

**OPTIMASI PENJADWALAN MATA PELAJARAN
MENGUNAKAN METODE *TABU SEARCH*
(STUDI KASUS: SMKN 2 SINGOSARI)**

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:

OLIVE KHOIRUL LUKLUIL MAKNUN AL FAISHOL

NIM: 0910960055



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016**

PENGESAHAN

OPTIMASI PENJADWALAN MATA PELAJARAN MENGGUNAKAN METODE *TABU SEARCH* (STUDI KASUS : SMKN 2 SINGOSARI)

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :

OLIVE KHOIRUL LUKLUIL MAKNUN AL FAISHOL
NIM: 0910960055

Skripsi ini telah diuji dan dinyatakan lulus pada
26 Agustus 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Agus Wahyu Widodo, S.T., M.Cs.
NIP. 19740805 200112 1 001

Budi Darma Setiawan, S.Kom. M.Cs.
NIP. 19841015 201404 1 002

Mengetahui,
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T., M.T., Ph.D.
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 26 Agustus 2016

Olive Khoirul Lukluil Maknun Al Faishol
NIM: 0910960055



KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah SWT, atas segala rahmat dan karuniaNya, penulis dapat menyelesaikan skripsi dengan judul “Optimasi Penjadwalan Mata Pelajaran Menggunakan Metode *Tabu Search* (Studi Kasus: SMKN 2 Singosari)”. Skripsi ini merupakan salah satu syarat untuk memenuhi persyaratan akademis untuk menyelesaikan studi di Program Studi Teknik Informatika Universitas Brawijaya. Selama melaksanakan skripsi ini, penulis mendapat bantuan dan dukungan dari banyak pihak. Untuk itu, penulis ingin mengucapkan terima kasih kepada:

1. Agus Wahyu Widodo,S.T.,M.Cs. selaku dosen pembimbing I yang telah bijaksana dan sabar dalam membimbing dan menyalurkan ilmu kepada penulis serta semua waktu dan nasehat yang telah diberikan dalam proses penyelesaian skripsi ini.
2. Budi Darma Setiawan,S.Kom.,M.Cs. selaku dosen pembimbing II yang telah bijaksana dan sabar dalam membimbing dan menyalurkan ilmu kepada penulis serta semua waktu dan nasehat yang telah diberikan dalam proses penyelesaian skripsi ini.
3. Segenap bapak dan ibu dosen yang telah mendidik dan mengamalkan ilmunya kepada penulis.
4. Segenap staff dan karyawan Fakultas Ilmu Komputer Universitas Brawijaya yang telah membantu kelancaran pengerjaan skripsi.
5. Mama, Papa, kedua kakak, kedua kakak ipar dan seluruh keluarga besar tercinta yang selalu memberikan doa, kasih sayang dan dukungan baik moral maupun materi sehingga penulis dapat menyelesaikan pendidikan dengan baik.
6. Hanny, Yoga, Tita, Lia, Candra, Vivin, Nining serta teman-teman seperjuangan Ilmu Komputer dan TIF angkatan 2009 dan seluruh warga Fakultas Ilmu Komputer, Universitas Brawijaya dalam perjalanan mencari ilmu.
7. Sufia A. dan A. Gladys
8. Dan semua pihak lain yang telah membantu terselesaikannya skripsi ini yang tidak bisa penulis sebutkan satu-persatu.

Semoga segala pertolongan dan kebaikan yang telah diberikan kepada penulis dibalas oleh Allah SWT. Penulis menyadari bahwa skripsi ini masih jauh dari sempurna, oleh karena itu kritik dan saran yang sifatnya membangun sangat diperlukan untuk memperbaiki mutu penulisan selanjutnya dan juga kebaikan penulis secara pribadi. Akhir kata, penulis berharap semoga skripsi ini dapat bermanfaat bagi banyak pihak.

Malang, 26 Agustus 2016

Penulis
olive.klma@gmail.com

ABSTRAK

Penjadwalan merupakan salah satu proses penting yang harus dilakukan oleh setiap organisasi untuk mencapai tujuan organisasi. Permasalahan penjadwalan dapat terjadi pada berbagai organisasi, terutama organisasi dengan sumber daya yang besar seperti perusahaan, pemerintahan, dan institusi pendidikan. Di SMKN 2 Singosari terdapat delapan jurusan, sehingga penjadwalan mata pelajaran menjadi salah satu masalah rumit yang selalu terjadi setiap awal semester. Selama ini proses penjadwalan yang berlangsung di SMKN 2 Singosari masih berjalan secara manual menggunakan bantuan Microsoft Excel. Sebuah sistem cerdas berbasis web dibuat untuk memudahkan proses penjadwalan di SMKN 2 Singosari. Sistem ini menggunakan algoritma *Tabu Search* untuk melakukan proses penjadwalan mata pelajaran. Dalam metode *Tabu Search*, solusi awal berupa jadwal dibangkitkan secara random, kemudian dicari solusi akhirnya dan yang menjadi *Tabu List* adalah kumpulan *move* berbentuk *array* yang merupakan solusi jadwal mata pelajaran dengan nilai total penalti paling kecil pada tiap iterasi. Hasil penjadwalan yang dibangkitkan memiliki total penalti sebesar 302 untuk penjadwalan pada 8 kelas.

Kata kunci: penjadwalan, mata pelajaran, *Tabu Search*



ABSTRACT

Scheduling is one of the important processes that must be done by each organization to achieve organizational goals. Scheduling problems can occur in a variety of organizations, especially organizations with large resources such as companies, governments, and educational institutions. At SMKN 2 Singosari there are eight majors, so scheduling the subjects is one of the complex problems that always occurred in the beginning of each semester. So far the subject scheduling process that takes place at SMKN 2 Singosari is still run manually using the help of Microsoft Excel. A web-based intelligent system designed to facilitate the subject scheduling process in SMKN 2 Singosari. The system uses Tabu Search algorithm to perform the scheduling process. In Tabu Search method, the initial solution is to generate random schedule, then searching for the final solutions and Tabu List is a collection of move in array form containing solutions to the schedule of subjects with smallest total penalty on each iteration. The results has total penalty of 302 for scheduling 8 classes.

Keywords: Scheduling, Subjects, Tabu Search



DAFTAR ISI

PENGESAHAN.....	ii
PERNYATAAN ORISINALITAS.....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	x
DAFTAR GAMBAR.....	xi
DAFTAR LAMPIRAN.....	xiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	2
1.4 Tujuan.....	3
1.5 Manfaat Penelitian.....	3
1.6 Sistematika Penulisan.....	3
BAB 2 LANDASAN KEPUSTAKAAN.....	4
2.1 Penjadwalan.....	4
2.2 Mata Pelajaran dan Kurikulum SMK.....	4
2.3 Metode <i>Tabu Search</i>	6
2.3.1 Intensifikasi dan Diversifikasi.....	7
2.3.2 Elemen Utama <i>Tabu Search</i>	8
2.3.3 <i>Pseudocode</i> Algoritma <i>Tabu Search</i>	8
2.3.4 <i>Tabu List</i>	9
2.3.5 Kriteria Pemberhentian (<i>Stopping Criterion</i>).....	9
2.4 Metode <i>Tabu Search</i> untuk Permasalahan Penjadwalan.....	10
2.5 Pengujian.....	11
BAB 3 METODOLOGI.....	12
3.1 Studi Literatur.....	12
3.2 Pengumpulan Data.....	13

3.3 Analisis Kebutuhan.....	14
3.4 Perancangan.....	14
3.5 Implementasi.....	14
3.6 Pengujian.....	14
3.7 Pengambilan Kesimpulan.....	14
BAB 4 PERANCANGAN.....	17
4.1 Deskripsi Sistem.....	17
4.2 Analisa Kebutuhan Perangkat.....	17
4.2.1 Identifikasi Aktor.....	17
4.2.2 Analisa Kebutuhan Masukan.....	18
4.2.3 Analisa Kebutuhan Proses.....	19
4.2.4 Analisa Kebutuhan Keluaran.....	19
4.3 Perancangan Sistem.....	20
4.3.1 <i>Flowchart</i> Sistem.....	20
4.3.2 Perancangan DFD Sistem.....	20
4.4 Perancangan Aplikasi Penjadwalan.....	21
4.4.1 Proses Penjadwalan dengan Metode <i>Tabu Search</i>	22
4.4.2 <i>Constraint</i>	23
4.4.3 Perhitungan Manual.....	26
4.5 Rancangan <i>User Interface</i>	33
4.6 Perancangan Pengujian.....	35
BAB 5 IMPLEMENTASI.....	36
5.1 Spesifikasi Lingkungan Uji Coba.....	36
5.1.1 Spesifikasi Perangkat Keras.....	36
5.1.2 Spesifikasi Perangkat Lunak.....	36
5.2 Batasan-batasan Implementasi.....	36
5.3 Implementasi Aplikasi Penjadwalan.....	37
5.3.1 Implementasi Basis Data.....	37
5.3.2 Implementasi Penjadwalan Menggunakan Metode <i>Tabu Search</i>	38

5.3.2.1 <i>Source Code</i> Penjadwalan dengan Metode <i>Tabu Search</i>	38
5.3.2.2 <i>Source Code</i> Proses Pemberian Penalti Pada Penjadwalan Mata Pelajaran	45
5.3.3 Implementasi Antarmuka.....	47
5.3.3.1 Antarmuka Melalui <i>User Administrasi</i>	48
5.3.3.2 Antarmuka Melalui <i>User Akademik</i>	49
5.3.3.3 Antarmuka Melalui <i>User Guru</i>	50
BAB 6 PENGUJIAN DAN EVALUASI.....	51
6.1 Hasil Pengujian.....	51
BAB 7 PENUTUP.....	54
7.1 Kesimpulan.....	54
7.2 Saran.....	54
DAFTAR PUSTAKA.....	55



DAFTAR TABEL

Tabel 2.1 Struktur Kurikulum SMK/MAK Bidang Keahlian Teknologi Informasi dan Komunikasi.....	5
Tabel 4.1 Data <i>Constraint</i>	23
Tabel 4.2 Alokasi Mata pelajaran dan Guru.....	27
Tabel 5.1 <i>Source Code</i> Proses Pemberian Penalti Pada Penjadwalan Mata Pelajaran.....	39
Tabel 5.2 <i>Source Code</i> Proses Pemberian Penalti Pada Penjadwalan Mata Pelajaran.....	45
Tabel 6.1 Hasil Pengujian dengan Parameter MaxIterasi.....	51
Tabel 6.2 Hasil Pengujian dengan Parameter Jumlah Kelas.....	52



DAFTAR GAMBAR

Gambar 2.1 Intensifikasi dan Diversifikasi.....	7
Gambar 2.2 Skema Move Mata Kuliah.....	11
Gambar 3.1 Diagram Alir Pelaksanaan Penelitian.....	13
Gambar 3.2 <i>Flow Map</i> Penjadwalan Mata Pelajaran yang Sedang Berlangsung.....	15
Gambar 3.3 <i>Flow Map</i> Penjadwalan Mata Pelajaran yang Diharapkan.....	16
Gambar 4.1 <i>Use Case</i> Diagram Sistem Penjadwalan Mata Pelajaran.....	18
Gambar 4.2 Bentuk <i>Array</i> Penyimpanan untuk Hasil Penjadwalan Mata Pelajaran.....	19
Gambar 4.3 Diagram Alir Penjadwalan Mata Pelajaran.....	20
Gambar 4.4 DFD Level 1 Sistem Penjadwalan Mata Pelajaran.....	21
Gambar 4.5 Konsep Arsitektur Sistem.....	21
Gambar 4.6 Diagram Alir Proses Penjadwalan dengan Metode Tabu Search.....	22
Gambar 4.7 Diagram Alir Proses Pembangkitan Solusi Awal Jadwal Mata Pelajaran	24
Gambar 4.8 Diagram Alir Proses Penyusunan Jadwal Mata Pelajaran.....	25
Gambar 4.9 Diagram Alir Proses Evaluasi Penjadwalan Mata Pelajaran.....	26
Gambar 4.10 Tabel Jadwal Kosong.....	29
Gambar 4.11 Tabel Jadwal dengan Mata Pelajaran yang Telah Dijadwalkan	30
Gambar 4.12 Tabel Solusi Awal Jadwal Mata Pelajaran.....	30
Gambar 4.13 Tabel Solusi Awal Jadwal Mata Pelajaran dengan Nilai Penalti.....	31
Gambar 4.14 Tabel Solusi Jadwal Mata Pelajaran Iterasi 1 Alternatif 0.....	32
Gambar 4.15 Halaman Perancangan <i>Home</i> Program Penjadwalan Melalui <i>User Administrasi</i>	33
Gambar 4.16 Gambar Perancangan Halaman <i>Tab</i> Tambah Data Kelas.....	33
Gambar 4.17 Gambar Perancangan Halaman Menu Buat Jadwal Mata Pelajaran.....	34
Gambar 4.18 Gambar Perancangan Halaman Hasil Penjadwalan Mata Pelajaran.....	34

Gambar 4.19 Gambar Perancangan Halaman <i>Home User Guru Tab</i> Jadwal Saya.....	35
Gambar 5.1 <i>Physical Diagram</i> Program Penjadwalan Menggunakan Metode Tabu Search.....	37
Gambar 5.2 Halaman Tambah Data Kelas.....	48
Gambar 5.3 Halaman Update Data Kelas.....	48
Gambar 5.4 Halaman Buat Jadwal Mata Pelajaran.....	49
Gambar 5.5 Tampilan Proses Penjadwalan yang Belum Disimpan.....	49
Gambar 5.6 Tampilan Hasil Akhir Penjadwalan yang Sudah Disimpan.....	50
Gambar 5.7 Halaman Jadwal Saya <i>User Guru</i>	50
Gambar 6.1 Perbandingan Total Penalti Pada Pengujian 1 dan 2 dengan Parameter MaxIterasi.....	52
Gambar 6.2 Perbandingan Total Penalti Pada Pengujian 1 dan 2 dengan Parameter Jumlah Kelas.....	53



DAFTAR LAMPIRAN



BAB 1 PENDAHULUAN

1.1 Latar Belakang

Penjadwalan merupakan salah satu proses penting yang harus dilakukan oleh setiap organisasi untuk mencapai tujuan organisasi. Sumber daya milik organisasi dapat digunakan secara efektif dan efisien dengan penjadwalan yang baik, sehingga dapat produktivitas organisasi meningkat. Di sisi lain, penjadwalan adalah salah satu masalah klasik yang secara terus menerus dipelajari oleh para ilmuwan sampai saat ini. Permasalahan penjadwalan dapat terjadi pada berbagai organisasi, terutama organisasi dengan sumber daya yang besar seperti perusahaan, pemerintahan, dan institusi pendidikan.

Pada institusi pendidikan, penjadwalan merupakan permasalahan yang sulit dipecahkan karena banyaknya batasan sehingga memerlukan waktu yang lama untuk mencapai solusi yang diharapkan. Salah satu institusi pendidikan yang mengalami permasalahan penjadwalan adalah SMK. Kurikulum SMK diarahkan agar dapat mencetak siswa yang siap kerja dan berani berkompetisi, sehingga jurusan / kompetensi keahlian yang terdapat di SMK disesuaikan dengan bidang keahlian yang terkait, dan tidak terbatas pada IPA, IPS, serta Bahasa saja.

Di SMKN 2 Singosari terdapat delapan jurusan, sehingga penjadwalan mata pelajaran menjadi salah satu masalah rumit yang selalu terjadi setiap awal semester. Selama ini proses penjadwalan yang berlangsung di SMKN 2 Singosari masih menggunakan bantuan Microsoft Excel. Dengan menggunakan Microsoft Excel, penjadwal harus menyusun jadwal untuk tiap-tiap kelas secara manual. Dengan sistem penjadwalan manual seperti ini, apabila terdapat satu perubahan pada jadwal, maka seluruh jadwal dapat terpengaruh. Selain itu, karena proses penggantian jadwal masih dilakukan secara manual, maka tidak ada sinkronisasi data yang dapat menyebabkan terjadinya duplikasi pada jadwal. Proses audit jadwal juga harus dilakukan oleh penjadwal secara manual, sehingga memungkinkan adanya *human error*. Jumlah kelas sangat berpengaruh pada lamanya proses penyusunan jadwal karena banyaknya variabel yang harus diperhitungkan dalam penyusunan jadwal seperti jumlah kelas, guru, dan kurikulum sekolah.

Berdasarkan permasalahan di atas, maka dibutuhkan sebuah sistem cerdas berbasis web untuk memudahkan proses penjadwalan yang ada di SMKN 2 Singosari. Sistem cerdas ini memberikan keluaran berupa beberapa alternatif solusi penjadwalan mata pelajaran. Sistem cerdas memberikan kontribusi untuk penjadwal dengan membantu proses penjadwalan berdasarkan solusi penjadwalan terbaik yang dihasilkan oleh sistem. Model sistem cerdas yang digunakan adalah dengan menggunakan metode *Tabu Search*.

Metode *Tabu Search* dipilih karena berdasarkan penelitian yang dilakukan oleh Gunawan et al. (2004), dalam menyelesaikan permasalahan penjadwalan mata kuliah, solusi hasil dari algoritma *Tabu Search* sedikit lebih baik dibandingkan dengan solusi hasil algoritma *Simulated Annealing* dan algoritma Genetika dalam perolehan nilai fungsi tujuan (*objective function value*). Selain

itu, secara umum unjuk kerja algoritma *Tabu Search* juga lebih baik dibandingkan algoritma *Simulated Annealing* dan algoritma Genetika dalam batas kualitas solusi yang dihasilkan. Namun algoritma *Simulated Annealing* mempunyai waktu komputasi yang lebih cepat dibandingkan waktu komputasi algoritma *Tabu Search* dan algoritma Genetika.

Metode *Tabu Search* adalah metode optimasi matematis yang akan memperbaiki performa dari *local search* dengan memanfaatkan penggunaan dari struktur memori. *Local search* adalah metode metaheuristik untuk memecahkan masalah optimasi komputasi yang sulit. *Local search* digunakan pada permasalahan yang dapat dirumuskan sebagai mencari solusi dengan cara memaksimalkan kriteria di antara sejumlah kandidat solusi. Sebagian solusi yang pernah dibangkitkan ditandai sebagai “tabu” sehingga algoritma *Tabu Search* tidak akan mengunjungi solusi tersebut berulang kali. Agar solusi terbaik tidak hilang, *Tabu Search* akan menyimpan solusi terbaik dan terus mencari berdasarkan solusi terakhir. Hal ini membuat algoritma *tabu search* menjadi lebih efisien dalam hal usaha dan waktu (Suyanto, 2010).

Penelitian mengenai penjadwalan menggunakan metode *Tabu Search* telah dilakukan oleh Fadillah (2013). Penelitian ini membahas tentang penjadwalan ujian komprehensif dengan studi kasus PTIUK UB. Pada penelitian tersebut, metode *Tabu Search* digunakan untuk menentukan penjadwalan ujian komprehensif baru di PTIUK. Pada penjadwalan ujian komprehensif tersebut, penjadwalan hanya dilakukan satu kali untuk setiap mahasiswa, dan terbatas pada periode yang telah ditetapkan sebagai tanggal untuk ujian komprehensif.

Berdasarkan paparan informasi di atas, penulis mengambil judul skripsi “Optimasi Penjadwalan Mata Pelajaran menggunakan Metode *Tabu Search* (Studi Kasus : SMKN 2 Singosari)”. Skripsi ini membahas mengenai sistem cerdas berbasis web, untuk penjadwalan mata pelajaran dengan studi kasus SMKN 2 Singosari menggunakan metode *Tabu Search*.

1.2 Rumusan Masalah

Berdasarkan permasalahan di atas, maka dibuat rumusan masalah yaitu: Bagaimana mengimplementasikan optimasi untuk penjadwalan mata pelajaran menggunakan metode *Tabu Search*?

1.3 Batasan Masalah

Agar permasalahan yang dirumuskan dapat lebih terfokus, maka pada penelitian ini dibatasi dalam hal :

1. Jadwal mata pelajaran dihasilkan dengan mempertimbangkan ketersediaan guru mata pelajaran, jumlah kelas, kurikulum sekolah, pemilihan waktu untuk mata pelajaran tertentu, serta alokasi waktu setiap mata pelajaran.
2. Penelitian ini hanya untuk optimasi penjadwalan mata pelajaran di SMKN 2 Singosari.

1.4 Tujuan

Tujuan dari penelitian ini adalah untuk mengimplementasikan optimasi untuk penjadwalan mata pelajaran menggunakan metode *Tabu Search*.

1.5 Manfaat Penelitian

Diharapkan dengan adanya penelitian ini dapat memberikan kontribusi bagi SMKN 2 Singosari dalam proses penyusunan penjadwalan mata pelajaran dan memudahkan penjadwal dalam menyusun jadwal mata pelajaran.

1.6 Sistematika Penulisan

Dalam penyusunan tugas akhir ini, secara garis besar meliputi beberapa bab, yaitu :

BAB I

Pendahuluan

Bab ini berisi latar belakang, rumusan masalah, tujuan, batasan masalah, manfaat tujuan, dan sistematika penulisan.

BAB II

Landasan Kepustakaan

Membahas teori dasar dan teori penunjang yang berhubungan dengan penjadwalan mata pelajaran, metode *tabu search* serta pengujian.

BAB III

Metodologi

Bab ini membahas metodologi yang digunakan pada sistem cerdas untuk penjadwalan mata pelajaran menggunakan metode *tabu search*.

BAB IV

Perancangan

Membahas tentang perancangan dari sistem.

BAB V

Implementasi

Membahas tentang implementasi dari sistem.

BAB VI

Pengujian dan Evaluasi

Bab ini membahas hasil dari pengujian dan evaluasi terhadap sistem yang telah diimplementasikan.

BAB VI

Penutup

Menguraikan kesimpulan yang diperoleh dari pembuatan dan pengujian aplikasi sistem, serta saran-saran untuk pengembangan lebih lanjut.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Penjadwalan

Jadwal memiliki arti pembagian waktu berdasarkan rencana pengaturan urutan kerja menurut penjelasan dari Kamus Besar Bahasa Indonesia ; daftar atau tabel kegiatan atau rencana kegiatan dng pembagian waktu pelaksanaan yg terperinci. Menurut Morton et al. (2001), penjadwalan merupakan suatu proses pengorganisasian, pemilihan, dan penentuan waktu penggunaan sumber daya yang ada untuk menghasilkan *output* seperti yang diharapkan dalam waktu yang diharapkan pula. Secara umum, masalah penjadwalan terdiri dari pengalokasian sejumlah *event* ke dalam periode waktu dan tempat terbatas dengan meminimalkan pelanggaran terhadap serangkaian *constraint*. Masalah penjadwalan yang berbeda memiliki *constraint* yang berbeda pula.

2.2 Mata Pelajaran dan Kurikulum SMK

Mata pelajaran menurut Kamus Besar Bahasa Indonesia berarti pelajaran yang harus diajarkan (dipelajari) untuk sekolah dasar atau sekolah lanjutan. Sedangkan kurikulum memiliki arti perangkat mata pelajaran yang diajarkan pada lembaga pendidikan. Undang-undang Nomor 20 Tahun 2003 tentang Sistem Pendidikan Nasional, menyebutkan bahwa kurikulum adalah seperangkat rencana dan pengaturan mengenai tujuan, isi, dan bahan pelajaran serta cara yang digunakan sebagai pedoman penyelenggaraan kegiatan pembelajaran untuk mencapai tujuan pendidikan tertentu. Berdasarkan pengertian tersebut terdapat dua dimensi kurikulum, yang pertama adalah rencana dan pengaturan mengenai tujuan, isi, dan bahan pelajaran, dan yang kedua adalah cara yang digunakan untuk kegiatan pembelajaran.

Kurikulum yang digunakan di SMK Negeri 2 Singosari untuk pengembangan sistem penjadwalan mata pelajaran adalah Kurikulum 2013. Menurut Permen Peraturan Menteri Pendidikan dan Kebudayaan No.70 Tahun 2013, Kurikulum 2013 ini menganut: (1) *Taught Curriculum* yang berarti pembelajaran dilakukan guru dalam bentuk proses yang dikembangkan berupa kegiatan pembelajaran di sekolah, kelas, dan masyarakat; dan (2) *Learned-curriculum* yang berarti pengalaman belajar langsung peserta didik sesuai dengan latar belakang, karakteristik, dan kemampuan awal peserta didik. Hasil belajar seluruh peserta didik akan menjadi hasil kurikulum, sementara pengalaman belajar langsung oleh individual peserta didik akan menjadi hasil belajar bagi dirinya. Kurikulum 2013 dikembangkan atas teori “pendidikan berdasarkan standar” (*standard-based education*), dan teori kurikulum berbasis kompetensi (*competency-based curriculum*). Mata pelajaran pilihan tersusun atas pilihan akademik dan vokasional untuk SMK/MAK. Mata pelajaran pilihan ini memberi corak kepada fungsi satuan pendidikan, dan terdapat pilihan sesuai dengan minat peserta didik didalamnya. Beban belajar SMK/MAK adalah sebanyak 48 jam pelajaran per minggu, dengan satu jam

belajar selama 45 menit. Struktur umum SMK/MAK terdiri dari tiga kelompok mata pelajaran, yaitu Kelompok A, B, dan C. Kelompok mata pelajaran yang substansinya dikembangkan oleh pusat termasuk ke kelompok A dan C. Kelompok mata pelajaran yang substansinya dikembangkan oleh pusat dan dapat dilengkapi dengan muatan lokal yang dikembangkan oleh pemerintah daerah masuk ke kelompok B. Pada saat peserta didik mendaftar pada SMK/MAK akan dilakukan pemilihan peminatan bidang keahlian dan program keahlian. Pilihan pendalaman peminatan keahlian dalam bentuk pilihan Paket Keahlian didasarkan pada nilai rapor dan/atau rekomendasi guru BK di SMK/MAK dan/atau hasil tes penempatan (placement test) oleh psikolog. Pada SMK/MAK, Mata Pelajaran Kelompok Peminatan (C) terdiri atas:

- Kelompok Mata Pelajaran Dasar Bidang Keahlian (C1);
- Kelompok Mata Pelajaran Dasar Program Keahlian (C2);
- Kelompok Mata Pelajaran Paket Keahlian (C3).

Direktorat Jenderal Pendidikan Menengah Kementerian Pendidikan dan Kebudayaan akan menetapkan mata pelajaran serta kompetensi dasar pada kelompok C2 dan C3 agar dapat menyesuaikan dengan perkembangan teknologi serta kebutuhan dunia usaha dan industri. Penerapan kurikulum 2013 disesuaikan dengan kebijakan dan kemampuan sekolah masing-masing. Pada penelitian ini mata pelajaran kelompok C disebut dengan mata pelajaran produksi. Berikut ini adalah contoh tabel struktur kurikulum mata pelajaran SMK/MAK bidang keahlian teknologi dan rekayasa yang dapat dilihat pada tabel 2.1.

Tabel 2.1 Struktur Kurikulum SMK/MAK Bidang Keahlian Teknologi Informasi dan Komunikasi

MATA PELAJARAN		KELAS DAN SEMESTER					
		X		XI		XII	
		1	2	1	2	1	2
Kelompok A (Wajib)							
1.	Pendidikan Agama dan Budi Pekerti	3	3	3	3	3	3
2.	Pendidikan Pancasila dan Kewarganegaraan	2	2	2	2	2	2
3.	Matematika	4	4	4	4	4	4
4.	Sejarah Indonesia	4	4	4	4	4	4
5.	Bahasa Inggris	2	2	2	2	2	2
Kelompok B (Wajib)							
6.	Seni Budaya	2	2	2	2	2	2
7.	Prakarya dan Kewirausahaan	2	2	2	2	2	2

Tabel 2.1 (lanjutan) Struktur Kurikulum SMK/MAK Bidang Keahlian Teknologi Informasi dan Komunikasi

8.	Pendidikan Jasmani, Olah Raga dan Kesehatan	3	3	3	3	3	3
Kelompok C (Peminatan)							
10.	Fisika	2	2	2	2	2	2
11.	Pemrograman Dasar	2	2	2	2	2	2
12.	Sistem Komputer	2	2	2	2	2	2
C2.	Dasar Program Keahlian	18	18	-	-	-	-
C3.	Paket Keahlian	-	-	18	18	24	24
Total		48	48	48	48	48	48

Sumber : Permen Peraturan Menteri Pendidikan dan Kebudayaan No.70 Tahun 2013

2.3 Metode *Tabu Search*

Tabu Search dikenal menjadi metaheuristik efisien dalam memecahkan berbagai masalah kombinatorial sulit, yang meliputi penjadwalan. Glover memperkenalkan *Tabu search* pertama kali sekitar tahun 1986 (Siregar, 2010). Menurut Glover dan Laguna (2013) kata *tabu* atau "*taboo*" berasal dari bahasa Tongan, suatu bahasa Polinesia yang digunakan oleh suku Aborigin pulau Tonga untuk mengindikasikan suatu hal yang tidak boleh "disentuh" karena kesakralannya. *Tabu* menunjukkan tidak boleh/terlarang untuk dilakukan (penting) yang berkaitan erat dengan memori sosial dari suatu kelompok masyarakat.

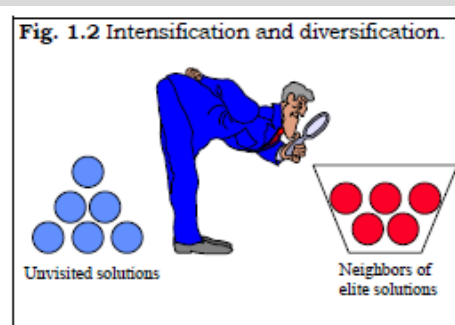
Tabu Search adalah suatu metode optimasi matematis yang termasuk ke dalam *local search*. *Tabu Search* memperbaiki performansi *local search* dengan memanfaatkan penggunaan struktur memori. Sebagian solusi yang pernah dibangkitkan ditandai sebagai "*tabu*" sehingga algoritma *Tabu Search* tidak akan mengunjungi solusi tersebut secara berulang-ulang. Solusi algoritma *Tabu Search* yang lebih buruk dari solusi saat ini tetap dapat diterima, sehingga algoritma ini akan menyimpan solusi terbaik dan terus mencari berdasarkan solusi terakhir agar solusi terbaik tidak hilang. Sebagian solusi yang pernah ditemui akan diingat dan solusi yang telah ditelusuri akan dilarang untuk diingat untuk menghindari pengulangan yang sia-sia. Hal-hal inilah yang membuat algoritma *Tabu Search* menjadi lebih efisien dalam hal usaha dan waktu yang dibutuhkan untuk menyelesaikan permasalahan (Suyanto, 2010). *Tabu Search* menggunakan *short-term memory* untuk menjaga agar proses pencarian tidak terjebak pada nilai optimum lokal (Kusumadewi, 2005)..

Tabu search menggabungkan *adaptive memory* dan *responsive exploration*. Dengan adanya kemampuan *adaptive memory*, memungkinkan adanya implementasi prosedur yang mampu untuk mencari lingkup solusi secara efisien dan ekonomis. Penggunaan *responsive exploration* pada *tabu search* sangat penting karena pengandaian bahwa pemilihan strategi yang buruk dapat menghasilkan lebih banyak informasi daripada sebuah pilihan baik yang diperoleh secara acak. *Responsive exploration* memiliki penggabungan prinsip dasar dari pencarian cerdas seperti selalu berusaha mencari solusi terbaik selama proses pencarian kemungkinan yang ada. Pencarian cara yang baru dan lebih efisien dalam mengambil keuntungan dari penggabungan mekanisme *adaptive memory* dan *responsive exploration*, akan dilakukan oleh *Tabu Search*. *Tabu Search* merupakan salah satu pilihan metode dalam merancang solusi dalam menyelesaikan masalah optimasi kombinatorial yang sulit (Glover, 2007).

2.3.1 Intensifikasi dan Diversifikasi

Metode Tabu Search dapat diatur dengan baik selama proses pencarian ketetangaan menggunakan cara intensifikasi dan diversifikasi. Untuk mengimplementasikan keduanya, dibutuhkan suatu tambahan pada fungsi objektif. Intensifikasi bisa dilakukan dengan memberikan penalti atau denda terhadap solusi yang jauh lebih buruk dari solusi saat ini. Sedangkan diversifikasi dilakukan dengan memberikan penalti pada solusi yang kualitasnya mendekati solusi saat ini. Tujuan dari intensifikasi adalah agar proses pencarian menjadi lebih sistematis dan fokus pada tetangga-tetangga yang lebih menjanjikan peningkatan kualitas. Diversifikasi membuat proses pencarian menjadi lebih meluas pada ruang solusi yang ada (Reeves, 1993).

Strategi intensifikasi berdasarkan pada modifikasi aturan pemilihan untuk mendorong atau menguatkan kombinasi pergerakan dan solusi yang mempunyai histori yang baik. Strategi ini juga dapat memberikan suatu nilai kepada daerah yang potensial untuk diproses lebih mendalam. Solusi elit harus dicatat untuk memeriksa tetangga terdekat mereka, memori eksplisit berkaitan erat dengan pelaksanaan strategi intensifikasi. Pada Gambar 2.1 menggambarkan perbedaan utama antara intensifikasi dan diversifikasi, yakni selama tahap intensifikasi pencarian berfokus pada proses pemeriksaan ketetangaan dari solusi elit.



Gambar 2.1 Intensifikasi dan Diversifikasi

Sumber: Glover (1997)

Strategi ini mencari “*neighbors*” dengan menggabungkan semua komponen dari solusi yang baik atau menggunakan evaluasi dari strategi yang telah dimodifikasi agar menjadi sebuah solusi yang dapat berkembang. Proses pencarian yang digunakan untuk menguji daerah yang tidak pernah dikunjungi/dibahas sebelumnya untuk menghasilkan suatu solusi yang berbeda dari alternatif-alternatif solusi yang pernah ada merupakan dasar dari diversifikasi. Pengukuran diversifikasi juga berhubungan dengan banyaknya perpindahan yang dibutuhkan untuk memindahkan satu solusi ke solusi yang lainnya. Diversifikasi juga mendorong agar proses pencarian mencoba daerah yang belum pernah dikunjungi dan menghasilkan solusi yang berbeda dalam banyak hal (Glover, 2013).

Pada banyak literatur, metode *Tabu Search* dapat menyelesaikan banyak masalah optimasi yang sulit sehingga dianggap efisien. *Tabu List* digunakan untuk menerima solusi yang tidak bisa memberikan peningkatan kualitas sehingga dapat lepas dari jebakan optimum lokal. Metode *Tabu Search* mungkin dapat menemukan optimum global, namun metode *Tabu Search* tidak menjamin ditemukannya optimum global karena banyak parameter yang harus ditetapkan secara hati-hati (Ji, 2004).

2.3.2 Elemen Utama *Tabu Search*

Terdapat 5 elemen utama dari *Tabu Search*, yaitu:

1. Mengklasifikasikan suatu subhimpunan langkah di dalam suatu ketetanggaan sebagai larangan atau *tabu* merupakan langkah utama untuk memanfaatkan memori di dalam *Tabu Search*.
2. Mengidentifikasi solusi-solusi tetangga yang dapat dicapai dari solusi saat ini merupakan tujuan dibangunnya ketetanggaan (*neighbourhood*).
3. Sejarah pencarian merupakan dasar klasifikasi, terutama kebaruan (*recency*) atau frekuensi (*frequency*) bahwa langkah atau komponen solusi tertentu (atribut) telah berpartisipasi pada pembangkitan solusi-solusi sebelumnya.
4. *Tabu Moves* adalah *Tabu List* yang digunakan untuk mencatat langkah-langkah terlarang
5. Kondisi atau kriteria pengabaian status *tabu* disebut kondisi aspirasi. Hal ini terjadi Ketika suatu langkah *tabu* memberikan suatu solusi yang lebih baik dibandingkan semua langkah terbaik sebelumnya, maka status *tabu* dari langkah tersebut bisa diabaikan (Suyanto, 2010).

2.3.3 Pseudocode Algoritma *Tabu Search*

Secara umum, *pseudocode* algoritma *Tabu Search* dapat dituliskan sebagai berikut (Rohini et al., 2016):

1. $s \leftarrow s_0$
2. $s_{\text{Best}} \leftarrow s$
3. $\text{tabuList} \leftarrow []$
4. while (not stoppingCondition())
5. $\text{candidateList} \leftarrow []$
6. $\text{bestCandidate} \leftarrow \text{null}$

7. for (sCandidate in sNeighborhood)
8. if ((not tabuList.contains(sCandidate)) and (fitness(sCandidate) > fitness(bestCandidate)))
9. bestCandidate ← sCandidate
10. end
11. end
12. s ← bestCandidate
13. if (fitness(bestCandidate) > fitness(sBest))
14. sBest ← bestCandidate
15. end
16. tabuList.push(bestCandidate);
17. if (tabuList.size > maxTabuSize)
18. tabuList.removeFirst()
19. end
20. end
21. return sBest

2.3.4 Tabu List

Tabu List merupakan struktur memori yang digunakan oleh *Tabu Search* untuk menyimpan atribut dari sebagian *move* (langkah transisi dari satu solusi ke solusi yang lain) yang telah diterapkan pada iterasi-iterasi sebelumnya. Fungsi lain dari *Tabu List* adalah untuk menolak solusi-solusi yang memenuhi atribut tertentu agar proses pencarian tidak dilakukan berulang kali di daerah solusi yang sama. Selain itu *Tabu List* juga digunakan untuk menuntun proses pencarian menelusuri solusi-solusi yang belum pernah dikunjungi (Suyanto, 2010).

Tabu List dapat disimpan dalam bentuk *array* atau *tree*, sesuai dengan masalah dan implementasi yang diinginkan. Ukuran *Tabu List* dapat mengikuti aturan berikut :

1. *Tabu List* statis. Ukuran *Tabu List* tidak berubah dan selalu sesuai dengan yang telah ditentukan sebelumnya.
2. Ukuran *Tabu List* disesuaikan dengan besar masalah. Semakin besar masalah, maka ukuran *Tabu List* yang dibutuhkan juga semakin besar.
3. Mengubah ukuran *Tabu List* sesuai dengan kualitas solusi. *Move* adalah suatu cara untuk menghasilkan solusi yang baru di dalam sebuah *neighborhood* (Trisnawati et al., 2011).

2.3.5 Kriteria Pemberhentian (*Stopping Criterion*)

Proses pencarian dengan *Tabu Search* akan dihentikan ketika sebuah solusi dengan nilai fungsi obyektif nol telah didapatkan. Begitu juga ketika jumlah iterasi maksimal yang ditentukan oleh pengguna telah tercapai dan solusi yang lebih baik dari solusi sebelumnya tidak tercapai, maka proses pencarian akan dihentikan (Aladag et al., 2007).

2.4 Metode Tabu Search untuk Permasalahan Penjadwalan

Setyadi (2012) melakukan penelitian mengenai aplikasi penjadwalan kelas pada sanggar belajar menggunakan tabu search, parameter yang digunakan untuk pembuatan jadwal kelas adalah data pelajaran, data tutor, *tabu rule*, dan kurikulum. Penjadwalan kelas memiliki perbedaan dengan penjadwalan mata pelajaran SMK, karena pada penjadwalan mata pelajaran yang diajarkan di SMK, setiap kelas harus mengikuti pelajaran sesuai dengan kurikulum jurusan.

Penelitian mengenai penjadwalan mata pelajaran dengan metode *Tabu Search* juga dilakukan oleh Khang et al. (2010) untuk SMA di Vietnam. Penelitian dilakukan dengan mengaplikasikan *Tabu Search* untuk masalah penjadwalan mata pelajaran, yang melibatkan penetapan mata pelajaran dengan lama jam pelajaran yang berbeda-beda ke periode yang sesuai. Algoritma yang diusulkan memiliki dua fase: fase inisialisasi menggunakan algoritma *Greedy* dan fase perbaikan menggunakan *Tabu Search*. Dalam algoritma *Tabu Search* untuk penjadwalan mata pelajaran ini, tiga jenis perubahan yang digunakan adalah: *single moves*, *swap moves* dan *block-changing moves*.

Membangkitkan jadwal awal (*initial timetable*) harus dilakukan dengan mudah dan cepat. Setiap *event* dijadwalkan ke *timeslot* dan ruang paling pagi yang tersedia. Solusi jadwal awal ini diusahakan untuk memenuhi semua *hard constraint*. Tentu saja, jadwal awal tidak sepenuhnya memenuhi semua *hard constraint*, dan *constraint* lain yang tidak terpenuhi akan dipenuhi melalui perbaikan dengan metode *Tabu Search*. Implementasi variasi *Tabu* diperlukan untuk masalah-masalah tertentu. Banyak dari variasi *Tabu* ini yang meliputi pemilihan struktur *move* yang memungkinkan, kriteria aspirasi, variasi ukuran *Tabu List*, penggunaan memori yang lebih lama, dan modifikasi kriteria pemberhentian (Mushi, 2006).

Valdes et al. (2001) melakukan penelitian mengenai *Tabu Search* untuk permasalahan organisasi di universitas. Permasalahan ini meliputi Penjadwalan Kuliah, Penjadwalan Kelas-Dosen, Penjadwalan Mahasiswa, Penjadwalan Dosen, dan Penjadwalan Kelas. Berdasarkan penelitian tersebut, *Tabu Search* tidak seperti metaheuristik lainnya, *Tabu Search* didasarkan pada ide bahwa pencarian cerdas harus menampilkan eksplorasi sistematis dari ruang solusi. Karena itu metode ini pada dasarnya bersifat deterministik. Solusi ketetanggaan dan *move* yang mendefinisikannya, adalah salah satu elemen yang paling penting. Karena kemampuan dari algoritma untuk mencari ruang solusi secara efisien bergantung pada hal tersebut. Tipe dari *move* untuk mendefinisikan setiap kasus bergantung pada ruang solusi yang didefinisikan oleh *hard constraint* dan terbagi menjadi 3 yaitu:

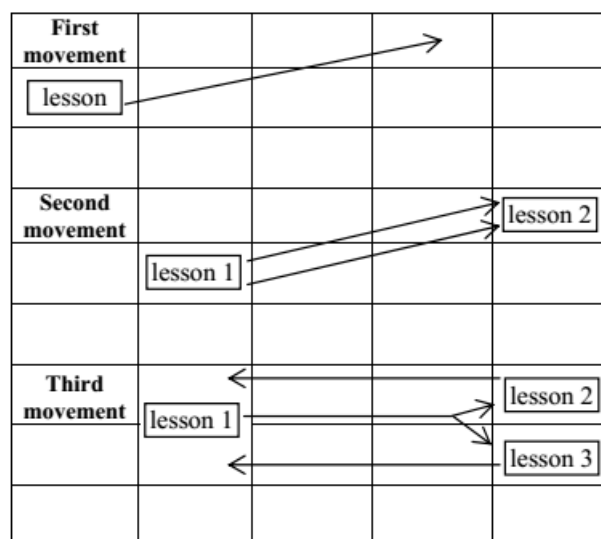
1. Kadang *move* dapat bersifat simpel. Seperti pada bagian Tugas Mahasiswa, setiap solusi terdiri dari satu tugas individu untuk setiap mahasiswa, *move* hanya akan mengubah satu dari banyak tugas individual ini. Pada penjadwalan Ujian, sebuah *move* sederhana berarti mengubah periode yang diperuntukkan bagi satu mata pelajaran dari satu periode ke periode lainnya. Walaupun demikian, *constraint* yang dikenakan pada permasalahan dapat mengakibatkan *move* sederhana ini menjadi tidak berguna. Misalnya, mengubah periode yang

diperuntukkan bagi sebuah mata kuliah akan menjadi sangat sulit dilakukan jika harus terdapat sebuah ruang kosong pada periode baru dan ruangan yang tersedia terbatas.

2. Move yang bersifat lebih kompleks adalah swap, melibatkan dua item yang menukar tugas mereka. Pada Penjadwalan Ujian, sebuah swap terdiri dari sepasang ujian yang diubah periodenya. Swap lainnya untuk problem yang sama didefinisikan dengan pertukaran daftar lengkap dari ujian-ujian yang ditetapkan pada periode tertentu dengan daftar ujian lainnya yang ditetapkan pada periode lainnya. Pada Penjadwalan Mata Kuliah, pertukaran periode antar mata kuliah mungkin mengeliminasi permasalahan pencarian ruang kelas, karena setiap mata kuliah diperuntukkan ke ruangan yang sebelumnya ditempati oleh mata kuliah lainnya.

3. Jika struktur solusi tidak mudah dimodifikasi dengan *move-move* ini, sebuah move baru dapat didefinisikan yang sifatnya lebih kompleks lagi. Misalnya, pada Penjadwalan Mata Kuliah, move yang digunakan adalah multiswap, dimana sebuah mata kuliah ditukar periodenya dengan satu set lebih dari satu mata kuliah. Move ini sangat berguna ketika mata kuliah yang berkaitan memiliki lama durasi pengajaran yang berbeda atau mata kuliah ini saling berkaitan sehingga harus dipindah bersamaan.

Skema dari move yang berbeda ini digambarkan untuk Penjadwalan Mata Kuliah ini terlihat pada Gambar 2.2



Gambar 2.2 Skema Move Mata Kuliah

Sumber: Valdes et al. (2001)

2.5 Pengujian

Pengujian merupakan tahap penting dalam pengembangan sebuah sistem karena tahap ini bertujuan untuk menemukan kesalahan yang masih terjadi setelah sebuah sistem diimplementasikan. Pengujian juga merupakan suatu proses yang dilakukan untuk menilai apakah penerapan suatu metode telah berjalan maksimal.

BAB 3 METODOLOGI

Bab ini menjelaskan mengenai langkah-langkah serta metode yang dilakukan dalam penelitian. Langkah-langkah yang dilakukan adalah sebagai berikut:

1. Studi Literatur

Studi literatur mengenai sistem yang sedang berjalan di SMK Negeri 2 Singosari dan penjadwalan menggunakan metode *Tabu Search* dilakukan pada tahap ini.

2. Analisis Kebutuhan

Analisis terhadap proses penjadwalan mata pelajaran yang telah berjalan di SMKN 2 Singosari dilakukan pada tahap ini. Menganalisis kebutuhan dari sistem dan mengidentifikasi kebutuhan dari pengguna terhadap sistem yang akan dibuat.

3. Perancangan Sistem

Pada tahap ini dilakukan perancangan dari hasil analisis kebutuhan yang telah dilakukan, meliputi proses perancangan penjadwalan mata pelajaran menggunakan metode *Tabu Search*, perancangan desain antar muka dari sistem, dan perancangan pengujian.

4. Implementasi

Pada tahap ini dilakukan implementasi hasil dari analisis dan perancangan yang telah dibuat di tahap perancangan sistem.

5. Uji Coba dan Evaluasi

Pada tahap ini dilakukan uji coba untuk mengetahui pengaruh parameter terhadap sistem yang telah dirancang dan dibangun.

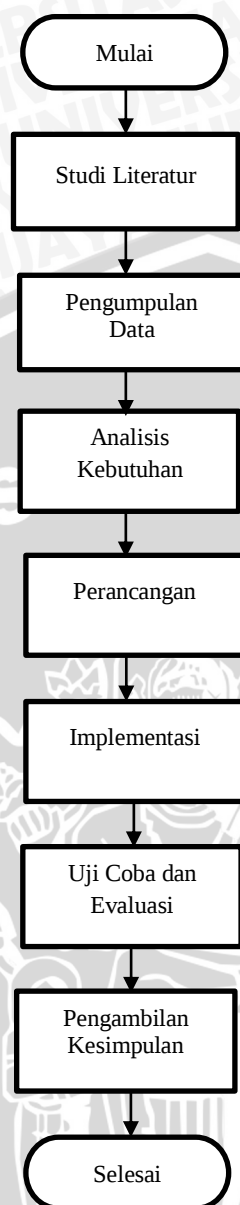
Diagram alir dari penelitian yang akan dilakukan ditunjukkan pada Gambar 3.1.

3.1 Studi Literatur

Studi literatur yang dilakukan dalam penelitian ini berupa studi pustaka. Bahan studi pustaka berupa buku, jurnal, maupun media elektronik (internet) digunakan untuk menambah pemahaman terhadap optimasi penjadwalan mata pelajaran menggunakan metode *Tabu Search*. Studi literatur memaparkan teori yang digunakan untuk penelitian dan penulisan skripsi ini. Berikut ini adalah teori-teori pendukung yang dibutuhkan :

- a. Penjadwalan
- b. Mata Pelajaran dan Kurikulum SMK
- c. Algoritma *Tabu Search*
- d. Pengujian

Literatur yang diperoleh kemudian diringkas dan dijelaskan pada bab 2.



Gambar 3.1 Diagram Alir Pelaksanaan Penelitian

3.2 Pengumpulan Data

Pada tahap pengumpulan data, variabel penelitian yang dibutuhkan adalah data mata pelajaran, data kurikulum, data user, data slot waktu, data hari, data penalti, data kelas serta bagaimana hasil optimasi penjadwalan menggunakan metode *Tabu Search*. Data diperoleh melalui studi literatur dan wawancara pada akademik yang bertugas menjadwalkan jadwal mata pelajaran di SMK Negeri 2 Singosari. Informasi yang diperoleh berupa deskripsi data, cara menentukan jadwal secara manual berdasarkan kurikulum dan batasan yang telah ditetapkan.

3.3 Analisis Kebutuhan

Analisis kebutuhan adalah proses menghasilkan spesifikasi kebutuhan sistem. Spesifikasi kebutuhan berisi mengenai kebutuhan dari pengguna, serta perancangan sistem yang dibuat berdasarkan dari kebutuhan pengguna tersebut. Secara umum sistem yang dirancang dalam penelitian ini adalah suatu sistem cerdas untuk penjadwalan mata pelajaran menggunakan metode *Tabu Search*. Dengan sistem ini, seorang penjadwal mata pelajaran akan memilih data kelas yang akan dijadwalkan, selanjutnya akan dilakukan penjadwalan menggunakan metode *Tabu Search* dan hasil akhir yang didapatkan adalah solusi jadwal mata pelajaran yang terbaik dari yang dihasilkan oleh sistem. Tujuan dari pembuatan sistem ini adalah memberikan solusi untuk penjadwalan mata pelajaran di SMKN 2 Singosari.

3.4 Perancangan

Pada tahap perancangan, dibangun sebuah sistem cerdas yang mengoptimalkan proses penjadwalan mata pelajaran menggunakan metode *Tabu Search*. Masukan dari sistem berupa data penyusun jadwal mata pelajaran serta paramater penjadwalan. Sedangkan keluaran dari sistem berupa tabel jadwal mata pelajaran yang telah *fix*.

Flow map penjadwalan mata pelajaran yang saat ini sedang berjalan di SMKN 2 Singosari ditunjukkan pada Gambar 3.2. Sedangkan *Flow map* penjadwalan mata pelajaran yang diharapkan berjalan ke depannya ditunjukkan pada Gambar 3.3.

3.5 Implementasi

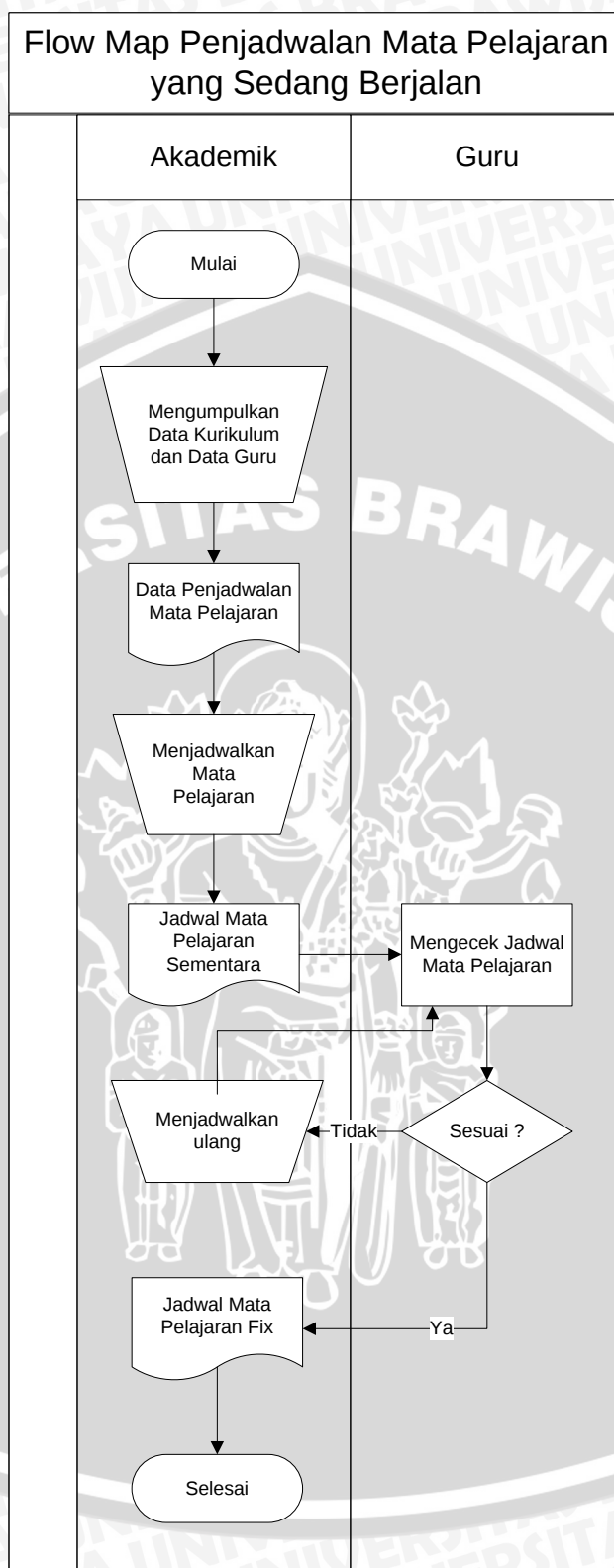
Implementasi dalam hal ini adalah pembuatan sistem cerdas berdasarkan arsitektur data dan arsitektur aplikasi yang telah dirancang sebelumnya. Program penjadwalan akan menggunakan bahasa pemrograman PHP dan dibuat menggunakan program Adobe Dreamweaver CS5 dan Sublime Text. Program akan ditampilkan melalui *web browser* Google Chrome dan XAMPP.

3.6 Pengujian dan Evaluasi

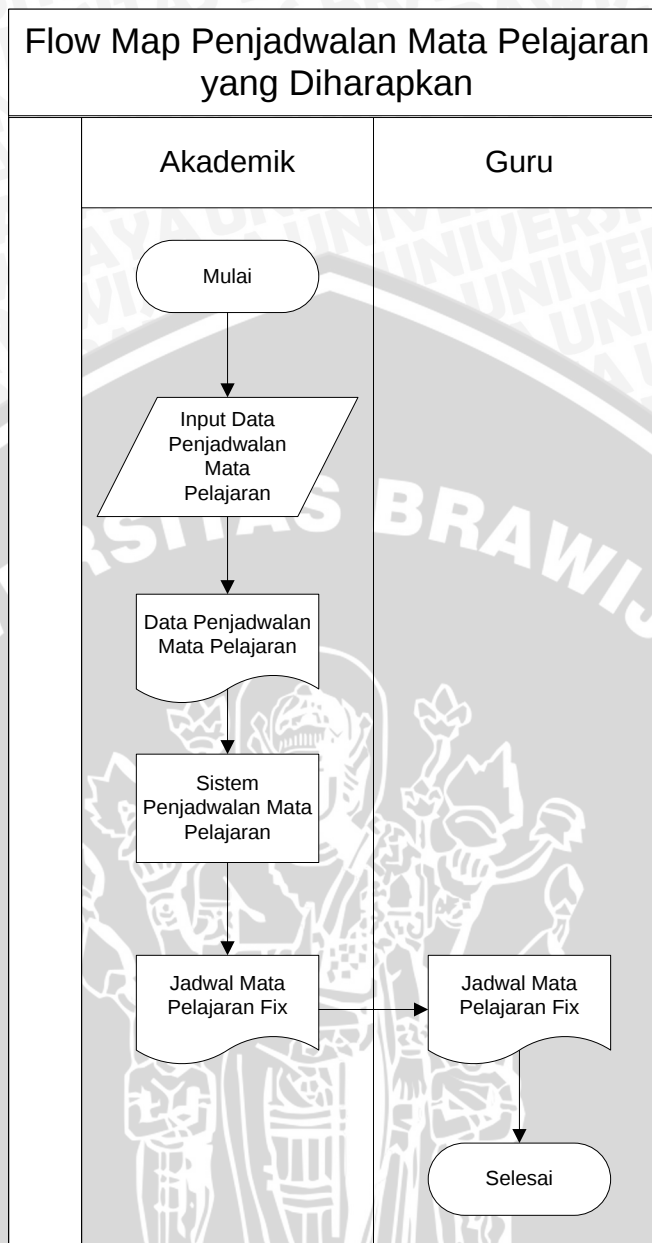
Pada tahap ini dilakukan pengujian dengan menggunakan observasi terhadap parameter yang mempengaruhi kinerja algoritma *Tabu Search* dalam penjadwalan mata pelajaran.

3.7 Pengambilan Kesimpulan

Pengambilan kesimpulan dari penelitian dilakukan setelah semua tahapan studi literatur, pengumpulan data, perancangan, Implementasi, dan pengujian metode yang diterapkan telah selesai dilakukan. Hasil pengujian akan menentukan kesimpulan yang didapat pada akhir penelitian. Tahap selanjutnya adalah adanya saran untuk memperbaiki dan mengembangkan penelitian selanjutnya.



Gambar 3.2 Flow Map Penjadwalan Mata Pelajaran yang Sedang Berlangsung



Gambar 3.3 Flow Map Penjadwalan Mata Pelajaran yang Diharapkan

BAB 4 PERANCANGAN

4.1 Deskripsi Sistem

Secara umum sistem yang dirancang dalam penelitian ini adalah suatu sistem cerdas berbasis web untuk optimasi penjadwalan mata pelajaran menggunakan metode *Tabu Search*. Sistem ini memiliki tiga tipe *user* yaitu Administrasi, Akademik, dan Guru. Administrasi bertugas memasukkan data masukan untuk proses penjadwalan mata pelajaran seperti data mata pelajaran, data kelas, data guru, dll. Akademik bertugas menentukan parameter penjadwalan dan menjadwalkan mata pelajaran menggunakan metode *Tabu Search* sedangkan *user* Guru hanya melihat hasil jadwal yang dihasilkan oleh sistem. Data masukan akan dimasukkan ke tabel jadwal mata pelajaran kemudian dibangkitkan hasilnya menggunakan metode *Tabu Search*. Mata pelajaran yang bertabrakan akan diberi nilai penalti dan dilakukan pengulangan pencarian hasil. Hasil keluaran dari sistem berupa tabel berisi jadwal mata pelajaran yang dianggap terbaik menurut sistem. Tujuan dari pembuatan sistem ini adalah memberikan optimasi pada sistem penjadwalan mata pelajaran yang berjalan di SMKN 2 Singosari.

4.2 Analisa Kebutuhan Perangkat

Analisis kebutuhan ini ditujukan untuk menggambarkan kebutuhan - kebutuhan yang harus disediakan oleh sistem agar dapat memenuhi kebutuhan pengguna dari sistem penjadwalan mata pelajaran ini. Analisis kebutuhan perangkat berisi identifikasi aktor-aktor yang terlibat dalam sistem penjadwalan, penjabaran kebutuhan masukan, proses dan keluaran. Berikut ini adalah kebutuhan yang digunakan dalam pembuatan sistem cerdas penjadwalan:

1. Kebutuhan Hardware, meliputi:
 - Laptop
2. Kebutuhan Software , meliputi:
 - Web Browser
 - Basis data MySQL
 - Bahasa Pemrograman PHP
3. Data yang dibutuhkan meliputi:
 - Data penyusun jadwal mata pelajaran
 - Data *constraint*

4.2.1 Identifikasi Aktor

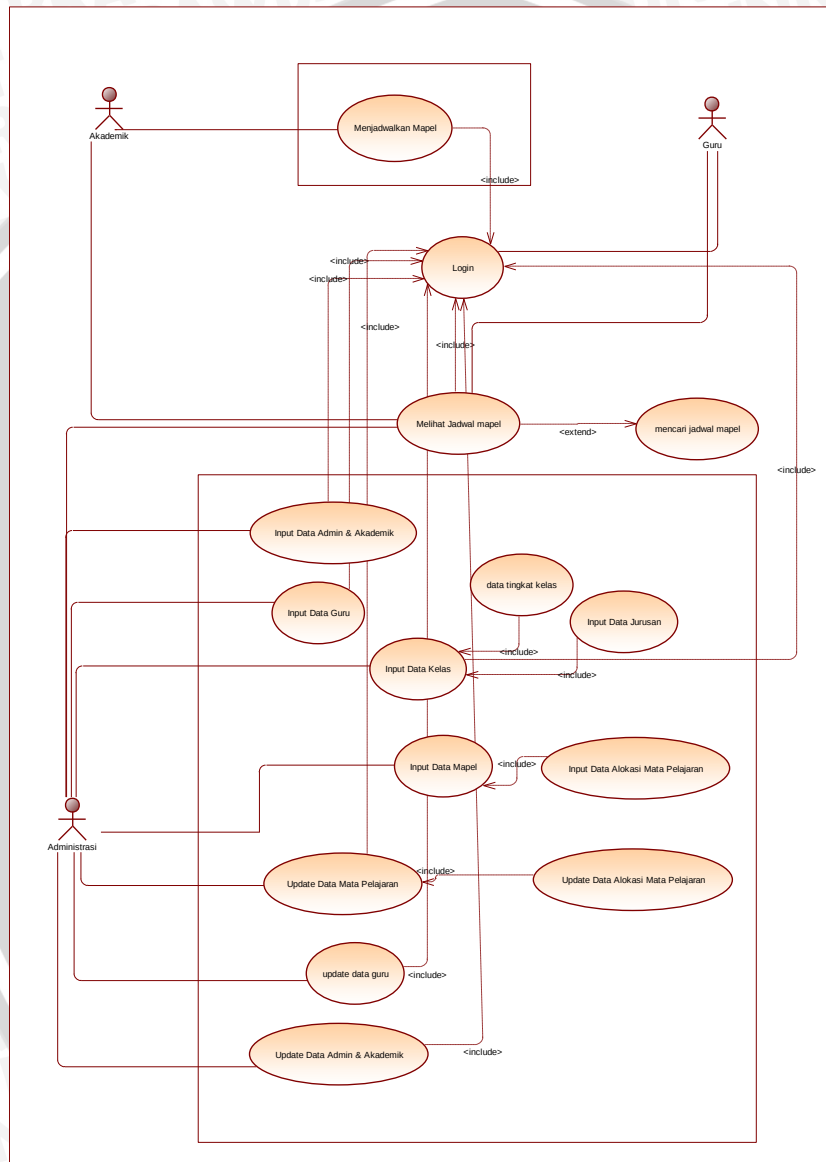
Sistem optimasi penjadwalan ini memiliki tiga aktor yang dapat mengakses sistem. Masing-masing aktor memiliki hak akses yang berbeda. Fungsi dari *user* Administrasi yaitu:

- Melihat, menambah, mengubah dan menghapus data *user*.
- Mengubah status aktif *user* akademik dan administrasi.
- Melihat, menambah, mengubah dan menghapus data alokasi jam mengajar guru.
- Melihat, mencari, menambah dan menghapus data kelas.

Sedangkan fungsi dari user Akademik yaitu:

- Melihat dan mengubah parameter penjadwalan
- Membuat jadwal mata pelajaran.
- Melihat, mencari data jadwal mata pelajaran.

User Guru hanya dapat melihat dan mencari jadwal mata pelajaran yang telah *fix*. Keterangan mengenai hal ini dijelaskan lebih lanjut pada use case diagram yang terdapat pada Gambar 4.1.



Gambar 4.1 Use Case Diagram Sistem Penjadwalan Mata Pelajaran

4.2.2 Analisa Kebutuhan Masukan

Sistem penjadwalan mata pelajaran ini memiliki masukan berupa data pendukung penjadwalan yang dimasukkan oleh Administrasi ke sistem. Data masukan inilah yang akan diproses untuk menghasilkan jadwal mata pelajaran di SMKN 2 Singosari. Data masukan tersebut adalah sebagai berikut:

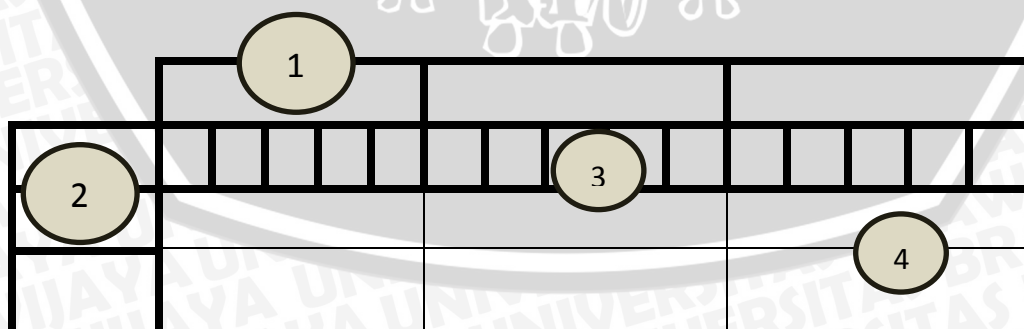
1. data mata pelajaran
2. data user
3. data hari
4. data alokasi mata pelajaran
5. data level *user*
6. data tahun ajaran
7. data semester
8. data jurusan
9. data guru mata pelajaran
10. data tingkat kelas
11. data kelas

4.2.3 Analisa Kebutuhan Proses

Proses inti yang dilakukan yaitu proses optimasi jadwal menggunakan metode tabu search. Dari data masukan dan batasan, dibuat jadwal mata pelajaran awal yang disimpan dalam bentuk array 3 dimensi dan dianggap sebagai solusi awal. Data solusi awal ini dicari total penaltinya, kemudian akan dilakukan penyusunan kembali jadwal mata pelajaran tersebut menggunakan algoritma tabu search sesuai dengan parameter penjadwalan yang telah ditentukan. Setiap iterasi akan membangkitkan 6 alternatif sebagai implemetasi dari proses diversifikasi. Proses intensifikasi dilakukan dengan membuat *forbidden list*, yaitu daftar move yang nilai *best so far*-nya lebih buruk daripada nilai GlobalMin solusi saat ini. TabuList dibuat dalam bentuk array 2 dimensi.

4.2.4 Analisa Kebutuhan Keluaran

Data keluaran berupa tabel hasil penjadwalan mata pelajaran menggunakan metode tabu search dengan total penalti seminimal mungkin. Hasil keluaran yaitu tabel jadwal mata pelajaran yang terdiri dari nama kelas, kode mata pelajaran, dan kode guru, serta keterangan tahun ajaran dan semester. Hasil penjadwalan mata pelajaran disimpan dalam bentuk array 3 dimensi dapat dilihat pada Gambar 4.3 :



Gambar 4.2 Bentuk Array Penyimpanan untuk Hasil Penjadwalan Mata Pelajaran

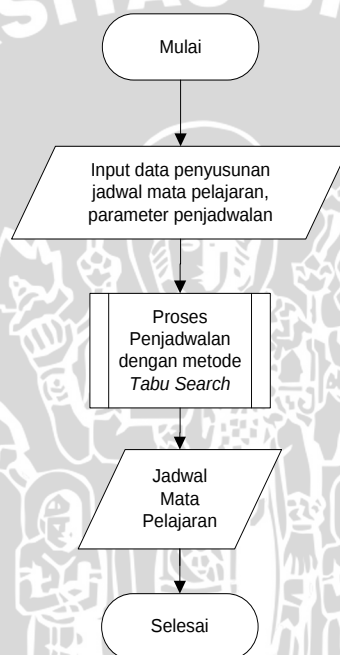
Keterangan :

- 1 : index data hari
- 2 : index data kelas
- 3: index data slot waktu
- 4: data index mata pelajaran dan guru

4.3 Perancangan Proses

4.3.1 Flowchart Sistem

Proses penjadwalan mata pelajaran pada penelitian ini terdiri dari beberapa tahap, yaitu *input* parameter penjadwalan, *input* data untuk pembuatan jadwal mata pelajaran, dan proses penjadwalan menggunakan metode *Tabu Search*. Diagram alir penjadwalan mata pelajaran ditunjukkan pada Gambar 3.4.



Gambar 4.3 Diagram Alir Penjadwalan Mata Pelajaran

Dari Gambar 4.3 diketahui terdapat dua macam masukan, yang pertama adalah berupa data untuk penyusunan jadwal mata pelajaran dan yang kedua adalah parameter penjadwalan yang berupa jumlah iterasi maksimal. Selanjutnya dilakukan proses penjadwalan menggunakan metode *Tabu Search*. Hasil akhir berupa tabel jadwal mata pelajaran dengan nilai penalti paling kecil berdasarkan dengan input parameter iterasi yang dimasukkan pengguna.

4.3.2 Perancangan DFD Sistem

Tampilan DFD Level 1 dari sistem ini dapat dilihat pada Gambar 4.4. DFD yang digunakan adalah DFD Level 1 dari sistem penjadwalan mata pelajaran.

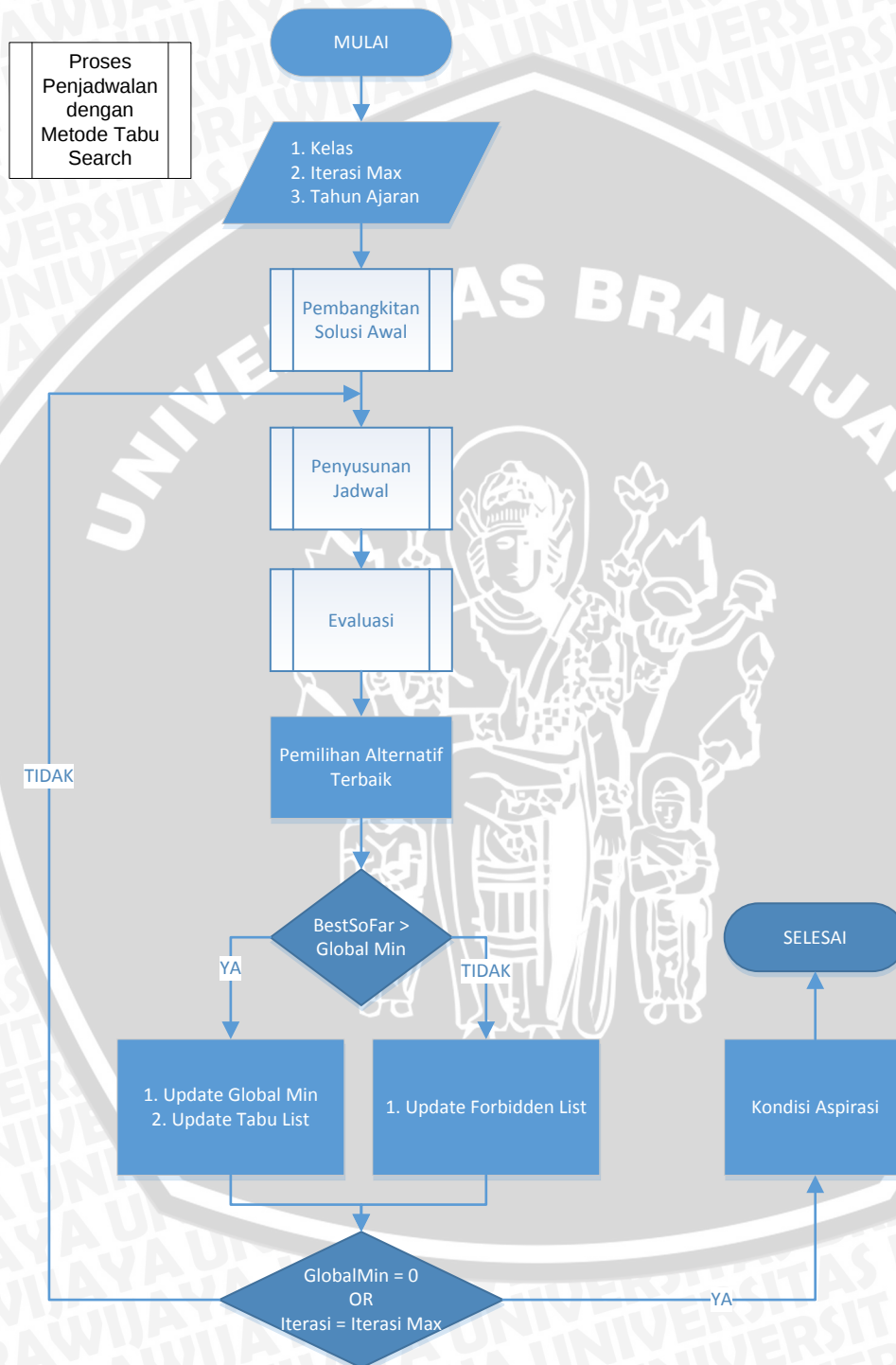


Aplikasi penjadwalan ini digunakan untuk optimasi penjadwalan mata pelajaran di SMK Negeri 2 Singosari. Metode *tabu search* digunakan untuk mempercepat proses penjadwalan yang selama ini berjalan secara manual menggunakan Microsoft Excel. Gambaran konsep arsitektur sistem ditunjukkan pada Gambar 4.5.



4.4.1 Proses Penjadwalan dengan Metode Tabu Search

Proses membangkitkan jadwal menggunakan algoritma Tabu Search dapat dilihat pada Gambar 4.6 yang berisi *flowchart* penjadwalan menggunakan metode *tabu search*.



Gambar 4.6 Diagram Alir Proses Penjadwalan dengan Metode Tabu Search

Dari Gambar 4.6, masukan berupa data parameter penjadwalan. Selanjutnya dibangkitkan solusi awal. Lalu dilakukan iterasi sebanyak iterasi maksimal. Satu kali proses iterasi akan membangkitkan 6 alternatif. Alternatif adalah solusi yang dihasilkan dengan membedakan awal hari untuk penyusunan jadwal. Selama proses optimasi, pada setiap iterasi, solusi akan dievaluasi. BestSoFar adalah nilai total penalti di tiap iterasi, sedangkan GlobalMin adalah nilai total penalti sampai akhir iterasi max. Solusi tiap iterasi dengan BestSoFar terkecil akan dimasukkan ke dalam *tabu list*. TabuList berukuran maksimal 5 solusi. Apabila sudah tidak ada lagi solusi yang tidak menjadi anggota *tabu list*, maka nilai terbaik yang baru saja diperoleh merupakan solusi sebenarnya. Proses optimasi akan dihentikan ketika GlobalMin=0 atau jumlah iterasi telah mencapai jumlah iterasi Max. Kondisi aspirasi merupakan kondisi dimana solusi terbaik masuk ke *Tabu List*, sehingga harus dikeluarkan dari *tabu list* terlebih dahulu agar menjadi solusi terbaik.

Intensifikasi pada proses penjadwalan dilakukan dengan cara memasukkan solusi yang nilai BestSoFar-nya di bawah GlobalMin ke dalam *Forbidden List*. *Forbidden List* ini akan di-reset ketika hasil solusi selanjutnya lebih baik daripada GlobalMin. Proses diversifikasi dilakukan dengan membangkitkan 6 alternatif tiap iterasinya.

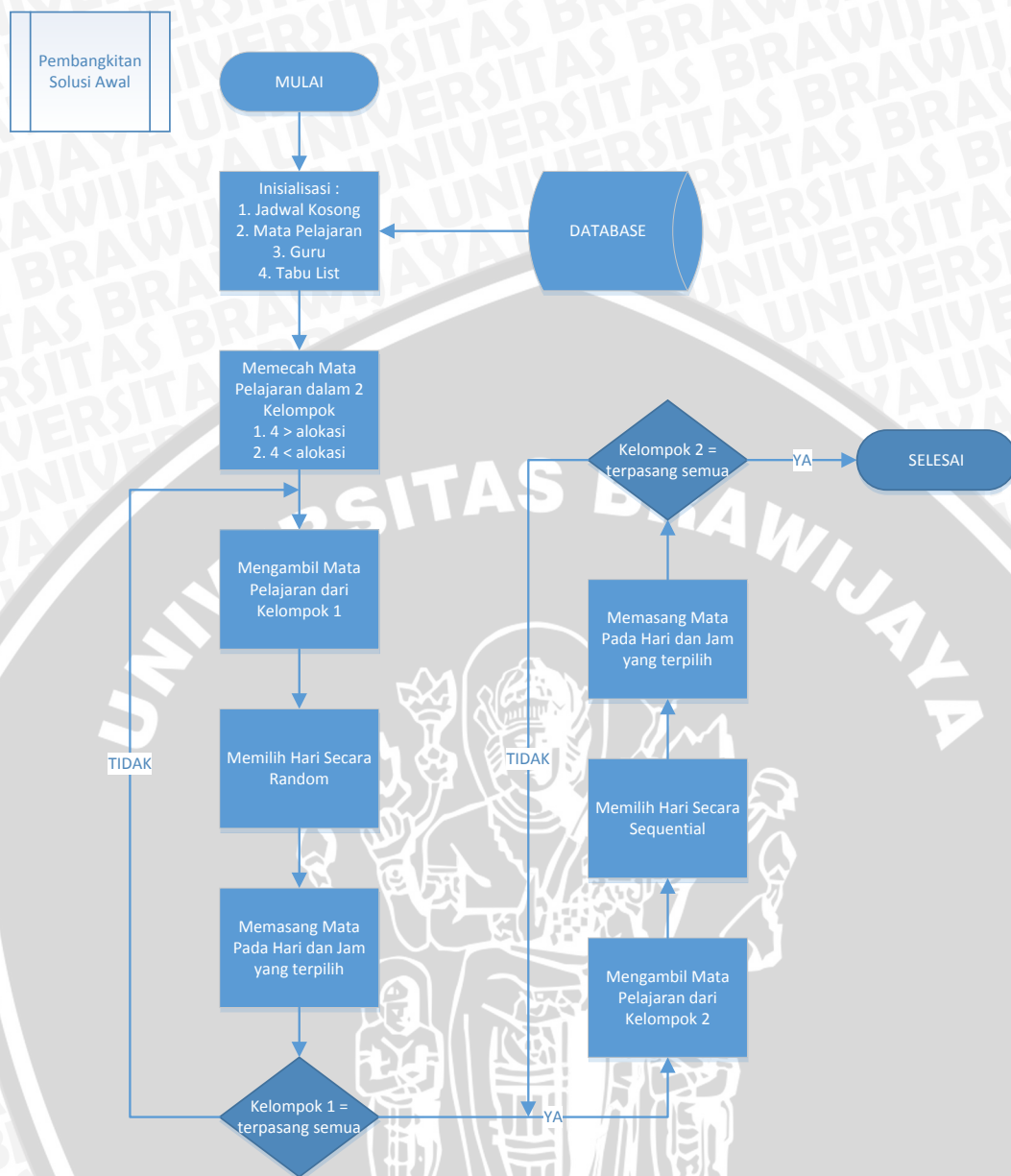
Gambar 4.7 menunjukkan diagram alir proses pembangkitan solusi awal jadwal mata pelajaran. Gambar 4.8 menunjukkan diagram alir proses penyusunan jadwal, dan Gambar 4.9 menunjukkan diagram alir proses evaluasi penjadwalan mata pelajaran.

4.4.2 Constraint

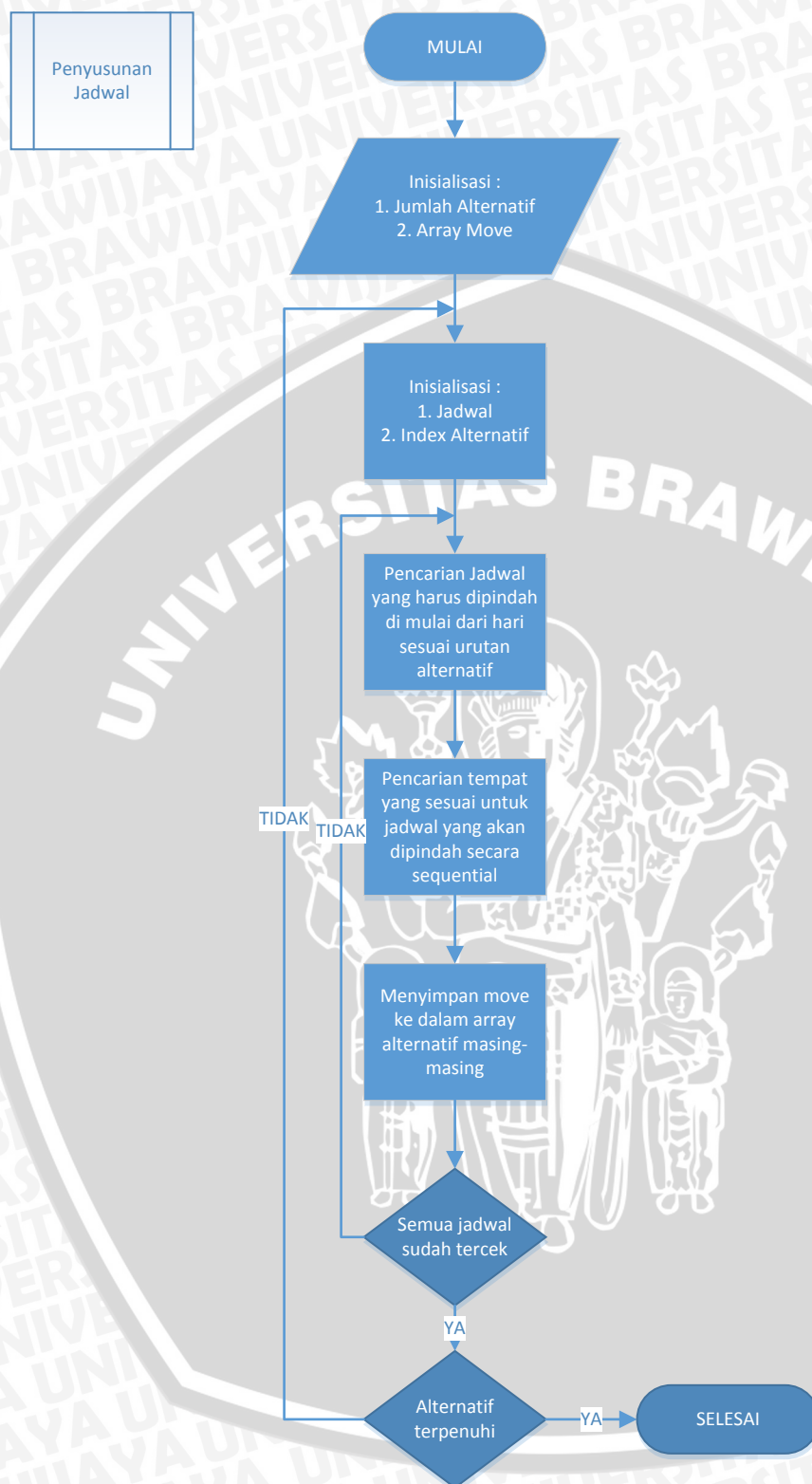
Data *constraint* digunakan sebagai penentu kualitas dari solusi hasil penjadwalan mata pelajaran dalam sistem yang akan dibangun. Data *constraint* yang digunakan dalam sistem ini ditunjukkan pada Tabel 4.4.

Tabel 4.1 Data Constraint

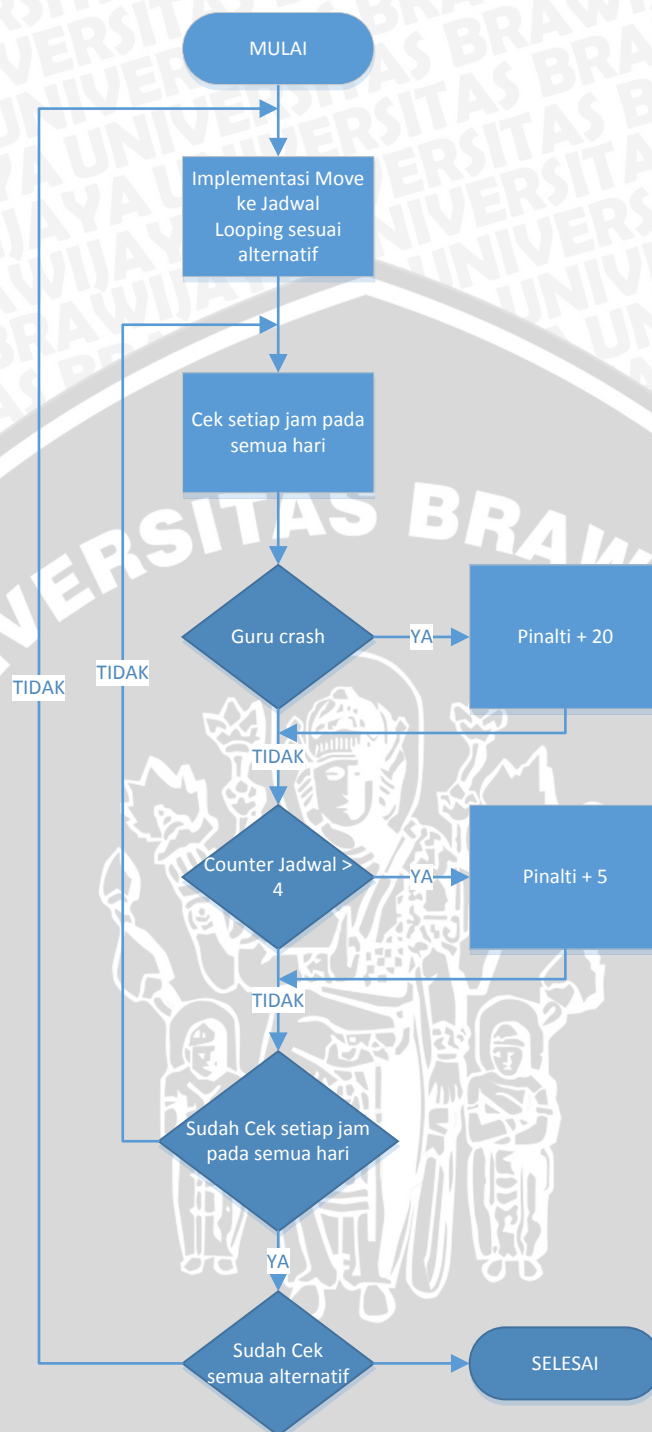
Constraint	Penalti
Bentrok Guru Mengajar Kelas Lain	20
Mata Pelajaran Diajarkan Lebih dari 4 Jam Mata Pelajaran Dalam Satu Hari	2



Gambar 4.7 Diagram Alir Proses Pembangkitan Solusi Awal Jadwal Mata Pelajaran



Gambar 4.8 Diagram Alir Proses Penyusunan Jadwal Mata Pelajaran



Gambar 4.9 Diagram Alir Proses Evaluasi Penjadwalan Mata Pelajaran

4.4.3 Perhitungan Manual

Perhitungan manual ini akan menggunakan 5 kelas untuk disusun jadwal mata pelajarannya. Tabel 4.1 menunjukkan daftar guru , jumlah jam mengajar, mata pelajaran, dan alokasi jam mata pelajaran untuk tiap kelas.

Tabel 4.2 Alokasi Mata Pelajaran dan Guru

Kelas	Mata Pelajaran		Jumlah Jam Pelajaran	Jumlah Jam Per Blok Mata Pelajaran	Kode Guru Mata Pelajaran
	Kode	Nama Mapel			
X AV X RPL X TKJ X MT X EI	OR	Olahraga	3	3	05
	JAWA	Bahasa Jawa	2	2	119
	BIND	Bahasa Indonesia	4	2	105
	MAT	Matematika	4	2	107
	BK	Bimbingan & Konseling	1	1	68
	KWU	Kewirausahaan	2	2	58
	AGM	Agama	3	3	108
	FIS	Fisika	2	2	82
	ING	Bahasa Inggris	2	2	88
	SI	Sejarah Indonesia	2	2	24
	SB	Seni Budaya	2	2	37
	PKN	Pendidikan Kewarganegaraan	2	2	106
X AV X EI X MT	KIM	Kimia	2	2	09
X AV	PROD AV	Mata Pelajaran Produktif AV	20	4	13
				4	17
				4	18
				4	69
X EI	PROD EI	Mata Pelajaran Produktif EI	20	4	04
				4	16
				4	19
				4	52
				4	

Tabel 4.1 Alokasi Mata Pelajaran dan Guru (lanjutan)

X MT	PROD MT	Mata Pelajaran Produktif MT	20	4	117
				4	20
				4	
				4	07
X RPL	PROD TKJ	Mata Pelajaran Produktif TKJ	23	4	03
				4	61
				4	84
				4	22
				4	116
				3	
X RPL	PROD RPL	Mata Pelajaran Produktif RPL	23	4	39
				4	39
				4	10
				4	22
				4	75
				3	

Sumber: Diadaptasi Kurikulum Akademik SMKN 2 Singosari Semester Ganjil Tahun Ajaran 2015/2016 (2015)

1. Membuat tabel jadwal mata pelajaran sesuai dengan kurikulum yang telah ditetapkan. Hari Senin-Kamis memiliki slot waktu sebanyak 10 jam mata pelajaran, sementara slot waktu hari Jumat dan Sabtu sebanyak 6 jam mata pelajaran. Gambar tabel jadwal kosong dapat dilihat pada Gambar 4.10

Kelas	Senin										Selasa										Rabu										Kamis										Jumat						Sabtu					
	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	I	II	III	IV	V	VI										
X AV																																																				
kode guru																																																				
X EI																																																				
kode guru																																																				
X MT																																																				
kode guru																																																				
X TKJ																																																				
kode guru																																																				
X RPL																																																				
kode guru																																																				

Gambar 4.10 Tabel Jadwal Kosong

2. Selanjutnya adalah membangkitkan solusi awal . Proses penyusunan jadwal mata pelajaran diawali dengan memasukkan mata pelajaran produksi beserta kode guru terlebih dahulu untuk memudahkan melakukan penyusunan jadwal. Kemudian dilanjutkan dengan memasukkan mata pelajaran lain yang jumlah bloknnya ganjil (3 dan 1). Selanjutnya mata pelajaran dengan panjang blok genap dimasukkan. Semua mata pelajaran harus dijadwalkan dan tidak boleh ada jam kosong antara satu mata pelajaran dengan mata pelajaran lainnya. Apabila terdapat mata pelajaran yang belum terjadwalkan maka penyusunan jadwal akan diulang sehingga semua mata pelajaran tersusun. Gambar 4.11 menunjukkan tabel jadwal dengan mata pelajaran produksi yang telah dijadwalkan. Gambar 4.12 menunjukkan tabel jadwal mata pelajaran yang telah lengkap dan menjadi solusi awal jadwal mata pelajaran.

Kelas	Senin										Selasa										Rabu										Kamis										Jumat						Sabtu					
	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	I	II	III	IV	V	VI										
X AV	PROD										PROD											PROD											PROD																			
kode guru	13	13	13	13							17	17	17	17							17	17	17	17							18	18	18	18					69	69	69	69										
X EI	PROD										PROD											PROD											PROD																			
kode guru	4	4	4	4							16	16	16	16							19	19	19	19							52	52	52	52					52	52	52	52										
X MT	PROD										PROD											PROD											PROD																			
kode guru	117	117	117	117							117	117	117	117							20	20	20	20							20	20	20	20			7	7	7	7												
X TKJ	PROD										PROD											PROD											PROD																			
kode guru	3	3	3	3							61	61	61	61							84	84	84	84							22	22	22	22			116	116	116	116			116	116	116							
X RPL	PROD										PROD											PROD											PROD																			
kode guru	39	39	39	39							39	39	39	39							10	10	10	10							22	22	22	22			75	75	75	75			75	75	75							

Gambar 4.11 Tabel Jadwal dengan Mata Pelajaran Produksi yang Telah dijadwalkan

27	Kelas	Senin										Selasa										Rabu										Kamis										Jumat						Sabtu							
28		I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	VII	VIII	IX	X	I	II	III	IV	V	VI	I	II	III	IV	V	VI												
29	X AV	PROD				OR		AGM				PROD				BK	BIND		MAT				PROD				KIM	KWU		JAWA		PROD				FIS		ING		SI		PROD				SB		PKN		BIND		MAT			
30	kode guru	13	13	13	13	5	5	5	108	108	108	17	17	17	17	68	105	105	107	107		17	17	17	17	9	9	58	58	119	119	18	18	18	18	82	82	88	88	24	24	69	69	69	69	27	27	106	106	105	105	107	107		
31	X EI	PROD				OR		AGM				PROD				BK	BIND		MAT				PROD				KIM	KWU		JAWA		PROD				FIS		ING		SI		PROD				SB		PKN		BIND		MAT			
32	kode guru	4	4	4	4	5	5	5	108	108	108	16	16	16	16	68	105	105	107	107		19	19	19	19	9	9	58	58	119	119	52	52	52	52	82	82	88	88	24	24	52	52	52	52	27	27	106	106	105	105	107	107		
33	X MT	PROD				OR		AGM				PROD				BK	BIND		MAT				PROD				KIM	KWU		JAWA		PROD				FIS		ING		SI		PROD				SB		PKN		BIND		MAT			
34	kode guru	117	117	117	117	5	5	5	108	108	108	117	117	117	117	68	105	105	107	107		20	20	20	20	9	9	58	58	119	119	20	20	20	20	82	82	88	88	24	24	7	7	7	7	27	27	106	106	105	105	107	107		
35	X TKJ	PROD				OR		AGM				PROD				BIND		MAT		MAT				PROD				KWU		FIS		JAWA		PROD				ING		SI		SB		PROD				PKN		PROD		BIND		BK	
36	kode guru	3	3	3	3	5	5	5	108	108	108	61	61	61	61	105	105	107	107	107		84	84	84	84	58	58	82	82	119	119	22	22	22	22	88	88	24	24	27	27	116	116	116	116	106	106	116	116	105	105	68			
37	X RPL	PROD				OR		AGM				PROD				BIND		MAT		MAT				PROD				KWU		FIS		JAWA		PROD				ING		SI		SB		PROD				PKN		PROD		BIND		BK	
38	kode guru	39	39	39	39	5	5	5	108	108	108	39	39	39	39	105	105	107	107	107		10	10	10	10	58	58	82	82	119	119	22	22	22	22	88	88	24	24	27	27	75	75	75	75	106	106	75	75	105	105	68			

Gambar 4.12 Tabel Solusi Awal Jadwal Mata Pelajaran

- Selanjutnya menghitung nilai total penalti sesuai dengan constraint yang telah ditetapkan sebelumnya. Mata Pelajaran yang gurunya bertabrakan di kelas lain diberi nilai penalti. Kelas paling atas (X AV) dijadikan acuan untuk dibandingkan dengan kelas lain di bawahnya. Bentrok Guru dicek setiap jam mata pelajaran. Buatlah TabuList dengan ukuran maksimal 5. Selengkapnya dapat dilihat pada Gambar 4.13

TOTAL PENALTI KESELURUHAN : 2340

Gambar 4.13 Tabel Solusi Awal Mata Pelajaran Dengan Nilai Penalti

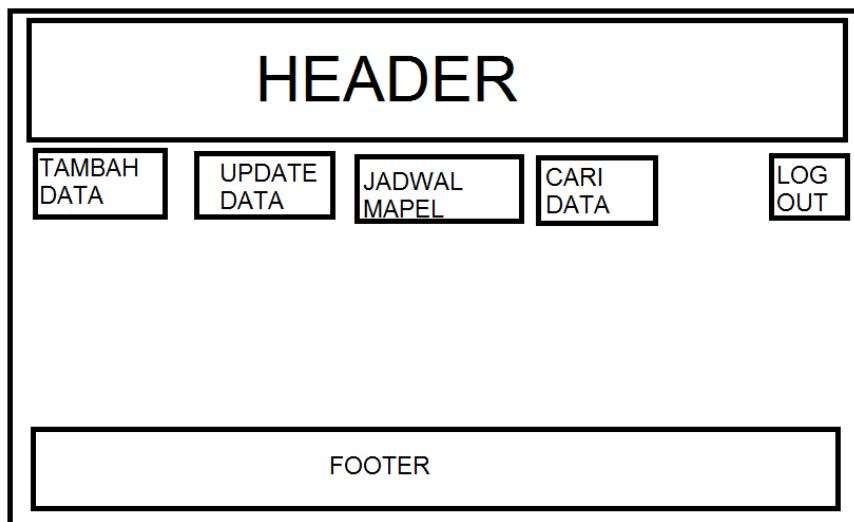
4. Hasil Solusi Awal ini akan dimasukkan TabuList sehingga sisa ruang di Tabu List = 4. Lalu tentukan nilai iterasi max yang akan digunakan untuk proses evaluasi. Selanjutnya dilakukan proses diversifikasi sehingga satu kali proses iterasi akan membangkitkan 6 alternatif jadwal yang berisi move. Alternatif ini merupakan hasil pengolahan jadwal dari solusi awal/ iterasi yang sebelumnya. Proses evaluasi dilakukan dengan mengecek tiap mata pelajaran yang memiliki tabrakan guru dengan jadwal lain pada jam yang sama. Kemudian lakukan proses swap antara mata pelajaran asal yang sama ke mata pelajaran target yang memiliki panjang blok yang sama, dalam satu kelas yang sama dan mata pelajaran target boleh terdiri dari mata pelajaran yang berbeda. Proses swap dilakukan dengan mata pelajaran pada hari selanjutnya (sekuensial). Jika ada mata pelajaran yang bertabrakan dan tidak bisa di-swap dengan mata pelajaran lainnya maka mata pelajaran tersebut akan dilewati. Evaluasi akan dihentikan ketika semua mata pelajaran dalam jadwal telah selesai dievaluasi. Hasil proses iterasi 1 alternatif 0 dapat dilihat pada Gambar 4.14

Gambar 4.14 Tabel Solusi Jadwal Mata Pelajaran Iterasi 1 Alternatif 0

32

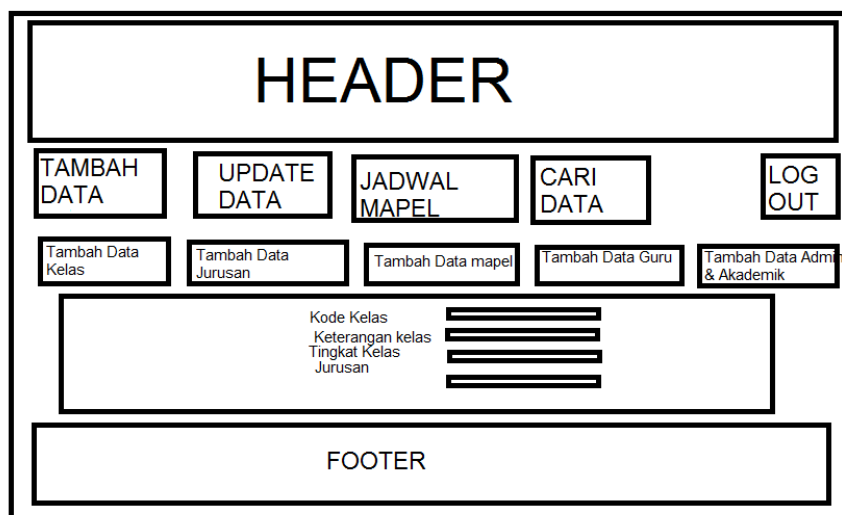
4.5 Rancangan *User Interface*

- Perancangan Antarmuka *Home User Administrasi*



Gambar 4.15 Halaman Perancangan *Home* Program Penjadwalan Melalui *User Administrasi*

- Perancangan Antarmuka *Tab* Tambah Data Kelas



Gambar 4.16 Gambar Perancangan Halaman *Tab* Tambah Data Kelas

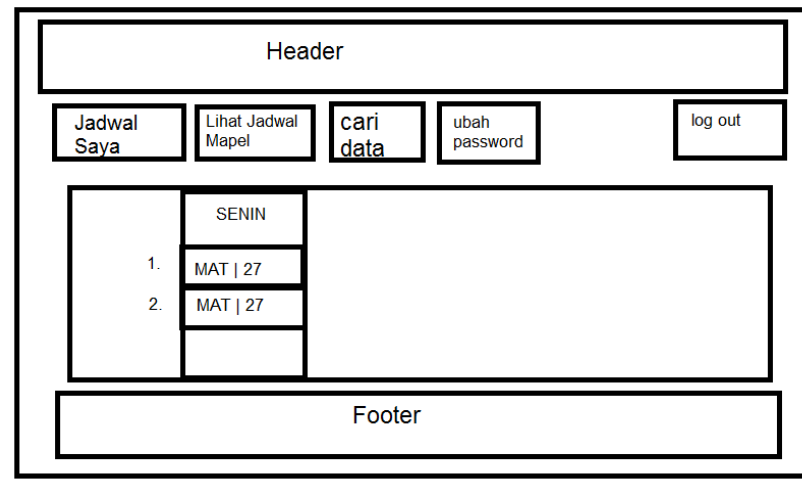
- **Perancangan Antarmuka Hasil Penjadwalan Mata Pelajaran**

Header											
Buat Jadwal Mapel			Lihat Jadwal Mata pelajaran			Ubah Password		Cari Data		Log out	
Main Content Area Simpan Hasil								Total Penalti:			

Footer

34

- Perancangan Antarmuka *Home User* Guru Tab Jadwal Saya



Gambar 4.19 Gambar Perancangan Halaman *Home User* Guru Tab Jadwal Saya

4.6 Perancangan Pengujian

Perancangan untuk pengujian dilakukan dengan mengobservasi parameter yang digunakan algoritma Tabu Search yaitu MaxIterasi dan jumlah kelas. Skenario pengujian MaxIterasi dilakukan dengan cara mengubah parameter jumlah iterasi yang digunakan untuk mengetahui total penalti yang dihasilkan. Masing-masing skenario pengujian dilakukan sebanyak 10 kali. Sementara skenario jumlah kelas dilakukan dengan cara mengubah parameter jumlah kelas yang akan dijadwalkan. Tujuan dari pengujian ini adalah untuk mengetahui pengaruh parameter penjadwalan terhadap hasil penjadwalan.

BAB 5 IMPLEMENTASI

Bab ini membahas mengenai implementasi proses berdasarkan hasil yang telah diperoleh dari analisis kebutuhan dan proses perancangan yang telah dibuat. Pembahasan terdiri dari penjelasan tentang spesifikasi lingkungan uji coba, batasan-batasan dalam implementasi, implementasi algoritma dan implementasi antarmuka/ *interface*.

5.1 Spesifikasi Lingkungan Uji Coba

Hasil analisis kebutuhan dan perancangan aplikasi yang telah diuraikan pada Bab 4 dijadikan acuan untuk membuat aplikasi penjadwalan mata pelajaran. Spesifikasi lingkungan uji coba diimplementasikan pada spesifikasi perangkat keras dan perangkat lunak berikut.

5.1.1 Spesifikasi Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan implementasi optimasi sistem ini adalah sebagai berikut:

1. Prosesor Intel Core i3
2. Memori RAM 2GB
3. Hardisk 250 GB
4. Keyboard
5. Monitor 14"
6. Kartu Grafis Nvidia GeForce

5.1.2 Spesifikasi Perangkat Lunak

Perangkat lunak yang digunakan dalam pengembangan implementasi optimasi sistem ini adalah sebagai berikut:

1. Sistem Operasi Microsoft Windows 7
2. Web Browser Google Chrome
3. Server LocalhostXAMPP 5.5.30-6
4. Tools pemrograman Adobe Dreamweaver CS5 dan Sublime Text
5. DBMS MySQL

5.2 Batasan-Batasan Implementasi

Beberapa batasan dalam mengimplementasikan aplikasi penjadwalan menggunakan metode Tabu Search adalah sebagai berikut :

- Masukan yang diterima oleh aplikasi berupa data-data yang berfungsi sebagai penyusun jadwal mata pelajaran, data tersebut adalah , data mata pelajaran, , data user, data hari, data alokasi mata pelajaran, data guru mata pelajaran, data level *user*, data tahun ajaran, data semester, dan data jurusan
- Keluaran yang diterima oleh pengguna berupa tabel hasil jadwal mata pelajaran yang telah dioptimasi menggunakan metode Tabu Search.
- Tiap proses iterasi akan membangkitkan 6 kandidat.
- Aplikasi yang digunakan berbasis web dengan basis data penyimpanan

menggunakan MySQL.

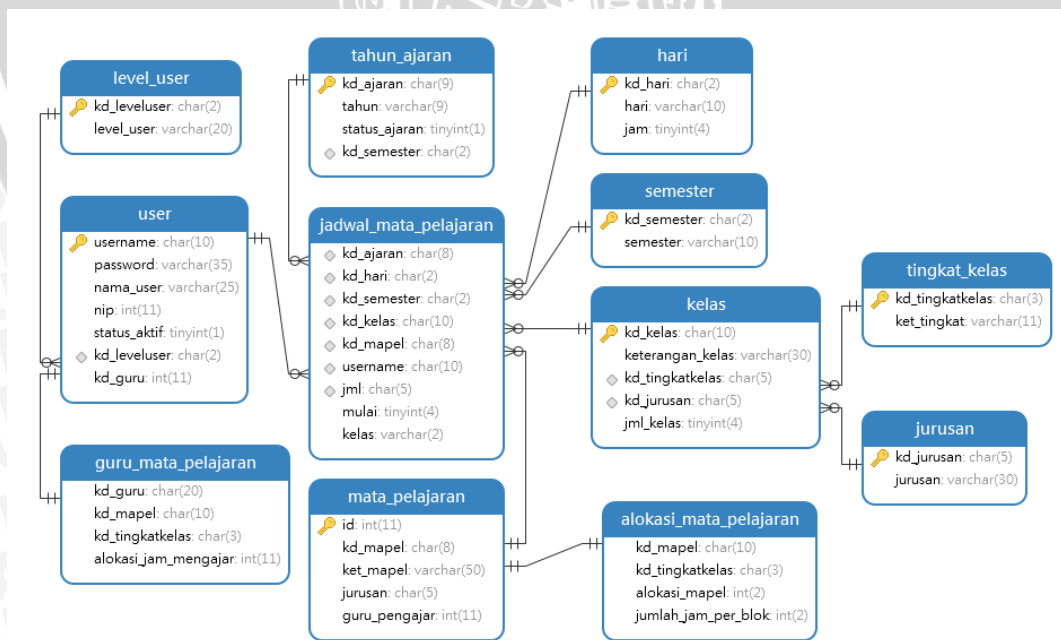
- Aplikasi penjadwalan ini dibuat berdasarkan kurikulum yang berlaku di SMK Negeri 2 Singosari
- Satu guru hanya mengajar satu mata pelajaran saja.
- Penggunaan dengan level *user* Akademik dapat membuat jadwal mata pelajaran, sedangkan pengguna dengan level *user* Administrasi hanya dapat memasukkan data penyusun jadwal mata pelajaran dan melihat hasil jadwal mata pelajaran. Pengguna dengan level *user* Guru hanya dapat melihat hasil jadwal mata pelajaran saja.

5.3 Implementasi Aplikasi Penjadwalan

Hasil perancangan aplikasi penjadwalan mata pelajaran yang telah diuraikan pada Bab dijadikan acuan untuk melakukan implementasi bagian sistemnya. Bagian aplikasi penjadwalan yang diimplementasikan yaitu implementasi basis data, implementasi penjadwalan dengan Tabu Search, dan implementasi antarmuka.

5.3.1 Implementasi Basis Data

Implementasi basis data digunakan untuk melakukan proses penyimpanan data. Proses ini diimplementasikan ke dalam Database Management System MySQL yang berupa script-script SQL. Implementasi basis data yang akan digunakan terdiri dari 24 tabel yang dapat dilihat pada *physical diagram* Gambar 5.1. Terdapat 12 tabel database yaitu tabel tingkat_kelas, jurusan, alokasi_mata_pelajaran, semester, hari, tahun_ajaran, guru_mata_pelajaran, level_user, user, jadwal_mata_pelajaran, mata_pelajaran, dan kelas.



Gambar 5.1 Physical Diagram Program Penjadwalan Menggunakan Metode Tabu Search

5.3.2 Implementasi Penjadwalan Menggunakan Metode *Tabu Search*

Pada implementasi algoritma, akan dibahas penggunaan bahasa pemrograman PHP dan database MySQL dengan mengimplementasikan metode *Tabu Search* untuk proses penjadwalan mata pelajaran SMK.

5.3.2.1 Source Code Penjadwalan dengan Metode *Tabu Search*

Tabel 5.1 berisi source code untuk proses pembangkitan jadwal menggunakan algoritma *Tabu Search*. Baris ke-2 hingga ke-18 digunakan memanggil inisialisasi variabel. Baris ke-20 hingga ke-24 digunakan untuk membuat kerangka penjadwalan. Baris ke 26-33 digunakan untuk mengambil data guru dari database. Baris ke 35-40 digunakan untuk mengurutkan guru sesuai alokasi jam mengajar. Baris ke 42-66 digunakan untuk melakukan penyusunan mata pelajaran dalam array global. Baris ke 68-88 untuk melakukan pemecahan mata pelajaran sesuai jumlah guru. Baris ke 90-93 digunakan untuk pembuatan kerangka jadwal. Baris ke 95-105 untuk melakukan pembagian blok jadwal mata pelajaran dalam *exception* (4 jam mata pelajaran per blok) dan normal. Baris ke 107-129 untuk melakukan proses memasukkan mata pelajaran yang termasuk *exception* ke jadwal. Baris ke 131-174 untuk mencari jam yang kosong. Baris ke 176-234 untuk melakukan penyusunan mata pelajaran dengan jumlah jam per blok 1,2, dan 3 (normal). Baris ke 236-241 digunakan untuk melakukan perhitungan penalti solusi awal. Baris ke 244-250 untuk menampilkan solusi jadwal awal. Baris ke 253-255 digunakan untuk melakukan iterasi sesuai masukan user Akademik. Baris ke 257-266 digunakan untuk membuat move sejumlah 6 alternatif. Baris ke 268-277 digunakan untuk mencari panjang blok jadwal yang akan dipindah. Baris ke 278-290 digunakan untuk mencari blok mata pelajaran yang tidak bertabrakan di sebelah kanan (hari selanjutnya). Baris ke 291-307 digunakan untuk mencari blok mata pelajaran yang tidak bertabrakan di sebelah kiri (hari sebelumnya). Baris ke 309-348 digunakan untuk mencari tempat blok mata pelajaran yang akan dijadikan tujuan *swap*. Baris ke 351-374 digunakan untuk melakukan proses *swap* data mata pelajaran. Baris ke 376-387 digunakan untuk menghitung nilai *cost* (penalti) tiap move pada setiap alternatif. Baris ke 389-391 digunakan untuk menyesuaikan ukuran *Tabu List*. Baris ke 394-396 digunakan untuk mengimplementasikan move pada jadwal yang akan digunakan pada iterasi berikutnya. Baris ke-388 digunakan untuk me-reset *forbidden list*. Baris ke 401 digunakan untuk memasukkan move yang dianggap tidak layak dikunjungi kembali ke *forbidden list*. Baris ke 403-420 digunakan untuk menampilkan solusi akhir penjadwalan mata pelajaran beserta keterangan lainnya.

Tabel 5.1 Source Code Penjadwalan Mata Pelajaran Menggunakan Metode Tabu Search

```

1  <?php
2  $start = microtime(true);
3  $maxIte = $_POST["iterasi"];
4  $kode = "ta".$_POST["semester"].$_POST["tahun"];
5  $t = $_POST["tahun"]+1;
6  $tahun = $_POST["tahun"].'/'.$t;
7  $mainTahun = array("kd_ajaran" => $kode, "tahun" => $tahun,
8  "semester" => $_POST["semester"]);
9  $addKelas = "WHERE ";
10 $delimiter = "";
11 foreach ($_POST["kelas"] as $key => $value) {
12     $addKelas .= $delimiter."kd_kelas = '".$value."'";
13     $delimiter = " OR ";
14 }
15 $GLOBALS["tabulist"] = array();
16 $dataHari = getData("hari");
17 $q4 = mysqli_query($connect,"SELECT * FROM hari");
18 $l = 0;
19 // membuat kerangka penjadwalan
20 $arrJadwal = array();
21 while($data = mysqli_fetch_assoc($q4)){
22     $l++;
23     $arrJadwal[$l] = array_fill(1, $data["jam"], 0);
24 }
25 // mengambil data guru dari database
26 $q2 = mysqli_query($connect,"SELECT * FROM guru_mata_pelajaran");
27 $GLOBALS["guru"] = array();
28 $i = 0;
29 while($data = mysqli_fetch_assoc($q2)){
30     $i++;
31     $data["jml_mengajar"] = 0;
32     $GLOBALS["guru"][$i] = $data;
33 }
34 // pengurutan guru sesuai dengan alokasi jam mengajar
35 aasort($GLOBALS["guru"],"alokasi_jam_mengajar");
36 $GLOBALS["susunan"] = array();
37 $time_start = microtime(true);
38 $query = mysqli_query($connect,"SELECT * FROM kelas ".$addKelas);
39 $intKel = range(1, 20);
40 $GLOBALS['kel_mp'] = array();
41 // penyusunan mata pelajaran disusun dalam array global kel_mapel
42 while($kelas = mysqli_fetch_assoc($query)){
43     $q3 = mysqli_query($connect,"SELECT * FROM mata_pelajaran,
44 alokasi_mata_pelajaran WHERE
45 mata_pelajaran.kd_mapel=alokasi_mata_pelajaran.kd_mapel AND
46 (mata_pelajaran.jurusan='UM' OR
47 mata_pelajaran.jurusan='".$kelas["kd_jurusan"]."') AND
48 alokasi_mata_pelajaran.kd_tingkatkelas =
49 '".$kelas["kd_tingkatkelas"]."");
50     $j = 0;
51     while($mapel = mysqli_fetch_assoc($q3)){
52         $j++;
53         if($kelas["jml_kelas"]>0){
54             for ($kel = 0; $kel < $kelas["jml_kelas"]; $kel++) {
55                 $temp = array();
56                 $temp = [
57                     "kd_mapel" => $mapel["kd_mapel"],
58                     "ket_mapel" => $mapel["ket_mapel"],
59                     "kelas" => $intKel[$kel],
60                     "kd_jurusan" => $kelas["kd_jurusan"],
61                     "kd_tingkatkelas" => $mapel["kd_tingkatkelas"],
62                     "jumlah jam per blok" => (int) $mapel["jumlah jam per blok"]

```

```

63 ];
64 $keyKelas = $temp["kd_jurusan"].' '.$temp["kd_tingkatkelas"].'
65 '.$temp["kelas"];
66 $mainKelas[] = $keyKelas;
67 // pemecahan mata pelajaran sesuai jumlah guru
68 $jml_jam = $mapel["alokasi_mapel"];
69 $lMapel = (int) ceil($jml_jam/$mapel["guru_pengajar"]);
70 for($c=0; $c<$mapel["guru_pengajar"];$c++){
71     $keyMapel = $c+$j;
72     if(($c+1) == $mapel["guru_pengajar"]){
73         $lMapel = $jml_jam;
74         if(($kel+1) == $kelas["jml_kelas"]){
75             $j += $c;
76         }
77     }else{
78         $jml_jam -= $lMapel;
79         $temp["kd_guru"] = getGuru($mapel["kd_mapel"],
80 $mapel["kd_tingkatkelas"], $lMapel, $intKel[$kel],
81 $kelas["kd_jurusan"]);
82         $temp["alokasi_mapel"] = $lMapel;
83         $GLOBALS["kel_mp"][$keyKelas][$keyMapel] = $temp;
84     }
85 }
86 }
87 }
88 }
89 // pembuatan kerangka jadwal
90 foreach ($mainKelas as $mainKey) {
91     $GLOBALS["jadwal"][$mainKey] = $arrJadwal;
92     $GLOBALS["pinalti"][$mainKey] = $arrJadwal;
93 }
94 // pembagian jadwal ke dalam exception dan normal
95 $nMapel = array();
96 $exception = array();
97 foreach ($GLOBALS["kel_mp"] as $key => $value) {
98     foreach ($value as $key2 => $value2) {
99         if($value2["jumlah_jam_per_blok"] >= 4){
100             $exception[$key][] = $key2;
101         }else{
102             $nMapel[$key][] = $key2;
103         }
104     }
105 }
106 //exception
107 foreach ($exception as $keyKelas => $keys) {
108     $jml_hari = count($arrJadwal);
109     shuffle($keys);
110     $jml_mapel = count($keys);
111     foreach ($keys as $index => $key) {
112         $selectedDay = mt_rand(1, $jml_hari);
113         $mapel = $GLOBALS["kel_mp"][$keyKelas][$key];
114 $loop= ceil($mapel["alokasi_mapel"]/$mapel["jumlah_jam_per_blok"]);
115 $total = $GLOBALS["kel_mp"][$keyKelas][$key]["alokasi_mapel"];
116         for ($k=1; $k <= $loop ; $k++) {
117             // penentuan panjang mapel
118             if($k == $loop){
119                 $jam = (int) $total;
120             }else{
121                 $jam = (int) $mapel["jumlah_jam_per_blok"];
122                 $total -= (int) $mapel["jumlah_jam_per_blok"];
123             }
124             $flag1 = true;
125             $flag2 = true;
126             $countLoop = 0;
127

```

```

128         while($flag1 == true){
129             $cJadwal = 0;
130             // mencari jam kosong
131             foreach ($GLOBALS["jadwal"][$keyKelas][$selectedDay] as $xJam =>
132                 $value) {
133                 if($value == 0){
134                     $cJadwal++;
135                     if($cJadwal == $jam) {
136                         break;
137                     }
138                 }
139             }
140             if($cJadwal == $jam){
141                 if(($xJam == 9 AND ($selectedDay != 6 AND $selectedDay != 5) OR
142                     ($xJam == 5 AND ($selectedDay == 6 OR $selectedDay == 5))) AND
143                     $jml_mapel != $index+1){
144                     $flag2 = false;
145                     $selectedDay++;
146                 }else{
147                     $flag1 = false;
148                     for($x = (int)$xJam - $jam + 1; $x <= (int)$xJam; $x++){
149                         $GLOBALS["jadwal"][$keyKelas][$selectedDay][$x] = $key;
150                     }
151                     if($flag2 == false){
152                         $selectedDay--;
153                     }
154                 }
155             }
156             if($flag1 == true AND $flag2 == true){
157                 $selectedDay++;
158             }
159             if($selectedDay > $jml_hari) { $selectedDay = 1; }
160             if($countLoop > $jml_hari){
161                 die(var_dump("JADWAL PENUH EXCEPTION",["kunci" => $keyKelas,
162                     "jadwal" => $GLOBALS["jadwal"][$keyKelas], "jumlah" => $jam]));
163             }
164             if(isset($GLOBALS["jadwal"][$keyKelas][$selectedDay-1][10])
165                 AND $GLOBALS["jadwal"][$keyKelas][$selectedDay-1][10] == 0){
166                 $selectedDay--;
167             }
168             $countLoop++;
169         }
170     }
171 }
172 }
173 $GLOBALS["main"] = $GLOBALS["jadwal"];
174 $loopJadwal = true;
175 // penyusunan mapel normal
176 while($loopJadwal){
177     $loopJadwal = false;
178     $GLOBALS["jadwal"] = $GLOBALS["main"];
179     foreach ($nMapel as $keyKelas => $keys) {
180         $jml_hari = count($arrJadwal);
181         shuffle($keys);
182         $keys = arrangeKeys($keys, $keyKelas);
183         $jml_mapel = count($keys);
184         foreach ($keys as $index => $key) {
185             $mapel = $GLOBALS["kel_mp"][$keyKelas][$key];
186             $loop= ceil($mapel["alokasi_mapel"]/$mapel["jumlah_jam_per_blok"]);
187             $total = $GLOBALS["kel_mp"][$keyKelas][$key]["alokasi_mapel"];
188             for ($k=1; $k <= $loop ; $k++) {
189                 $selectedDay = 1;
190                 if($k == $loop){
191                     $jam = (int) $total;
192                 }else{

```

```

193     $jam = (int) $mapel["jumlah_jam_per_blok"];
194     $total -= (int) $mapel["jumlah_jam_per_blok"];
195     }
196     $flag = true;
197     $countLoop = 0;
198     while($flag == true){
199         $cJadwal = 0;
200         // mencari jam kosong
201     foreach ($GLOBALS["jadwal"][$keyKelas][$selectedDay] as $xJam =>
202     $value) {
203         if($value == 0){
204             $cJadwal++;
205             if($cJadwal == $jam) {
206                 break;
207             }
208         }
209     }
210     if($cJadwal == $jam){
211     if(($xJam == 9 AND ($selectedDay != 6 AND $selectedDay != 5) OR
212     ($xJam == 5 AND ($selectedDay == 6 OR $selectedDay == 5))) AND
213     $jml_mapel != $index+1){
214         $selectedDay++;
215     }else{
216         $flag = false;
217         for($x = (int)$xJam - $jam + 1; $x <= (int)$xJam; $x++){
218             $GLOBALS["jadwal"][$keyKelas][$selectedDay][$x] = $key;
219         }
220     }
221     }else{
222         $selectedDay++;
223     }
224     if($selectedDay > $jml_hari) { $selectedDay = 1; }
225     if($countLoop > $jml_hari){
226         $loopJadwal = true;
227         break;
228     }
229     $countLoop++;
230     }
231     }
232     }
233     }
234 }
235 // penghitungan cost solusi awal
236 fCost();
237 $GLOBALS["globalMin"] = $GLOBALS["data"]["val"];
238 $GLOBALS["detail_main"]["pinalti"] = $GLOBALS["data"]["val"];
239 $GLOBALS["detail_main"]["iterasi"] = "Awal";
240 $GLOBALS["detail_main"]["hard"] = $GLOBALS["data"]["hard"];
241 $GLOBALS["detail_main"]["soft"] = $GLOBALS["data"]["soft"];
242
243 //JADWAL AWAL
244 $GLOBALS["fixed"] = $GLOBALS["jadwal"];
245 $GLOBALS["fixed-pinalti"] = $GLOBALS["pinalti"];
246 $GLOBALS["main"] = $GLOBALS["jadwal"];
247
248 echo "<h5>Jadwal Awal</h5><br/>";
249 print_jadwal($addKelas);
250 $jml_hari = count($arrJadwal);
251
252 // mulai iterasi
253 for ($iteration=1; $iteration <= $maxIte; $iteration++) {
254     $time_start = microtime(true);
255     $GLOBALS["move"] = array();
256     // membuat move sejumlah alternatif
257     for ($alter=0; $alter < $jml_hari ; $alter++) {

```

```

258     $GLOBALS["jadwal"] = $GLOBALS["main"];
259     foreach ($GLOBALS["pinalti"] as $key => $value) {
260     for ($keyMain2=1; $keyMain2 <= count($value) ; $keyMain2++) {
261     $key2 = ($keyMain2 + $alter) > $jml_hari ? $alter : $keyMain2 +
262     $alter;
263
264         $value2 = $value[$key2];
265         $maxKey3 = count($value2);
266         for ($key3=1; $key3<=$maxKey3;$key3++) {
267             $value3 = $value2[$key3];
268             // mencari panjang dari jadwal yang akan dipindah
269             if($value3 > 0 and $GLOBALS["jadwal"][$key][$key2][$key3] !=0){
270             $mapel=$GLOBALS["kel_mp"][$key][$GLOBALS["jadwal"][$key][$key2][$key3]];
271
272                 $flag = true;
273                 $y=1;
274                 $dir = "right";
275                 $max = $key3;
276                 $min = $key3;
277                 while($flag){
278                     //mencari ke arah kanan
279                     if($dir=="right"ANDisset($GLOBALS["jadwal"][$key][$key2][$key3+$y])
280 ) {
281
282                         if($GLOBALS["jadwal"][$key][$key2][$key3+$y]==
283 $GLOBALS["jadwal"][$key][$key2][$key3]){
284                             $max++;
285                         }else{
286                             $dir = "left";
287                         }
288                     }else if($dir == "right" AND
289 !isset($GLOBALS["jadwal"][$key][$key2][$key3+$y])){
290                         $dir = "left";
291                     }
292                     // mencari ke arah kiri
293                     if($dir == "left" AND isset($GLOBALS["jadwal"][$key][$key2][$max-
294 $y])){
295                         if($GLOBALS["jadwal"][$key][$key2][$max-$y] ==
296 $GLOBALS["jadwal"][$key][$key2][$key3]){
297                             $min--;
298                         }else{
299                             $flag = false;
300                         }
301                     }else if($dir == "left" AND
302 !isset($GLOBALS["jadwal"][$key][$key2][$max-$y])){
303                         $flag = false;
304                     }
305                     $y++;
306                 }
307                 $key3 = $max;
308                 $jam = $max - $min + 1;
309             // mencari tempat yang akan dijadikan tujuan pemindahan
310             if($flag == false){
311                 $countDays = 1;
312                 $day = ($key2 + 1) > $jml_hari ? 1 : $key2+1;
313                 $search = true;
314                 while($search){
315                     $tempHari = $dataHari[$day];
316                     $cJadwal = 0;
317                     for ($h=1; $h <= $tempHari["jam"]; $h++) {
318                         $cJadwal++;
319                     }
320                     foreach ($GLOBALS["jadwal"] as $x1 => $v1) {
321                         if($x1 != $key){
322                             $temp=

```

```

323 $GLOBALS["kel_mp"][$x1][$GLOBALS["jadwal"][$x1][$day][$h]]["kd_guru
324 "] : NULL;
325     if($stemp == $mapel["kd_guru"]){
326         $cJadwal = 0;
327         break;
328     }
329     }
330 }
331 if($cJadwal == $jam){
332     $keyMin = $h-$jam+1;
333     if(isset($GLOBALS["jadwal"][$key][$day][$keyMin-1]) AND
334 $GLOBALS["jadwal"][$key][$day][$keyMin] ==
335 $GLOBALS["jadwal"][$key][$day][$keyMin-1]){
336         $cJadwal--;
337     }else if(!isset($GLOBALS["jadwal"][$key][$day][$keyMin-1]))
338         if(isset($GLOBALS["jadwal"][$key][$day][$h+1]) AND
339 $GLOBALS["jadwal"][$key][$day][$h] ==
340 $GLOBALS["jadwal"][$key][$day][$h+1]){
341             $cJadwal--;
342         }
343     }
344     if($cJadwal == $jam){
345         break;
346     }
347 }
348 }
349 if($cJadwal == $jam){
350 // lakukan swap data
351 for ($t=0; $t < $jam ; $t++) {
352     $x1 = $h-$t;
353     $x2 = $max-$t;
354     $GLOBALS["move"][$alter][$key][$key2.' '.$x2] = $day.'
355 '.$x1;
356     $stemp = $GLOBALS["jadwal"][$key][$day][$x1];
357     $GLOBALS["jadwal"][$key][$day][$x1] =
358     $GLOBALS["jadwal"][$key][$key2][$x2];
359     $GLOBALS["jadwal"][$key][$key2][$x2] = $stemp;
360 }
361 $search = false;
362 }else{
363     $day++;
364     if($day > $jml_hari) { $day = 1; }
365     if($countDays == $jml_hari) { $search = false;}
366 }
367 $countDays++;
368 }
369 }
370 }
371 }
372 }
373 }
374 }
375 // hitung nilai cost tiap move pada setiap kandidat
376 fCostEachMove();
377 // mencari key dengan nilai penalti terkecil (best So Far)
378 $keyBest = bestAlternatif();
379 $bestAlternatif = $GLOBALS["calculation"][$keyBest];
380 if($GLOBALS["globalMin"] > $bestAlternatif["val"]){
381     $GLOBALS["detail_main"]["pinalti"] = $bestAlternatif["val"];
382     $GLOBALS["detail_main"]["iterasi"] = "Ke - ".$iteration;
383     $GLOBALS["detail_main"]["kandidat"] = $keyBest;
384     $GLOBALS["detail_main"]["hard"] = $bestAlternatif["hard"];
385     $GLOBALS["detail_main"]["soft"] = $bestAlternatif["soft"];
386     $GLOBALS["globalMin"] = $bestAlternatif["val"];
387     $GLOBALS["tabulist"][] = $GLOBALS["move"][$keyBest];

```

```

388 // penyesuaian ukuran tabulist
389     if ($GLOBALS["sizeTabuList"] < count($GLOBALS["tabulist"])){
390         array_shift($GLOBALS["tabulist"]);
391     }
392 // mengimplementasikan move pada jadwal yang akan digunakan pada
393 iterasi berikutnya
394     $GLOBALS["main"] = implementMove($keyBest);
395     $GLOBALS["fixed-pinalti"] = $bestAlternatif["array"];
396     $GLOBALS["fixed"] = $GLOBALS["main"];
397 // reset fobidden list
398     $GLOBALS["forbidden"] = array();
399     }else{
400 // memasukkan move yang dianggap tidak layak dikunjungi kembali
401     insertForbidden();
402     }
403     $time_exec = (microtime(true) - $time_start);
404     echo "Iterasi - ".$iteration."<br/><br/>";
405     var_dump(implementMove($keyBest));
406     echo "<br/><br/>Nilai penalti total :
407     ".$bestAlternatif["val"]."<br/>";
408     echo "Kandidat yang dipilih : ".$keyBest."<br/>";
409     echo "Soft Constraints : ".$bestAlternatif['soft']. "<br/>";
410     echo "Hard Constraints : ".$bestAlternatif['hard']. "<br/>";
411     echo "Durasi Eksekusi : ".number_format($time_exec, 4). "
412     detik<br/><br/>";
413     echo "<hr/>";
414     if ($GLOBALS["globalMin"] == 0){
415         array_pop($GLOBALS["tabulist"]);
416         break;
417     }else if ($iteration == $maxIte){
418         array_pop($GLOBALS["tabulist"]);
419     }
420 }

```

5.3.2.2 Source Code Proses Pemberian Penalti Pada Penjadwalan Mata Pelajaran

Pada Tabel 5.2 menampilkan *source code* untuk proses pemberian penalti pada penjadwalan mata pelajaran dengan metode *Tabu Search*. Baris ke 3-17 untuk proses pengurutan array. Baris ke 21-65 untk menghitung nilai penalt. Baris ke 35-49 untuk mengecek dan memberi penalti pada pelanggaran *hard constraint*. Baris ke 51-55 untuk mengecek dan memberi penalti pada pelanggaran *soft constraint*. Baris ke 67-80 untuk melakukan proses evaluasi. Baris ke 82-99 untuk menghitung total nilai penalti. Baris ke 101-113 untuk mencari alternatif dengan nilai total penalti paling kecil. Baris ke 115-123 untuk mencari nilai prosentase total nilai penalti.

Tabel 5.2 Source Code Pemberian Penalti Pada Penjadwalan Mata Pelajaran

```

1 <?php
2 // mengurutkan array
3 function aasort (&$array, $key, $order = "DESC") {
4     $sorter=array();
5     $ret=array();
6     reset($array);
7     foreach ($array as $ii => $va) {
8         $sorter[$ii]=$va[$key];
9     }

```

**Tabel 5.2 Source Code Pemberian Penalti Pada Penjadwalan Mata Pelajaran
(lanjutan)**

```

10     if($order == "ASC"){
11         asort($sorter);
12     }else if($order == "DESC"){
13         arsort($sorter);
14     }
15     foreach ($sorter as $ii => $va) {
16         $ret[$ii]=$array[$ii];
17     }
18     $array=$ret;
19 }
20 // mencari nilai kos
21 function fCost(){
22     $source = array();
23     $score = 0;
24     $hard = 0;
25     $soft = 0;
26     $mengajar = array();
27     foreach ($GLOBALS["jadwal"] as $key => $value) {
28         foreach ($value as $key2 => $value2) {
29             $temp = array();
30             foreach ($value2 as $key3 => $value3) {
31                 $GLOBALS["pinalti"][$key][$key2][$key3] = 0;
32                 if($value3 != 0){
33                     $mapel = $GLOBALS["kel_mp"][$key][$value3];
34                     // cek hard constraint
35                     if(isset($mengajar[$mapel["kd_guru"]])){
36                         foreach ($mengajar[$mapel["kd_guru"]] as $k => $v) {
37                             $args = $key2."/".$key3;
38                             if($args == $v){
39                                 $GLOBALS["pinalti"][$key][$key2][$key3] += 20;
40                                 $score += 20;
41                                 $hard++;
42                             }
43                         }
44                     }
45                     $mengajar[$mapel["kd_guru"]][] = $key2.'/'.$key3;
46                     if(!isset($temp[$value3])){
47                         $temp[$value3] = 0;
48                     }
49                     $temp[$value3]++;
50                     // cek soft constraint
51                     if($temp[$value3] >4){
52                         $GLOBALS["pinalti"][$key][$key2][$key3] += 2;
53                         $score += 2;
54                         $soft++;
55                     }
56                 }
57             }
58         }
59     }
60     $GLOBALS["data"] = ["val" => $score, "soft" => $soft, "hard" =>
61     $hard];
62     $source = ["array" => $GLOBALS["pinalti"], "val" => $score, "soft"
63     => $soft, "hard" => $hard];
64     return $source;
65 }
66 // evaluasi data
67 function evaluation(){
68     foreach ($GLOBALS["data"] as $key => $value) {
69         if(isset($min)){

```

```

70         if($min > $value["val"]){
71             $min = $value["val"];
72             $scand = $key;
73         }
74     }else{
75         $min = $value["val"];
76         $scand = $key;
77     }
78 }
79 return [$GLOBALS["data"], "min" => $min, "cand" => $scand];
80 }
81 // hitung pada setiap move
82 function fCostEachMove(){
83     $GLOBALS["calculation"] = array();
84     foreach ($GLOBALS["move"] as $keyAlter => $alter) {
85         foreach ($alter as $keyKelas => $moves) {
86             $GLOBALS["temp"] = $GLOBALS["main"];
87             foreach ($moves as $base => $target) {
88                 $base = explode(" ", $base);
89                 $target = explode(" ", $target);
90                 $tempData = $GLOBALS["temp"][$keyKelas][$base[0]][$base[1]];
91                 $GLOBALS["temp"][$keyKelas][$base[0]][$base[1]] =
92                 $GLOBALS["temp"][$keyKelas][$target[0]][$target[1]];
93                 $GLOBALS["temp"][$keyKelas][$target[0]][$target[1]] =
94                 $tempData;
95             }
96         }
97         $GLOBALS["calculation"][$keyAlter] = fCost();
98     }
99 }
100 // mencari alternatif terbaik
101 function bestAlternatif(){
102     foreach ($GLOBALS["calculation"] as $key => $value) {
103         if(isset($max)){
104             if($max < $value["val"]){
105                 $pickedKey = $key;
106             }
107         }else{
108             $pickedKey = $key;
109             $max = $value["val"];
110         }
111     }
112     return $pickedKey;
113 }
114 // presentasi nilai conflict
115 function precentage($input){
116     $max = "";
117     foreach ($GLOBALS["jadwal"] as $key => $value) {
118         foreach ($value as $key2 => $value2) {
119             $max += count($value2) * 22;
120         }
121     }
122     return 100-((($input/$max)*100);
123 }
124 ?>

```

5.3.3 Implementasi Antarmuka

Antarmuka aplikasi digunakan oleh pengguna untuk berinteraksi dengan aplikasi sehingga pengguna melakukan proses penjadwalan mata pelajaran dan/ melihat hasil jadwal mata pelajaran. Implementasi antarmuka aplikasi ini terbatas hanya pada beberapa antarmuka, diantaranya yaitu :

1. Antarmuka melalui user Administrasi

- Antarmuka halaman tambah data kelas
 - Antarmuka halaman update data kelas
2. Antarmuka pada halaman Akademik
 - Antarmuka halaman buat jadwal
 - Antarmuka halaman jadwal mata pelajaran
 3. Antarmuka melalui *user* Guru
 - Antarmuka halaman home *user* Guru

5.3.3.1 Antarmuka Melalui *User* Administrasi

SMK NEGERI 2 SINGOSARI

Tambah Data Update Data Lihat Jadwal Mata Pelajaran Cari Data

Data Jurusan Data Kelas Data Mata Pelajaran Data Guru Data Administrator & Akademik

Kode Kelas :

Keterangan Kelas :

Data Tingkat Kelas : Kelas 1

Data Jurusan : Audio Video

Jumlah Kelas :

SIMPAN

Gambar 5.2 Halaman Tambah Data Kelas

SMK NEGERI 2 SINGOSARI

Tambah Data Update Data Lihat Jadwal Mata Pelajaran Cari Data

Data Jurusan Data Kelas Data Mata Pelajaran Data Guru Data Administrator & Akademik

Kode Kelas	Keterangan Kelas	Tingkat Kelas	Kode Jurusan	Jumlah Kelas	
X AV	X Audio Video	TK1	AV	2	EDIT DELETE
X EI	X Elektronika Industri	TK1	EI	2	EDIT DELETE
X MM	X Multimedia	TK1	MM	2	EDIT DELETE
X MT	X Mekatronika	TK1	MT	2	EDIT DELETE
X TKJ	X Teknik Komputer Jaringan	TK1	TKJ	2	EDIT DELETE

Gambar 5.3 Halaman Update Data Kelas

SMK NEGERI 2 SINGOSARI

Buat Jadwal Mata Pelajaran | Lihat Jadwal Mata Pelajaran | Ubah Password | Cari Data

Jadwal 2014/2015

Daftar jadwal yang tersimpan :

X Audio Video - 1

-	Senin	Selasa	Rabu	Kamis	Jumat	Sabtu
Jam - 1	PRODUKSI AUDIO VIDEO 14	PRODUKSI AUDIO VIDEO 18	PRODUKSI AUDIO VIDEO 19	PRODUKSI AUDIO VIDEO 18	KEWIRAUSAHAAN 37	PRODUKSI AUDIO VIDEO 18
Jam - 2	PRODUKSI AUDIO VIDEO 14	PRODUKSI AUDIO VIDEO 18	PRODUKSI AUDIO VIDEO 19	PRODUKSI AUDIO VIDEO 18	KEWIRAUSAHAAN 37	PRODUKSI AUDIO VIDEO 18
Jam - 3	PRODUKSI AUDIO VIDEO 14	PRODUKSI AUDIO VIDEO 18	PRODUKSI AUDIO VIDEO 19	PRODUKSI AUDIO VIDEO 18	BAHASA JAWA 11	PRODUKSI AUDIO VIDEO 18
Jam - 4	PRODUKSI AUDIO VIDEO 14	PRODUKSI AUDIO VIDEO 18	PRODUKSI AUDIO VIDEO 19	PRODUKSI AUDIO VIDEO 18	BAHASA JAWA 11	PRODUKSI AUDIO VIDEO 18
Jam - 5	BIMBINGAN DAN KONSELING 3	OLAHRAGA 13	BAHASA INGGRIS 16	BAHASA INDONESIA 1	SENI BUDAYA 5	FISIKA 15
Jam - 6	AGAMA ISLAM 20	OLAHRAGA 13	BAHASA INGGRIS 16	BAHASA INDONESIA 1	SENI BUDAYA 5	FISIKA 15
Jam - 7	AGAMA ISLAM 20	OLAHRAGA 13	MATEMATIKA 12	BAHASA INDONESIA 1	-	-

Gambar 5.6 Tampilan Hasil Akhir Penjadwalan yang Sudah Disimpan

5.3.3.3 Antarmuka Melalui *User Guru*

Jadwal 2016/2017

X Audio Video - 1

-	Senin	Selasa	Rabu	Kamis	Jumat	Sabtu
Jam - 1	-	-	-	-	-	-
Jam - 2	-	-	-	-	-	-
Jam - 3	-	-	-	-	-	-
Jam - 4	-	-	-	-	-	-
Jam - 5	-	-	-	-	BAHASA INGGRIS 16	-
Jam - 6	-	-	-	-	BAHASA INGGRIS 16	-
Jam - 7	-	-	-	-	-	-
Jam - 8	-	-	-	-	-	-
Jam - 9	-	-	-	-	-	-
Jam - 10	-	-	-	-	-	-

X Audio Video - 2

-	Senin	Selasa	Rabu	Kamis	Jumat	Sabtu
Jam - 1	-	-	-	BAHASA INGGRIS 16	-	-

Gambar 5.7 Halaman Jadwal Saya *User Guru*

BAB 6 PENGUJIAN DAN EVALUASI

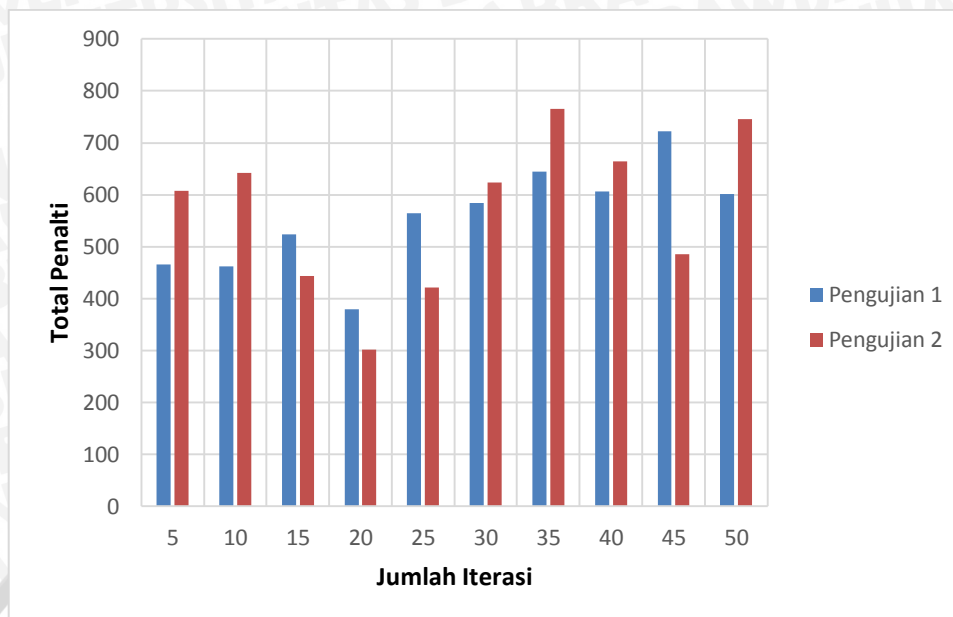
6.1 Hasil Pengujian

Pada hasil pengujian akan menampilkan hasil dari pengujian yang telah dilakukan, yaitu uji MaxIterasi. Dilakukan pengujian dengan parameter jumlah iterasi yang berbeda sebanyak 10 kali dan dilakukan sebanyak 2 set. Tabel hasil pengujian dapat dilihat pada Tabel 6.1 sedangkan grafik hasil pengujian dapat dilihat pada Gambar 6.1.

Tabel 6.1 Hasil Pengujian dengan Parameter MaxIterasi

Jumlah Kelas	Jumlah Iterasi	Total Penalti	
		Pengujian 1	Pengujian 2
8	5	466	608
8	10	462	642
8	15	524	444
8	20	380	302
8	25	564	422
8	30	584	624
8	35	644	766
8	40	606	664
8	45	722	486
8	50	602	746

Hasil dilakukan pada 8 kelas dengan jumlah iterasi sebanyak 10 iterasi dan dilakukan pengujian masing-masing iterasi sebanyak dua kali. Hasil pengujian 1 dan 2 menunjukkan bahwa dengan parameter iterasi dan jumlah kelas yang sama, total penalti dari hasil penjadwalan memiliki perbedaan. Hal ini disebabkan karena untuk setiap iterasi yang dilakukan, metode *Tabu Search* akan membangkitkan solusi awal secara acak yang kemudian akan dievaluasi dan dioptimasi menggunakan metode *Tabu Search*. Hasil pengujian terbaik terdapat pada percobaan ke 4 dengan iterasi 20, pengujian kedua dengan nilai penalti total 302.



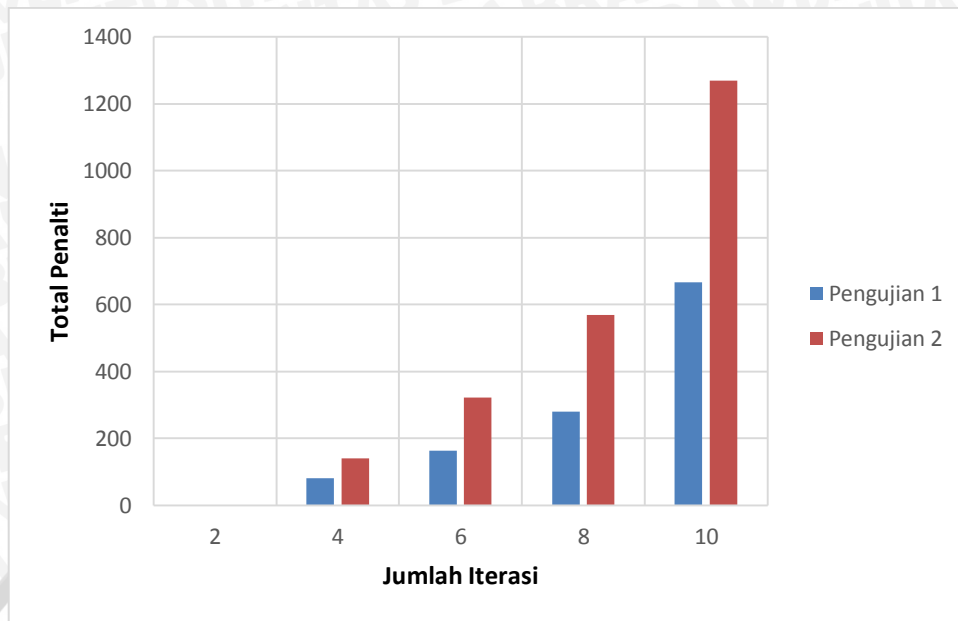
Gambar 6.1 Perbandingan Total Penalti Pada Pengujian 1 dan 2 dengan Parameter MaxIterasi

Pengujian kedua dilakukan dengan nilai iterasi yang sama , namun dengan jumlah kelas yang berbeda. Hasil dari pengujian kedua dapat dilihat pada Tabel 6.2 , dan grafik pengujian kedua dapat dilihat pada Gambar 6.2.

Tabel 6.2 Hasil Pengujian dengan Parameter Jumlah Kelas

Jumlah Kelas	Jumlah Iterasi	Total Penalti	
		Pengujian 1	Pengujian 2
2	10	0	0
4	10	80	140
6	10	162	322
8	10	280	568
10	10	666	1270

Dari hasil pengujian kedua, hasil penjadwalan dengan nilai penalti terbaik terdapat pada percobaan 1, dimana jumlah kelas yang dijadwalkan hanya 2 kelas saja dengan hasil total nilai penalti = 0 baik pada pengujian 1 maupun 2. Pada pengujian selanjutnya, semakin banyak jumlah kelas yang dijadwalkan, maka jumlah total penalti yang didapat juga semakin meningkat. Hal ini disebabkan kompleksitas penyusunan dan sumber daya yang harus dijadwalkan meningkat, sedangkan jumlah guru mata pelajaran tertentu dan slot waktu yang dijadwalkan tetap.



Gambar 6.2 Perbandingan Total Penalti Pada Pengujian 1 dan 2 dengan Parameter Jumlah Kelas



BAB 7 PENUTUP

7.1 Kesimpulan

Berdasarkan hasil implementasi dan pengujian penjadwalan mata pelajaran dengan metode *Tabu Search*, maka dapat diambil kesimpulan bahwa:

1. Hasil dari pengujian menunjukkan bahwa total nilai penalti yang diperoleh dipengaruhi oleh hasil pembangkitan solusi awal jadwal mata pelajaran dan jumlah kelas yang dijadwalkan.
2. Dalam metode *Tabu Search*, solusi awal berupa jadwal dibangkitkan secara random, kemudian dicari solusi akhirnya dan yang menjadi *Tabu List* adalah kumpulan *move* berbentuk *array* yang merupakan solusi jadwal mata pelajaran dengan nilai total penalti paling kecil pada tiap iterasi.

7.2 Saran

Berdasarkan penelitian yang telah dilakukan, saran yang dapat dikembangkan untuk penelitian ini meliputi:

1. Aplikasi ini dapat dikembangkan lebih lanjut dengan menambahkan penalti pelanggaran lainnya, seperti guru yang bersertifikasi harus mengajar minimal 24 jam mata pelajaran.
2. Menggabungkan metode *Tabu Search* dengan metode heuristik lainnya untuk mencapai solusi penjadwalan mata pelajaran yang lebih optimal.

DAFTAR PUSTAKA

- Aladag ,C.H. dan Hocaoglu ,G.. 2007.*A Tabu Search Algorithm To Solve A Course Timetabling Problem*. [e-journal] Tersedia melalui: <<http://dergipark.ulakbim.gov.tr/hujms/article/download/5000017516/5000017780>> [Diakses 20 Juni 2016]
- Fadillah,T.2013.*Sistem Pendukung Keputusan Ujian Komprehensif Menggunakan Algoritma Tabu Search*.S1. Universitas Brawijaya.
- Glover, F. dan Laguna, M. 2013. *Tabu Search*. [pdf] Ming Chuan University. Tersedia melalui: <http://www.cs.mcu.edu.tw/~s9170446/research/Tabu_Search/TABU%20SEARCH.pdf> [Diakses 20 Februari 2016].
- Glover, F. Dan Laguna, M. Dan Marti, R. 2007. *Principles Of Tabu Search*. University Of Colorado, Colorado.
- Gunawan, A; Ong, H.L. dan Ng., K.M. 2004. *Applying Metaheuristics for the Course Scheduling Problem*. [e-journal] Tersedia melalui :<https://www.researchgate.net/profile/Aldy_Gunawan/publication/267836506_APPLYING_METAHEURISTICS_FOR_THE_COURSE_SCHEDULING_PROBLEM/links/55933dc708ae1e9cb42990b9.pdf> [Diakses 3 Maret 2016]
- Ji,M., Tang,H.2004.Global Optimizations and Tabu Search Based on Memory. Dalam: Suyanto.2010. *Algoritma Optimasi Deterministik atau Probabilistik*. Yogyakarta :Graha Ilmu. pp. 138.
- Kamus Besar Bahasa Indonesia dalam jaringan (KBBI daring): <http://bahasa.kemdiknas.go.id/kbbi/index.php>
- Khang, N.T., Nguyen, D.T.T, dan Trieu, Trang . *Using Tabu Search for Solving a High School Timetabling Problem*. [pdf] University of Ho Chi Minh. Tersedia di:<link.springer.com/chapter/10.1007%2F978-3-642-12090-9_26> [Diakses 10 Juni 2016]
- Kusumadewi, S. dan Purnomo, H. 2005. *Penyelesaian Masalah Optimasi Menggunakan Teknik-teknik Heuristik*. Yogyakarta: Graha Ilmu.
- Mushi,A.R. 2006. *Tabu Search Heuristic For University Course Timetabling Problem*. [pdf]University of Dar Es Salaam.Tersedia di : < <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.730.9259&rep=rep1&type=pdf>> [Diakses 10 Juni 2016]
- Peraturan Menteri Pendidikan dan Kebudayaan No.70 Tahun 2013. Jakarta : Kementrian Pendidikan Negara Republik Indonesia.
- Reeves.C.R.,1999.Modern Heuristic Techniques for Combinatorial Problem. Dalam: Suyanto.2010. *Algoritma Optimasi Deterministik atau Probabilistik*. Yogyakarta :Graha Ilmu. pp. 137.

Rohini,V., Natarajan,A.M., 2016. *Comparison of Genetic Algorithm with Particle Swarm Optimisation, Ant Colony Optimisation, and Tabu Search Based on University Course Scheduling System*. [e-journal] Tersedia melalui : <<http://www.indjst.org/index.php/indjst/article/download/85379/70133>> [Diakses 13 Juli 2016]

Setyadi, D. 2012. *Aplikasi Penjadwalan Mata Pelajaran dengan Algoritma Tabu Search (Studi Kasus pada Sanggar*, [online] Tersedia melalui : <repository.uksw.edu/handle/123456789 > [Diakses 5 Maret 2016]

Suyanto.2010. *Algoritma Optimasi Deterministik atau Probabilistik*. Yogyakarta :Graha Ilmu.

Siregar,D.P.2011.*Optimasi Penjadwalan Kuliah dengan Metode Tabu Search*. [e-journal] Tersedia melalui : <[repository.usu.ac.id/bitstream /123456789 /21792/7/Cover.pdf](http://repository.usu.ac.id/bitstream/123456789/21792/7/Cover.pdf)> [Diakses 5 Maret 2016]

Trisnawati,A.,Sangadji,B.M.,Karmila,S.,2011.*Implementasi Tabu Search untuk Penjadwalan Kelas*, [e-journal] Tersedia melalui : <<http://portal.kopertis3.or.id/bitstream/123456789/1344/1/IMPLEMENTASI%20METODE%20TABU%20SEARCH%20UNTUK%20PENJADWALAN%20KELAS.pdf>> [Diakses 10 April 2016]

Valdes, A.R.,Crespo, E., Tamarit, J., 2001. *Tabu Search : An Efficient Heuristic For University Organization Problem*, [pdf] University of Valencia.Tersedia melalui : <<https://rev-inv-ope.univ-paris1.fr/files/22201/IO-22201-3.pdf>> [Diakses 1 Juli 2016]