

SKRIPSI

**PENERAPAN ALGORITMA *PARTICLE SWARM OPTIMIZATION* (PSO)
DALAM PENJADWALAN MATA KULIAH DI JURUSAN TEKNIK
INFORMATIKA UNKHAIR**



**OLEH
Husen Albaar
07351811062**

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS KHAIRUN
TERNATE
2024**

LEMBAR PENGESAHAN

PENERAPAN ALGORITMA *PARTICLE SWARM OPTIMIZATION* (PSO) DALAM PENJADWALAN MATA KULIAH DI JURUSAN TEKNIK INFORMATIKA UNKHAIR

Oleh
Husen Albaar
07351811062

Skripsi ini telah disahkan
Tanggal 24 Juli 2024

Menyetujui
Tim Penguji

Ketua Penguji

MUHAMMAD SABRI AHMAD, S.Kom., M.Kom.
NIP. 198905092019031013

Pembimbing I

Dr. MUHAMMAD RIDHA ALBAAR S.Kom., M.Kom
NIP. 198504232008031001

Anggota Penguji

ACHMAD FUAD, S.T., M.T.
NIP. 197606182005011001

Pembimbing II

Ir. SALKIN LUTFI, S.Kom., M.T.
NIP. 198601112014041002

Anggota Penguji

SYARIFUDDIN N. KARITÁ, S.Pd., M.Si.
NIP. 199103122024211001

Mengetahui/Menyetujui

Koordinator Program Studi
Informatika

ROSIHAN S.T., M.Cs.
NIP. 197607192010121001



Dekan Fakultas Teknik
Universitas Khairun

Ir. ENDAH HARISUN, S.T., M.T., CRP.
NIP. 197511302005011013

LEMBAR PERNYATAAN KEASLIAN

Yang bertanda tangan di bawah ini:

Nama : Husen Albaar
NPM : 07351811062
Fakultas : Teknik
Jurusan/Program Studi : Informatika
Judul : Penerapan Algoritma *Particle Swarm Optimization*
(PSO) Dalam Penjadwalan Mata Kuliah Di Jurusan
Teknik Informatika Unkhair

Dengan ini menyatakan bahwa penulisan Skripsi yang telah saya buat ini merupakan hasil karya sendiri dan benar keasliannya. Apabila ternyata di kemudian hari penulisan Skripsi ini merupakan hasil plagiat atau penjiplakan terhadap karya orang lain, maka saya bersedia mempertanggung jawabkan sekaligus bersedia menerima sanksi berdasarkan aturan tata tertib di Universitas Khairun.

Demikian pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.

Penulis



Husen Albaar

HALAMAN PERSEMBAHAN

Bismillahirrohmannirrahim

Dengan rahmat Allah SWT yang maha pengasih lagi maha penyayang dengan ini saya persembahkan skripsi ini untuk

Keluarga tercinta terutama umi dan aba, terima kasih atas limpahan doa dan kasih sayang yang tak terhingga dan selalu memberikan dorongan baik secara moril maupun material sehingga dapat menyelesaikan skripsi ini.

Dosen-dosen dan staf pengurus prodi informatika yang telah membagikan ilmu dan pengalamannya hingga penulis dapat sampai ke tahap ini.

Teman-teman angkatan 2018, kakak senior dan juga adik junior yang selalu memberikan warna dalam perkuliahan, selalu memberikan dukungan dan semangat untuk sampai ke tahap ini.

MOTTO

“Jangan katakan tidak mungkin,
jika kamu belum mati dalam mencobanya”

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah Subhanahu wata'ala yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga penulis mampu menyelesaikan laporan Skripsi “Penerapan Algoritma *Particle Swarm Optimization (PSO)* Dalam Penjadwalan Mata Kuliah di Jurusan Teknik Informatika Unkhair”

Penyusunan laporan skripsi ini adalah untuk memenuhi salah satu persyaratan kelulusan pada Universitas Khairun Ternate Fakultas Teknik Program Studi Informatika. Penyusunan skripsi ini dapat terlaksana dengan baik berkat dukungan dari banyak pihak, untuk itu pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Dr., M. Ridha Ajam, M.Hum., selaku Rektor Universitas Khairun Ternate.
2. Bapak Ir., Endah Harisun, S.T., M.T., CRP., selaku Dekan Fakultas Teknik Universitas Khairun Ternate.
3. Bapak Rosihan, S.T., M.Cs., selaku Koordinator Program Studi Informatika.
4. Bapak Dr. Muhammad Ridha Albaar S.Kom., M.Kom., selaku pembimbing I yang selalu memberikan motivasi, kritik dan saran yang membangun selama penyusunan skripsi ini.
5. Bapak Ir. Salkin Lutfi, S.Kom., M.T., selaku pembimbing II yang telah meluangkan waktu dan pikiran di tengah kesibukan beliau untuk membimbing dan mengarahkan penulis dalam pengerjaan skripsi ini.
6. Para Dosen Program Studi Informatika Universitas Khairun Ternate yang telah memberikan bekal ilmu kepada penulis.
7. Kedua Orang Tua Tercinta dan seluruh keluarga atas segala bentuk dukungan bimbingan dan doa yang diberikan kepada penulis.
8. Teman-teman seperjuangan terkhususnya Andhika, Rifaldi dan Novit atas dukungan dan bantuannya selama ini.
9. Seluruh angkatan 2018 yang telah memberikan bantuan dan dukungannya dalam penyelesain skripsi ini.
10. Semua pihak yang penulis tidak dapat sebutkan satu persatu yang telah membantu penulis baik langsung maupun tidak langsung dalam menyelesaikan laporan skripsi ini.

Penulis menyadari sepenuhnya bahwa laporan skripsi ini masih jauh dari sempurna, untuk itu semua jenis saran, kritik, dan masukan yang bersifat membangun dari semua pihak terkait sangat penulis harapkan. Akhir kata semoga penyusunan laporan skripsi ini dapat memberikan manfaat dan wawasan tambahan bagi para pembaca dan khususnya bagi penulis sendiri.

Ternate, 24 Juli 2024

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PERNYATAAN KEASLIAN	iii
HALAMAN PERSEMBAHAN	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL	xi
ABSTRAK.....	xiii
 BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Penelitian.....	3
1.5. Manfaat Penelitian.....	4
1.6. Sistematika Penulisan	4
 BAB II TINJAUAN PUSTAKA	
2.1. Penelitian Terkait.....	5
2.2. Penjadwalan.....	7
2.3. Penjadwalan Mata Kuliah	7
2.4. Algoritma	8
2.5. Algoritma Penjadwalan	8
2.5.1. Algoritma Genetika (AG)	8
2.5.2. <i>Ant-Colony Optimization</i> (ACO)	9
2.5.3. <i>Tabu Search</i> (TS)	10
2.5.4. <i>Graph Coloring</i> (GC).....	10
2.5.5. <i>Particle Swarm Optimization</i> (PSO)	11
2.6. <i>Flowchart</i>	15
2.7. <i>Data Flow Diagram</i> (DFD)	15

2.8.	<i>Website</i>	16
2.9.	<i>Database</i>	17
2.10.	<i>MySql</i>	17
2.11.	<i>Hypertext Markup Language (HTML)</i>	17
2.12.	<i>Hypertext Preprocessor (PHP)</i>	18
2.13.	<i>Casecading Style Sheet (CSS)</i>	18
2.14.	<i>Visual Studio Code</i>	18
2.15.	<i>XAMPP</i>	19
2.16.	<i>Prototype</i>	19
2.17.	<i>Pengujian Black Box</i>	21
BAB III METODE PENELITIAN		
3.1.	<i>Metode Prototype</i>	23
3.2.	<i>Metode Pengumpulan Data</i>	24
3.3.	<i>Alat dan Bahan Penelitian</i>	24
3.4.1.	<i>Perangkat Keras (Hardware)</i>	25
3.4.2.	<i>Perangkat Lunak (Software)</i>	25
3.4.	<i>Perancangan Sistem</i>	25
3.5.1.	<i>Sistem Yang Sedang Berjalan</i>	26
3.5.2.	<i>Sistem Yang Diusulkan</i>	27
3.5.3.	<i>Data Flow Diagram (DFD)</i>	28
3.5.	<i>Implementasi Algoritma Particle Swarm Optimization (PSO)</i>	29
3.6.	<i>Desain Interface</i>	37
3.7.	<i>Pengujian Black Box</i>	40
BAB IV HASIL DAN PEMBAHASAN		
4.1.	<i>Pengujian dan Analisis</i>	41
4.2.	<i>Implementasi Sistem</i>	46
4.3.	<i>Pengujian Sistem</i>	50
4.4.	<i>Pengujian Particle Swarm Optimization (PSO)</i>	52
BAB V PENUTUP		
5.1.	<i>Kesimpulan</i>	79
5.2.	<i>Saran</i>	79
DAFTAR PUSTAKA		

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Model PSO.....	12
Gambar 2.2. Model <i>Prototype</i>	20
Gambar 3.1 Metode <i>Prototype</i>	23
Gambar 3.2 Sistem Yang Sedang Berjalan.....	26
Gambar 3.3 Sistem Yang Diusulkan	27
Gambar 3.4 DFD Level 0	28
Gambar 3.5 DFD Level 1	28
Gambar 3.6 Tampilan <i>Login</i>	38
Gambar 3.7 Tampilan Registrasi	38
Gambar 3.8 Tampilan <i>Dashboard</i>	39
Gambar 3.9 Tampilan Jadwal	39
Gambar 3.10 Tampilan <i>User</i>	40
Gambar 4.1 Diagram Hasil Pengujian 1	42
Gambar 4.2 Diagram Hasil Pengujian 2	43
Gambar 4.3 Diagram Hasil Pengujian 3	44
Gambar 4.4 Diagram Hasil Pengujian 4	45
Gambar 4. 5 Diagram Hasil Pengujian 5	46
Gambar 4.6 <i>Flowchart Login</i>	47
Gambar 4.7 Tampilan <i>Login</i>	47
Gambar 4.8 <i>Flowchart</i> Tambah Akun	48
Gambar 4.9 Tampilan Tambah Akun	48
Gambar 4.10 Tampilan <i>Dashboard Admin</i>	49
Gambar 4.11 <i>Flowchart</i> Jadwal	49
Gambar 4.12 Tampilan Jadwal	50
Gambar 4.13 Tampilan Data <i>User</i>	50
Gambar 4.14 Tampilan <i>Login</i> Berhasil	51
Gambar 4.15 Tampilan <i>Login</i> Gagal	51
Gambar 4.16 Tampilan <i>Error</i> Pada Inputan Kosong	52

Gambar 4.17 Grafik Algoritma PSO	76
Gambar 4.18 Tabel Sampel Awal	77
Gambar 4.19 Tabel Sampel Baru	77
Gambar 4. 20 Notifikasi <i>Fitness</i> 1	77
Gambar 4. 21 Notifikasi <i>Fitness</i> 2	78

DAFTAR TABEL

	Halaman
Tabel 2.1. Penelitian Terkait	5
Tabel 2.2. Simbol <i>Flowchart</i>	15
Tabel 2.3. Dimbol DFD	16
Tabel 3.1. Spesifikasi <i>Hardware</i>	25
Tabel 3.2. Spesifikasi <i>Software</i>	25
Tabel 3.3. Data Perkuliahan	29
Tabel 3.4. Slot Perkuliahan	29
Tabel 3.5. Nilai Posisi (x) Awal.....	31
Tabel 3.6. Parameter PSO.....	31
Tabel 3.7. Data Perkuliahan Setelah di Acak	31
Tabel 3.8. Data Perkuliahan Acak di Masukkan ke Dalam Slot	31
Tabel 3.9. <i>Fitness</i>	33
Tabel 3.10. Nilai Kecepatan (v) Awal	33
Tabel 3.11. Nilai Kecepatan (v) Baru	34
Tabel 3.12. Nilai Posisi (x) Baru.....	35
Tabel 3.13. Posisi Data Perkuliahan Baru	36
Tabel 4.1. Hasil Pengujian 1	41
Tabel 4.2. Hasil Pengujian 2	42
Tabel 4.3. Hasil Pengujian 3	43
Tabel 4.4. Hasil Pengujian 4	44
Tabel 4.5. Hasil Pengujian 5	45
Tabel 4.6. Pengujian Halaman <i>Login</i>	51
Tabel 4.7. Pengujian Halaman <i>Menu Data</i>	52
Tabel 4.8. Slot Perkuliahan	53
Tabel 4.9. Nilai Posisi (x) Awal.....	56
Tabel 4.10. Parameter PSO.....	57
Tabel 4.11. Data Perkuliahan Acak di Masukkan ke Dalam Slot	57
Tabel 4.12. <i>Fitness</i>	63

Tabel 4.13. Nilai Kecepatan (v) Awal	64
Tabel 4.14. Nilai Kecepatan (v) Baru	65
Tabel 4.15. Nilai Posisi (x) Baru	68
Tabel 4.16. Posisi Data Perkuliahan Baru	70

ABSTRAK

PENERAPAN ALGORITMA *PARTICLE SWARM OPTIMIZATION* (PSO) DALAM PENJADWALAN MATA KULIAH DI JURUSAN TEKNIK INFORMATIKA UNKHAIR

Husen Albaar¹, Muhammad Ridha Albaar², Salkin Lutfi³

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Khairun

Jl. Jati Metro, Kota Ternate

E-mail: husenalbaar92@gmail.com¹, mridha.albaar84@gmail.com², salkin.lutfi@unkhair.ac.id³

Penjadwalan mata kuliah yang dilakukan di Teknik informatika unkhair umumnya masih bersifat manual dimana penjadwalan yang dilakukan masih menggunakan perangkat lunak *Microsoft Excel* atau *Spreadsheet* untuk mengatur jadwal perkuliahan. Cara manual tersebut akan membutuhkan waktu yang cukup lama dalam mengatur jadwal. Oleh karena untuk memecahkan masalah tersebut diperlukan suatu sistem yang dapat membantu mempermudah dan mempercepat proses pembuatan jadwal mata kuliah di jurusan Teknik Informatika Unkhair. Algoritma yang digunakan untuk penelitian ini yaitu *Particle Swarm Optimization* (PSO). Adapun Algoritma PSO ini dipilih karena merupakan algoritma optimasi yang mudah dipahami, cukup sederhana, dan memiliki unjuk kerja yang sudah terbukti handal. Algoritma dasar PSO terdiri dari tiga tahap, yaitu pembangkitan posisi serta kecepatan partikel, *update velocity* (*update* kecepatan), *update position* (*update* posisi). Implementasi Algoritma *Particle Swarm Optimization* (PSO), belum bisa dikatakan berhasil sepenuhnya diterapkan untuk mengoptimasi permasalahan penjadwalan mata kuliah di jurusan Teknik Informatika Unkhair, karena hanya mendapatkan 1 hasil yang optimal dari beberapa kali percobaan yang dilakukan. Hasil dari penjadwalan mata kuliah menggunakan algoritma PSO yang paling optimal yaitu menggunakan nilai parameter $w=0.9$, $c_1=1$, $c_2=1$ dengan mendapat Tingkat ke optimalan 100%.

Kata kunci: Penjadwalan, Mata kuliah, Algoritma PSO

BAB I

PENDAHULUAN

1.1. Latar Belakang

Universitas merupakan tempat bagi seseorang yang hendak menimba ilmu, baik itu untuk program diploma, sarjana, magister maupun *doctor*. Universitas Khairun (Unkhair) Ternate merupakan salah satu universitas negeri yang berada di Provinsi Maluku Utara yang memiliki berbagai macam jurusan, salah satunya jurusan informatika.

Informatika adalah salah satu jurusan di fakultas teknik yang mempelajari bagaimana penerapan logika matematika dalam pengelolaan informasi yang dapat meliputi transformasi data atau pengolahan fakta-fakta simbolik dengan memanfaatkan teknologi komputer seoptimal mungkin. Mahasiswa jurusan ini juga dibekali keterampilan agar mampu melakukan pemrograman dan komputasi, maupun merancang perangkat lunak.

Penjadwalan adalah suatu proses yang berkaitan dengan pengaturan waktu. Pengaturan waktu sendiri memiliki arti dimana suatu kegiatan diatur sedemikian mungkin agar dapat berlangsung secara lancar dan tidak terjadi bentrok dengan kegiatan lainnya. Untuk itu diperlukan susunan jadwal kegiatan agar dapat terkoordinir dengan baik. Namun kebanyakan orang masih mengabaikan penyusunan jadwal ini dikarenakan prosesnya yang sangat membingungkan dan memakan waktu lama.

Penjadwalan mata kuliah merupakan kegiatan rutin yang dilakukan di Informatika unkhair pada setiap awal semester. Jadwal mata kuliah harus disesuaikan setiap awal semester karena kemungkinan jumlah mahasiswa yang berubah setiap semester. Berdasarkan data yang diperoleh dari *website* PDDIKTI Kemendikbud menunjukan mahasiswa unkhair jurusan informatika pada semester ganjil 2020 berjumlah 745.

mahasiswa, semsester genap 2020 berjumlah 713 mahasiswa, semester ganjil 2021 berjumlah 839 mahasiswa, semester genap 2021 berjumlah 779 mahasiswa, dan semester ganjil 2022 berjumlah 529 mahasiswa. Dengan melihat jumlah mahasiswa yang berubah maka harus menyesuaikan kelas atau ruang yang dibutuhkan.

Berdasarkan hasil wawancara yang penulis lakukan dengan koprodi informatika bapak Rosihan mengatakan bahwa, dalam pembuatan jadwal mata kuliah di Informatika unkhair harus memperhatikan jadwal dosen artinya, pada hari dan jam yang sama dosen tidak memiliki jadwal yang berbeda lebih dari satu, setelah itu harus memperhatikan jadwal ruang yang dimana tidak boleh lebih dari satu mata kuliah yang menggunakan ruang yang sama di hari dan jam yang sama pula, dan harus memperhatikan jadwal mata kuliah untuk satu semester di mahasiswa yang sama agar mahasiswa memilih mata kuliah yang tidak bertabrakan.

Berdasarkan penelitian yang dilakukan oleh Meiton Boru untuk optimasi penjadwalan mata kuliah menggunakan algoritma *Ant Colony Optimization*. Dari salah satu hasil pengujian penjadwalan 38 mata kuliah, 4 angkatan kelas, 6 lokasi perkuliahan dan 12 dosen pengampu mata kuliah, algoritma yang diusulkan dapat memperoleh fitness 0,0092. Tidak terdapat dosen yang memiliki frekuensi mengajar tinggi pada suatu hari, tetapi terdapat 4 mata kuliah yang dijadwalkan berdekatan dan 12 jadwal kuliah yang menyebabkan suatu kelas mengikuti frekuensi perkuliahan yang tinggi (Boru, 2020).

Selama ini, penjadwalan mata kuliah yang dilakukan di Teknik informatika unkhair umumnya masih bersifat manual dimana penjadwalan yang dilakukan masih menggunakan perangkat lunak *Microsoft Excel* atau *Spreadsheet* untuk mengatur jadwal perkuliahan. Cara manual tersebut akan membutuhkan waktu yang cukup lama dalam mengatur jadwal,

karena harus memperhatikan kembali secara teliti jadwal yang sudah dibuat untuk memastikan tidak ada jadwal yang bertabrakan.

Berdasarkan penelitian dan masalah di atas, maka penulis bertujuan untuk membuat penerapan algoritma *Particle Swarm Optimization (PSO)* untuk penjadwalan mata kuliah, untuk menghasilkan suatu sistem komputasi yang dapat menggantikan cara manual tersebut dan dapat menghasilkan hasil yang lebih bagus dan waktu yang lebih singkat. Tugas akhir ini nantinya diharapkan dapat mengatur jam mengajar dosen dan juga jadwal perkuliahan mahasiswa, sehingga menghindari adanya bentrokan jadwal.

1.2. Rumusan Masalah

Berdasarkan latar belakang diatas, maka dapat diidentifikasi masalah yaitu bagaimana membuat penerapan algoritma *Particle Swarm Optimization (PSO)* untuk penjadwalan mata kuliah dan pembagian dosen serta ruang kelas di Prodi Informatika Unkhair.

1.3. Batasan Masalah

Adapun batasan masalah dalam penelitian ini sebagai berikut:

1. Mata kuliah yang akan dijadwalkan dalam penelitian ini hanya jadwal semester genap pada jurusan Teknik Informatika Unkhair.
2. Atribut yang digunakan dalam pembuatan jadwal mata kuliah ini meliputi dosen, mahasiswa, ruang kelas, SKS dan mata kuliah.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk menerapkan algoritma *Particle Swarm Optimization (PSO)* untuk penjadwalan mata kuliah dan pembagian dosen serta ruang kelas di Prodi Informatika Unkhair.

1.5. Manfaat Penelitian

Penelitian ini diharapkan dapat membantu mempermudah dan mempercepat proses pembuatan jadwal mata kuliah di jurusan Teknik Informatika Unkhair.

1.6. Sistematika Penulisan

Dalam suatu laporan perulisan ataupun proposal tentunya wajib menggunakan sistematika penulisan yang benar dan baik. Sehingga laporan dan proposal penilitian iebih efektif dan mudah untuk dipahami oleh pembaca Berikut sistematika penulisan.

BAB I PENDAHULUAN

Di dalam bab I ini yang akan dibahas adalah tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan sitematika penulisan.

BAB II TINJAUAN TEORI

Di dalam bab II ini yang akan dibahas adalah mengenai teori-teori dan penelitian sebelumnya yang berkaitan dengan penilitian yang diangkat.

BAB III METODE PENELITIAN

pembahasan bab III lini membahas mengenai alur penelitian seperti, algoritma yang dipakai, rancangan program, lokasi penelitian, data-data penelitian dan lain-lain.

BAB IV HASIL DAN PEMBAHASAN,

didalam bab IV ini akan ditampilkan hasil dari penelitian serta penjelasan dari hasil tersebut.

BAB V PENUTUP,

bab penutup ini merupakan bab akhir dari sistematika penulisan yang dipakai. Didalam bab ini terdapat kesimpulan dan saran dari semua penjelasan yang ada di bab I-IV.

BAB II

TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian ini bukan yang pertama kali dilakukan namun terdapat beberapa penelitian terkait mengenai topik tentang penjadwalan. Pada bab ini akan dipaparkan beberapa landasan dapat dilihat pada tabel 2.1.

Tabel 2.1 Penelitian Terkait

No	Nama dan Tahun	Judul	Hasil
1	Kuncoro Renaldy Surya (2021)	Optimasi Penjadwalan Jam Kerja Di Masa Pandemi Covid-19 Menggunakan Metode <i>Particle Swarm Optimization</i>	Dengan hasil uji coba perhitungan yang sudah dilakukan pada penelitian ini dengan metode PSO sangat dibutuhkan sekali karena mendapatkan keakuratan yang tinggi dalam menentukan penjadwalan yang ideal untuk para karyawan atau pegawai. Diharapkan dengan adanya penelitian ini dapat membantu banyak perusahaan dan pelaku usaha dalam meningkatkan produktifitasnya dan mengurangi defisit kerugian yang diterima.
2	A. Muhajir nasir (2022)	Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika Di Jurusan Sistem Informasi ITS	Penyusunan jadwal perkuliahan dapat dioptimalkan dengan menerapkan pewarnaan simpul graf dengan algoritma <i>Welch-Powell</i> , sehingga tidak terjadi jadwal mata kuliah yang bentrok. Permasalahan utama dalam penyusunan jadwal perkuliahan adalah ketika ada mahasiswa tingkat bawah yang mengambil mata kuliah ditingkat atas atau sebaliknya.

3	Danu Rizki (2020)	Desain Sistem Basis Data Simulasi Jadwal Ujian Nasional Pada Murid Sekolah Menengah Pertama Menggunakan Algoritma Genetik	Pada Penyelesaian optimasi, sistem penjadwalan pengawas pada Sekolah Menengah Pertama Kota Bengkulu, menggunakan algoritma genetika telah berhasil dilakukan dengan data yang akurat dan efisien.
4	Choirur Roziqin, Imam Cholissodin, Bayu Rahayudi (2019)	Optimasi Penjadwalan Perkuliahan menggunakan <i>Hybrid Discrete Particle Swarm Optimization</i> (Studi Kasus: STAI Ma'had Aly Al-Hikam Malang)	Dijelaskan bahwa implementasi metode <i>Hybrid Discrete Particle Swarm Optimization</i> (HDPSSO) belum dapat diterapkan dalam penjadwalan perkuliahan. Dapat disimpulkan bahwa penelitian ini menggunakan metode HDPSSO belum dapat menghasilkan sebuah solusi yang efisien dan optimal, maka peneliti menggunakan metode PSO yang diharapkan bisa memberikan hasil dalam menentukan solusi terbaik dan optimal.
5	Tri Handayani (2020)	Kajian Algoritma Optimasi Penjadwalan Mata Kuliah	Berdasarkan hasil analisis, adanya temuan bahwa Algoritma <i>Greedy</i> belum mampu memperoleh solusi penjadwalan yang optimal. Hasil yang diperoleh hanya 2% dari nilai optimal. Hal ini dikarenakan jumlah data untuk pengujian yang cukup besar sehingga hasil yang diperoleh kurang maksimal dan membutuhkan waktu cukup lama.
6	Lusiana Paranduk (2018)	Sistem Informasi Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika Berbasis <i>Web</i>	Penerapan algoritma genetika yang digunakan pada penyusunan jadwal mata kuliah khususnya di STMIK PPKIA belum dapat dikatakan optimal. Dikarenakan dari beberapa uji coba yang dilakukan, dari 100 kromosom yang dibangkitkan hanya mendapat 35 solusi yang optimal dengan <i>permutation</i>

			rate sebesar 0,75, <i>mutation</i> sebesar 0.40 dan jumlah generasi 10000.
7	Meiton Boru (2020)	Algoritma <i>Ant Colony Optimization</i> untuk Optimasi Penjadwalan Mata Kuliah	Dari salah satu hasil pengujian penjadwalan 38 mata kuliah, 4 angkatan kelas, 6 lokasi perkuliahan dan 12 dosen pengampu mata kuliah, algoritma yang diusulkan dapat memperoleh fitness 0,0092. Tidak terdapat dosen yang memiliki frekuensi mengajar tinggi pada suatu hari, tetapi terdapat 4 mata kuliah yang dijadwalkan berdekatan dan 12 jadwal kuliah yang menyebabkan suatu kelas mengikuti frekuensi perkuliahan yang tinggi.

2.2. Penjadwalan

Penjadwalan merupakan kegiatan perencanaan untuk menentukan kapan dan dimana setiap operasi sebagai bagian dari pekerjaan secara keseluruhan harus dilakukan dan ada di setiap bidang pekerjaan dari akademis maupun non akademis, namun banyak hal yang harus diperhatikan sebelum membuat jadwal, salah satunya adalah pembuatan jadwal mata kuliah. Jadwal merupakan kumpulan dari beberapa pertemuan di satu waktu tertentu, pertemuan yang mengkombinasikan beberapa hal seperti ruangan, orang yang terlibat dalam kegiatan dan lain sebagainya. (Roihan, 2022).

2.3. Penjadwalan Mata Kuliah

Penyusunan jadwal perkuliahan menjadi bagian penting dari kegiatan yang rutin dilakukan setiap semester pada sebuah perguruan tinggi. Penjadwalan perkuliahan merupakan suatu cara mengalokasikan mahasiswa yang mengambil mata kuliah, dosen yang mengampu mata kuliah dan ruang yang digunakan untuk perkuliahan pada slot waktu

yang tersedia. Kegiatan perkuliahan dilakukan dengan mempertimbangkan jumlah dosen, ruang kelas dan slot waktu yang tersedia. Seringkali pada proses penyusunan jadwal perkuliahan terjadi konflik atau bentrok mata kuliah/dosen/ruang kuliah.(Nasir, 2022).

2.4. Algoritma

Pengertian algoritma adalah suatu metode khusus yang tepat dan terdiri dari serangkaian langkah yang terstruktur dan dituliskan secara matematis, yang akan dikerjakan untuk menyelesaikan suatu masalah dengan bantuan komputer. (Tachir Hasibuan, 2023).

2.5. Algoritma Penjadwalan

Ada beberapa algoritma yang bisa digunakan dalam membuat sistem penjadwalan. Berikut adalah beberapa diantaranya.

2.5.1. Algoritma Genetika (AG)

Algoritma Genetika sebagai cabang dari algoritma Evolusi merupakan metode *adaptive* yang bisa digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah optimasi. Algoritma Genetika menggunakan *analogy* secara langsung dari kebiasaan yang alami yaitu seleksi alam. (Afira, 2021).

Algoritma ini bekerja dengan sebuah populasi yang terdiri dari individu – individu, yang masing–masing individu mempresentasikan sebuah solusi yang mungkin bagi persoalan yang ada. Dalam kaitan ini, individu dilambangkan dengan sebuah nilai *fitness* yang akan digunakan untuk mencari solusi terbaik dari persoalan yang ada.(Afira, 2021).

Kelebihan dari algoritma genetika adalah Algoritma genetika bekerja dengan memanipulasi kode-kode set parameter, bukan dengan hasil manipulasi nilai parameter itu sendiri. Algoritma genetika bebas untuk mengkodekan masalah dengan berbagai cara

sehingga algoritma genetika tidak dibatasi dengan batasan dari metode lainnya. Algoritma genetika menggunakan aturan perpindahan probabilistik, bukan deterministik. (Afira, 2021)

2.5.2. *Ant-Colony Optimization (ACO)*

Ant-Colony Optimization (ACO) termasuk dalam kelompok *Swarm Intelligence*, yang merupakan salah satu jenis pengembangan paradigma yang digunakan untuk menyelesaikan masalah optimasi di mana inspirasi yang digunakan untuk memecahkan masalah tersebut berasal dari perilaku kumpulan atau kawanan (*swarm*) serangga. ACO adalah teknik probabilitas untuk menyelesaikan permasalahan, berdasarkan tingkah laku semut dalam sebuah koloni yang mencari sumber makanan. ACO biasanya digunakan untuk menyelesaikan *discrete optimization problems* dan persoalan yang kompleks dimana terdapat banyak variabel. Hasil yang diperoleh dengan menggunakan ACO, walaupun tidak optimal namun mendekati optimal. (Boru, 2020).

Ant-based techniques pertama kali digunakan oleh Dorigo et. al [1996] dengan menggunakan ACO untuk menyelesaikan *Traveling Salesman Problem (TSP)*. ACO sudah diterapkan di berbagai masalah seperti *Vehicle Routing Problems*, penjadwalan proyek dengan sumber daya terbatas, data mining, penjadwalan pekerjaan (*job scheduling*) dan beberapa masalah kombinatorial yang lain. (Boru, 2020).

Kelebihan algoritma ACO adalah Algoritma semut menggunakan metode umpan balik yang baik sehingga dapat mencapai solusi yang terbaik. Algoritma semut mempunyai sistem kerjasama yang baik. Ditunjukan dengan tingkat keefektifan sebuah pencarian yang melibatkan kerjasama antara koloni semut dalam penyelesaian pada solusi terbaik. Penggunaan struktur yang lebih luas dalam algoritma semut membantu dalam menemukan solusi yang dapat diterima pada tahap proses penelitian. (Boru, 2020).

2.5.3. *Tabu Search* (TS)

Tabu Search adalah sebuah metode optimasi yang berbasis pada *local search*. Proses pencarian bergerak dari satu solusi ke solusi berikutnya, dengan cara memilih solusi terbaik *neighbourhood* solusi sekarang (*current*) yang tidak tergolong solusi terlarang (*tabu*). Ide dasar dari algoritma *tabu search* adalah mencegah proses pencarian dari *local search* agar tidak melakukan pencarian ulang pada ruang solusi yang sudah pernah ditelusuri, dengan memanfaatkan suatu struktur memori yang mencatat sebagian jejak proses pencarian yang telah dilakukan. Kelebihan dari *tabu search* adalah memiliki waktu yang relatif lebih singkat daripada algoritma genetika dikarenakan solusi yang sudah optimal tidak akan dihitung lagi. *Tabu search* juga memiliki kekurangan yaitu terkait dengan penentuan seberapa besar ukuran *tabu list* yang optimal. (Paranduk, 2018)

Menurut *Fred Glover*, *tabu search* memiliki tiga hal pokok yaitu :

1. Penggunaan Struktur memori yang fleksibel, yang memungkinkan evaluasi *criteria* dan informasi historis bisa eksploitasi lebih baik dibandingkan dengan struktur memori yang statis.
2. Penggunaan mekanisme *control* yang didasarkan pada interaksi antara kondisi yang membatasi dan kondisi yang mendukung proses pencarian.
3. Penggabungan fungsi memori yang memiliki rentang waktu yang berbeda, dari memori jangka pendek sampai dengan memori jangka panjang, untuk menerapkan strategi intensifikasi dan diversifikasi.

2.5.4. *Graph Coloring* (GC)

Teori graf merupakan sebuah pokok bahasan yang sudah tua usianya namun memiliki banyak terapan sampai saat ini. Graf digunakan untuk merepresentasikan objek-

objek diskrit dan hubungan antara objek-objek tersebut. Graf pertama kali digunakan untuk memecahkan masalah jembatan *Königsberg* pada tahun 1736. Pada tahun tersebut, seorang matematikawan Swiss bernama L. Euler berhasil memecahkan masalah jembatan *Königsberg* tersebut. Ia memodelkan masalah ini ke dalam bentuk graf dengan daratan (titik-titik yang dihubungkan oleh jembatan) dimodelkan sebagai noktah atau *vertex* dan jembatan dinyatakan sebagai garis atau *edge* (Nasir, 2022).

Definisi sebuah graf adalah sebagai pasangan himpunan (V, E) yang dalam hal ini:

V = himpunan tidak kosong dari simpul-simpul (*vertices* atau *node*) dan

E = himpunan sisi (*edges* atau *arcs*) yang menghubungkan sepasang simpul.

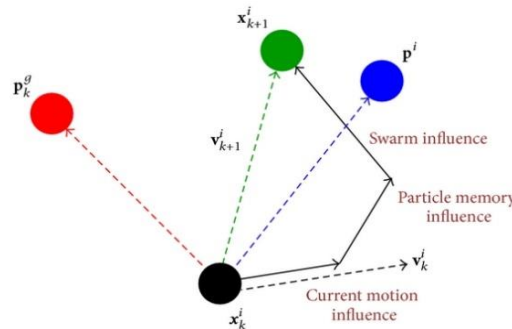
Pewarnaan graf adalah metode pewarnaan elemen sebuah graf yang terdiri dari pewarnaan *vertex* (simpul), sisi (*edge*), dan wilayah (*region*). Penerapan pewarnaan graf dalam kehidupan sehari-hari dapat diterapkan pada penyusunan sebuah jadwal. Sebuah jadwal yang ada mula-mula dipetakan menjadi bentuk graf terlebih dahulu. Proses pewarnaan graf ini nantinya akan dilakukan pada graf yang terbentuk. (Nasir, 2022).

2.5.5. Particle Swarm Optimization (PSO)

Algoritma PSO diperkenalkan oleh Dr. Eberhart dan Dr. Kennedy pada tahun 1995, merupakan algoritma optimasi yang meniru proses yang terjadi dalam kehidupan populasi burung dan ikan dalam bertahan hidup. Cara kerja PSO menyerupai sekumpulan burung atau ikan dalam menemukan sumber makanan dimana setiap individu disebut partikel dan populasi disebut *swarm* (koloni). PSO diinisialisasi dengan sekumpulan partikel sebagai kandidat solusi pada posisi acak. Setiap partikel diberikan posisi awal dan kecepatan awal. Ketika suatu partikel menemukan arah sumber makanan, partikel lainnya akan mengikuti menuju sumber makanan. Salah satu aplikasi PSO adalah untuk menemukan pendekatan

solusi optimal pada masalah optimisasi dengan tidak berkendala maupun dengan kendala.

(Roziqin, 2019) Model PSO dapat dilihat pada gambar 2.1.



Gambar 2.1 Model PSO

PSO memiliki beberapa kesamaan dengan teknik komputasi evolusioner seperti algoritma genetika. Dibandingkan dengan algoritma genetika, PSO memiliki beberapa kelebihan, antara lain mudah diimplementasikan dan memiliki lebih sedikit fungsi operasi dan parameter yang harus ditentukan. Sejak diperkenalkan pertama kali, algoritma PSO berkembang cukup pesat, baik dari sisi aplikasi maupun dari sisi pengembangan metode yang digunakan pada algoritma tersebut. Hal ini disebabkan, algoritma PSO merupakan algoritma optimasi yang mudah dipahami, cukup sederhana, dan memiliki unjuk kerja yang sudah terbukti handal. Algoritma PSO dapat digunakan pada berbagai masalah optimasi baik kontinu maupun diskrit, linier maupun nonlinier. (Muharni, 2019).

Algoritma dasar PSO terdiri dari tiga tahap, yaitu pembangkitan posisi serta kecepatan partikel, *update velocity* (*update* kecepatan), *update position* (*update* posisi). Partikel berubah posisinya dari suatu perpindahan (*iterasi*) ke posisi lainnya berdasarkan pada *update velocity*. Pertama posisi X_k^i , dan kecepatan V_k^i dari kumpulan partikel dibangkitkan secara *random* dengan memasang mata kuliah dengan ruang kelas yang telah disediakan. Nilai awal posisi dan kecepatan awal diinisialisasi sama dan ditentukan secara acak. (Muharni, 2019).

Posisi dan kecepatan direpresentasikan dalam bentuk *vektor* dimana n dimensi *vektor* merepresentasikan jumlah dari desain variabel partikel, dengan *superscript* dan *subscript* menotasikan partikel ke i (Indeks partikel) pada waktu ke k (Iterasi). Dengan proses inisialisasi ini maka kumpulan partikel dapat terdistribusi secara *random* pada *design space*. Vektor seperti ditunjukkan persamaan 2.1 dan 2.2.

$$X_k^i = (X_k^{i1}, X_k^{i2}, \dots, X_k^{in})^T \dots\dots\dots (2.1)$$

$$V_k^i = (V_k^{i1}, V_k^{i2}, \dots, V_k^{in})^T \dots\dots\dots (2.2)$$

Evaluasi *fitness* nilai *fitness* digunakan untuk mengetahui posisi partikel. Baik atau buruknya posisi partikel dilihat dari nilai *fitnessnya*. Faktor-faktor yang mempengaruhi evaluasi *fitness* terhadap *alternative* solusi adalah sebagai berikut:

- a. Banyaknya dosen yang bentrok jam mengajar.
- b. Banyaknya ruang yang bentrok.
- c. Banyaknya mata kuliah yang sama berada pada hari dan jam yang bersamaan.

Dari nilai *fitness* yang didapat akan menentukan pelanggaran – pelanggaran konstrain dari penjadwalan kuliah. Apabila pada tiap - tiap partikel terjadi pelanggaran terhadap konstrain – konstrain diatas maka nilai *fitness* masing – masing partikel akan di-increment sebanyak satu untuk tiap pelanggaran yang terjadi.

Langkah kedua adalah *update velocity* (kecepatan) untuk semua partikel pada waktu $k + 1$ menggunakan fungsi objektif atau nilai *fitness* posisi partikel saat ini pada *design space* saat waktu ke k . Dari nilai *fitness* dapat ditentukan partikel mana yang memiliki nilai global terbaik (*global best*) pada *swarm* saat ini, P_k^g , dan juga dapat ditentukan posisi terbaik dari tiap partikel pada semua waktu yang sekarang dan sebelumnya, P^i . Perumusan *update velocity* menggunakan dua informasi tersebut untuk semua partikel pada kumpulan dengan pengaruh perpindahan yang sekarang, V_k^i , untuk memberikan arah pencarian, V_{k+1}^i , untuk

generasi selanjutnya. Perumusan *update velocity* mencakup beberapa parameter *random*, *rnd*, untuk mendapatkan cakupan yang baik pada *design space*, tiga parameter yang mempengaruhi arah pencarian, yaitu *inertia factor* (*w*), *self confidence* (*c1*), *swarm confidence* (*c2*) akan digabungkan dalam satu penyajian, seperti yang ditunjukkan persamaan 2.3.

$$V_{k+1}^i = W * V_k^i + C1 * Rnd * (P_k^i - X_k^i) + C2 * Rnd * (P_k^g - X_k^i) \dots\dots(2.3)$$

Keterangan:

P_k^i = *Fitness saat iterasi*

P_k^g = *Fitness terbaik*

V_k^i = *Kecepatan*

X_k^i = *Posisi*

i = *Indeks partikel*

k = *Iterasi*

g = *Global best*

Dimana *w* adalah koefisien inersia yang mengontrol kontribusi kecepatan sebelumnya, nilai *w* umumnya berada dalam rentang (0.4 – 1.2). *c1* dan *c2* adalah koefisien yang mengontrol kontribusi posisi terbaik individu dan *global*, nilai *c1* dan *c2* biasanya dalam rentang (1.0 – 2.5). *Rnd* adalah nilai acak antara 0 dan 1 (Rizki, 2021)

langkah terakhir dari setiap iterasi adalah *update* posisi tiap partikel dengan vektor *velocity*, seperti yang ditunjukkan pada persamaan 2.4.

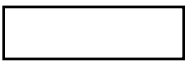
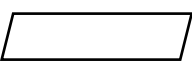
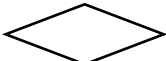
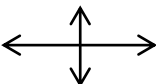
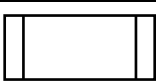
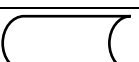
$$X_{k+1}^i = X_k^i + V_{k+1}^i \dots\dots\dots(2.4)$$

Tiga tahapan diatas akan diulang sampai kriteria kekonvergenan terpenuhi, kriteria kekonvergenan sangat penting dalam menghindari penambahan fungsi evaluasi setelah solusi optimum didapatkan, namun kriteria kekonvergenan tidak selalu mutlak diperlukan, penetapan jumlah iterasi maksimal juga dapat digunakan sebagai *stopping condition* dari algoritma. (Rizki, 2021).

2.6. Flowchart

Flowchart merupakan Teknik analitis yang digunakan untuk menjelaskan aspek-aspek sistem informasi secara jelas, tepat dan logis. *Flowchart* menggunakan serangkaian simbol standar untuk menguraikan aliran data dalam sebuah sistem. (Gunawan, 2021) Simbol *flowchart* bisa dilihat pada tabel 2.2.

Tabel 2.2 Simbol *Flowchart*

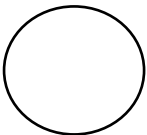
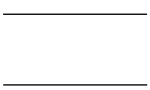
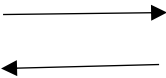
No	Simbol	Fungsi
1.		Terminal, untuk memulai dan mengakhiri suatu proses kegiatan.
2.		Proses, suatu yang menunjukkan setiap pengolahan yang dilakukan oleh komputer.
3.		<i>Input</i> , untuk memasukkan hasil dari suatu proses.
4.		<i>Decision</i> , suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban atau pilihan.
5.		<i>Display, output</i> yang ditampilkan dilayar terminal.
6.		<i>Connector</i> , suatu prosedur akan masuk atau keluar melalui simbol ini dalam lembar yang sama.
7.		<i>Off Page Connector</i> , merupakan simbol masuk atau keluarnya suatu prosedur pada kertas lembar lain.
8.		Arus <i>flow</i> , simbol ini digunakan untuk menggambarkan arus proses dari suatu kegiatan lain.
9.		<i>Hard disk storage, input output</i> yang menggunakan <i>hard disk</i> .
10.		<i>Predifined process</i> , untuk menyatakan sekumpulan langkah proses yang ditulis sebagai prosedur.
11.		<i>Stored data, input, output</i> yang menggunakan disket
12.		<i>Printer</i> , simbol ini digunakan untuk menggambarkan suatu dokumen atau kegiatan untuk mencetak suatu informasi dengan mesin <i>printer</i> .

2.7. Data Flow Diagram (DFD)

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau

sistem baru yang dikembangkan secara logika tanpa memepertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan. DFD merupakan alat yang digunakan pada metdologi pengembangan sistem yang terstruktur. (Gunawan, 2021). Simbol DFD dapat dilihat pada tabel 2.3.

Tabel 2.3 Dimbol DFD

Simbol	Keterangan
	<i>External Entity</i> , merupakan kesatuan di lingkungan luar sistem yang bisa berupa orang, organisasi atau sistem lain.
	Proses, merupakan proses seperti perhitungan aritmatik penulisan suatu formula atau pembuatan laporan
	<i>Data Store</i> , dapat berupa suatu <i>file</i> atau <i>database</i> pada <i>system computer</i> atau catatan manual
	<i>Data Flow</i> , arus data ini mengalir diantara proses, simpan data dan kesatuan luar

2.8. Website

Menurut Rohi Abdulloh *Website* atau disingkat *web*, dapat diartikan sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital baik berupa *text*, gambar, video, audio, dan animasi lainnya yang disediakan melalui jalur koneksi *internet* (Gunawan, 2021).

Menurut Tim *EMS Website* adalah apa yang anda lihat *via browser*, sedangkan yang disebut *web* sebenarnya adalah sebuah aplikasi *web*, karena melakukan *action* tertentu dan membantu anda melakukan kegiatan tertentu (Gunawan, 2021).

2.9. Database

Basis data (*database*) merupakan suatu kumpulan data yang disimpan didalam *computer* atau perangkat lunak yang dapat diolah atau dimanipulasi dan kemudian ditampilkan menjadi sebuah informasi (Andy, 2019).

Database merupakan susunan *record* data operasional lengkap dari suatu organisasi atau perusahaan, yang diorganisir dan disimpan secara terintegrasi dengan menggunakan metode tertentu sehingga mampu memenuhi informasi yang optimal yang dibutuhkan oleh para pengguna (Andy, 2019).

2.10. MySQL

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL (Gunawan, 2021).

2.11. Hypertext Markup Language (HTML)

Hypertext Markup Language (HTML) adalah suatu bahasa yang digunakan untuk membentuk struktur dan konten dari sebuah dokumen yang akan ditampilkan pada sebuah *browser*. Dokumen disimpan dengan ekstensi *.html*, didalamnya terdapat dua bagian utama yaitu *head* dan *body*. Bagian *body* yang biasanya berisi konten yang akan ditampilkan pada browser. HTML itu bahasa yang fleksibel karena tidak tergantung pada suatu *platform* (sistem operasi) tertentu. HTML terdiri dari *tag-tag* yang mendefenisikan elemen tertentu pada sebuah halaman *web*. HTML merupakan bahasa yang tidak *case sensitive*, tidak

seperti bahasa pemrograman *server-side* seperti PHP atau ASP. (Herpendi, 2020).

2.12. Hypertext Preprocessor (PHP)

PHP adalah salah satu bahasa pemrograman *web* yang *open source*, kode PHP dapat digabungkan dengan kode HTML untuk membentuk suatu halaman *web* yang dinamis, dalam konsep *web*, jika suatu *web browser* mengakses file dengan kode HTML murni maka *web server* akan mengirimkan *file* terakses secara utuh tanpa ada proses terlebih dahulu terhadap kode-kode yang digunakan dalam *file* tersebut, berbeda dengan *file* PHP, *web server* akan mengirimkan *file* PHP tersebut ke suatu program kompilasi (dalam hal ini adalah PHP kompilasi) untuk segera diproses, PHP *compiler* tersebut akan mengirimkan hasil prosesnya ke *web server* untuk segera dikirimkan kembali ke program pengakses. Hal ini akan memudahkan pembuat *website* untuk mengembangkan *website* dinamis, dalam hal ini *website* yang dapat merespon apa yang diminta oleh masing-masing program pengakses. (Herpendi, 2020).

2.13. Cascading Style Sheet (CSS)

Cascading Style Sheet (CSS) merupakan aturan untuk mengendalikan beberapa komponen dalam sebuah *web* sehingga akan lebih terstruktur dan seragam. CSS dapat mengendalikan ukuran gambar, warna *body teks*, warna tabel, ukuran *border*, warna *border*, warna *hyperlink*, warna *mouse over*, spasi antar paragraf, spasi antar teks, *margin* kiri/kanan/atas/bawah, dan parameter lainnya. CSS adalah bahasa *style sheet* yang digunakan untuk mengatur tampilan dokumen. CSS memungkinkan kita untuk menampilkan halaman yang sama dengan format yang berbeda (Andy, 2019).

2.14. Visual Studio Code

Visual Studio Code adalah kode editor sumber yang dikembangkan oleh *Microsoft*

untuk *Windows*, *Linux* dan *macOS*. Ini termasuk dukungan untuk *debugging*, kontrol git yang tertanam dan *GitHub*, penyorotan sintaksis, penyelesaian kode cerdas, *snippet*, dan *refactoring* kode. Ini sangat dapat disesuaikan, memungkinkan pengguna untuk mengubah tema, pintasan *keyboard*, preferensi, dan menginstal ekstensi yang menambah fungsionalitas tambahan (Joni, 2019).

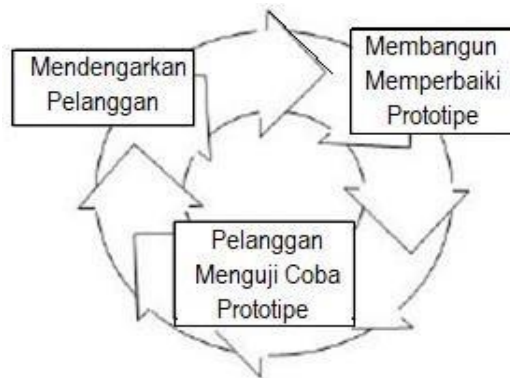
2.15. XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (*localhost*) yang terdiri atas *Apache*, *HTTP*, *Server*, *MySQL*, *Database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan *Perl*. Program ini tersedia dalam *General Public License (GNU)* dan bebas, merupakan *web server* yang mudah digunakan yang dapat melayani tampilan halaman *web* yang dinamis. XAMPP dapat berfungsi sebagai aplikasi untuk mengatur database tanpa koneksi internet. (Nirsal, 2020).

2.16. *Prototype*

Metode pengembangan sistem yang digunakan pada penelitian ini yaitu *prototype*, Sebuah *prototype* adalah versi awal dari sistem perangkat lunak yang digunakan untuk mendemonstrasikan konsep-konsep, percobaan rancangan, dan menemukan lebih banyak masalah dan solusi yang memungkinkan. Sistem dengan model *prototype* memperbolehkan pengguna untuk mengetahui bagaimana sistem berjalan dengan baik. metode *prototyping* yang digunakan di dalam penelitian ini bertujuan untuk mendapatkan gambaran aplikasi yang akan dibangun melalui rancangan aplikasi *prototype* terlebih dahulu kemudian akan dievaluasi oleh *user*. Aplikasi *prototype* yang telah dievaluasi oleh user selanjutnya akan dijadikan acuan untuk membuat aplikasi yang dijadikan produk akhir sebagai *output* dari

penelitian ini. (Maulana et al., 2020) Model prototype dapat dilihat pada gambar 2.2.



Gambar 2.2 Model *Prototype*

Tahapan-tahapan dalam *Prototype* adalah sebagai berikut:

1. Pengumpulan kebutuhan

Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

2. Membangun *Prototype*

Membangun *prototype* dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat *input* dan format *output*)

3. Evaluasi *Prototype*

Evaluasi ini dilakukan oleh pelanggan apakah *prototype* yang sudah dibangun sudah sesuai dengan keinginan pelanggan. Jika sudah sesuai maka langkah 4 akan diambil. Jika tidak *prototype* direvisi dengan mengulangi langkah 1, 2, dan 3

4. Mengkodekan Sistem

Dalam tahap ini *prototype* yang sudah disetujui diterjemahkan ke dalam bahasa pemrograman yang sesuai.

5. Menguji Sistem

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan. Pengujian ini dilakukan dengan *White Box*, *Black Box*, *Basis Path*, pengujian arsitektur dan lain-lain.

6. Evaluasi Sistem

Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan. Jika ya, langkah 7 dilakukan, jika tidak, ulangi langkah 4 dan 5.

7. Menggunakan Sistem

Perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan.

2.17. Pengujian *Black Box*

Black box testing merupakan pengujian kualitas perangkat lunak yang berfokus pada fungsionalitas perangkat lunak. Pengujian *black box testing* bertujuan untuk menemukan fungsi yang tidak benar, kesalahan antarmuka, kesalahan pada struktur data, kesalahan performansi, kesalahan inisialisasi dan terminasi, dalam pengujian *black box testing* digunakan alat untuk pengumpulan data yang disebut dengan *user acceptance test*, dokumen ini terdiri deskripsi indikator dari prosedur-prosedur pengujian fungsionalitas dari perangkat lunak (Gunawan, 2021).

Metode *Black box* memungkinkan perekrut perangkat lunak mendapatkan serangkaian kondisi *input*. *Black Box* adalah suatu pengujian memeriksa suatu aplikasi atau sistem apakah berjalan seperti yang diharapkan atau tidak. Untuk mengetahui kualitas suatu kemampuan aplikasi, mengetahui apakah aplikasi atau sistem memiliki kinerja yang baik atau gagal. (Herpendi, 2020) *Black box* dapat menemukan kesalahan dalam kategori berikut:

1. Fungsi yang tidak benar atau hilang.

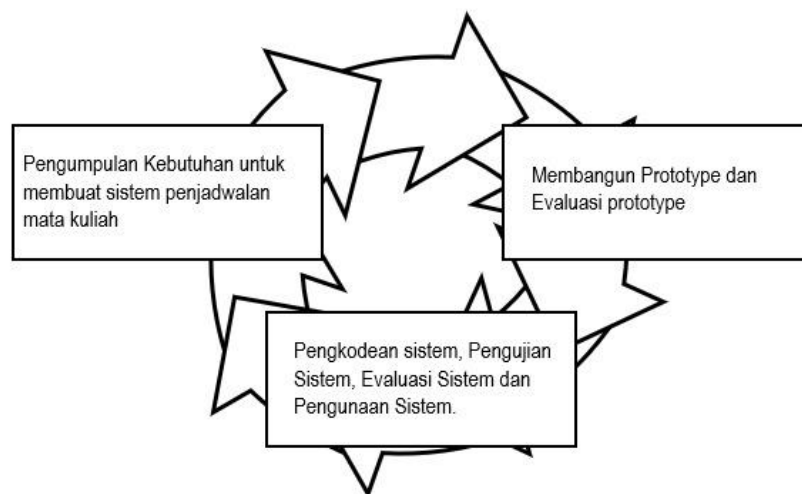
2. Kesalahan dalam struktur data atau akses basisdata eksternal.
3. Inisialisasi dan kesalahan terminasi.
4. Validitas fungsional.
5. Kesensitifan sistem terhadap nilai *input* tertentu.
6. Batasan dari suatu data.

BAB III

METODE PENELITIAN

3.1. Metode *Prototype*

Berikut ini merupakan langkah-langkah dalam penelitian sesuai dengan tahapan metode *ptototype* seperti yang terlihat pada gambar 3.1.



Gambar 3.1 Metode *Prototype*

1. Pengumpulan kebutuhan, pada tahapan ini peneliti mendengarkan kebutuhan-kebutuhan untuk membuat sistem penjadwalan mata kuliah Program Studi Informatika UNKHAIR.
2. Perancangan yang dimaksud ialah pembuatan rancangan sistem penjadwalan mata kuliah dimana tahapan ini termasuk kegiatan analisis sistem, perancangan sistem dan pembuatan program, menggunakan *Data Flow Diagram (DFD)* dan *Moockup*
3. Pada tahapan akhir peneliti melakukan pengujian terhadap sistem yang dibuat. Pengujian dilakukan untuk mengevaluasi *prototype* yang bertujuan untuk memperjelas kebutuhan sistem. Pengujian sistem menggunakan *Black Box Testing*

3.2. Metode Pengumpulan Data

Tahapan pengumpulan data untuk memperoleh data sesuai dengan kebutuhan penelitian:

1. Studi Pustaka

Studi pustaka yang dilakukan dalam penelitian ini ialah mengenai *Particle Swarm Optimization (PSO)*, data yang digunakan dan perancangan sistem yang sesuai akan kebutuhan dari penelitian. Studi pustaka dapat diperoleh melalui jurnal-jurnal ilmiah dan sumber ilmiah lain seperti laman web dan media lain yang berhubungan dengan pembahasan masalah.

2. Wawancara

Teknik wawancara (*interview*), dilakukan untuk mengetahui hal-hal yang bersifat mendalam, peneliti melakukan tanya jawab dengan Koordinator Prodi Program Studi Informatika UNKHAIR terkait penjadwalan mata kuliah. Berdasarkan hasil wawancara yang penulis lakukan dengan koprodi informatika bapak Rosihan mengatakan bahwa, dalam pembuatan jadwal mata kuliah di Informatika unkhair harus memperhatikan jadwal dosen artinya, pada hari dan jam yang sama dosen tidak memiliki jadwal yang berbeda lebih dari satu, setelah itu harus memperhatikan jadwal ruang yang dimana tidak boleh lebih dari satu mata kuliah yang menggunakan ruang yang sama di hari dan jam yang sama pula, dan harus memperhatikan jadwal mata kuliah untuk satu semester di mahasiswa yang sama agar mahasiswa memilih mata kuliah yang tidak bertabrakan.

3.3. Alat dan Bahan Penelitian

Adapun alat dan bahan pada penelitian ini, terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*) sebagai penunjang selama penelitian.

3.4.1. Perangkat Keras (*Hardware*)

Adapun spesifikasi perangkat keras (*hardware*) yang digunakan pada penelitian ini sebagai penunjang proses perancangan dan pembuatan sistem selama penelitian berlangsung. Spesifikasi *hardware* dapat dilihat pada tabel 3.1.

Tabel 3.1 Spesifikasi *Hardware*

Jenis	Spesifikasi
Model	<i>Asus x441u</i>
<i>Processor</i>	<i>Intel(R) Core (TM) i3-7020U CPU @ 2.30GHz 2.30 GHz</i>
<i>Memory (RAM)</i>	4 GB
<i>Hardisk</i>	1 TB
<i>System Type</i>	64-bit

3.4.2. Perangkat Lunak (*Software*)

Selain kebutuahn *hardware* penulis juga membutuhkan *software* untuk melakukan perancangan dan pembuatan sistem. Spesifikasi *software* dapat dilihat pada tabel 3.2.

Tabel 3.2 Spesifikasi *Software*

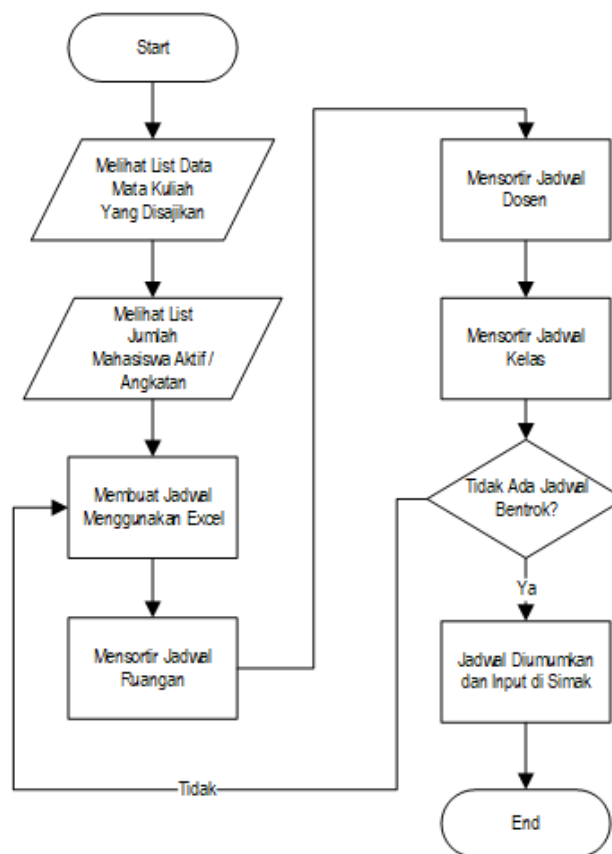
Jenis	Tipe	Keterangan
OS	<i>Windows 10 Home Single Language</i>	Sistem operasi yang digunakan selama penelitian.
Bahasa Pemrograman	PHP, HTML, CSS, <i>Java Script</i>	Bahasa pemrograman yang digunakan dalam pembuatan sistem.
<i>Text Editor</i>	<i>Visual Studio Code</i>	Alat yang digunakan untuk menulis kode program pembuatan sistem.
<i>Database</i>	XAMPP	Digunakan sebagai <i>server desktop</i> .

3.4. Perancangan Sistem

Pada tahapan ini akan dilakukan pemodelan sistem untuk keperluan proses, perancangan yang dimaksud ialah perancangan sistem yang sedang berjalan, sistem yang di usulkan dan *Data Flow Diagram (DFD)*.

3.5.1. Sistem Yang Sedang Berjalan

Sistem yang sedang berjalan adalah proses atau langkah-langkah yang menunjukkan tahap pembuatan jadwal mata kuliah di jurusan informatika unkhair. Berikut adalah sistem penjadwalan mata kuliah yang sedang berjalan di jurusan informatika unkhair, Dapat dilihat pada gambar 3.2.

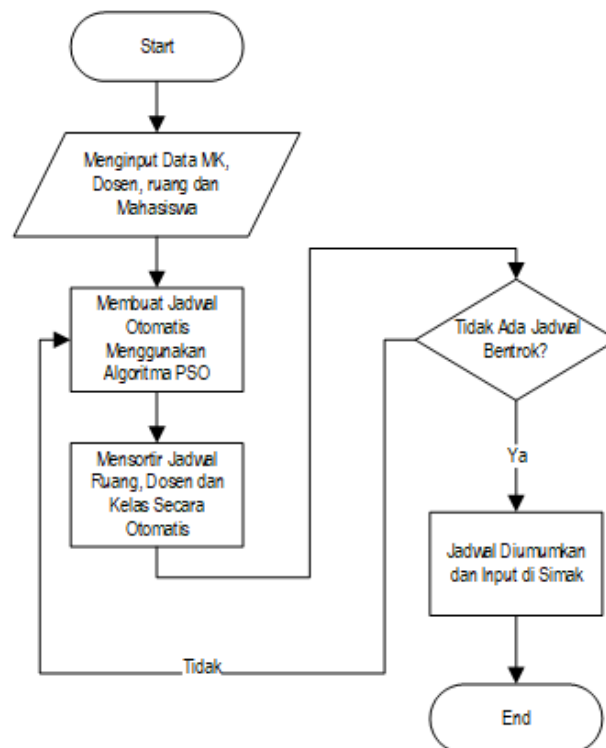


Gambar 3.2 Sistem Yang Sedang Berjalan

Berdasarkan gambar di atas proses penjadwalan mata kuliah di jurusan informatika unkhair dimulai dari koprodi melihat list data matakuliah, kemudian melihat list mahasiswa. Setelah itu koprodi membuat jadwal secara manual menggunakan *Microsoft Excel*, Kemudian koprodi akan mensortir secara berurutan jadwal ruangan, dosen dan kelas. Jika masih terdapat jadwal yang bentrok maka koprodi mengedit jadwal tersebut, jika sudah maka langsung di umumkan dan di *input* di Simak.

3.5.2. Sistem Yang Diusulkan

Sistem yang diusulkan adalah proses atau langkah-langkah yang menunjukkan tahap pembuatan jadwal yang di rekomendasikan. Berikut adalah sistem penjadwalan mata kuliah yang akan dibuat dan diusulkan oleh penulis, Dapat dilihat pada gambar 3.3.



Gambar 3.3 Sistem Yang Diusulkan

Keterangan:

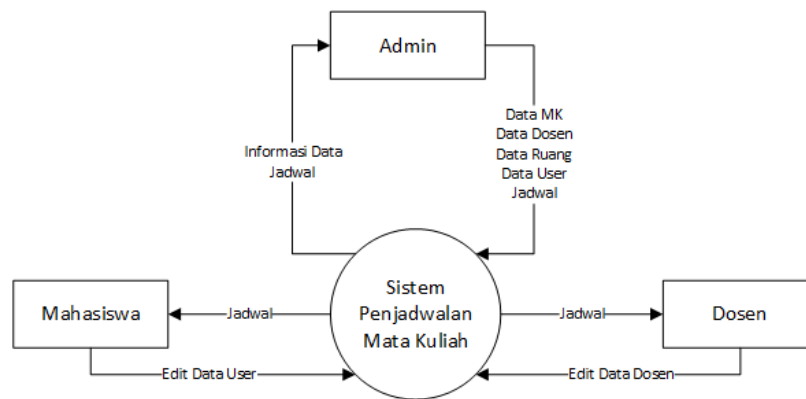
1. Koprodi yang akan bertindak sebagai *admin* akan menginput data-data yang diperlukan.
2. Membuat jadwal secara otomatis menggunakan sistem penjadwalan yang di usulkan penulis.
3. Mensortir jadwal secara otomatis dan jika terdapat jadwal bentrok maka *admin* bisa langsung mengeditnya.
4. Jadwal langsung di umumkan dan di *input* di Simak.

3.5.3. Data Flow Diagram (DFD)

1. DFD level 0

DFD mempunyai level-level mulai dari yang terkecil, yaitu level 0. DFD level 0 merupakan gambaran paling umum dari sistem dan prosesnya mewakili seluruh sistem.

Berikut adalah diagram level 0 dapat dilihat pada gambar 3.4:

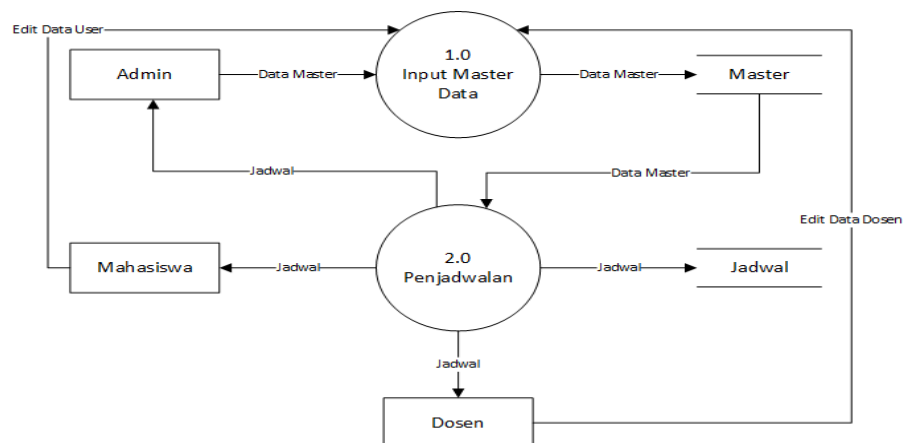


Gambar 3.4 DFD Level 0

Pada DFD level 0 diatas admin dapat memasukkan dan mengedit data-data penjadwalan yang di butuhkan kedalam sistem dan membuat jadwal. Dosen dan mahasiswa hanya dapat melihat data jadwal yang sudah jadi.

2. DFD level 1

Berikut adalah diagram level 1 proses *input master data*, dilihat pada gambar 3.5:



Gambar 3.5 DFD Level 1

Pada DFD *level 1 admin* tidak hanya dapat memasukkan dan mengedit data *master* tetapi juga dapat membuat jadwal perkuliahan baru, data *master* yang dimasukan akan tersimpan dalam *database*, Data *master* tersebut yang akan dibuat menjadi jadwal mata kuliah dan jadwal tersebut akan tersimpan dalam *database*. Dosen dan mahasiswa hanya dapat melihat jadwal yang sudah jadi.

3.5. Implementasi Algoritma *Particle Swarm Optimization (PSO)*

Proses algoritma *Particle Swarm Optimization* terdiri dari tiga tahap yaitu pembangkitan posisi serta kecepatan partikel, *Update velocity* dan *Update* posisi. Ketika tiga tahap tersebut sudah di lalui maka dilakukan pengecekan apakah hasil sudah optimum atau tidak. Bila hasil sudah optimum maka proses akan berhenti, namun bila hasil belum optimum maka proses akan berulang sampai mendapatkan hasil optimum atau sampai proses pengulangan maksimal, tahapan dari algoritma PSO sebagai berikut:

1. Buat suatu basis data yang menampung data perkuliahan, contoh seperti tabel 3.3.

Tabel 3.3 Data Perkuliahan

MK	SKS	Dosen	Ruang
Mk1 f1	3	Ds1	R1
Mk1 f2	3	Ds2	R2
Mk2 f1	4	Ds3	
Mk2 f2	4		
Mk3 f1	3		
Mk3 f2	3		

2. Buat suatu slot yang akan menampung posisi dan *velocity* awal, dapat dilihat pada tabel 3.4.

Tabel 3.4 Slot Perkuliahan

Hari	Jam	MK	Dosen	Ruang
Senin				
	Slot 0			

	Slot 1			
	Slot 2			
Selasa				
	Slot 3			
	Slot 4			
	Slot 5			
Rabu	Slot 6			
	Slot 7			
	Slot 8			
	Slot 9			
Kamis				
	Slot 10			
	Slot 11			
	Slot 12			
Jumat	Slot 13			
	Slot 14			
	Slot 15			
	Slot 16			
	Slot 17			
	Slot 18			
	Slot 19			

Berdasarkan slot di atas maka didapat nilai posisi (x) awal, seperti pada tabel 3.5.

Tabel 3.5 Nilai Posisi (x) Awal

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16	x17	x18	x19
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19

3. Tentukan parameter-parameter PSO dimisalkan seperti pada tabel 3.6:

Tabel 3.6 Parameter PSO

Iterasi	1
w	0,5
c1	1,5
c2	1,5
r1	0,1
r2	0,2

W = Inertia factor

$C1$ = Self confidence

$C2$ = Swarm confidence

$R1$ dan $R2$ = Nilai random dari 0-1

4. Acak data perkuliahan dan masukkan ke dalam slot perkuliahan, dapat dilihat pada tabel 3.7 dan 3.8.

Tabel 3.7 Data Perkuliahan Setelah di Acak

Acak				
Jam	MK	SKS	Dosen	Ruang
07:30-10:50	Mk2 f2	4	Ds2	R1
09:10-12:30	Mk2 f1	4	Ds2	R2
10:00-12:30	Mk3 f1	3	Ds1	R2
10:00-12:30	Mk1 f1	3	Ds3	R2
12:30-15:00	Mk3 f2	3	Ds2	R1
13:20-15:50	Mk1 f2	3	Ds3	R1

Tabel 3.8 Data Perkuliahan Acak di Masukkan ke Dalam Slot

Hari	Jam	MK	Dosen	Ruang
Senin	07:30-10:50	Mk2 f2	Ds2	R1
	Slot 0			
	09:10-12:30	Mk2 f1	Ds2	R2

	Slot 1			
	10:00-12:30	Mk3 f1	Ds1	R2
	Slot 2			
	10:00-12:30	Mk1 f1	Ds3	R2
	Slot 3			
Selasa	12:30-15:00	Mk3 f2	Ds2	R1
	Slot 4			
	13:20-15:50	Mk1 f2	Ds3	R1
	Slot 5			
	Slot 6			
	Slot 7			
Rabu				
	Slot 8			
	Slot 9			
	Slot 10			
Kamis	Slot 11			
	Slot 12			
	Slot 13			
	Slot 14			
Jumat				
	Slot 15			
	Slot 16			
	Slot 17			
	Slot 18			
	Slot 19			

Setelah memasukkan data perkuliahan ke dalam slot terdapat beberapa bentrokan jadwal atau *fitness* dapat dilihat pada tabel 3.9:

Tabel 3.9 *Fitness*

<i>fitness</i>	
f(1)	2
f(2)	1
f(3)	2

F(1) = Banyaknya kelas yang sama berada pada hari dan jam yang bersamaan.

F(2) = Banyaknya dosen yang bentrok jam mengajar.

F(3) = Banyaknya ruang yang bentrok.

Dari nilai *fitness* yang sudah didapat pada tabel 3.9, kita bisa menentukan nilai *local best* dan *global best*. Adapun nilai *local best* adalah *fitness* yang mempunyai nilai bentrokan / pelanggaran paling sedikit dan nilai *global best* adalah nilai *local best* paling baik dari setiap iterasi sehingga didapat nilai *local best* sebagai berikut:

Local best = 1

Setelah didapat nilai *local best* kita dapat menentukan nilai *global best*. Pada perhitungan manual ini jumlah iterasi = 1 dimaksudkan sebagai iterasi awal. Maka pada iterasi awal nilai *local best* sama dengan nilai *global best*, sehingga didapat nilai *global best* sebagai berikut:

Global best = 1

5. *Update Velocity*

Setiap partikel yang berpindah dari satu posisi ke posisi yang lain dipengaruhi oleh suatu *velocity* (kecepatan) yang menggambarkan perpindahan posisi. Nilai posisi (x) awal yang telah ditentukan dimasukkan ke dalam persamaan *velocity* sehingga didapat nilai *velocity* sebagai berikut pada tabel 3.10:

Tabel 3.10 Nilai Kecepatan (v) Awal

V0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15	v16	v17	v18	v19
----	----	----	----	----	----	----	----	----	----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
---	---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----

Kemudian melakukan update kecepatan menggunakan persamaan 2.3 berikut:

$$V_{k+1}^i = W * V_k^i + C1 * Rnd * (P_k^i - X_k^i) + C2 * Rnd * (P_k^g - X_k^i)$$

$$W = (0.4 - 1.4)$$

$$P_k^i = \text{Fitness saat iterasi}$$

$$C1 = (1.5 - 2.0)$$

$$P_k^g = \text{Fitness terbaik}$$

$$C2 = (1.5 - 2.0)$$

$$V_k^i = \text{Kecepatan}$$

$$X_k^i = \text{Posisi}$$

$$i = \text{Indeks partikel}$$

$$k = \text{Iterasi}$$

$$g = \text{Global best}$$

Berdasarkan rumus diatas maka perhitungannya sebagai berikut:

$$1. \quad V_0^0 = 0.5 * 0 + 1.5 * 0.1 * (1 - 0) + 1.5 * 0.2 * (1 - 0)$$

$$V_0^0 = 0.5 * 0 + 1.5 * 0.1 * 1 + 1.5 * 0.2 * 1$$

$$V_0^0 = 0 + 0.15 + 0.3$$

$$V_0^0 = 0.45$$

$$2. \quad V_0^1 = 0.5 * 1 + 1.5 * 0.1 * (1 - 1) + 1.5 * 0.2 * (1 - 1)$$

$$V_0^1 = 0.5 + 1.5 * 0.1 * 0 + 1.5 * 0.2 * 0$$

$$V_0^1 = 0.5 + 1.5 * 0 + 1.5 * 0$$

$$V_0^1 = 0.5$$

$$3. \quad V_0^2 = 0.5 * 2 + 1.5 * 0.1 * (1 - 2) + 1.5 * 0.2 * (1 - 2)$$

$$V_0^2 = 1 + 1.5 * 0.1 * (-1) + 1.5 * 0.2 * (-1)$$

$$V_0^2 = 1 - 0.15 - 0.3$$

$$V_0^2 = 0.55$$

Sehingga di dapat nilai kecepatan baru dapat dilihat pada tabel 3.11:

Tabel 3.11 Nilai Kecepatan (v) Baru

v0	0,45
v1	0,5
v2	0,55
v3	0,6
v4	0,65
v5	0,7
v6	0,75

v7	0,8
v8	0,85
v9	0,9
v10	0,95
v11	1
v12	1,05
v13	1,1
v14	1,15
v15	1,2
v16	1,25
v17	1,3
v18	1,35
v19	1,4

6. *Update* Posisi

Menggunakan persamaan 2.4 yaitu:

$$X_{k+1}^i = X_k^i + V_{k+1}^i$$

Berdasarkan rumus diatas maka perhitungannya sebagai berikut:

$$1. \quad X_0^0 = 0 + 0.45$$

$$X_0^0 = 0.45$$

$$2. \quad X_0^1 = 1 + 0.5$$

$$X_0^1 = 1.5$$

$$3. \quad X_0^2 = 2 + 0.55$$

$$X_0^2 = 2.55$$

Dalam *update* posisi ini, partikel atau yang dalam kasus jadwal ini adalah mata kuliah yang berada pada posisi yang salah (mengalami bentrokan jadwal) maka akan melakukan perpindahan posisi sesuai hasil *update* posisi, nilai posisi baru dapat dilihat pada tabel 3.12.

Tabel 3.12 Nilai Posisi (x) Baru

x0	0,45	0
x1	1,5	2
x2	2,55	3
x3	3,6	4

x4	4,65	5
x5	5,7	6
x6	6,75	7
x7	7,8	8
x8	8,85	9
x9	9,9	10
x10	10,95	11
x11	12	12
x12	13,05	13
x13	14,1	14
x14	15,15	15
x15	16,2	16
x16	17,25	17
x17	18,3	18
x18	19,35	19
x19	20,4	20

Jika posisi baru yang didapatkan ditampilkan ke dalam bentuk representasi posisi, maka dapat dilihat pada tabel 3.13 berikut.

Tabel 3.13 Posisi Data Perkuliahan Baru

Hari	Jam	MK	Dosen	Ruang
Senin	07:30-10:50	Mk2 f2	Ds2	R1
	Slot 0			
	Slot 1			
	09:10-12:30	Mk2 f1	Ds2	R2
	Slot 2			
	10:00-12:30	Mk3 f1	Ds1	R2
Selasa	Slot 3			
	10:00-12:30	Mk1 f1	Ds3	R2
	Slot 4			
	12:30-15:00	Mk3 f2	Ds2	R1
	Slot 5			
	13:20-15:50	Mk1 f2	Ds3	R1
	Slot 6			
Rabu				
	Slot 7			
	Slot 8			

	Slot 9			
	Slot 10			
Kamis	Slot 11			
	Slot 12			
	Slot 13			
	Slot 14			
Jumat	Slot 15			
	Slot 16			
	Slot 17			
	Slot 18			
	Slot 19			

7. Kondisi selesai

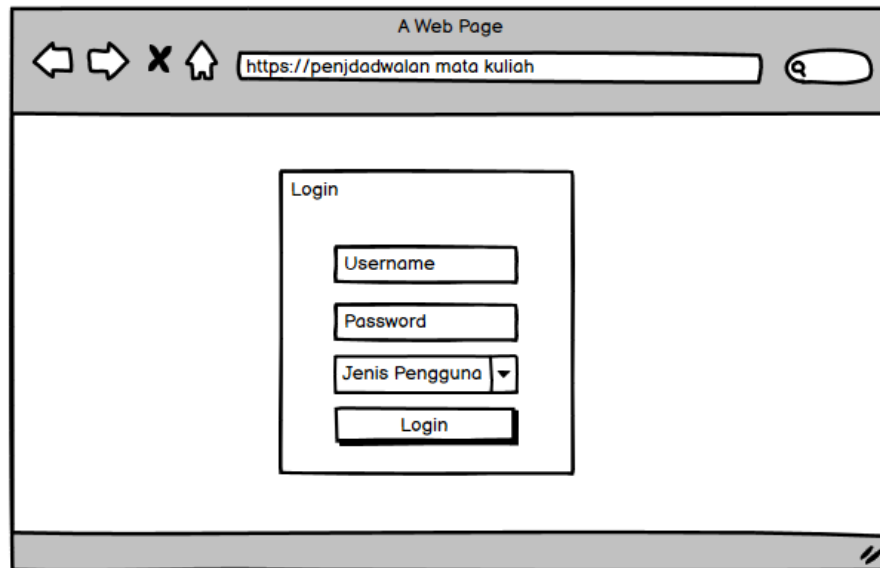
Terdapat suatu kondisi selesai yang dapat menghentikan proses algoritma *Particle Swarm Optimization* yaitu jika nilai *fitness* terbaik telah dicapai. Jika saat kondisi diatas telah diperoleh oleh program, maka iterasi akan dihentikan dan menampilkan hasil dari penjadwalan yang telah diperoleh.

3.6. Desain *Interface*

Perancangan desain *Interface* sebuah sistem dapat berfungsi untuk membuat tampilan *input/output* sebuah sistem yang nanti akan dibuat, berikut beberapa tampilan desain sistem.

1. Tampilan *Login*

Desain tampilan halaman *login* berupa *username*, *password* dan jenis pengguna dapat dilihat pada gambar 3.6.

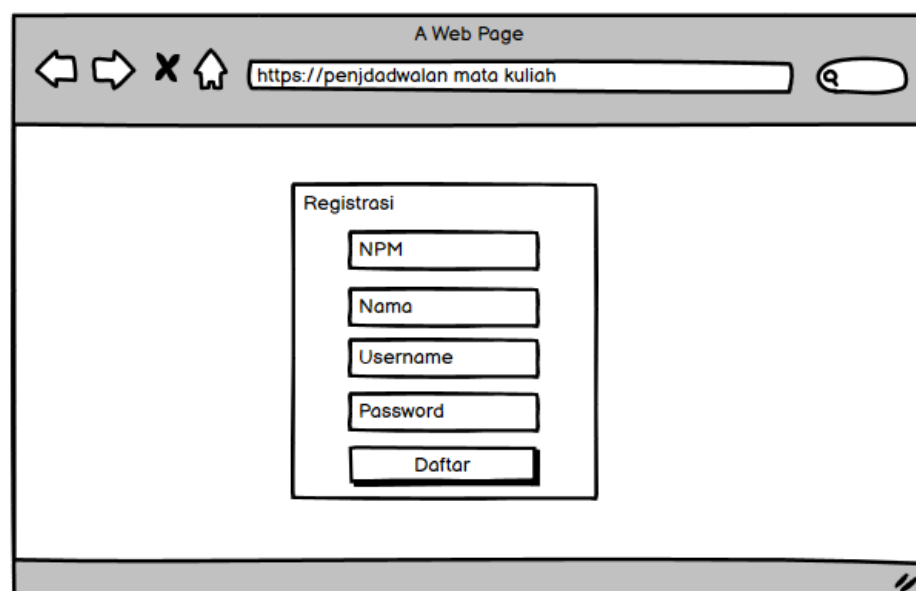


The image shows a web browser window titled "A Web Page". The address bar contains the URL "https://penjadwalan mata kuliah". The main content area displays a "Login" form. The form consists of four input fields stacked vertically: "Username", "Password", "Jenis Pengguna" (a dropdown menu), and a "Login" button. The browser window has standard navigation buttons (back, forward, stop, home) and a search icon.

Gambar 3.6 Tampilan *Login*

2. Tampilan Registrasi

Pada Desain Tampilan Registrasi terdapat beberapa inputan yakni npm, nama, *username* dan *password*. Tampilan registrasi dapat dilihat pada gambar 3.7.

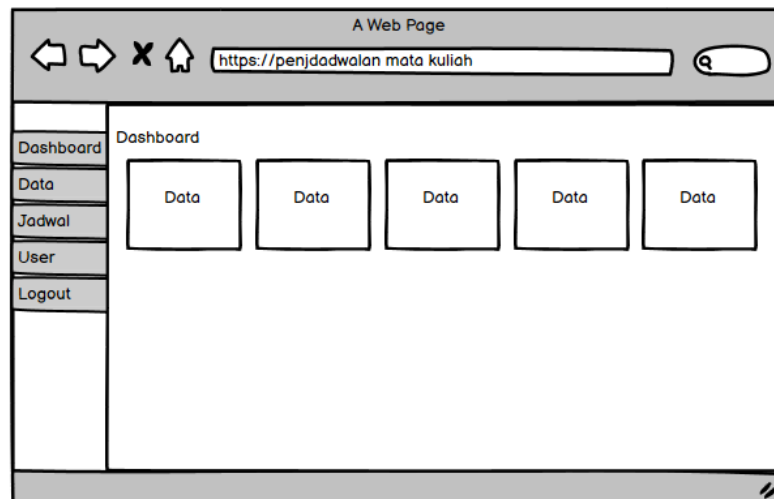


The image shows a web browser window titled "A Web Page". The address bar contains the URL "https://penjadwalan mata kuliah". The main content area displays a "Registrasi" form. The form consists of five input fields stacked vertically: "NPM", "Nama", "Username", "Password", and a "Daftar" button. The browser window has standard navigation buttons (back, forward, stop, home) and a search icon.

Gambar 3.7 Tampilan Registrasi

3. Tampilan *Dashboard*

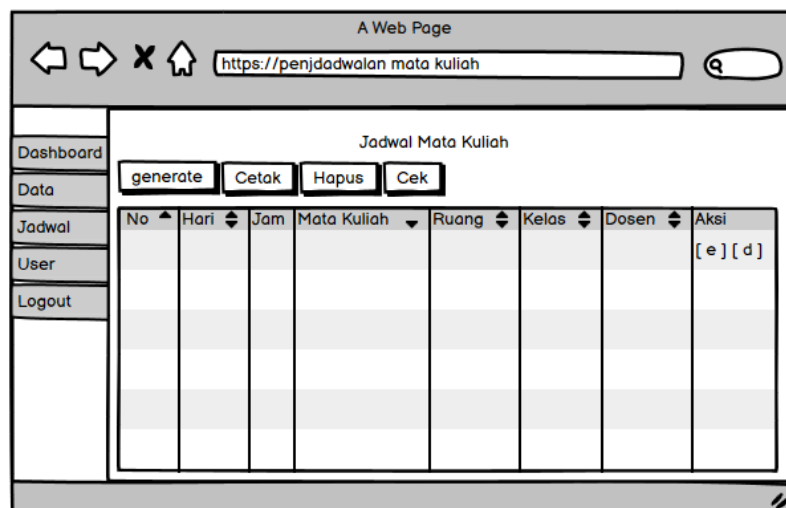
Pada Desain Tampilan *Dashboard* terdapat beberapa *menu* yakni data, jadwal dan *user*. Pada *menu* data terdapat *sub menu* yaitu dosen, mata kuliah, ruang dan kelas. Tampilan *dashboard* dapat dilihat pada gambar 3.8.



Gambar 3.8 Tampilan Dashboard

4. Tampilan *Menu Jadwal*

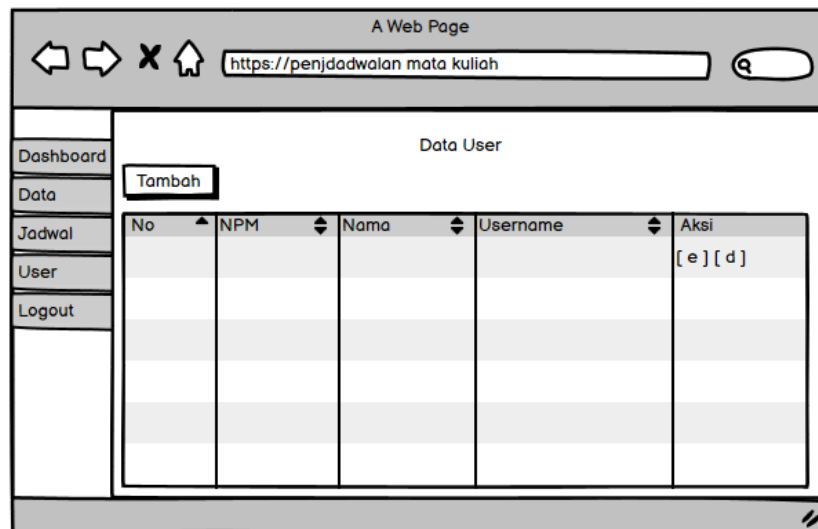
Pada tampilan jadwal, untuk membuat jadwal yang harus dilakukan ialah mengklik *button generate* yang dapat dilihat pada gambar 3.9.



Gambar 3.9 Tampilan Jadwal

5. Tampilan *User*

Pada tampilan *User* ialah data *user* yaitu no, npm, nama dan *username* yang dapat dilihat pada gambar 3.10.



Gambar 3.10 Tampilan *User*

3.7. Pengujian *Black Box*

Metode yang digunakan dalam melakukan pengujian pada sistem ialah metode *Black Box*. Metode *Black Box* ialah metode pengujian yang dilakukan untuk mengamati hasil eksekusi antarmuka (*interface*) melalui data uji dan memeriksa fungsional dari sistem yang dibuat. Pengujian sistem bertujuan untuk mengevaluasi hasil kerja dengan melakukan uji kelayakan sistem dan melakukan pengecekan apakah terjadi kesalahan atau tidak dalam penerapan metode yang digunakan dalam penelitian

BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini menjelaskan tentang Penerapan Algoritma *Particle Swarm Optimization* (PSO) Untuk Penjadwalan Mata Kuliah di Teknik Informatika Unkhair. Hasil yang dibahas ialah menentukan atau membuat jadwal mata kuliah dan pembagian dosen serta ruang kelas di Prodi Informatika Unkhair.

4.1. Pengujian dan Analisis

Pengujian ini dilakukan untuk melihat apakah algoritma *Particle Swarm Optimization* (PSO) digunakan dalam permasalahan penjadwalan mata kuliah akan mendapatkan hasil yang optimal atau tidak, dengan menggunakan fungsi algoritma dan fungsi penjadwalan mata kuliah. Pada pengujian ini penulis menggunakan penjadwalan mata kuliah untuk semester genap di jurusan informatika unkhair.

1. Pengujian menggunakan parameter 1

Pengujian akan dilakukan 5 kali percobaan jumlah iterasi yang digunakan adalah 50 dan pada pengujian 1 menggunakan nilai parameter $w = 0.4$, $c_1 = 1.0$ dan $c_2 = 1.0$. Hasil pengujian 1 dapat dilihat pada tabel 4.1 berikut.

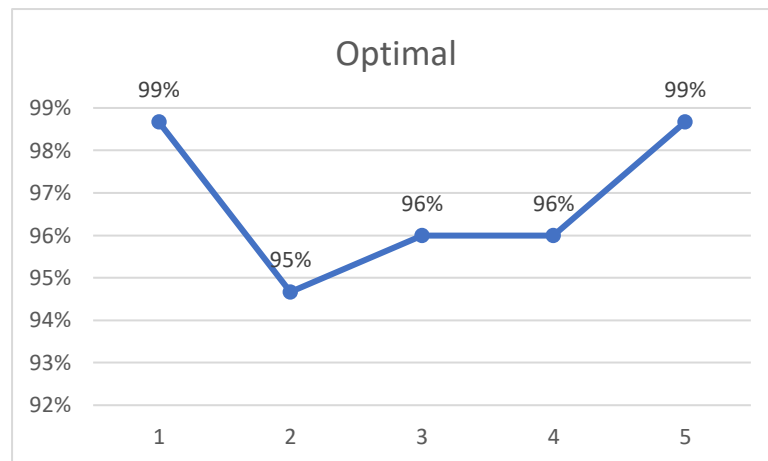
Tabel 4.1 Hasil Pengujian 1

percobaan	tb ruang	tb dosen	tb kelas	Optimal
1	1	0	0	99%
2	3	0	1	95%
3	1	0	2	96%
4	1	0	2	96%
5	0	0	1	99%

Pada pengujian menggunakan parameter 1 hasil yang di dapat yaitu belum mendapat jadwal yang optimal, Dimana masih terdapat beberapa tabrakan jadwal. Pada

Pengujian pertama ini mendapat tingkat ke optimalan jadwal paling tinggi yaitu 99% dan yang paling rendah yaitu 95%, Dimana terdapat 3 tabrakan ruang dan 1 tabrakan kelas.

Diagram hasil pengujian 1 dapat dilihat pada gambar 4.1.



Gambar 4.1 Diagram Hasil Pengujian 1

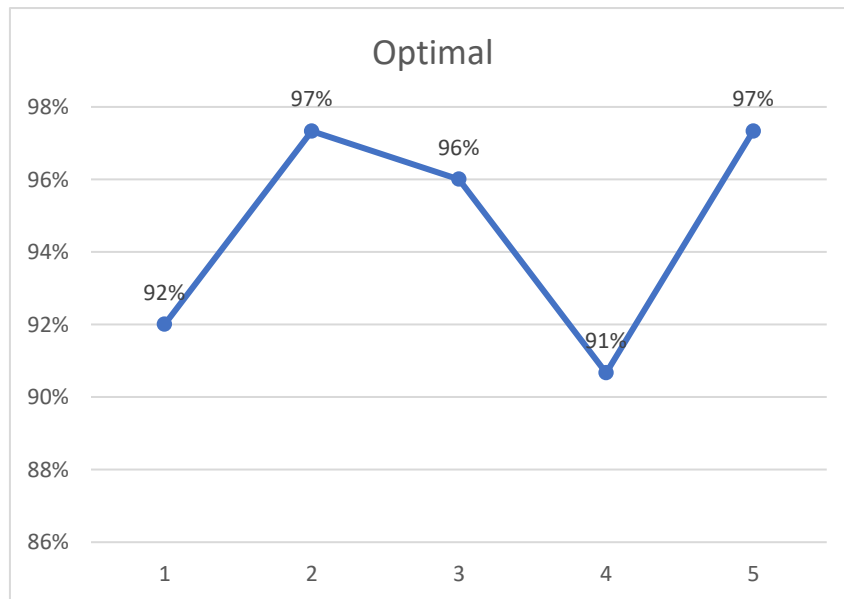
2. Pengujian menggunakan parameter 2

Pengujian akan dilakukan 5 kali percobaan jumlah iterasi yang digunakan adalah 50 dan pada pengujian 2 menggunakan nilai parameter $w = 0.9$, $c1 = 1.5$ dan $c2 = 1.5$. Hasil pengujian 2 dapat dilihat pada tabel 4.2 berikut.

Tabel 4.2 Hasil Pengujian 2

percobaan	tb ruang	tb dosen	tb kelas	Optimal
1	3	1	2	92%
2	0	0	2	97%
3	1	0	2	96%
4	3	1	3	91%
5	0	0	2	97%

Pada pengujian menggunakan parameter 2 hasil yang di dapat yaitu belum mendapat jadwal yang optimal, Dimana masih terdapat beberapa tabrakan jadwal. Pada pengujian kedua mendapat Tingkat ke optimalan jadwal yang paling tinggi yaitu 97% dan yang paling rendah yaitu 91%. Diagram hasil pengujian 2 dapat dilihat pada gambar 4.2.



Gambar 4.2 Diagram Hasil Pengujian 2

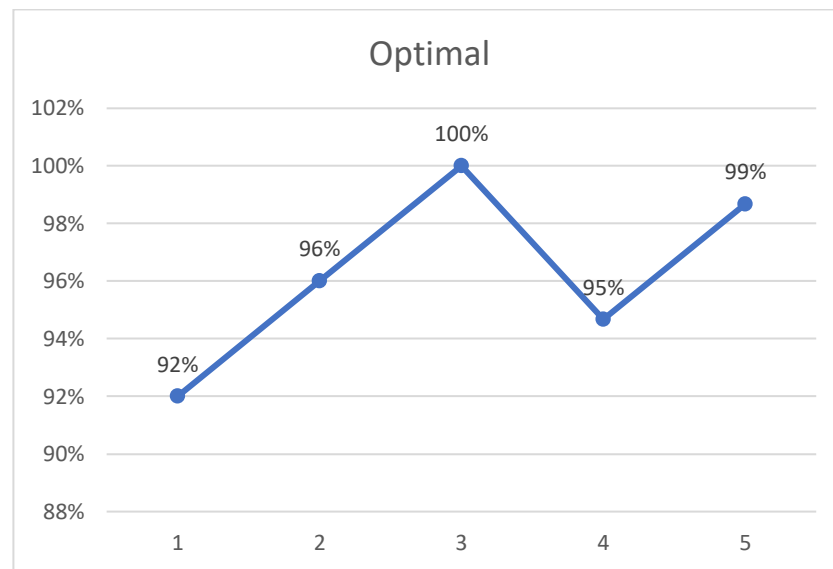
3. Pengujian menggunakan parameter 3

Pengujian akan dilakukan 5 kali percobaan jumlah iterasi yang digunakan adalah 50 dan pada pengujian 3 menggunakan nilai parameter $w = 0.9$, $c1 = 1$ dan $c2 = 1$. Hasil pengujian 3 dapat dilihat pada tabel 4.3 berikut.

Tabel 4.3 Hasil Pengujian 3

percobaan	tb ruang	tb dosen	tb kelas	Optimal
1	2	1	3	92%
2	1	1	1	96%
3	0	0	0	100%
4	2	0	2	95%
5	0	0	1	99%

Pada pengujian menggunakan parameter 3 hasil yang di dapat yaitu pada 5 kali percobaan terdapat 1 hasil jadwal yang optimal, dimana tidak terdapat tabrakan jadwal. Pada pengujian ketiga mendapat Tingkat ke optimalan jadwal yang paling tinggi yaitu 100% dan yang paling rendah yaitu 92%, Dimana terdapat 2 tabrakan ruang, 1 tabrakan dosen dan 3 tabrakan kelas. Diagram hasil pengujian 3 dapat dilihat pada gambar 4.3.



Gambar 4.3 Diagram Hasil Pengujian 3

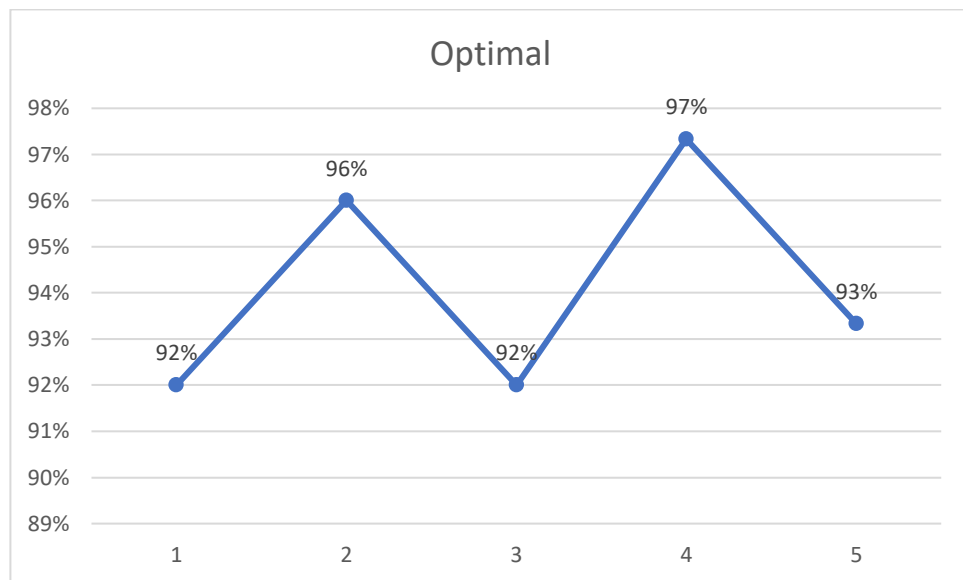
4. Pengujian menggunakan parameter 4

Pengujian akan dilakukan 5 kali percobaan jumlah iterasi yang digunakan adalah 50 dan pada pengujian 4 menggunakan nilai parameter $w = 0.7$, $c_1 = 2.5$ dan $c_2 = 2.5$. Hasil pengujian 4 dapat dilihat pada tabel 4.4 berikut.

Tabel 4.4 Hasil Pengujian 4

percobaan	tb ruang	tb dosen	tb kelas	Optimal
1	2	1	3	92%
2	2	0	1	96%
3	2	0	4	92%
4	1	0	1	97%
5	2	0	3	93%

Pada pengujian menggunakan parameter 4 hasil yang di dapat yaitu belum mendapat jadwal yang optimal, Dimana masih terdapat beberapa tabrakan jadwal. Pada pengujian keempat mendapat Tingkat ke optimalan jadwal yang paling tinggi yaitu 97% dan yang paling rendah yaitu 92%, Dimana terdapat 2 tabrakan ruang, 1 tabrakan dosen dan 3 tabrakan kelas. Diagram hasil pengujian 4 dapat dilihat pada gambar 4.4.



Gambar 4.4 Diagram Hasil Pengujian 4

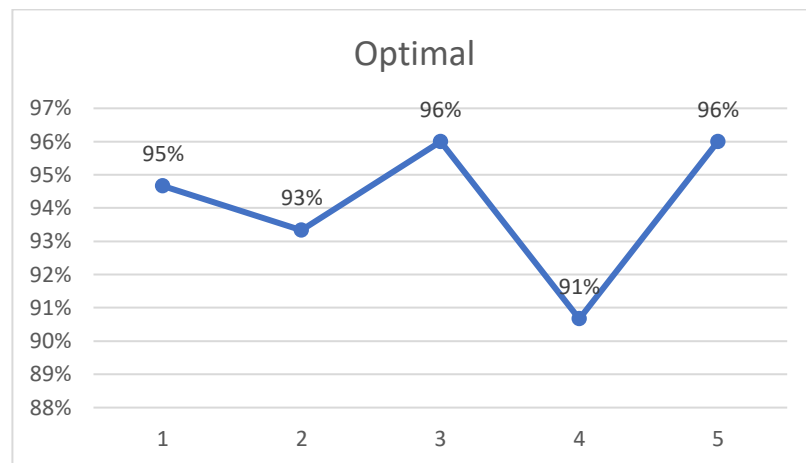
5. Pengujian menggunakan parameter 5

Pengujian akan dilakukan 5 kali percobaan jumlah iterasi yang digunakan adalah 50 dan pada pengujian 5 menggunakan nilai parameter $w = 1$, $c1 = 2$ dan $c2 = 2$. Hasil pengujian 5 dapat dilihat pada tabel 4.5 berikut.

Tabel 4.5 Hasil Pengujian 5

percobaan	tb ruang	tb dosen	tb kelas	Optimal
1	2	1	1	95%
2	2	0	3	93%
3	1	0	2	96%
4	3	1	3	91%
5	2	0	1	96%

Pada pengujian menggunakan parameter 5 hasil yang di dapat yaitu belum mendapat jadwal yang optimal, Dimana masih terdapat beberapa tabrakan jadwal. Pada pengujian kelima mendapat Tingkat ke optimalan jadwal yang paling tinggi yaitu 96% dan yang paling rendah yaitu 91%, Dimana terdapat 3 tabrakan ruang, 1 tabrakan dosen dan 3 tabrakan kelas. Diagram Hasil pengujian 5 dapat dilihat pada gambar 4.5.



Gambar 4. 5 Diagram Hasil Pengujian 5

Dari percobaan yang telah dilakukan diatas, dapat dilakukan analisa terhadap parameter-parameter yang digunakan. Analisa yang didapat sebagai berikut.

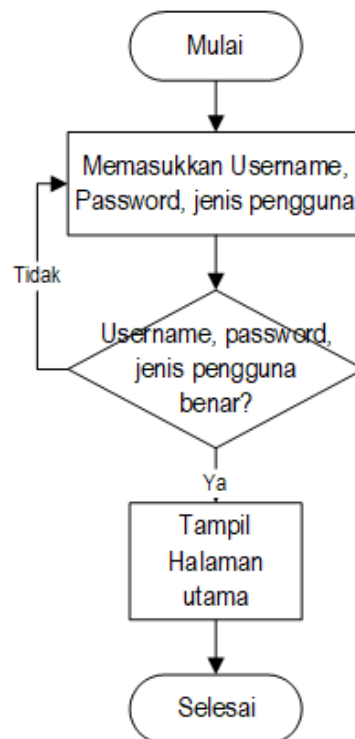
1. Pada pengujian menggunakan parameter 3 terdapat 1 hasil percobaan yang mendapat hasil paling optimal yaitu 100%, sedangkan pada pengujian menggunakan parameter 1 mendapat hasil yang paling mendekati optimal, Dimana pada setiap percobaan Tingkat ke optimalan jadwal diatas 95%.
2. Pada pengujian menggunakan nilai parameter yang berbeda mendapat hasil yang tidak terlalu berbeda, Dimana semua hasil yang didapat memiliki nilai ke optimalan di atas 90%.
3. Algoritma *Particle Swarm Optimization* (PSO) belum dapat menghasilkan jadwal yang selalu optimal, dibutuhkan beberapa kali percobaan untuk mendapatkan hasil yang optimal.

4.2. Implementasi Sistem

Dalam proses pengkodean sistem dibagi menjadi 2 *level user* yaitu: *level admin* (Koordinator Program Studi) dan *level user* mahasiswa. Adapun hak akses yang diberikan dari masing-masing *level user*.

1. Tampilan Login

Berikut ini merupakan *Flowchart Login* dan Tampilan *Login* ketika sistem pertama dijalankan. Tampilan ini yang menghubungkan *user* dengan aksesnya masing-masing sesuai dengan *level user* yang dipilih terlihat pada gambar 4.6 dan 4.7.



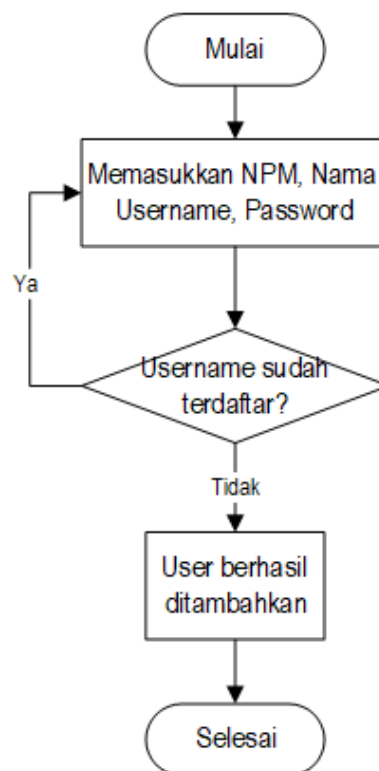
Gambar 4.6 *Flowchart Login*

The screenshot shows a web-based login form. At the top, there is a green header bar with a circular logo on the left and the word 'Login' in white text on the right. Below the header, the form contains three input fields: 'Username' with a person icon, 'Password' with a lock icon, and 'Jenis Pengguna' (User Type) which is a dropdown menu currently showing '-Pilih-'. Below these fields is a large green button labeled 'Login'. At the bottom of the form, there is a link that says 'Belum Punya Akun?Daftar Sekarang' (Don't have an account? Register Now).

Gambar 4.7 Tampilan *Login*

2. Tampilan Tambahkan Akun

Sebelum masuk pada halaman *login user* Mahasiswa harus membuat akun terlebih dahulu. *Flowchart* buat akun dan tampilan buat akun dapat dilihat pada gambar 4.8 dan 4.9.

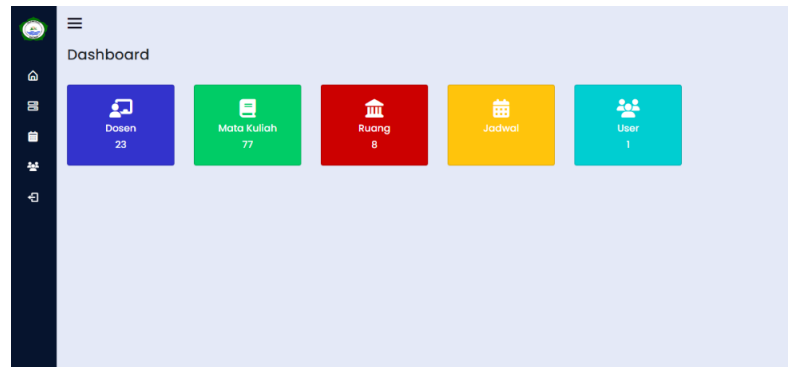


Gambar 4.8 *Flowchart* Tambah Akun

Gambar 4.9 Tampilan Tambah Akun

3. Dashboard *Admin/Koordinator*

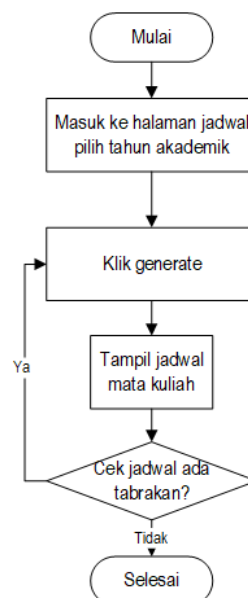
Pada tampilan *dashboard/beranda admin* terdapat beberapa *menu* yakni data dosen, data mata kuliah, data ruang, data jadwal dan data pengguna/*user*. Untuk tampilannya bisa dilihat pada gambar 4.10.



Gambar 4.10 Tampilan *Dashboard Admin*

4. Tampilan Jadwal

Berikut adalah *flowchart* dan tampilan jadwal, admin bisa membuat/*generate* jadwal, cetak jadwal, mengedit, menghapus jadwal dan mengecek apakah ada jadwal yang bertabrakan atau tidak. Dapat dilihat pada gambar 4.11 dan 4.12.



Gambar 4.11 *Flowchart Jadwal*

Waktu Eksekusi: 299 detik

Generate Cetak Hapus Cek

Show 10 entries Search:

No.	Hari	Jam	Mata kuliah	Ruang	Kelas	Dosen	Aksi
1	Kamis	07:30:00 - 09:10:00	Bahasa Inggris	Lab Jarkom	2if3	Abdulhalim Daud S.Pd, M.Pd	
2	Kamis	08:20:00 - 10:50:00	Data Mining	JT-05	6iflp	Muhammad Fhadli S.Kom., M.Sc.	
3	Rabu	09:10:00 - 11:40:00	Rekayasa Perangkat Lunak	Lab Jarkom	4if2	Hairil Kurniadi Siradjuddin S.Kom., M.Kom.	
4	Jumat	10:00:00 - 10:50:00	Seminar 1	Lab RPL	8ifkb	Rosihan S.T., M.Cs.	
5	Kamis	10:50:00 - 12:30:00	Digital Forensic	JT-03	6iflp	Yasir Muin S.T., M.Kom.	
6	Rabu	11:40:00 - 14:10:00	Aljabar Linier	Lab RPL	2if4	Munazat Salmin S.Pd., M.Cs.	
7	Rabu	12:30:00 - 15:50:00	Jaringan Komputer [+praktikum]	JT-06	4if3	Muhammad Sabri Ahmad S.Kom.,	

Gambar 4.12 Tampilan Jadwal

5. Tampilan Data *User/Mahasiswa*

Pada tampilan data *user/mahasiswa admin* dapat menambahkan data, mengedit dan melihat data *user/mahasiswa* yaitu NPM, Nama dan Username. Untuk tampilannya bisa dilihat pada gambar 4.13.

+ Tambah

Show 10 entries Search:

NO.	NPM	Nama	Username	Aksi
1	0735181062	Husen	husen	

Showing 1 to 1 of 1 entries Previous Next

Gambar 4.13 Tampilan Data *User*

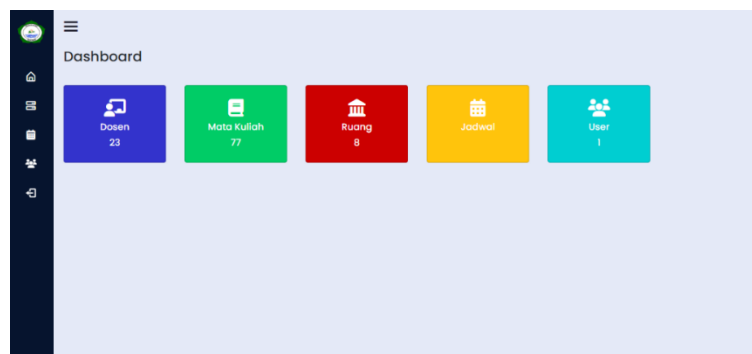
4.3. Pengujian Sistem

Pada tahapan pengujian sistem ini, sistem akan diuji menggunakan pengujian *black box*, dimana pengujian *black box* tidak sampai ke algoritma sistem yang dibangun akan Tetapi pengujian *black box* akan menguji fungsi-fungsi dari setiap *menu* pada sistem, apakah sudah sesuai dengan fungsinya atau belum.

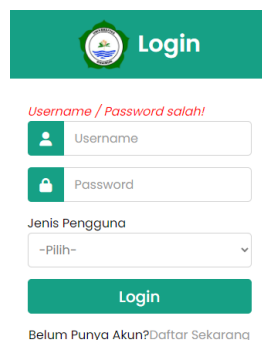
1. Halaman *Login*Tabel 4.6 Pengujian Halaman *Login*

Kasus dan Hasil Uji Coba (data yang dimasukkan benar)			
Data Masukkan	Yang diharapkan	Pengamatan	Kesimpulan
Username: <i>admin</i> Password:123 Jenis pengguna: <i>admin</i>	Ketika <i>admin</i> memilih tombol <i>login</i> akan diarahkan ke halaman <i>dashboard</i>	Dapat masuk ke halaman <i>dashboard</i>	Diterima
Kasus dan Hasil Uji Coba (data yang dimasukkan salah)			
Username: <i>admin</i> Password:123 Jenis pengguna: mahasiswa	Ketika <i>admin</i> memilih tombol <i>login</i> , sistem akan menampilkan peringatan " <i>Username / Password salah!</i> "	Tidak dapat masuk ke halaman <i>dashboard</i>	Diterima

Pada gambar 4.14 menampilkan proses jika berhasil *login* maka akan menampilkan halaman *dashboard*.

Gambar 4.14 Tampilan *Login* Berhasil

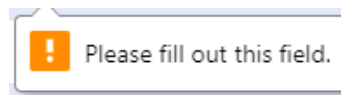
Pada gambar 4.15 menampilkan proses *login* gagal apabila memasukan *username*, *password* dan jenis pengguna yang salah.

Gambar 4.15 Tampilan *Login* Gagal

2. Halaman *Menu Data*

Tabel 4.7 Pengujian Halaman *Menu Data*

Kasus dan Hasil Uji Coba (data yang dimasukkan benar)			
Data Masukkan	Yang diharapkan	Pengamatan	Kesimpulan
Memilih <i>menu data</i>	Menampilkan halaman <i>menu data</i>	Dapat menampilkan halaman data	Diterima
Memilih tombol tambah data	Menampilkan halaman tambah data	Dapat menampilkan halaman tambah data	Diterima
Memilih tombol edit data	Menampilkan halaman edit data	Dapat menampilkan halaman edit data	Diterima
Memilih tombol hapus data	Menampilkan pesan "Yakin Ingin menghapus data ini?"	Data yang dipilih dihapus	Diterima
Kasus dan Hasil Uji Coba (data yang dimasukkan salah)			
Menekan <i>enter</i> pada <i>field input</i> data dan dibiarkan kosong.	Menampilkan pesan " <i>please fill out this field</i> "	Dapat menampilkan pesan <i>error</i> .	Diterima



Gambar 4.16 Tampilan *Error* Pada Inputan Kosong

Pada gambar 4.16 merupakan tampilan *error* jika admin tidak memasukan data dan di biarkan kosong maka akan tampil pesan seperti gambar diatas.

4.4. ***Pengujian Particle Swarm Optimization (PSO)***

Pada penelitian ini akan dilakukan pengujian perhitungan manual dengan menggunakan algoritma *Particle Swarm Optimization* (PSO). Pada algoritma *Particle Swarm Optimization* (PSO) terdiri dari tiga tahap yaitu pembangkitan posisi serta kecepatan partikel, *Update velocity* dan *Update* posisi.

1. Buat suatu slot yang akan menampung posisi dan *velocity* awal, dapat dilihat pada

tabel 4.8.

Tabel 4. 8 Slot Perkuliahan

Hari	Jam	MK	Dosen	Ruang
Senin				
	Slot 0			
	Slot 1			
	Slot 2			
	Slot 3			
	Slot 4			
	Slot 5			
	Slot 6			
	Slot 7			
	Slot 8			
	Slot 9			
	Slot 10			
	Slot 11			
	Slot 12			
	Slot 13			
	Slot 14			
Selasa				
	Slot 15			
	Slot 16			
	Slot 17			

	Slot 18			
	Slot 19			
	Slot 20			
	Slot 21			
	Slot 22			
	Slot 23			
	Slot 24			
	Slot 25			
	Slot 26			
	Slot 27			
	Slot 28			
	Slot 29			
Rabu				
	Slot 30			
	Slot 31			
	Slot 32			
	Slot 33			
	Slot 34			
	Slot 35			
	Slot 36			
	Slot 37			
	Slot 38			

	Slot 39			
	Slot 40			
	Slot 41			
	Slot 42			
	Slot 43			
	Slot 44			
Kamis				
	Slot 45			
	Slot 46			
	Slot 47			
	Slot 48			
	Slot 49			
	Slot 50			
	Slot 51			
	Slot 52			
	Slot 53			
	Slot 54			
	Slot 55			
	Slot 56			
	Slot 57			
	Slot 58			

Jumat	Slot 59			
	Slot 60			
	Slot 61			
	Slot 62			
	Slot 63			
	Slot 64			
	Slot 65			
	Slot 66			
	Slot 67			
	Slot 68			
	Slot 69			
	Slot 70			
	Slot 71			
	Slot 72			
	Slot 73			
	Slot 74			

Berdasarkan slot di atas maka didapat nilai posisi (x) awal, seperti pada tabel 4.9.

Tabel 4. 9 Nilai Posisi (x) Awal

x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	x10	x11	x12	x13	x14	x15	x16
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
x17	x18	x19	x20	x21	x22	x23	x24	x25	x26	x27	x28	x29	x30	x31	x32	x33
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
x34	x35	x36	x37	x38	x39	x40	x41	x42	x43	x44	x45	x46	x47	x48	x49	x50

34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
x51	x52	x53	x54	x55	x56	x57	x58	x59	x60	x61	x62	x63	x64	x65	x66	x67
51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67
x68	x69	x70	x71	x72	x73	x74										
68	69	70	71	72	73	74										

2. Tentukan parameter-parameter PSO dimisalkan seperti pada tabel 4.10:

Tabel 4. 10 Parameter PSO

w	0,5
c1	1,5
c2	1,5
r1	0,1
r2	0,2

W = Inertia factor

$C1$ = Self confidence

$C2$ = Swarm confidence

$R1$ dan $R2$ = Nilai random dari 0-1

3. Masukkan data perkuliahan kedalam slot perkuliahan secara acak, dapat dilihat pada tabel 4.11.

Tabel 4. 11 Data Perkuliahan Acak di Masukkan ke Dalam Slot

Hari	Jam	MK	Dosen	Ruang	Kelas
Senin	08:00 - 09:40	Pendidikan Kewarganegaraan	Fatoni Achmad, M.Pd.I. & Roy Nurfan, S.Pd., M.Pd.	JT-01	2IF1
	Slot 0				
	10:50 - 14:10	Dasar Pemrograman [+Praktikum]	Yasir Muin, S.T., M.Kom.	Lab. RPL	2IF1
	Slot 1				
	07:30 - 10:50	Pemrograman Berorientasi Objek [+Praktikum]	Ir. Salkin Lutfi, S.Kom., M.T., IPM	Lab. RPL	4IF1
	Slot 2				
	13:20 - 15:50	Rekayasa Perangkat Lunak	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-05	4IF1
Slot 3					

	10:50 - 14:10	Jaringan Komputer [+Praktikum]	Saiful Do. Abdullah, S.T., M.T.	Lab. JAK	4IF2
	Slot 4				
	07:30 - 10:00	Desain dan Analisis Algoritma	Muhammad Fhadli, S.T., M.Sc.	JT-04	4IF3
	Slot 5				
	14:10 - 16:40	Sistem Informasi Geografis [Kepulauan]	Ir. Arifandy Mario M., S.Kom., M.Kom., IPM	Lab. RPL	6IF1
	Slot 6				
	10:50 - 12:30	Riset Teknologi Informasi	Ir. Arifandy Mario M., S.Kom., M.Kom., IPM	JT-05	6IF2
	Slot 7				
	13:20 - 15:50	Sistem Terdistribusi	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-06	6IF2
	Slot 8				
	08:00 - 10:30	Teori Informasi	Achmad Fuad Assagaf, S.T., M.T.	JT-05	6IF1
	Slot 9				
	07:30 - 10:50	Jaringan Komputer [+Praktikum]	Saiful Do. Abdullah, S.T., M.T.	Lab. JAK	4IF4
	Slot 10				
	09:10 - 10:50	Pendidikan Kewarganegaraan	Fatoni Achmad, M.Pd.I. & Roy Nurfan, S.Pd., M.Pd.	JT-01	2IF3
	Slot 11				
	07:30 - 09:10	Etika Profesi	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	JT-06	4IF2
	Slot 12				
	07:30 - 10:00	Logika Matematika	Ir. Amal Khairan, S.T., M.Eng., IPM	JT-02	2IF2
	Slot 13				
	07:30 - 10:00	Matematika Diskrit	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-03	2IF4
	Slot 14				
Selasa	10:10 - 11:50	Etika Profesi	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	JT-06	4IF3
	Slot 15				
	13:20 - 15:50	Aljabar Linier	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-04	2IF2
	Slot 16				

	13:20 - 15:50	Aljabar Linier	Muhammad Natsir, S.T., M.T. & Aisjah Rachmawaty Ryadin, Ph.D.	JT-01	2IF4
	Slot 17				
	07:30 - 10:00	Technopreneur	Alfanugrah Hi. Usman, S.T., M.Kom.	JT-06	6IF2
	Slot 18				
	10:00 - 12:30	Aljabar Linier	Achmad Fuad Assagaf, S.T., M.T.	JT-02	2IF3
	Slot 19				
	07:30 - 10:00	Matematika Diskrit	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-04	2IF1
	Slot 20				
	07:30 - 10:00	Desain dan Analisis Algoritma	Muhammad Fhadli, S.T., M.Sc.	JT-02	4IF1
	Slot 21				
	07:30 - 10:50	Pemrograman Berorientasi Objek [+Praktikum]	Ir. Salkin Lutfi, S.Kom., M.T., IPM	Lab. RPL	4IF2
	Slot 22				
	13:20 - 15:50	Rekayasa Perangkat Lunak	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-06	4IF2
	Slot 23				
	10:00 - 12:30	Keamanan Data	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-06	4IF4
	Slot 24				
	10:30 - 12:10	Sistem Pendukung Keputusan	Alfanugrah Hi. Usman, S.T., M.Kom.	JT-05	6IFP
	Slot 25				
	07:30 - 10:00	Logika Matematika	Ir. Amal Khairan, S.T., M.Eng., IPM	JT-01	2IF4
	Slot 26				
	07:30 - 10:00	Kecerdasan Buatan	Dr. M. Ridha Albaar, S.Kom., M.Kom.	JT-03	4IF3
	Slot 27				
	13:20 - 15:00	Etika Profesi	Rosihan, S.T., M.Cs.	JT-05	4IF4
	Slot 28				
	07:30 - 10:00	Arsitektur dan Organisasi Komputer	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	JT-05	2IF2

Rabu	Slot 29				
	15:00 - 16:40	Bahasa Inggris	Yetty, S.S., M.Hum.	JT-01	2IF2
	Slot 30				
	13:20 - 15:00	Bahasa Inggris	Abdulhalim Daud, S.Pd., M.Pd.	JT-02	2IF1
	Slot 31				
	13:20 - 15:00	Bahasa Inggris	Yetty, S.S., M.Hum.	JT-01	2IF3
	Slot 32				
	10:00 - 12:30	Kecerdasan Buatan	Dr. M. Ridha Albaar, S.Kom., M.Kom.	JT-03	4IF1
	Slot 33				
	13:20 - 15:50	Keamanan Data	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-03	4IF3
	Slot 34				
	15:00 - 16:40	Bahasa Inggris	Abdulhalim Daud, S.Pd., M.Pd.	JT-02	2IF4
	Slot 35				
	13:20 - 15:50	Manajemen Proyek Perangkat Lunak	Dr. Assaf Arief, S.T., M.Eng.	JT-01	6IF1
	Slot 36				
	07:30 - 10:00	Teori Informasi	Achmad Fuad Assagaf, S.T., M.T.	JT-06	6IF2
	Slot 37				
	07:30 - 10:00	Kecerdasan Buatan	Dr. M. Ridha Albaar, S.Kom., M.Kom.	JT-02	4IF2
	Slot 38				
	07:30 - 10:00	Matematika Diskrit	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-01	2IF2
	Slot 39				
	10:50 - 14:10	Dasar Pemrograman [+Praktikum]	Yasir Muin, S.T., M.Kom.	Lab. RPL	2IF3
	Slot 40				
	10:00 - 12:30	Keamanan Data	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-06	4IF1
	Slot 41				
	13:20 - 15:50	Rekayasa Perangkat Lunak	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-03	4IF3
	Slot 42				

	07:30 - 10:00	Desain dan Analisis Algoritma	Muhammad Fhadli, S.T., M.Sc.	JT-04	4IF4
	Slot 43				
	13:20 - 15:00	Sistem Informasi	Rosihan, S.T., M.Cs.	JT-06	6IFP
	Slot 44				
Kamis	09:30 - 12:00	Sistem Terdistribusi	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-05	6IF1
	Slot 45				
	07:30 - 09:10	Etika Profesi	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	JT-05	4IF1
	Slot 46				
	13:20 - 15:50	Keamanan Data	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-04	4IF2
	Slot 47				
	07:30 - 10:00	Logika Matematika	Ir. Amal Khairan, S.T., M.Eng., IPM	JT-03	2IF3
	Slot 48				
	14:10 - 16:40	Sistem Informasi Geografis [Kepulauan]	Ir. Arifandy Mario M., S.Kom., M.Kom., IPM	Lab. RPL	6IF2
	Slot 49				
	07:30 - 10:50	Pemrograman Berorientasi Objek [+Praktikum]	Ir. Salkin Lutfi, S.Kom., M.T., IPM	Lab. RPL	4IF3
	Slot 50				
	10:50 - 12:30	Jaringan Terkini	Ir. Arifandy Mario M., S.Kom., M.Kom., IPM	JT-04	6IFP
	Slot 51				
	10:00 - 12:30	Kecerdasan Buatan	Dr. M. Ridha Albaar, S.Kom., M.Kom.	JT-02	4IF4
	Slot 52				
	10:50 - 14:10	Dasar Pemrograman [+Praktikum]	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	Lab. RPL	2IF2
	Slot 53				
	07:30 - 10:50	Pemrograman Berorientasi Objek [+Praktikum]	Ir. Salkin Lutfi, S.Kom., M.T., IPM	Lab. RPL	4IF4
	Slot 54				
	07:30 - 10:00	Logika Matematika	Ir. Amal Khairan, S.T., M.Eng., IPM	JT-04	2IF1

	Slot 55				
	07:30 - 10:00	Matematika Diskrit	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-02	2IF3
	Slot 56				
	10:50 - 12:30	Riset Teknologi Informasi	Dr. Assaf Arief, S.T., M.Eng.	JT-01	6IF1
	Slot 57				
	13:20 - 15:00	Sistem Multimedia	Saiful Do. Abdullah, S.T., M.T.	JT-03	6IFP
	Slot 58				
	07:30 - 10:00	Technopreneur	Alfanugrah Hi. Usman, S.T., M.Kom.	JT-01	6IF1
	Slot 59				
Jumat	13:20 - 15:50	Manajemen Proyek Perangkat Lunak	Dr. Assaf Arief, S.T., M.Eng.	JT-02	6IF2
	Slot 60				
	13:20 - 15:50	Arsitektur dan Organisasi Komputer	Seh Turuy, S.T., M.Eng.	JT-06	2IF3
	Slot 61				
	07:30 - 10:50	Jaringan Komputer [+Praktikum]	Saiful Do. Abdullah, S.T., M.T.	Lab. JAK	4IF3
	Slot 62				
	10:50 - 14:10	Jaringan Komputer [+Praktikum]	Ir. Amal Khairan, S.T., M.Eng., IPM	Lab. JAK	4IF1
	Slot 63				
	07:30 - 09:10	Data Mining & Big Data Analytic	Muhammad Fhadli, S.T., M.Sc.	JT-06	6IFP
	Slot 64				
	10:00 - 11:40	Pendidikan Kewarganegaraan	Hasmawati, S.Ag., M.Pd.+Abd. Firman Bunta, S.Pd., M.Pd	JT-01	2IF4
	Slot 65				
	10:50 - 14:10	Dasar Pemrograman [+Praktikum]	Yasir Muin, S.T., M.Kom.	Lab. RPL	2IF4
	Slot 66				
	09:20 - 11:00	Expert System	Muhammad Fhadli, S.T., M.Sc.	JT-06	6IFP
	Slot 67				
	13:20 - 15:00	Digital Forensic	Yasir Muin, S.T., M.Kom.	JT-04	6IFP
	Slot 68				

	09:30 - 12:00	Arsitektur dan Organisasi Komputer	Seh Turuy, S.T., M.Eng.	JT-02	2IF1
	Slot 69				
	09:10 - 10:50	Pendidikan Kewarganegaraan	Hasmawati, S.Ag., M.Pd.+Abd. Firman Bunta, S.Pd., M.Pd	JT-01	2IF2
	Slot 70				
	13:20 - 15:50	Rekayasa Perangkat Lunak	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-01	4IF4
	Slot 71				
	08:00 - 10:30	Desain dan Analisis Algoritma	Mohammad Taufik, S.Kom., M.Sc.	JT-04	4IF2
	Slot 72				
	14:00 - 16:30	Aljabar Linier	Ir. Suyuti, S.T., M.T., Ph.D., IPM	JT-01	2IF1
	Slot 73				
	07:30 - 10:00	Arsitektur dan Organisasi Komputer	Seh Turuy, S.T., M.Eng.	JT-02	2IF4
	Slot 74				

Setelah memasukkan data perkuliahan ke dalam slot terdapat beberapa bentrokan jadwal atau *fitness* dapat dilihat pada tabel 4.12:

Tabel 4. 12 *Fitness*

<i>Fitness</i>	
$f(1)$	2
$f(2)$	3
$f(3)$	1

$F(1)$ = Banyaknya dosen yang bentrok jam mengajar.

$F(2)$ = Banyaknya ruang yang bentrok.

$F(3)$ = Banyaknya kelas yang sama berada pada hari dan jam yang bersamaan.

Dari nilai *fitness* yang sudah didapat pada tabel 4.12, kita bisa menentukan nilai *local best* dan *global best*. Adapun nilai *local best* adalah *fitness* yang mempunyai nilai bentrokan / pelanggaran paling sedikit dan nilai *global best* adalah nilai *local best*

paling baik dari setiap iterasi sehingga didapat nilai *local best* sebagai berikut:

Local best = 1

Setelah didapat nilai *local best* kita dapat menentukan nilai *global best*. Pada perhitungan manual ini jumlah iterasi = 1 dimaksudkan sebagai iterasi awal. Maka pada iterasi awal nilai *local best* sama dengan nilai *global best*, sehingga didapat nilai *global best* sebagai berikut:

Global best = 1

4. *Update Velocity*

Setiap partikel yang berpindah dari satu posisi ke posisi yang lain dipengaruhi oleh suatu *velocity* (kecepatan) yang menggambarkan perpindahan posisi. Nilai posisi (x) awal yang telah ditentukan dimasukkan ke dalam persamaan *velocity* sehingga didapat nilai *velocity* sebagai berikut pada tabel 4.13:

Tabel 4. 13 Nilai Kecepatan (v) Awal

v0	v1	v2	v3	v4	v5	v6	v7	v8	v9	v10	v11	v12	v13	v14	v15	v16
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
v17	v18	v19	v20	v21	v22	v23	v24	v25	v26	v27	v28	v29	v30	v31	v32	v33
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33
v34	v35	v36	v37	v38	v39	v40	v41	v42	v43	v44	v45	v46	v47	v48	v49	v50
34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
v51	v52	v53	v54	v55	v56	v57	v58	v59	v60	v61	v62	v63	v64	v65	v66	v67
51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67
v68	v69	v70	v71	v72	v73	v74										
68	69	70	71	72	73	74										

Kemudian melakukan update kecepatan menggunakan persamaan 2.3 berikut:

$$V_{k+1}^i = W * V_k^i + C1 * Rnd * (P_k^i - X_k^i) + C2 * Rnd * (P_k^g - X_k^i)$$

$$W = (0.4 - 1.4)$$

$$P_k^i = \text{Fitness saat iterasi}$$

$$C1 = (1.5 - 2.0)$$

$$P_k^g = \text{Fitness terbaik}$$

$$C2 = (1.5 - 2.0)$$

$$V_k^i = \text{Kecepatan}$$

$$X_k^i = \text{Posisi}$$

$$i = \text{Indeks partikel}$$

$$k = \text{Iterasi}$$

$$g = \text{Global best}$$

Berdasarkan rumus diatas maka perhitungannya sebagai berikut:

$$1. \quad V_0^0 = 0.5 * 0 + 1.5 * 0.1 * (6 - 0) + 1.5 * 0.2 * (6 - 0)$$

$$V_0^0 = 0.5 * 0 + 1.5 * 0.1 * 6 + 1.5 * 0.2 * 6$$

$$V_0^0 = 0 + 0.9 + 1.8$$

$$V_0^0 = 2.7$$

$$2. \quad V_0^1 = 0.5 * 1 + 1.5 * 0.1 * (6 - 1) + 1.5 * 0.2 * (6 - 1)$$

$$V_0^1 = 0.5 + 1.5 * 0.1 * 5 + 1.5 * 0.2 * 5$$

$$V_0^1 = 0.5 + 1.5 * 0.5 + 1.5 * 1$$

$$V_0^1 = 2.75$$

$$3. \quad V_0^2 = 0.5 * 2 + 1.5 * 0.1 * (6 - 2) + 1.5 * 0.2 * (6 - 2)$$

$$V_0^2 = 1 + 1.5 * 0.1 * 4 + 1.5 * 0.2 * 4$$

$$V_0^2 = 1 + 0.6 + 1.2$$

$$V_0^2 = 2.8$$

Sehingga di dapat nilai kecepatan baru dapat dilihat pada tabel 4.14:

Tabel 4. 14 Nilai Kecepatan (v) Baru

v0	2,7
v1	2,75
v2	2,8
v3	2,85
v4	2,9
v5	2,95
v6	3
v7	3,05
v8	3,1
v9	3,15
v10	3,2
v11	3,25
v12	3,3
v13	3,35
v14	3,4

v15	3,45
v16	3,5
v17	3,55
v18	3,6
v19	3,65
v20	3,7
v21	3,75
v22	3,8
v23	3,85
v24	3,9
v25	3,95
v26	4
v27	4,05
v28	4,1
v29	4,15
v30	4,2
v31	4,25
v32	4,3
v33	4,35
v34	4,4
v35	4,45
v36	4,5
v37	4,55
v38	4,6
v39	4,65
v40	4,7
v41	4,75
v42	4,8
v43	4,85
v44	4,9
v45	4,95
v46	5
v47	5,05
v48	5,1
v49	5,15
v50	5,2
v51	5,25
v52	5,3
v53	5,35
v54	5,4
v55	5,45

v56	5,5
v57	5,55
v58	5,6
v59	5,65
v60	5,7
v61	5,75
v62	5,8
v63	5,85
v64	5,9
v65	5,95
v66	6
v67	6,05
v68	6,1
v69	6,15
v70	6,2
v71	6,25
v72	6,3
v73	6,35
v74	6,4

5. Update Posisi

Menggunakan persamaan 2.4 yaitu:

$$X_{k+1}^i = X_k^i + V_{k+1}^i$$

Berdasarkan rumus diatas maka perhitungannya sebagai berikut:

$$1. \quad X_0^0 = 0 + 2.7$$

$$X_0^0 = 2.7$$

$$2. \quad X_0^1 = 1 + 2.75$$

$$X_0^1 = 3.75$$

$$3. \quad X_0^2 = 2 + 2.8$$

$$X_0^2 = 4.8$$

Dalam *update* posisi ini, partikel atau yang dalam kasus jadwal ini adalah mata kuliah yang berada pada posisi yang salah (mengalami bentrokan jadwal) maka akan melakukan perpindahan posisi sesuai hasil *update* posisi, nilai posisi baru dapat

dilihat pada tabel 4.15.

Tabel 4. 15 Nilai Posisi (x) Baru

x0	2,7	3
x1	3,75	4
x2	4,8	5
x3	5,85	6
x4	6,9	7
x5	7,95	8
x6	9	9
x7	10,05	10
x8	11,1	11
x9	12,15	12
x10	13,2	13
x11	14,25	14
x12	15,3	15
x13	16,35	16
x14	17,4	17
x15	18,45	18
x16	19,5	20
x17	20,55	21
x18	21,6	22
x19	22,65	23
x20	23,7	24
x21	24,75	25
x22	25,8	26
x23	26,85	27
x24	27,9	28
x25	28,95	29
x26	30	30
x27	31,05	31
x28	32,1	32
x29	33,15	33
x30	34,2	34
x31	35,25	35
x32	36,3	36
x33	37,35	37
x34	38,4	38
x35	39,45	39

x36	40,5	41
x37	41,55	42
x38	42,6	43
x39	43,65	44
x40	44,7	45
x41	45,75	46
x42	46,8	47
x43	47,85	48
x44	48,9	49
x45	49,95	50
x46	51	51
x47	52,05	52
x48	53,1	53
x49	54,15	54
x50	55,2	55
x51	56,25	56
x52	57,3	57
x53	58,35	58
x54	59,4	59
x55	60,45	60
x56	61,5	62
x57	62,55	63
x58	63,6	64
x59	64,65	65
x60	65,7	66
x61	66,75	67
x62	67,8	68
x63	68,85	69
x64	69,9	70
x65	70,95	71
x66	72	72
x67	73,05	73
x68	74,1	74
x69	75,15	75
x70	76,2	76
x71	77,25	77
x72	78,3	78
x73	79,35	79
x74	80,4	80

Jika posisi baru yang didapatkan ditampilkan ke dalam bentuk representasi posisi,
maka dapat dilihat pada tabel 4.16 berikut.

Tabel 4.16 Posisi Data Perkuliahan Baru

Hari	Jam	MK	Dosen	Ruang	Kelas
Senin	07:30 - 10:00	Arsitektur dan Organisasi Komputer	Seh Turuy, S.T., M.Eng.	JT-02	2IF4
	Slot 0				
	14:00 - 16:30	Aljabar Linier	Ir. Suyuti, S.T., M.T., Ph.D., IPM	JT-01	2IF1
	Slot 1				
	08:00 - 10:30	Desain dan Analisis Algoritma	Mohammad Taufik, S.Kom., M.Sc.	JT-04	4IF2
	Slot 2				
	08:00 - 09:40	Pendidikan Kewarganegaraan	Fatoni Achmad, M.Pd.I. & Roy Nurfan, S.Pd., M.Pd.	JT-01	2IF1
	Slot 3				
	10:50 - 14:10	Dasar Pemrograman [+Praktikum]	Yasir Muin, S.T., M.Kom.	Lab. RPL	2IF1
	Slot 4				
	07:30 - 10:50	Pemrograman Berorientasi Objek [+Praktikum]	Ir. Salkin Lutfi, S.Kom., M.T., IPM	Lab. RPL	4IF1
	Slot 5				
	13:20 - 15:50	Rekayasa Perangkat Lunak	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-05	4IF1
	Slot 6				
	10:50 - 14:10	Jaringan Komputer [+Praktikum]	Saiful Do. Abdullah, S.T., M.T.	Lab. JAK	4IF2
	Slot 7				
	07:30 - 10:00	Desain dan Analisis Algoritma	Muhammad Fhadli, S.T., M.Sc.	JT-04	4IF3
	Slot 8				
	14:10 - 16:40	Sistem Informasi Geografis [Kepulauan]	Ir. Arifandy Mario M., S.Kom., M.Kom., IPM	Lab. RPL	6IF1
	Slot 9				
	10:50 - 12:30	Riset Teknologi Informasi	Ir. Arifandy Mario M., S.Kom., M.Kom., IPM	JT-05	6IF2
	Slot 10				

	13:20 - 15:50	Sistem Terdistribusi	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-06	6IF2
	Slot 11				
	08:00 - 10:30	Teori Informasi	Achmad Fuad Assagaf, S.T., M.T.	JT-05	6IF1
	Slot 12				
	07:30 - 10:50	Jaringan Komputer [+Praktikum]	Saiful Do. Abdullah, S.T., M.T.	Lab. JAK	4IF4
	Slot 13				
	09:10 - 10:50	Pendidikan Kewarganegaraan	Fatoni Achmad, M.Pd.I. & Roy Nurfan, S.Pd., M.Pd.	JT-01	2IF3
Selasa	Slot 14				
	07:30 - 09:10	Etika Profesi	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	JT-06	4IF2
	Slot 15				
	07:30 - 10:00	Logika Matematika	Ir. Amal Khairan, S.T., M.Eng., IPM	JT-02	2IF2
	Slot 16				
	07:30 - 10:00	Matematika Diskrit	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-03	2IF4
	Slot 17				
	10:10 - 11:50	Etika Profesi	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	JT-06	4IF3
	Slot 18				
	13:20 - 15:50	Rekayasa Perangkat Lunak	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-01	4IF4
	Slot 19				
	13:20 - 15:50	Aljabar Linier	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-04	2IF2
	Slot 20				
	13:20 - 15:50	Aljabar Linier	Muhammad Natsir, S.T., M.T. & Aisjah Rachmawaty Ryadin, Ph.D.	JT-01	2IF4
	Slot 21				
	07:30 - 10:00	Technopreneur	Alfanugrah Hi. Usman, S.T., M.Kom.	JT-06	6IF2
	Slot 22				
	10:00 - 12:30	Aljabar Linier	Achmad Fuad Assagaf, S.T., M.T.	JT-02	2IF3

	Slot 23				
	07:30 - 10:00	Matematika Diskrit	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-04	2IF1
	Slot 24				
	07:30 - 10:00	Desain dan Analisis Algoritma	Muhammad Fhadli, S.T., M.Sc.	JT-02	4IF1
	Slot 25				
	07:30 - 10:50	Pemrograman Berorientasi Objek [+Praktikum]	Ir. Salkin Lutfi, S.Kom., M.T., IPM	Lab. RPL	4IF2
	Slot 26				
	13:20 - 15:50	Rekayasa Perangkat Lunak	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-06	4IF2
	Slot 27				
	10:00 - 12:30	Keamanan Data	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-06	4IF4
	Slot 28				
	10:30 - 12:10	Sistem Pendukung Keputusan	Alfanugrah Hi. Usman, S.T., M.Kom.	JT-05	6IFP
	Slot 29				
	07:30 - 10:00	Logika Matematika	Ir. Amal Khairan, S.T., M.Eng., IPM	JT-01	2IF4
Rabu	Slot 30				
	07:30 - 10:00	Kecerdasan Buatan	Dr. M. Ridha Albaar, S.Kom., M.Kom.	JT-03	4IF3
	Slot 31				
	13:20 - 15:00	Etika Profesi	Rosihan, S.T., M.Cs.	JT-05	4IF4
	Slot 32				
	07:30 - 10:00	Arsitektur dan Organisasi Komputer	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	JT-05	2IF2
	Slot 33				
	15:00 - 16:40	Bahasa Inggris	Yetty, S.S., M.Hum.	JT-01	2IF2
	Slot 34				
	13:20 - 15:00	Bahasa Inggris	Abdulhalim Daud, S.Pd., M.Pd.	JT-02	2IF1
	Slot 35				
	13:20 - 15:00	Bahasa Inggris	Yetty, S.S., M.Hum.	JT-01	2IF3
	Slot 36				
	10:00 - 12:30	Kecerdasan Buatan	Dr. M. Ridha Albaar, S.Kom., M.Kom.	JT-03	4IF1

	Slot 37				
	13:20 - 15:50	Keamanan Data	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-03	4IF3
	Slot 38				
	15:00 - 16:40	Bahasa Inggris	Abdulhalim Daud, S.Pd., M.Pd.	JT-02	2IF4
	Slot 39				
	09:10 - 10:50	Pendidikan Kewarganegaraan	Hasmawati, S.Ag., M.Pd.+Abd. Firman Bunta, S.Pd., M.Pd	JT-01	2IF2
	Slot 40				
	13:20 - 15:50	Manajemen Proyek Perangkat Lunak	Dr. Assaf Arief, S.T., M.Eng.	JT-01	6IF1
	Slot 41				
	07:30 - 10:00	Teori Informasi	Achmad Fuad Assagaf, S.T., M.T.	JT-06	6IF2
	Slot 42				
	07:30 - 10:00	Kecerdasan Buatan	Dr. M. Ridha Albaar, S.Kom., M.Kom.	JT-02	4IF2
	Slot 43				
	07:30 - 10:00	Matematika Diskrit	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-04	2IF2
Kamis	Slot 44				
	10:50 - 14:10	Dasar Pemrograman [+Praktikum]	Yasir Muin, S.T., M.Kom.	Lab. RPL	2IF3
	Slot 45				
	10:00 - 12:30	Keamanan Data	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-06	4IF1
	Slot 46				
	13:20 - 15:50	Rekayasa Perangkat Lunak	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-03	4IF3
	Slot 47				
	07:30 - 10:00	Desain dan Analisis Algoritma	Muhammad Fhadli, S.T., M.Sc.	JT-04	4IF4
	Slot 48				
	13:20 - 15:00	Sistem Informasi	Rosihan, S.T., M.Cs.	JT-06	6IFP
	Slot 49				
	09:30 - 12:00	Sistem Terdistribusi	Ir. Abdul Mubarak, S.Kom., M.T., IPM	JT-05	6IF1
Slot 50					

	07:30 - 09:10	Etika Profesi	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	JT-05	4IF1
	Slot 51				
	13:20 - 15:50	Keamanan Data	Muhammad Sabri Ahmad, S.Kom., M.Kom.	JT-04	4IF2
	Slot 52				
	07:30 - 10:00	Logika Matematika	Ir. Amal Khairan, S.T., M.Eng., IPM	JT-03	2IF3
	Slot 53				
	14:10 - 16:40	Sistem Informasi Geografis [Kepulauan]	Ir. Arifandy Mario M., S.Kom., M.Kom., IPM	Lab. RPL	6IF2
	Slot 54				
	07:30 - 10:50	Pemrograman Berorientasi Objek [+Praktikum]	Ir. Salkin Lutfi, S.Kom., M.T., IPM	Lab. RPL	4IF3
	Slot 55				
	10:50 - 12:30	Jaringan Terkini	Ir. Arifandy Mario M., S.Kom., M.Kom., IPM	JT-04	6IFP
	Slot 56				
	10:00 - 12:30	Kecerdasan Buatan	Dr. M. Ridha Albaar, S.Kom., M.Kom.	JT-02	4IF4
	Slot 57				
	10:50 - 14:10	Dasar Pemrograman [+Praktikum]	Hairil Kurniadi Sirajuddin, S.Kom., M.Kom.	Lab. RPL	2IF2
	Slot 58				
	07:30 - 10:50	Pemrograman Berorientasi Objek [+Praktikum]	Ir. Salkin Lutfi, S.Kom., M.T., IPM	Lab. RPL	4IF4
	Slot 59				
Jumat	07:30 - 10:00	Logika Matematika	Ir. Amal Khairan, S.T., M.Eng., IPM	JT-04	2IF1
	Slot 60				
	09:30 - 12:00	Arsitektur dan Organisasi Komputer	Seh Turuy, S.T., M.Eng.	JT-02	2IF1
	Slot 61				
	07:30 - 10:00	Matematika Diskrit	Syarifuddin N. Kapita, S.Pd., M.Si.	JT-02	2IF3
	Slot 62				
	10:50 - 12:30	Riset Teknologi Informasi	Dr. Assaf Arief, S.T., M.Eng.	JT-01	6IF1
	Slot 63				

	13:20 - 15:00	Sistem Multimedia	Saiful Do. Abdullah, S.T., M.T.	JT-03	6IFP
	Slot 64				
	07:30 - 10:00	Technopreneur	Alfanugrah Hi. Usman, S.T., M.Kom.	JT-01	6IF1
	Slot 65				
	13:20 - 15:50	Manajemen Proyek Perangkat Lunak	Dr. Assaf Arief, S.T., M.Eng.	JT-02	6IF2
	Slot 66				
	13:20 - 15:50	Arsitektur dan Organisasi Komputer	Seh Turuy, S.T., M.Eng.	JT-06	2IF3
	Slot 67				
	07:30 - 10:50	Jaringan Komputer [+Praktikum]	Saiful Do. Abdullah, S.T., M.T.	Lab. JAK	4IF3
	Slot 68				
	10:50 - 14:10	Jaringan Komputer [+Praktikum]	Ir. Amal Khairan, S.T., M.Eng., IPM	Lab. JAK	4IF1
	Slot 69				
	07:30 - 09:10	Data Mining & Big Data Analytic	Muhammad Fhadli, S.T., M.Sc.	JT-06	6IFP
	Slot 70				
	10:00 - 11:40	Pendidikan Kewarganegaraan	Hasmawati, S.Ag., M.Pd.+Abd. Firman Bunta, S.Pd., M.Pd	JT-01	2IF4
	Slot 71				
	10:50 - 14:10	Dasar Pemrograman [+Praktikum]	Yasir Muin, S.T., M.Kom.	Lab. RPL	2IF4
	Slot 72				
	09:20 - 11:00	Expert System	Muhammad Fhadli, S.T., M.Sc.	JT-06	6IFP
	Slot 73				
	13:20 - 15:00	Digital Forensic	Yasir Muin, S.T., M.Kom.	JT-04	6IFP
	Slot 74				

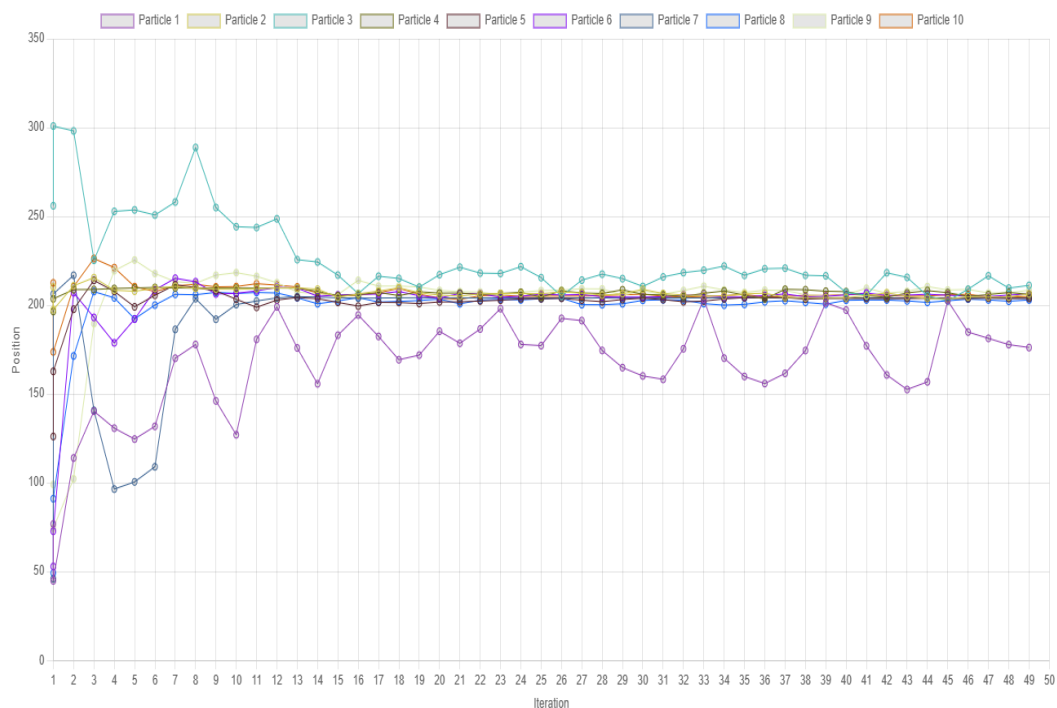
6. Kondisi Berhenti

Terdapat suatu kondisi selesai yang dapat menghentikan proses algoritma *Particle Swarm Optimization* yaitu jika nilai *fitness* terbaik telah dicapai. Jika saat kondisi diatas telah diperoleh oleh program, maka iterasi akan dihentikan dan menampilkan

hasil dari penjadwalan yang telah diperoleh.

7. Grafik Algoritma PSO

Berikut adalah grafik yang menggambarkan proses algoritma pso bergerak ketika melakukan update posisi pada setiap iterasi, dalam grafik tersebut terdapat 10 *particle* dan 50 iterasi. Pada grafik tersebut terlihat bahwa setiap particle pada iterasi awal memiliki posisi yang berjauhan tetapi pada akhir iterasi setiap particle memiliki posisi yang saling berdekatan. Grafik Algoritma PSO dapat dilihat pada gambar 4.17.



Gambar 4.17 Grafik Algoritma PSO

8. Contoh Sampel Perubahan Jadwal

Pada gambar berikut menampilkan sampel perubahan jadwal, pada gambar terlihat pada tabel jadwal awal bapak Salkin lutfi untuk kelas 4IF4 mendapatkan jadwal pada jam 08:00 pagi, namun pada saat update jadwal mendapatkan jadwal pada jam 07:30

pagi. Bapak Achmad fuad untuk kelas 6IF1 pada jadwal awal mendapatkan jadwal jam 08:30 pagi, namun pada saat update jadwal mendapatkan jadwal jam 10:00.

Dapat dilihat pada gambar 4.18 dan 4.19.

Mata Kuliah	Kelas	Ruang	Hari	Dosen	Start Time	End Time
Keamanan Data	4IF1	JT-03	Rabu	Muhammad Sabri Ahmad, S.Kom., M.Kom.	07:30:00	10:00:00
Kecerdasan Buatan	4IF4	Lab RPL	Rabu	Dr. M. Ridha Albaar, S.Kom., M.Kom.	07:40:00	10:10:00
Keamanan Data	4IF4	JT-02	Jumat	Muhammad Sabri Ahmad, S.Kom., M.Kom.	07:50:00	10:20:00
Pemrograman Berorientasi Objek [+Praktikum]	4IF4	JT-01	Kamis	Ir. Salkin Lutfi, S.Kom., M.T., IPM	08:00:00	11:20:00
Kecerdasan Buatan	4IF3	JT-05	Jumat	Dr. M. Ridha Albaar, S.Kom., M.Kom.	08:10:00	10:40:00
Aljabar Linier	2IF3	JT-02	Selasa	Achmad Fuad Assagaf, S.T., M.T.	08:20:00	10:50:00
Teori Informasi	6IF1	Lab Jarkom	Selasa	Achmad Fuad Assagaf, S.T., M.T.	08:30:00	10:10:00
Matematika Diskrit	2IF1	JT-01	Senin	Syarifuddin N. Kapita, S.Pd., M.Si.	08:40:00	11:10:00
Pemrograman Berorientasi Objek [+Praktikum]	4IF3	JT-04	Kamis	Ir. Salkin Lutfi, S.Kom., M.T., IPM	08:50:00	12:10:00
Aljabar Linier	2IF2	JT-06	Senin	Syarifuddin N. Kapita, S.Pd., M.Si.	09:00:00	11:30:00

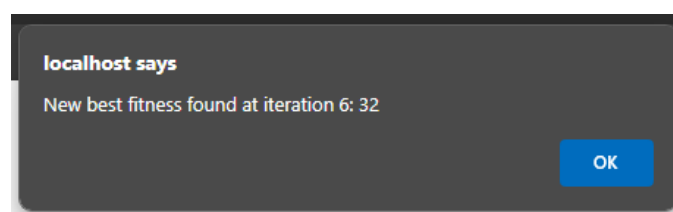
Gambar 4.18 Tabel Sampel Awal

Mata Kuliah	Kelas	Ruang	Hari	Dosen	Start Time	End Time
Keamanan Data	4IF1	JT-03	Rabu	Muhammad Sabri Ahmad, S.Kom., M.Kom.	07:30:00	10:00:00
Kecerdasan Buatan	4IF4	Lab RPL	Rabu	Dr. M. Ridha Albaar, S.Kom., M.Kom.	07:30:00	10:00:00
Keamanan Data	4IF4	JT-02	Jumat	Muhammad Sabri Ahmad, S.Kom., M.Kom.	07:30:00	10:00:00
Pemrograman Berorientasi Objek [+Praktikum]	4IF4	JT-01	Kamis	Ir. Salkin Lutfi, S.Kom., M.T., IPM	07:30:00	10:50:00
Kecerdasan Buatan	4IF3	JT-05	Jumat	Dr. M. Ridha Albaar, S.Kom., M.Kom.	07:30:00	10:00:00
Aljabar Linier	2IF3	JT-02	Selasa	Achmad Fuad Assagaf, S.T., M.T.	07:30:00	10:00:00
Teori Informasi	6IF1	Lab Jarkom	Selasa	Achmad Fuad Assagaf, S.T., M.T.	10:00:00	11:40:00
Matematika Diskrit	2IF1	JT-01	Senin	Syarifuddin N. Kapita, S.Pd., M.Si.	07:30:00	10:00:00
Pemrograman Berorientasi Objek [+Praktikum]	4IF3	JT-04	Kamis	Ir. Salkin Lutfi, S.Kom., M.T., IPM	10:50:00	14:10:00
Aljabar Linier	2IF2	JT-06	Senin	Syarifuddin N. Kapita, S.Pd., M.Si.	10:00:00	12:30:00

Gambar 4.19 Tabel Sampel Baru

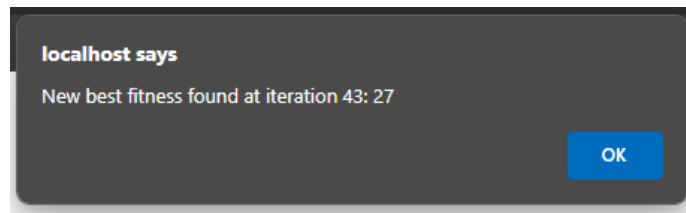
9. Tampilan Notifikasi Fitness

Pada gambar berikut menampilkan notifikasi yang akan tampil ketika mendapat nilai fitness. Dapat dilihat pada gambar 4.20 dan 4.21.



Gambar 4. 20 Notifikasi fitness 1

Pada gambar 4.20 diatas manampilkan notifikasi fitness pada iterasi ke 6 yaitu mendapatkan nilai fitness 32.



Gambar 4. 21 Notifikasi Fitness 2

Pada gambar 4.21 diatas menampilkan notifikasi fitness pada iterasi ke 43 yaitu mendapatkan nilai fitnes

BAB V

PENUTUP

5.1. Kesimpulan

Dari hasil percobaan dan analisis penerapan algoritma *Particle Swarm Optimization* (PSO) untuk permasalahan penjadwalan mata kuliah, dapat disimpulkan sebagai berikut:

1. Implementasi Algoritma *Particle Swarm Optimization* (PSO), belum bisa dikatakan berhasil sepenuhnya diterapkan untuk mengoptimasi permasalahan penjadwalan mata kuliah di jurusan Teknik Informatika Unkhair, karena hanya mendapatkan 1 hasil yang optimal dari beberapa kali percobaan yang dilakukan.
2. Hasil dari penjadwalan mata kuliah menggunakan algoritma PSO yang paling optimal yaitu menggunakan nilai parameter $w=0.9$, $c1=1$, $c2=1$ dengan mendapat Tingkat ke optimalan 100%.
3. Berdasarkan hasil pengujian performa algoritma PSO menggunakan variasi nilai parameter, tidak mendapatkan perbedaan yang signifikan pada hasil penjadwalan mata kuliah, Dimana Tingkat ke optimalan jadwal selalu diatas 90%.

5.2. Saran

Dari beberapa kesimpulan yang telah diambil, maka dapat ditemukan saran-saran yang akan sangat membantu untuk pengembangan sistem ini selanjutnya.

1. Menggunakan metode perhitungan yang berbeda agar dapat membandingkan hasilnya karena semua metode memiliki kelebihan dan kekurangan masing-masing.
2. Konstrain dapat ditambah dan disesuaikan dengan kebutuhan dan persyaratan pembuatan jadwal yang berlaku.

DAFTAR PUSTAKA

- Afira, R., & Wijaya, R., 2021, Penjadwalan Mata Pelajaran Dengan Algoritma Genetika (Studi Kasus Di Smk Negeri 1 Padang). Jurnal KomtekInfo, 8(2).
<https://doi.org/10.35134/komtekinfo.v7i4>
- Andy Antonius Setiawan. (2019). Rancang Bangun Aplikasi Unsrat *E-Catalog*. Jurnal Teknik Informatika, 14(4), 1–9.
- Boru, M., 2020, Algoritma *Ant Colony Optimization* untuk Optimasi Penjadwalan Mata Kuliah. Telekontran: Jurnal Ilmiah Telekomunikasi, Kendali Dan Elektronika Terapan, 7(2), 136–144.
<https://doi.org/10.34010/telekontran.v7i2.2632>
- Gunawan, R., Suherman, Y., & Auliya, N. Z., 2021, Perancangan Sistem Infomasi Pengadaan Barang Berbasis *Web* Pada PT. Sintas Kurama Perdana Karawang. Jurnal Ilmiah Ekonomi Dan Bisnis, 14(1), 101–113.
<http://journal.stekom.ac.id/index.php/E-Bisnis>
- Herpendi, H., Yuliyanti, W., & Aprilia, C., 2020, Aplikasi Amandemen Laporan Survei Ekspor pada PT. Sucofindo Persero Cabang Banjarmasin Berbasis *Web*. Jurnal Sains Dan Informatika, 6(1), 82–91.
<https://doi.org/10.34128/jsi.v6i1.202>
- Joni Kurniawan, W. (2019). Sistem *E-Learning* Do'a dan Iqro' dalam Peningkatan Proses Pembelajaran pada TK Amal Ikhlas. Jurnal Mahasiswa Aplikasi Teknologi Komputer Dan Informasi, 1(3), 154–159.
- Maulana, H., Kasmawi, K., & Enda, D., 2020, Buku Penghubung Berbasis *Android* Menggunakan Metode *Prototyping*. Jurnal Teknik Informatika Dan Sistem Informasi, 6(3).
<https://doi.org/10.28932/jutisi.v6i3.2993>
- Muharni, Y., 2019, Minimasi Makespan Pada Penjadwalan *Flow Shop* Mesin *Paralel* Produk *Steel Bridge* B-60 Menggunakan Metode *Longest Processing Time* Dan *Particle Swarm Optimization*. Jurnal Industrial Services, 4.
- Nasir, A. M., 2022, Optimalisasi Penjadwalan Mata Kuliah Menggunakan Teori Pewarnaan Graf. Jurnal Penelitian Matematika Dan Pendidikan Matematika, 5.

- Nirsal, 2020, Desain Dan Implementasi Sistem Pembelajaran Berbasis *E-Learning* Pada Sekolah Menengah Pertama Negeri 1 Pakue Tengah. *Jurnal Ilmiah d'Computare*, 10.
- Paranduk, L., Indriani, A., Hafid, M., & Sistem Informasi STMIK PPKIA Tarakanita Rahmawati Tarakan, J. (2018). Sistem Informasi Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika Berbasis *Web*. In Seminar Nasional Aplikasi Teknologi Informasi (SNATi).
- Rizki, A. M., & Nurlaili, A. L., 2021, *Algoritme Particle Swarm Optimization* (PSO) untuk Optimasi Perencanaan Produksi Agregat *Multi-Site* pada Industri Tekstil Rumahan. *Journal of Computer, Electronic, and Telecommunication*, 1(2).
<https://doi.org/10.52435/complete.v1i2.73>
- Roihan, A., Nasution, K., & Siambaton, Mhd. Z., 2022, Implementasi Algoritma *Greedy* Kombinasi dengan Perulangan pada Aplikasi Penjadwalan Praktikum. *Sudo Jurnal Teknik Informatika*, 1(2), 42–50.
<https://doi.org/10.56211/sudo.v1i2.8>
- Roziqin, A. C., Cholissodin, I., & Rahayudi, B., 2019, Optimasi Penjadwalan Perkuliahan menggunakan *Hybrid Discrete Particle Swarm Optimization* (Studi Kasus: STAI Ma'had Aly Al-Hikam Malang) (Vol. 3, Issue 11).
<http://j-ptiik.ub.ac.id>
- Tachir Hasibuan, T., 2023, Penerapan Algoritma Adaptif Dalam Mengenali Wajah Manusia Berdasarkan Penjajaran *Minutiae*. *Teknologi Dan Informasi*, 1(1), 1–7.
<https://journal.grahamitra.id/index.php/jurikti>

LAMPIRAN

Source Code

```
<?php
session_start();
include 'config.php';
ini_set('max_execution_time', 1000);

// PSO parameters
$numParticles = 40;
$maxIterations = 50;
$inertiaWeight = 1;
$c1 = 2; // cognitive coefficient
$c2 = 2; // social coefficient

// Record start time for calculating duration
$starttime = microtime(true);

// Other parameters for course scheduling
$resultMK = mysqli_query($koneksi, "SELECT id, kode_mengajar, mata_kuliah, sks, kelas
FROM pengampu");
$resultRuang = mysqli_query($koneksi, "SELECT id, ruang FROM ruang");
$resultHari = mysqli_query($koneksi, "SELECT id, nama FROM hari");
$resultJam = mysqli_query($koneksi, "SELECT id, jam FROM jam");

if (!$resultMK || !$resultRuang || !$resultHari || !$resultJam) {
    echo "Error executing query: " . mysqli_error($koneksi);
    exit;
}

$numMK = mysqli_num_rows($resultMK);
$numRuang = mysqli_num_rows($resultRuang);
$numHari = mysqli_num_rows($resultHari);
$numJam = mysqli_num_rows($resultJam);

$sksValues = [];
while ($row = mysqli_fetch_assoc($resultMK)) {
    $sksValues[$row['id']] = $row['sks'];
}

// Initialize particles
for ($i = 0; $i < $numParticles; $i++) {
    $particle = [];
    for ($j = 0; $j < $numMK; $j++) {
        $particle[$j] = rand(0, $numMK * $numHari - 1);
        $possibleIndices = range(0, $numMK - 1);
        shuffle($possibleIndices);
        $particle = array_slice($possibleIndices, 0, $numMK);
    }
}
```

```

    }
    $particles[] = [
        'position' => $particle,
        'velocity' => array_fill(0, $numMK, 0),
        'bestPosition' => $particle,
        'bestFitness' => cekFitness($particle, $numRuang, $sksValues, $koneksi, $numMK,
$numJam),
    ];
}

```

```

$bestFitness = PHP_INT_MAX;
$bestIteration = -1;
$bestParticle = null;

```

// PSO loop

```

for ($iteration = 0; $iteration < $maxIterations; $iteration++) {
    foreach ($particles as $particleIndex => &$particle) {
        $fitness = cekFitness($particle['position'], $numRuang, $sksValues, $koneksi,
$startTimes, $numJam);

```

```

        if ($fitness < $particle['bestFitness']) {
            $particle['bestPosition'] = $particle['position'];
            $particle['bestFitness'] = $fitness;
        }

```

```

        if ($fitness < $bestFitness) {
            $bestFitness = $fitness;
            $bestIteration = $iteration;
            $bestParticle = $particle['position'];
        }

```

```

        for ($j = 0; $j < $numMK; $j++) {
            $r1 = rand() / getrandmax();
            $r2 = rand() / getrandmax();
            $newVelocity =
                $inertiaWeight * $particle['velocity'][$j] +
                $c1 * $r1 * ($particle['bestPosition'][$j] - $particle['position'][$j]) +
                $c2 * $r2 * ($particles[array_search(getBestParticle($particles),
$particles)]['bestPosition'][$j] - $particle['position'][$j]);

```

```

            $newPosition = $particle['position'][$j] + $newVelocity;
            $newPosition = max(0, min($newPosition, $numMK * $numHari - 1));

```

```

            $particle['velocity'][$j] = $newVelocity;
            $particle['position'][$j] = $newPosition;
        }
    }
}

```

```

if ($bestFitness == 0) {

```

```

        break;
    }
}

```

```

$endtime = microtime(true);
$durationSecond = $endtime - $starttime;
$durations = round($durationSecond);

```

```

function getBestParticle($particles) {
    $bestParticle = $particles[0];
    foreach ($particles as $particle) {
        if ($particle['bestFitness'] < $bestParticle['bestFitness']) {
            $bestParticle = $particle;
        }
    }
    return $bestParticle;
}

```

```

// Display the best schedule
$bestParticle = getBestParticle($particles);
$bestJadwal = $bestParticle['bestPosition'];

```

```

$assignedSlots = [];
$startTimes = [];
while ($row = mysqli_fetch_assoc($resultJam)) {
    $startTimes[] = $row['jam'];
}

```

```

$endTimeLimit = strtotime("18:20");
$startTimeLimit = strtotime("07:30");

```

```

for ($i = 0; $i < $numMK; $i++) {
    $mkIndex = $bestJadwal[$i] % $numMK;
    $ruangIndex = floor($bestJadwal[$i] % $numRuang);
    $hariIndex = $bestJadwal[$i] % $numHari;

```

```

    $mkResult = mysqli_fetch_assoc(mysqli_query($koneksi, "SELECT kode_mengajar,
mata_kuliah, sks, kelas FROM pengampu WHERE id = '$mkIndex'"));
    $ruangResult = mysqli_fetch_assoc(mysqli_query($koneksi, "SELECT ruang FROM
ruang WHERE id = '$ruangIndex'"));
    $hariResult = mysqli_fetch_assoc(mysqli_query($koneksi, "SELECT nama FROM hari
WHERE id = '$hariIndex'"));
    $dosenResult = mysqli_fetch_assoc(mysqli_query($koneksi, "SELECT dosen FROM
pengampu WHERE kode_mengajar = '{$mkResult['kode_mengajar']}'"));
    $duration = $mkResult['sks'] * 50;
    $endTime = strtotime("+" . $duration . " minutes", $startTime);

```

```

$jadwalDosenQuery = "SELECT end_time FROM jadwal WHERE dosen =
'{$dosenResult['dosen']}' AND hari = '{$hariResult['nama']}'";
$jadwalDosenResult = mysqli_query($koneksi, $jadwalDosenQuery);
if ($jadwalDosenResult) {
    $jadwalDosen = mysqli_fetch_all($jadwalDosenResult, MYSQLI_ASSOC);
    $DosenEndTime = null;
    foreach ($jadwalDosen as $jadwal) {
        $jadwalEndTime = strtotime($jadwal['end_time']);
        if ($DosenEndTime === null || $jadwalEndTime > $DosenEndTime) {
            $DosenEndTime = $jadwalEndTime;
        }
    }
    $startTime = $DosenEndTime ?? strtotime($startTimes[$startTimeIndex]);
    if ($startTime < $startTimeLimit) {
        $startTime = $startTimeLimit;
    }
    if ($startTime > $endTimeLimit) {
        $startTime = $endTimeLimit;
    }
    $startTimeIndex = array_search(date('H:i:s', $startTime), $startTimes);
} else {
    $startTime = strtotime($startTimes[$startTimeIndex]);
}

```

```

$jadwalRuangQuery = "SELECT end_time FROM jadwal WHERE ruang =
'{$ruangResult['ruang']}' AND hari = '{$hariResult['nama']}'";
$jadwalRuangResult = mysqli_query($koneksi, $jadwalRuangQuery);
if ($jadwalRuangResult) {
    $jadwalRuang = mysqli_fetch_all($jadwalRuangResult, MYSQLI_ASSOC);
    $ruangEndTime = null;
    foreach ($jadwalRuang as $jadwal) {
        $jadwalEndTime = strtotime($jadwal['end_time']);
        if ($ruangEndTime === null || $jadwalEndTime > $ruangEndTime) {
            $ruangEndTime = $jadwalEndTime;
        }
    }
    $startTime = max($ruangEndTime, $startTime);
    if ($startTime < $startTimeLimit) {
        $startTime = $startTimeLimit;
    }
    if ($startTime > $endTimeLimit) {
        $startTime = $endTimeLimit;
    }
    $startTimeIndex = array_search(date('H:i:s', $startTime), $startTimes);
} else {

```

```

        $startTime = strtotime($startTimes[$startTimeIndex]);
    }

    $jadwalKelasQuery = "SELECT jam, end_time FROM jadwal WHERE kelas =
'{$mkResult['kelas']}' AND hari = '{$hariResult['nama']}'";
    $jadwalKelasResult = mysqli_query($koneksi, $jadwalKelasQuery);
    $kelasEndTime = null;
    if ($jadwalKelasResult) {
        $jadwalKelas = mysqli_fetch_all($jadwalKelasResult, MYSQLI_ASSOC);
        foreach ($jadwalKelas as $jadwal) {
            $jadwalEndTime = strtotime($jadwal['end_time']);
            if ($kelasEndTime === null || $jadwalEndTime > $kelasEndTime) {
                $kelasEndTime = $jadwalEndTime;
            }
        }
        $startTime = max($kelasEndTime, $startTime);
        if ($startTime < $startTimeLimit) {
            $startTime = $startTimeLimit;
        }
        if ($startTime > $endTimeLimit) {
            $startTime = $endTimeLimit;
        }
        $startTimeIndex = array_search(date('H:i:s', $startTime), $startTimes);
    } else {
        $startTime = strtotime($startTimes[$startTimeIndex]);
    }

    $endTime = strtotime("+" . $duration . " minutes", $startTime);

    if ($endTime < $startTimeLimit) {
        $endTime = $startTimeLimit;
    }
    if ($endTime > $endTimeLimit) {
        $endTime = $endTimeLimit;
    }

    $assignedSlots[$ruangResult['ruang']][$hariResult['nama']] = true;

    $result = mysqli_query($koneksi, "INSERT INTO jadwal (hari, jam, end_time,
mata_kuliah, ruang, kelas, dosen)
VALUES ('{$hariResult['nama']}', " . date('H:i:s', $startTime) . ", " . date('H:i:s',
$endTime) . ", '{$mkResult['mata_kuliah']}', '{$ruangResult['ruang']}', '{$mkResult['kelas']}',
'{$dosenResult['dosen']}')");
    if (!$result) {
        echo "Error executing query: " . mysqli_error($koneksi);
        exit;
    }

```

```

    }
    $startTimeIndex++;
    if ($startTimeIndex >= count($startTimes)) {
        $startTimeIndex = 0;
    }
}

$unassignedCourses = [];
while ($rowMk = mysqli_fetch_assoc($resultMK)) {
    $mkId = $rowMk['id'];

    $resultScheduledDays = mysqli_query($koneksi, "SELECT hari FROM jadwal WHERE
mata_kuliah = '$mkId'");

    if (mysqli_num_rows($resultScheduledDays) == 0) {
        $unassignedCourses[] = $mkId;
    }
}

foreach ($unassignedCourses as $mkId) {
    $assignedDays = [];
    $resultAssignedDays = mysqli_query($koneksi, "SELECT DISTINCT hari FROM
jadwal");
    while ($rowAssignedDay = mysqli_fetch_assoc($resultAssignedDays)) {
        $assignedDays[] = $rowAssignedDay['hari'];
    }

    $resultAllDays = mysqli_query($koneksi, "SELECT id FROM hari");
    while ($rowAllDays = mysqli_fetch_assoc($resultAllDays)) {
        $hariId = $rowAllDays['id'];

        if (!in_array($hariId, $assignedDays)) {
            $resultInsertDay = mysqli_query($koneksi, "INSERT INTO jadwal (mata_kuliah,
hari) VALUES ('$mkId', '$hariId')");
            if (!$resultInsertDay) {
                echo "Error executing query: " . mysqli_error($koneksi);
                exit;
            }
            break;
        }
    }
    mysqli_data_seek($resultAssignedDays, 0);
    mysqli_data_seek($resultAllDays, 0);
}
for ($ruangIndex = 0; $ruangIndex < $numRuang; $ruangIndex++) {
    for ($hariIndex = 0; $hariIndex < $numHari; $hariIndex++) {

```

```

        if (empty($assignedSlots[$ruangIndex][$hariIndex])) {
            }
        }
    }
}

```

// Function to check fitness

```

function cekFitness($jadwal, $numHari, $sksValues, $koneksi) {
    $fitness = 0;

```

// Menyimpan jadwal berdasarkan ruang, dosen, dan kelas

```

    $existingSchedule = [
        'ruang' => [],
        'dosen' => [],
        'kelas' => []
    ];

```

```

    foreach ($jadwal as $mkIndex => $position) {
        $ruangIndex = floor($position / $numHari);
        $hariIndex = $position % $numHari;

```

// Mendapatkan SKS dan durasi

```

        $sks = $sksValues[$mkIndex];
        $duration = $sks * 50; // 50 menit per SKS

```

// Mendapatkan nama dosen dan kelas

```

        $resultDosen = mysqli_query($koneksi, "SELECT dosen FROM pengampu
WHERE id = '$mkIndex'");
        $dosen = mysqli_fetch_assoc($resultDosen)['dosen'];

```

```

        $resultKelas = mysqli_query($koneksi, "SELECT kelas FROM pengampu WHERE
id = '$mkIndex'");
        $kelas = mysqli_fetch_assoc($resultKelas)['kelas'];

```

// Cek tumpang tindih ruang

```

        if (isset($existingSchedule['ruang'][$ruangIndex][$hariIndex])) {
            $fitness+= 1; // Conflict in ruang
        } else {
            $existingSchedule['ruang'][$ruangIndex][$hariIndex] = true;
        }

```

// Cek tumpang tindih dosen

```

        if (isset($existingSchedule['dosen'][$dosen][$hariIndex])) {
            $fitness+= 1; // Conflict in dosen
        } else {
            $existingSchedule['dosen'][$dosen][$hariIndex] = true;
        }
    }
}

```

```

// Cek tumpang tindih kelas
if (isset($existingSchedule['kelas'][$kelas][$hariIndex])) {
    $fitness+= 1; // Conflict in kelas
} else {
    $existingSchedule['kelas'][$kelas][$hariIndex] = true;
}
}
return $fitness;
}

```

// Adjust schedules to avoid overlaps

```

function adjustSchedules($koneksi, $startTimes) {
    $jadwalQuery = mysqli_query($koneksi, "SELECT * FROM jadwal ORDER BY dosen,
kelas, ruang, hari, jam");
    if (!$jadwalQuery) {
        echo "Error executing query: " . mysqli_error($koneksi);
        exit;
    }
    $schedules = mysqli_fetch_all($jadwalQuery, MYSQLI_ASSOC);
    foreach ($schedules as $i => $schedule) {
        $startTimes = strtotime($schedule['jam']);
        $end_time = strtotime($schedule['end_time']);
        // Check for the next schedule with the same dosen, class, or room on the same day
        for ($j = $i + 1; $j < count($schedules); $j++) {
            if (
                $schedules[$j]['hari'] == $schedule['hari'] &&
                ($schedules[$j]['dosen'] == $schedule['dosen'] ||
                $schedules[$j]['kelas'] == $schedule['kelas'] ||
                $schedules[$j]['ruang'] == $schedule['ruang'])
            ) {
                $next_start_time = strtotime($schedules[$j]['jam']);
                $next_end_time = strtotime($schedules[$j]['end_time']);
                // Adjust if there is an overlap
                if ($next_start_time < $end_time) {
                    $new_start_time = $end_time;
                    $new_end_time = strtotime("+" . (($next_end_time - $next_start_time) / 60) . "
minutes", $new_start_time);
                    // Update the schedule in the database
                    $updateQuery = "UPDATE jadwal SET jam = " . date('H:i:s', $new_start_time)
. ", end_time = " . date('H:i:s', $new_end_time) . " WHERE id = {"$schedules[$j]['id']}";
                    $updateResult = mysqli_query($koneksi, $updateQuery);
                    if (!$updateResult) {
                        echo "Error executing query: " . mysqli_error($koneksi);
                        exit;
                    }
                }
                // Update the times in the array

```



```
$schedules[$j]['jam'] = date('H:i:s', $new_start_time);  
$schedules[$j]['end_time'] = date('H:i:s', $new_end_time);  
}  
}  
}  
}
```

```
adjustSchedules($koneksi, $startTimes);
```

```
$_SESSION['bestFitness'] = $bestFitness;
$_SESSION['bestIteration'] = $bestIteration;
$_SESSION['durations'] = $durations;
```

```
header("Location: ?page=tampil_genap");
exit;
```

```
return mysqli_affected_rows($koneksi);
?>
```



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : RABU, 24 JULI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR

NPM : 07351811062

Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION
(PSO) DALAM PENJADWALAN MATA KULIAH DI JURUSAN
TEKNIK INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Perbaiki Sesuai catatan dari penguji

Pelajari Perhitungan Vektor

Agc
Ridha

Dosen Pembimbing I,

Ridha

Dr. MUHAMMAD RIDHA ALBAAR, S.Kom., M.Kom.
NIP. 198504232008031001



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : RABU, 24 JULI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR

NPM : 07351811062

Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION
(PSO) DALAM PENJADWALAN MATA KULIAH DI JURUSAN
TEKNIK INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Revisi catatan dari para penguji

Kuasai konsep pemrograman dasar

Kuasai aplikasi office

30/8/2024

Acc Revisi
Huse

Dosen Pembimbing II,

Ir. SALKIN LUTFI, S.Kom., M.T.
NIP. 198601112014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : RABU, 24 JULI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR

NPM : 07351811062

Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION
(PSO) DALAM PENJADWALAN MATA KULIAH DI JURUSAN
TEKNIK INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Selesaikan sesuai permintaan saya ketika ujian

Albaar
Chap

Dosen Penguji

Chap

MUHAMMAD SABRI AHMAD, S.Kom., M.Kom.
NIP. 198905092019031013



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : RABU, 24 JULI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR

NPM : 07351811062

Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION
(PSO) DALAM PENJADWALAN MATA KULIAH DI JURUSAN
TEKNIK INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

1. perhatikan semua masukan saat ujian skripsi! ✓
2. masih terdapat perancangan dan hasil penerapan pso yang belum tepat/kurang! ✓
3. perjelas tahapan algoritma termasuk rumus-rumusnya pada setiap hasil yang dicapai! ✓
4. perbaiki kesimpulan, format penulisan dan lengkapi diagram penjadwalan! ✓

Acc Perbaikan Skripsi
22/08/2024

Gth & Hth

Dosen Penguji II,

ACHMAD FUAD, S.T., M.T.

NIP. 197606182005011001



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN UJIAN SKRIPSI/TUTUP

Dengan ini dinyatakan bahwa pada

Hari / tanggal : RABU, 24 JULI 2024

Pukul : 09:00 - 10:30

Tempat : RUANG PRODI

telah berlangsung Ujian Skripsi/Tutup dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR

NPM : 07351811062

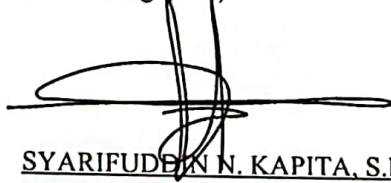
Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION
(PSO) DALAM PENJADWALAN MATA KULIAH DI JURUSAN
TEKNIK INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

1. Iterasi Belu jelas karena gunakan logika yang bukan dari metode
2. pelajari lagi metodenya karena blum paham rumus-rumus, belum tau prosesnya dan harus sama dengan program kemudian jelaskan ke saya

Ag, 19/08/2024

Dosen Penguji III,



SYARIFUDDIN N. KAPITA, S.Pd., M.Si.
NIP. 199103122024211001



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 21 JUNI 2024
Pukul : 13:30 - 15:30
Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR
NPM : 07351811062
Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION (PSO)
DALAM PENJADWALAN MATA KULIAH DI JURUSAN TEKNIK
INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Perbaiki Hasil Pengujian

ACC
Ridha
16-7-2024

Dosen Pembimbing I,

Dr. MUHAMMAD RIDHA ALBAAR, S.Kom., M.Kom.
NIP. 198504232008031001



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 21 JUNI 2024
Pukul : 13:30 - 15:30
Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR
NPM : 07351811062
Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION (PSO)
DALAM PENJADWALAN MATA KULIAH DI JURUSAN TEKNIK
INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

Revisi catatan dari para penguji

Kuasai pemrograman yang digunakan

Kuasai metode yang digunakan

18/7/2024

As revisi
AM

Dosen Pembimbing II,

Ir. SALKIN LUTFI, S.Kom., M.T.
NIP. 198601112014041002



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 21 JUNI 2024

Pukul : 13:30 - 15:30

Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR

NPM : 07351811062

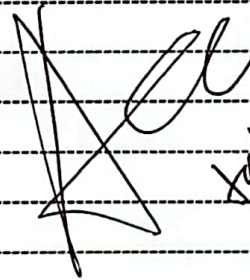
Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION (PSO)
DALAM PENJADWALAN MATA KULIAH DI JURUSAN TEKNIK
INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

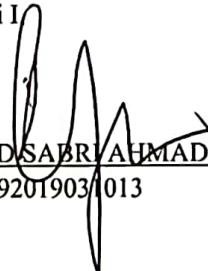
Tambah proses inputan

NIP tidak boleh ada karakter, NIP + NIDN batas

Rumus C_1, C_2, r_1, r_2

 7/15/2024

Dosen Penguji I


MUHAMMAD SABRI AHMAD, S.Kom., M.Kom.
NIP. 198905092019031013



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 21 JUNI 2024
Pukul : 13:30 - 15:30
Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR
NPM : 07351811062
Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION (PSO)
DALAM PENJADWALAN MATA KULIAH DI JURUSAN TEKNIK
INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

1. perbaiki format penulisan ✓
2. revisi desain yang keliru; perhatikan metode, data-data, kriteria — ✓
3. perbaiki flowchart — ?
4. tambahkan sequens dan activity diagram — ? → tdk dlm (opsi)
5. hasil PSO dalam pejadwalan masih belum tuntas dan tambahkan diagram akhir dengan uraian yang tepat ? ✓

Ace Perbin Hani
15/07/2024

Dosen Penguji II,

ACHMAD FUAD, S.T., M.T.
NIP. 197606182005011001



UNIVERSITAS KHAIRUN
FAKULTAS TEKNIK
PROGRAM STUDI INFORMATIKA

DAFTAR PERBAIKAN SEMINAR HASIL SKRIPSI

Dengan ini dinyatakan bahwa pada

Hari / tanggal : JUMAT, 21 JUNI 2024
Pukul : 13:30 - 15:30
Tempat : RUANG PRODI

telah berlangsung Seminar Hasil Skripsi dengan Peserta:

Nama Mahasiswa : HUSEN ALBAAR
NPM : 07351811062
Judul : PENERAPAN ALGORITMA PARTICLE SWARM OPTIMIZATION (PSO)
DALAM PENJADWALAN MATA KULIAH DI JURUSAN TEKNIK
INFORMATIKA UNKHAIR

dinyatakan HARUS menyelesaikan perbaikan, yaitu:

1. slide untuk tutup diganti
2. perbaiki manual penjadwalan karena yg mau di buat penjadwalan bukan evaluasi jadwal buat seperti yg di sistem
3. sitasi harus 5 tahun terkahir

100 15/07/2024

Dosen Penguji III,

SYARIFUDDIN N. KAPITA, S.Pd., M.Si.
NIP. 199103122024211001



KARTU BIMBINGAN HASIL

[illegible]



KARTU BIMBINGAN HASIL

Nama Mahasiswa : Husen Albaar
NIM : 07351811062
Dosen Pembimbing II : Salkin Lutfi, S.Kom., M.T.
Judul : Penerapan algoritma particle swarm optimization (PSO) dalam penjadwalan mata kuliah di jurusan teknik informatika unkhair

NO	Tanggal	Uraian	Paraf
	7 - 5 - 2024	Tambahkan beberapa dasar teori	
		Hasil jangan terlalu banyak bahas sistem, fokus di penerapan algoritma pro	
	19 - 5 - 2024	Buktikan rumus yang di sistem sama dengan rumus di simulasi	
		Simulasi menggunakan data real	
	17 - 5 - 2024	Sesuaikan langkah-langkah simulasi dengan sistem	
		Pelajari simulasinya	
	10 - 6 - 2024	Pelajari kodingan	
		Perbaiki looping algoritma	
		Soemmo Hani	