hal yang diciptakan Allah pasti bermanfaat karena itu merupakan tanda kekuasaan Allah. Manusia diperintahkan untuk berfikir tentang manfaat segala sesuatu yang ada di dunia ini. Semua manfaat di dunia ini akan dapat diambil ketika manusia tersebut mau berfikir tentang tanda-tanda kebesaran Allah swt.

Pada penelitian kali ini, menggunakan depth-first search sebagai metode untuk menyelesaikan jadwal kuliah yang tidak melanggar *hard constraints*. Pada graf terdiri dari simpul dan sisi, sisi disini berfungsi untuk menghubungkan sepasang simpul. Selain itu, terdapat teori graf mengenai pewarnaan simpul. Pewarnaan simpul ini bertujuan untuk memberikan warna pada

masing-masing simpul sehingga simpul-simpul yang bertetangga mempunyai warna yang berbeda. Implementasi warna pada masing-masing simpul dapat merepresentasikan solusi dari sebuah permasalahan di dunia nyata.

Teori graf yang telah dijelaskan pada keterangan di atas juga ada dalam firman Allah swt yang menjelaskan tentang pelaksanaan shalat, yaitu surat An-Nisa' ayat 103 yang sudah dituliskan pada paragraf sebelumnya. Shalat fardhu dalam islam dilaksanakan 5 kali dalam sehari dengan waktu yang tidak berbenturan antara shalat fardhu yang satu dengan shalat fardhu yang lain. Hubungan teori graf dengan pelaksanaan shalat adalah pada graf terdapat simpul, yang bisa kita representasikan sebagai lima macam shalat fardhu, kemudian sisi merupakan hubungan atau keterkaitan antara shalat fardhu yang satu dengan shalat fardhu yang lainnya. Kemudian warna simpul merepresentasikan waktu pelaksanaan dari masing-masing simpul. Sehingga karena setiap simpul atau shalat fardhu ini saling berkaitan maka warna atau waktu pelaksanaan dari masing-masing shalat fardhu pun berbeda. seperti itulah kajian penjadwalan dan teori tentang graf dalam Alquran. Alquran merupakan kitab yang isinya adalah apa-apa yang ada di bumi dan menjadi petunjuk bagi semua umat manusia serta tidak ada sedikitpun keraguan baik dalam tulisan maupun arti dari Alquran.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah melalui uji coba, maka jadwal kuliah yang dihasilkan terbukti tidak melanggar *hard constraints* dan data matakuliah yang terjadwal adalah 2 sks. Walaupun aplikasi ini hanya menggunakan sampel data pada satu jurusan saja di fakultas sains & teknologi. Namun, dalam uji coba aplikasi ini masih terdapat matakuliah yang belum terjadwal dengan menggunakan depth-first search. Maka, matakuliah tersebut dijadwalkan dengan menelusuri slot waktu yang bisa digunakan. Oleh karena itu penggunaan algoritma depth-first search ini dirasa masih belum optimal.

5.2. Saran

Setelah melakukan penelitian mulai dari tahap awal sampai uji coba. Ada beberapa saran yang dapat dijadikan pertimbangan oleh peneliti selanjutnya yang akan mengembangkan aplikasi serupa dalam penelitiannya. Beberapa saran tersebut sebagai berikut.

- a. Menggunakan algoritma tambahan agar pendistribusian hari dan ruang merata.
- b. Mengembangkan obyek penelitian menjadi satu perguruan tinggi atau kampus.

DAFTAR PUSTAKA

- Aldasht, M.M., Saheb, M., Najjar, Tamimi, M.H., Takruru, T.O. (2005). University Course Scheduling Using Parallel Multi-Objective Evolutionary Algorithms. *Journal of Theoretical and Applied Information Technology*, www.jatit.org, 129-139.
- Amirthagadeswaran, K. S. , Arunachalam, V.P.,. (2006). Improved solutions for job shop scheduling problems through genetic algorithm with a different method of schedule de duction. *International Journal of Advanced Manufacturing Technology*, 532-540.
- Jamnezhad, M.E., Javidan, R., Dezfouli , M.A. (2011). Optimization of timetabling structures based on evolutionary algorithms. 5th Symposium on Advances in Science and Technology (5thsastech) Mashhad, Iran.
- Pinedo, M. L. (2012). *Scheduling: Theory, Algorithms, and Systems*. New York: Springer.
- Xia, W., Wu, Z. (2006). A hybrid particle swarm optimization approach for the jobshop scheduling problem. *International Journal of Advanced Manufacturing Technology*, 360366.
- Oner, A., Ozcan, S., Dengi, D. (2011). Optimization Of University Course Scheduling Problem With A Hybrid Artificial Bee Colony Algorithm. *Evolutionary Computation (CEC)*, 2011 IEEE Congress on, 339-345.
- Breadth-First Search, http://en.wikipedia.org/wiki/Breadth-first_search, diakses padataanggal 09 Maret 2014.
- Marc Lipson, Ph.D, Seymour Lischutz, Ph.D. 2008. *Matematika Diskrit Edisi Ketiga*. Jakarta: Erlangga.
- Eppstein, David. *BFS and DFS*. <http://www.ics.uci.edu/~eppstein/161/960215.html>. diakses padataanggal 07 Maret 2014.
- Basic Graph Algorithms*. cs.anu.edu.au/student/comp3600/apac/basic_graph_algorithms.pdf. diakses padataanggal 07 Maret 2014.