

---

## Sistem Penjadwalan Sidang Skripsi Dengan Algoritma Genetika Pada Program Studi Informatika Universitas Muhammadiyah Jambi

Muhammad fauzi<sup>1\*</sup>, Zulfikri Akbar<sup>2</sup>, Kevin Kurniawansyah<sup>3</sup>

<sup>1\*,2,3</sup> Fakultas Sains dan Teknologi/Universitas Muhammadiyah Jambi

E-mail: [fauzim876@gmail.com](mailto:fauzim876@gmail.com)<sup>1</sup>, [zulfikriakbar8668@gmail.com](mailto:zulfikriakbar8668@gmail.com)<sup>2</sup>, [kevin.kurniawansyah4h@gmail.com](mailto:kevin.kurniawansyah4h@gmail.com)<sup>3</sup>

\*) untuk penulis corespondensi

---

### Abstract

*A thesis is a requirement for students to obtain a bachelor's and diploma degree at every higher education institution, both public and private. Several processes of the thesis include the submission of a thesis proposal, proposal seminar, thesis guidance, and thesis defense. The large number of students and the availability of lecturers and few rooms cause difficulties in finding schedules that do not collide so that the scheduling process will take a long time. This research aims to design a web-based thesis defense scheduling system for the Informatics Study Program at Universitas Muhammadiyah Jambi. This research uses the waterfall method and UML diagram modeling. In the system design, the PHP (Hypertext Pre-Processor) programming language and MySQL are used as the database. The result of this research is the application of a genetic algorithm in thesis defense scheduling into a designed system that can provide recommendations to the Head of the Study Program in effectively scheduling thesis defenses based on relevant parameters such as students, availability of lecturers, and rooms.*

**Keywords**— Thesis Examination Scheduling, Genetic Algorithm, Waterfall, UML, Website, PHP

---

### Abstrak

Skripsi merupakan sebuah persyaratan bagi mahasiswa untuk mendapatkan gelar sarjana dan diploma di setiap Perguruan Tinggi, baik negeri maupun swasta. Beberapa proses skripsi antara lain pengajuan proposal skripsi, seminar proposal, bimbingan skripsi dan sidang skripsi. Banyaknya jumlah mahasiswa dan tersedianya dosen serta ruangan yang sedikit menyebabkan muncul kesulitan menemukan jadwal yang tidak bertabrakan sehingga proses penjadwalan akan memakan waktu lama. Penelitian ini bertujuan untuk merancang sistem penjadwalan sidang skripsi pada Program Studi Informatika Universitas Muhammadiyah Jambi berbasis web. Penelitian ini menggunakan metode *waterfall* dan permodelan diagram UML. Pada perancangan sistem menggunakan bahasa pemrograman PHP (*Hypertext Pre-Processor*) dan MySQL sebagai databasenya. Hasil dari penelitian ini adalah menerapkan algoritma genetika dalam penjadwalan sidang skripsi ke dalam suatu sistem yang dirancang sehingga dapat memberikan rekomendasi kepada Ketua Program Studi dalam menjadwalkan sidang skripsi secara efektif berdasarkan parameter yang relevan seperti mahasiswa, tersedianya dosen, dan ruang.

**Kata kunci**— Penjadwalan Sidang Skripsi, Algoritma Genetika, *Waterfall*, UML, Website, PHP

---

1. PENDAHULUAN

Pada era globalisasi seperti sekarang, seluruh aspek dalam kehidupan manusia sudah mengalami perkembangan pesat terutama dari segi teknologi.[1] Berkembangnya teknologi ini tentunya mendukung kemajuan di segala bidang terutama dibidang perkuliahan khususnya penjadwalan sidang skripsi.

Skripsi merupakan sebuah persyaratan bagi mahasiswa untuk mendapatkan gelar sarjana dan diploma di setiap Perguruan Tinggi, baik negeri maupun swasta. Beberapa proses skripsi antara lain pengajuan proposal skripsi, seminar proposal, bimbingan skripsi dan sidang skripsi.[2][3]

Penjadwalan sidang adalah masalah menempatkan waktu, ruangan dan penguji sidang kepada mahasiswa yang diuji [4][5]. Jumlah mahasiswa di Universitas Muhammadiyah Jambi sebagai salah satu perguruan tinggi di kota Jambi akan terus bertambah seiring dengan pesatnya pembangunan yang sedang dilakukan. Untuk mendapatkan gelar sarjana, mahasiswa diwajibkan untuk mengambil mata kuliah skripsi yang didalamnya mencakup kegiatan seminar proposal dan sidang akhir skripsi.

Setiap ketua program studi dan dosen di Universitas Muhammadiyah Jambi yang memiliki cukup banyak mahasiswa akan dihadapkan pada sebuah permasalahan setiap periode pendaftaran sidang proposal dan sidang skripsi, yaitu permasalahan dalam pembuatan jadwal sidang skripsi. Banyaknya jumlah mahasiswa dan tersedianya dosen serta ruangan yang sedikit menyebabkan muncul kesulitan menemukan jadwal yang tidak bertabrakan sehingga proses penjadwalan akan memakan waktu lama.

Di Program Studi Informatika Universitas Muhammadiyah Jambi, terdapat beberapa pihak yang terlibat di dalam menjadwalkan sidang skripsi diantaranya ketua program studi, dosen, dan mahasiswa. Ketua program studi memerlukan informasi data mahasiswa termasuk kelengkapan berkas-berkasnya, data dosen untuk menentukan penguji, serta kesesuaian keterampilan dosen penguji 1 dengan judul skripsi mahasiswa sehingga dapat dilakukan penyusunan jadwal dan selanjutnya diserahkan ke Fakultas. Dosen memerlukan informasi mengenai mahasiswa yang

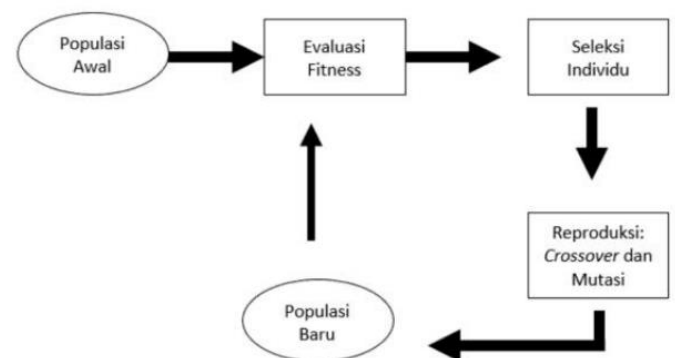
akan diuji serta jadwal sidang apa saja yang akan diikuti. Mahasiswa memerlukan informasi agenda sidang dan informasi mengenai dosen penguji.

Proses penjadwalan sidang skripsi membutuhkan waktu yang lama karena kompleksitasnya yang mempertimbangkan berbagai komponen seperti, mahasiswa, jadwal, dosen pembimbing, dosen penguji dan ruangan,[4] sehingga hal ini menjadi kendala di Program Studi Informatika.

Untuk memudahkan dalam proses penjadwalan tersebut, diperlukan adanya sistem yang dapat memudahkan penjadwalan sidang skripsi. Penelitian ini menggunakan algoritma genetika sebagai metode untuk mencari solusi yang terbaik dari suatu permasalahan penjadwalan sidang skripsi yang berkaitan dengan optimalisasi ruangan yang terbatas dengan kesediaan dosen penguji yang terbatas.

Algoritma Genetika merupakan suatu algoritma *heuristic* yang didasarkan atas mekanisme evolusi biologis. Keberagaman pada evolusi biologis adalah variasi dari kromosom antar individu organisme. Variasi kromosom ini akan mempengaruhi laju produksi dan tingkat kemampuan organisme untuk hidup.[6]

Menurut David Goldberg dalam buku Artificial Intelligence :Konsep Dasar Dan Kajian Praktis, siklus dari Algoritma Genetika dapat dilihat pada gambar:



Gambar 1. Siklus algoritma genetika [7]

a. Inisialisasi Populasi

Inisialisasi populasi awal yaitu tahap untuk membentuk individu awal yang direpresentasikan dalam sejumlah gen (kromosom). Kumpulan kromosom ditempatkan pada penampungan yang disebutkan populasi.

b. Evaluasi *Fitness*

Setiap individu dalam populasi dievaluasi berdasarkan tingkat kecocokannya terhadap masalah yang ingin diselesaikan. Individu dengan tingkat kecocokan yang lebih baik akan mendapatkan nilai *fitness* yang lebih tinggi.

Perhitungan *fitness* pada masing-masing individu dilakukan berdasarkan peraturan, yaitu *constraint*. Berikut adalah perhitungan *fitness* :

$$F = \frac{1}{1 + \sum_{i=1}^n X_i} \quad (1)$$

Ket :

F = *fitness*

x = *penalty*

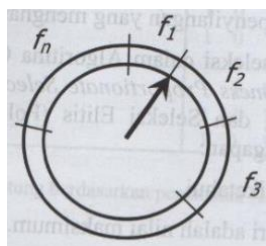
n = banyaknya *penalty* [8]

c. Seleksi

Proses seleksi diterapkan untuk memilih individu dari himpunan populasi dan *offspring* yang dipertahankan hidup pada generasi berikutnya. Makin besar nilai *fitness* sebuah kromosom maka makin besar peluangnya untuk terpilih. Hal ini dilakukan agar terbentuk generasi berikutnya yang lebih baik dari generasi sekarang.[9] Metode seleksi yang sering digunakan adalah *roulette wheel* dan *tournament*.

Metode *Roulette Wheel Selection* ialah proses seleksi menggunakan probabilitas nilai *fitness* sehingga nilai *fitness* yang besar memiliki kesempatan untuk terpilih lebih besar dibandingkan dengan nilai *fitness* yang memiliki nilai lebih kecil. Caranya adalah membangkitkan bilangan acak R dari 0 sampai 1 sebanyak jumlah populasi kromosom.[10][11]

Berikut adalah ilustrasi dari roda rolet dan perumusan dari fungsi distribusi linier, terdapat pada gambar dan persamaan di bawah ini [12]



**Gambar 2. Ilustrasi dari roulette wheel**

Probabilitas *Roulette Wheel* dapat dihitung melalui persamaan berikut:

$$P = \frac{fitness(x)}{\sum_{j=1}^M fitness(x)} \quad (2)$$

j = index atau deret ke-1

M = banyaknya kromosom [13]

d. Reproduksi

Proses reproduksi dengan melakukan *crossover* dan mutasi. *Crossover* ialah operator genetik yang menggabungkan dua kromosom untuk menghasilkan kromosom baru. Ide di baliknya adalah bahwa Kromosom baru mungkin lebih baik daripada kromosom induk jika mewarisi karakteristik terbaik dari masing-masing kromosom. Ada beberapa metode *crossover* yang dapat digunakan yaitu ; Metode *Partially Mapped Crossover*, Metode *One-cut-point Crossover*.

Konsep *crossover* adalah setiap keturunan harus mewarisi sebagian susunan gen (kromosom) dari induknya. Representasi kromosom dapat dilihat pada susunan kromosom di bawah ini:

P2 [01110001]

P5 [01010010]

Diperoleh

C1 [01110010]

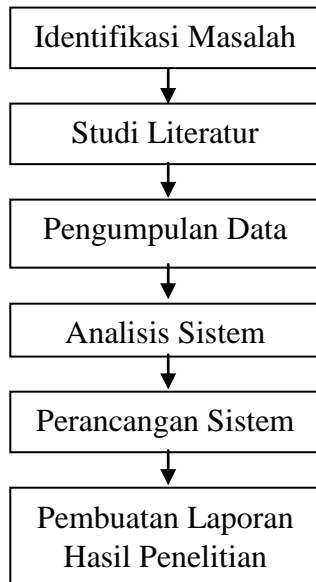
C2 [01010001]

Perhatikan 3 bit pertama dari C1 didapatkan dari P2 dan sisanya dari P5. C2 mewarisi 3 bit awal dari P5 dan sisanya dari P2. Metode ini selanjutnya digunakan dalam penelitian yaitu one-cut-point crossover. Titik potong tukar silang ini ditentukan secara acak.[9]

Selanjutnya selain persilangan genetik (*crossover*), pada pembentukan kromosom baru bisa juga terjadi proses mutasi. Mutasi sebagai aktivitas acak kecil dalam kromosom yang bertujuan untuk memperoleh solusi baru. Proses mutasi diterapkan agar menghasilkan nilai optimum global dan tidak menghasilkan optimum lokal pada proses *random*. Probabilitas mutasi yang baik berada pada kisaran 0 sampai 0,3. Nilai probabilitas mutasi yang terlalu kecil dapat menyebabkan nilai terjebak dalam nilai optimum lokal, dan probabilitas mutasi yang terlalu besar dapat mempersulit dalam memperoleh nilai konvergen.[14]

**2. METODE PENELITIAN**

Kerangka penelitian bertujuan untuk membuat sistem penjadwalan sidang skripsi pada Program Studi Informatika Universitas Muhammadiyah Jambi. Berikut adalah bentuk kerangka penelitian:



**Gambar 3. Kerangka penelitian**

Berdasarkan Kerangka Penelitian diatas, dapat diuraikan sebagai berikut:

a) Identifikasi Masalah

Pada tahap ini, penelitian melakukan identifikasi masalah apa saja yang terjadi saat melakukan penunjukkan penguji dan ketua sidang. Identifikasi masalah dilakukan dengan wawancara pada ketua program studi yang mengatur penunjukkan penguji dan ketua sidang.

b) Studi Literatur

Penulis mempelajari dan memahami teori yang menjadi pedoman dan referensi penelitian-penelitian yang relevan guna penyelesaian masalah. Dimana referensi dan pedoman tersebut penulis ambil dari beberapa sumber yaitu e-book, buku panduan, jurnal, article, laporan-laporan, yang ada hubungannya dengan masalah yang dipecahkan seperti perancangan sistem berbasis web, pengelolaan sidang skripsi, penjadwalan otomatis dan pemrograman, metode penelitian yang digunakan, permodelan sistem yang digunakan, serta *tools* sistem yang digunakan.

c) Pengumpulan Data

1. Observasi

Peneliti melakukan pengamatan secara langsung bagaimana proses pendaftaran sampai pelaksanaan sidang. Hal ini bertujuan untuk mengetahui bagaimana proses sidang skripsi di Program Studi Informatika Universitas Muhammadiyah Jambi.

2. Wawancara

wawancara yang penulis terapkan ialah wawancara tidak terstruktur, dimana tahap ini ialah tahapan peneliti melakukan wawancara dengan ketua Program Studi mengenai proses persidangan skripsi dan membahas tentang jadwal waktu, ruangan seminar dan sidang. Wawancara dilakukan untuk memperoleh informasi mengenai objek penelitian. Wawancara ini bertujuan untuk mengetahui masalah yang dialami Program Studi Informatika, sehingga peneliti dapat menentukan kebutuhan yang diinginkan pengguna pada sistem.

d) Analisis Sistem

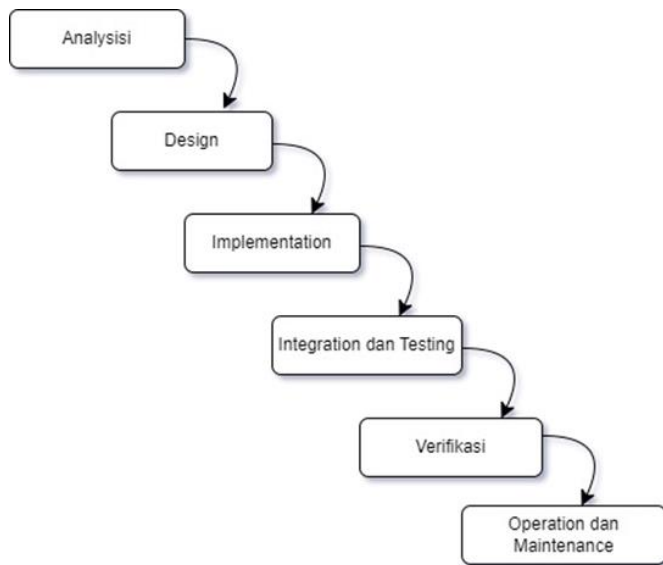
Analisis sistem dilakukan untuk mengidentifikasi permasalahan pada sistem dan proses penjadwalan sidang skripsi pada Program Studi Informatika Universitas Muhammadiyah Jambi. Banyaknya jumlah mahasiswa dan tersedianya dosen serta ruang yang sedikit menyebabkan muncul kesulitan menemukan jadwal yang tidak bertabrakan sehingga proses penjadwalan akan memakan waktu lama.

Setelah menemukan permasalahan yang terjadi, maka pada tahapan analisis sistem ini akan di berikan usulan sistem yang baru guna mengatasi permasalahan yang telah ditemukan pada tahapan analisis kelemahan sistem yang berjalan. Pada tahapan analisis sistem ini. sistem yang diusulkan penulis yaitu sistem penjadwalan sidang skripsi pada program studi informatika Universitas Muhammadiyah Jambi.

e) Perancangan Sistem

Dalam membuat tampilan sistem, penulis merancang dengan menggunakan metode *waterfall* berdasarkan permodelan sistem UML. Pada permodelan UML tersebut terdapat diagram-diagram seperti diagram *use case*, diagram *activity*, dan diagram *class*. Dan metode pengembangan sistem yang digunakan adalah model *waterfall*.

Metode *waterfall* atau model air terjun adalah suatu pendekatan dalam pengembangan perangkat lunak yang menggambarkan pengembangan linier dan berurutan metode ini terdiri dari lima hingga tujuh fase, dimana keseluruhan fase didefinisikan oleh tugas dan tujuan yang berbeda untuk menggambarkan siklus hidup perangkat lunak hingga pengirimannya.[15] Berikut mode dari metode *waterfall* di bawah ini :



Gambar 4. Metode *waterfall* [15]

f) Laporan Hasil Penelitian

Tahap ini merupakan bentuk pembuatan laporan hasil dari penelitian yang telah dilakukan berupa laporan hasil penelitian yang menghasilkan sistem penjadwalan sidang skripsi pada Program Studi Informatika Universitas Muhammadiyah Jambi.

3. HASIL DAN PEMBAHASAN

A. Analisis Permasalahan Sistem

Dalam penjadwalan sidang skripsi memerhatikan kegiatan- kegiatan dosen yang akan menguji dan kesesuaian keterampilan dosen penguji 1 dengan judul skripsi mahasiswa. Pada Program Studi Informatika masih melakukan proses penjadwalan manual serta belum efisien dari segi waktu. Petugas harus memeriksa dan mencoba untuk membuat jadwal untuk setiap dosen ketua penguji, dosen penguji 1, dan dosen penguji 2 serta ruangan agar tidak terjadi bentrok.

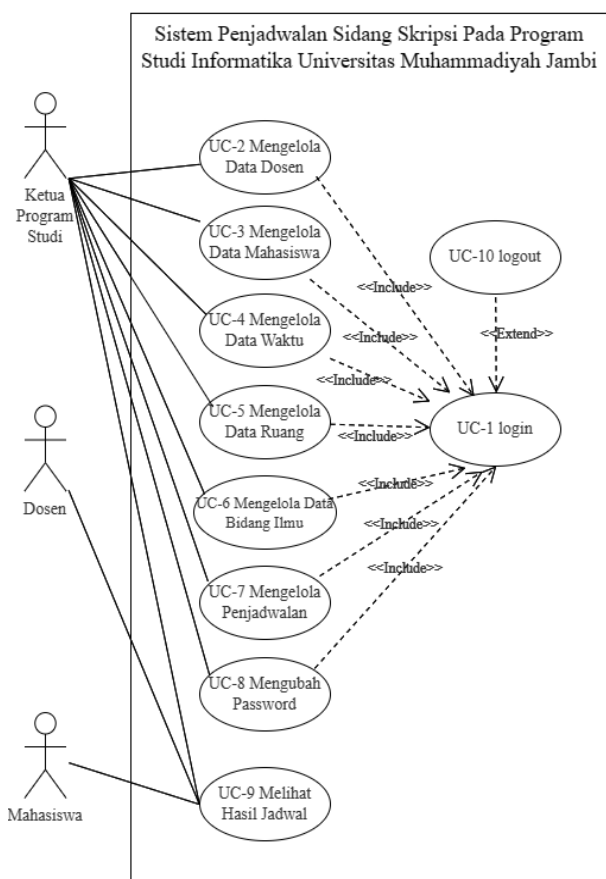
B. Solusi Permasalahan Sistem

Solusi permasalahan sistem yang penulis gunakan dalam menyelesaikan masalah yang ada dapat dijelaskan sebagai berikut:

1. Sebagai pengolahan jadwal sidang skripsi, solusi yang dapat diusulkan yaitu perancangan sistem penjadwalan sidang skripsi berbasis web.
2. Halaman web terbagi menjadi 2 bagian yaitu halaman mahasiswa dan dosen yang akan menampilkan data hasil jadwal yang telah dibuat, dan halaman ketua program studi yang akan menampilkan semua data berupa data mahasiswa, data dosen, data ruang, data waktu, data bidang ilmu, data pengguna, data hasil jadwal.
3. Sebagai penyimpanan data yang besar dan dapat *backup* maka solusi yang diusulkan yaitu penyimpanan *database* pada MySQL.
4. Sistem dapat *generate* secara otomatis jadwal sidang skripsi berdasarkan mahasiswa, ruang, dan dosen yang dapat menguji.
5. Sebagai aspek keamanan sistem terdapat *form* login terlebih dahulu untuk dosen dan ketua program studi.

C. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Analisis kebutuhan sistem Penjadwalan Sidang Skripsi dapat dilihat melalui fungsionalitas yang terdapat pada gambar dibawah ini :



Gambar 5. Diagram use case

Dari gambar diatas dapat diketahui bahwa terdapat 3 pengguna yang akan menggunakan sistem ini dan dapat mengelola data sesuai dengan hak aksesnya masing-masing

**D. Penerapan Algoritma Genetika**

Sebelum menuju ke langkah bagaimana algoritma genetika dalam penjadwalan sidang skripsi, agar jadwal dapat dibuat dengan lebih efektif, maka sejumlah kendala harus diperhatikan. Program Studi Informatika Universitas Muhammadiyah Jambi mempunyai aturan dalam penjadwalan sidang skripsi yaitu:

1. Dalam satu sesi sidang skripsi terdiri dari 3 dosen yaitu dosen ketua penguji, dosen penguji 1, dan dosen penguji 2.
2. Dosen penguji 1 sesuai keterampilan bidang ilmu pada topik judul skripsi.
3. Dosen pembimbing tidak boleh menguji mahasiswa yang dibimbingnya.
4. Sidang skripsi dilakukan pada rentang hari yang telah ditentukan.

5. Terdapat minimal satu ruangan sidang skripsi untuk satu mahasiswa.
6. Setiap sesi sidang skripsi berlangsung 1jam.

Kromosom yang terbentuk berdasarkan variabel seperti di bawah ini:

a. Variabel Bidang Ilmu

Variabel bidang ilmu merupakan Id ilmu yang diambil dari data bidang ilmu sesuai. Variabel ini dikodekan dengan *character* yang direpresentasikan dalam struktur tabel bidang ilmu pada *database*.

**Tabel 1. Variabel bidang ilmu**

Kode Ilmu	Nama Ilmu
B01	Data Mining
B02	Game
B03	Sistem Pakar
B04	Kecerdasan Buatan
B05	Sistem Pendukung Keputusan
----	-----

b. Variabel Dosen

Variabel dosen merupakan Id dosen yang diambil dari data dosen yang akan menguji sidang skripsi. Variabel ini dikodekan dengan *character* yang direpresentasikan dalam struktur tabel dosen yang berelasi pada tabel bidang dosen pada *database*.

**Tabel 2. Variabel dosen**

Kode Dosen	Nama Dosen	Bidang Dosen
D001	Heri Santoso, S.Kom., M.Kom	B02,B05
D002	Dr. Hetty Rohayani, S.T., M.Kom	B01,B03,B04,B05
D003	Zulfikri Akbar, S.Kom., M.S.I	B01,B03,B05
D004	Helmina, S.Kom., M.S.I	B01,B02,B03,B04,B05
D005	Kevin Kurniawansyah, S.Kom., M.Kom	B02,B04,B05
----	-----	-----

c. Variabel Mahasiswa

Variabel mahasiswa merupakan Id mahasiswa yang diambil dari data mahasiswa yang akan melakukan sidang skripsi. Variabel ini dikodekan dengan *character* yang direpresentasikan dalam struktur tabel mahasiswa pada *database*.

**Tabel 3. Variabel mahasiswa**

Kode Mhs	Topik	Nama Mahasiswa	PS 1	PS 2
001	B01	Iqbal ardian	D001	D004
002	B01	Rts. Afrilia	D001	D003
003	B04	Raja aldi M.	D005	D001
004	B05	Rendi efdiansyah	D002	D001
005	B02	M. choirul umam	D002	D003
006	B02	Mia fitriyani	D002	D001
007	B04	M. Fauzi	D003	D005
008	B03	M. Alfareza	D005	D003
009	B05	Dwi nopriyani	D005	D001
010	B05	Safira agustina	D003	D005
----	-----	-----	-----	-----

d. Variabel Ruang

Variabel ruang merupakan Id ruang yang diambil dari data ruang pelaksanaan sidang skripsi. Variabel ini dikodekan dengan *character* yang direpresentasikan dalam struktur tabel ruang pada *database*.

**Tabel 4. Variabel ruang**

Kode Ruang	Nama Ruang
R01	Ruang 1
----	-----

e. Variabel Waktu

Variabel waktu merupakan Id waktu yang diambil dari data waktu dilakukan sidang skripsi yang berisi *slot* kesediaan dosen yang dapat menguji. Variabel ini dikodekan dengan *character* yang direpresentasikan dalam struktur tabel waktu pada *database*.

**Tabel 5. Variabel waktu**

Kode Waktu	Nama Waktu	Sesi Waktu
1	Selasa	I (09.00–10.00)
2		II (10.00-11.00)
3		III (11.00-12.00)
4	Rabu	I (09.00–10.00)
5		II (10.00-11.00)
6		III (11.00-12.00)
7	Kamis	I (09.00–10.00)
8		II (10.00-11.00)
9		III (11.00-12.00)
10	Jumat	I (09.00–10.00)
11		II (10.00-11.00)

Kode Waktu	Nama Waktu	Sesi Waktu
12		III (11.00-12.00)

1. Pembangkitan Populasi Awal

Proses inisialisasi populasi awal dalam penjadwalan sidang skripsi nantinya data- data yang diperoleh akan dikodekan dalam proses algoritma genetika. Data tersebut meliputi data mahasiswa dikodekan (001), data ruang dikodekan (R01), data waktu dikodekan (1), data bidang ilmu dikodekan (B01), data dosen yang memuat ketua penguji dikodekan (D001), penguji 1 dikodekan (D002) dan penguji 2 dikodekan (D003). Sehingga rancangan kromosomnya akan terbentuk seperti gambar di bawah ini:

001	R01	1	B01	D001	D002	D003
-----	-----	---	-----	------	------	------

**Gambar 6. Rancangan kromosom**

Membangkitkan populasi awal diperoleh secara *random* sehingga didapatkan solusi awal. Dimana banyaknya individu/*parent* diasumsikan sebanyak 4 individu sebagai generasi awal dengan kromosom sebanyak jumlah mahasiswa 10 orang. Pembangkitan populasi awal dapat dilihat seperti dibawah ini:

Individu[0]:

([001,R01,9,B01,D005,D002,D003],[002,R01,7,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

Individu[1]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu [2]:



([001,R01,12,B01,D005,D003,D002],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

Individu[3]:

([001,R01,11,B01,D003,D005,D002],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

2. Evaluasi *fitness*

Pada proses ini, individu yang dibangkitkan sebagai solusi penjadwalan sidang skripsi diatas dievaluasi dengan menghitung nilai *fitness* masing-masing individu. Setiap terjadi bentrok pada kromosom maka nilai *penalty* ditambah 1.

Tabel 6. Evaluasi *fitness*

Populasi awal	$\frac{fitness}{1 + \sum_{i=1}^n \chi_i}$
Individu [0]	1/1+(0+1) 0.5
Individu [1]	1/1+(0+1) 0.5
Individu [2]	1/1+(0+1) 0.5
Individu [3]	1/1+(1+1) 0.333
Total <i>fitness</i>	1.833

3. Seleksi

Setelah evaluasi dilakukan, proses seleksi digunakan dalam memilih individu baru untuk reproduksi. Metode seleksi *Roulette Wheel* menggunakan probabilitas

nilai *fitness* yang akan diputar seperti roda rolet sebanyak jumlah populasi/ individu yang nantinya menjadi individu baru untuk proses reproduksi. Dibawah ini proses seleksi *Roulette Wheel* :

Tabel 7. Proses seleksi *roulette wheel*

Populasi	Probabilitas $\frac{fitness x}{\sum_{j=1}^M fitness x_j}$	Probabilitas Kumulatif
Individu[0]	0.273	0.273
Individu[1]	0.273	0.546
Individu[2]	0.273	0.819
Individu[3]	0.181	1
Total	1	

Rolet diatas diputar *random* sebanyak 4 kali, yaitu:

- R[0] : 0.28906986084258
- R[1] : 0.52036975907179
- R[2] : 0.95744712369398
- R[3] : 0.20947720353002

Berdasarkan perandoman menghasilkan individu baru yang terseleksi berdasarkan rolet yang diputar yaitu:

- Individu [0] = Individu [1]
- Individu [1] = Individu [1]
- Individu [2] = Individu [3]
- Individu [3] = Individu [0]

Individu [0]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu [1]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu [2]:

([001,R01,11,B01,D003,D005,D002],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B0



2,D003,D004,D005],[007,R01,9,B04,D004,D002 ,D001],[008,R01,5,B03,D001,D002,D004],[009, R01,2,B05,D004,D003,D002],[010,R01,7,B05,D 002,D001,D004])

Individu [3]:

([001,R01,9,B01,D005,D002,D003],[002,R01,7, B01,D005,D004,D002],[003,R01,12,B04,D002,D 004,D003],[004,R01,3,B05,D005,D003,D004],[0 05,R01,11,B02,D005,D004,D001],[006,R01,8,B0 2,D003,D004,D005],[007,R01,9,B04,D004,D002 ,D001],[008,R01,5,B03,D001,D002,D004],[009, R01,1,B05,D002,D003,D004],[010,R01,10,B05, D004,D002,D001])

Perhitungan *fitness* setelah dilakukan seleksi individu baru:

**Tabel 8. Fitness tahap seleksi**

Populasi Baru	<i>fitness</i>	
	1	$\frac{1}{1 + \sum_{i=1}^n \chi_i}$
Individu [0]	1/ 1+( 0+1)	0.5
Individu [1]	1/ 1+( 0+1)	0.5
Individu [2]	1/1+(1+1)	0.333
Individu [3]	1/ 1+( 0+1)	0.5
Total <i>fitness</i>		1.833

4. Reproduksi

1) *Crossover*

Individu yang telah terseleksi kemudian dilakukan proses *crossover* secara acak sesuai dengan nilai populasi yaitu : 0-3 dengan syarat *crossover ratio* : 0,7. Sehingga didapat individu yang akan dilakukan *crossover* seperti berikut:

Individu [0] : individu [1]

Individu [1] : Individu [2]

Individu [2] : Individu [3]

Individu [3] : Individu [1]

Dibawah ini merupakan proses *crossover* dengan metode *one-cut-point crossover*.

*Crossover offspring*

individu [0] :

([001,R01,4,B01,D005,D002,D003],[002,R01,4, B01,D005,D004,D002],[003,R01,12,B04,D002,D 004,D003],[004,R01,6,B05,D003,D004,D005],[0 05,R01,11,B02,D005,D004,D001],[006,R01,8,B0 2,D003,D004,D005],[007,R01,9,B04,D004,D002 ,D001],[008,R01,5,B03,D001,D002,D004],[009, R01,2,B05,D004,D003,D002],[010,R01,7,B05,D 002,D001,D004])

,D001],[008,R01,5,B03,D001,D002,D004],[009, R01,2,B05,D004,D003,D002],[010,R01,7,B05,D 002,D001,D004])

x

individu [1]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4, B01,D005,D004,D002],[003,R01,12,B04,D002,D 004,D003],[004,R01,6,B05,D003,D004,D005],[0 05,R01,11,B02,D005,D004,D001],[006,R01,8,B0 2,D003,D004,D005],[007,R01,9,B04,D004,D002 ,D001],[008,R01,5,B03,D001,D002,D004],[009, R01,2,B05,D004,D003,D002],[010,R01,7,B05,D 002,D001,D004])

individu [0] :

([001,R01,4,B01,D005,D002,D003],[002,R01,4, B01,D005,D004,D002],[003,R01,12,B04,D002,D 004,D003],[004,R01,6,B05,D003,D004,D005],[0 05,R01,11,B02,D005,D004,D001],[006,R01,8,B0 2,D003,D004,D005],[007,R01,9,B04,D004,D002 ,D001],[008,R01,5,B03,D001,D002,D004],[009, R01,2,B05,D004,D003,D002],[010,R01,7,B05,D 002,D001,D004])

*Crossover offspring*

individu [1]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4, B01,D005,D004,D002],[003,R01,12,B04,D002,D 004,D003],[004,R01,6,B05,D003,D004,D005],[0 05,R01,11,B02,D005,D004,D001],[006,R01,8,B0 2,D003,D004,D005],[007,R01,9,B04,D004,D002 ,D001],[008,R01,5,B03,D001,D002,D004],[009, R01,2,B05,D004,D003,D002],[010,R01,7,B05,D 002,D001,D004])

x

individu [2]:

([001,R01,11,B01,D003,D005,D002],[002,R01,6, B01,D005,D004,D002],[003,R01,12,B04,D002,D 004,D003],[004,R01,3,B05,D005,D003,D004],[0 05,R01,11,B02,D005,D004,D001],[006,R01,8,B0 2,D003,D004,D005],[007,R01,9,B04,D004,D002 ,D001],[008,R01,5,B03,D001,D002,D004],[009, R01,2,B05,D004,D003,D002],[010,R01,7,B05,D 002,D001,D004])

individu [1] :

([001,R01,4,B01,D005,D002,D003],[002,R01,4, B01,D005,D004,D002],[003,R01,12,B04,D002,D 004,D003],[004,R01,6,B05,D003,D004,D005],[0

05,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

*Crossover offspring*

individu [2]:

([001,R01,11,B01,D003,D005,D002],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

x

individu [3]:

([001,R01,9,B01,D005,D002,D003],[002,R01,7,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

individu [2] :

([001,R01,11,B01,D003,D005,D002],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

*Crossover offspring*

Individu [3]:

([001,R01,9,B01,D005,D002,D003],[002,R01,7,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

X

Individu [1]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu[3]:

([001,R01,9,B01,D005,D002,D003],[002,R01,7,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu[0]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu[1]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu[2]:

([001,R01,11,B01,D003,D005,D002],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

Individu[3]:  
 ([001,R01,9,B01,D005,D002,D003],[002,R01,7,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Perhitungan *fitness* kembali

**Tabel 9. Fitness tahap crossover**

Populasi Baru	<i>fitness</i>	
	1	$\frac{1}{1 + \sum_{i=1}^n \chi^i}$
Individu [0]	1/ 1+ (0+1)	0.5
Individu [1]	1/ 1+ (0+1)	0.5
Individu [2]	1/ 1+ (0+1)	0.5
Individu [3]	1/ 1+ (0+2)	0.333
Total <i>fitness</i>		1.833

Berdasarkan perhitungan *fitness* diatas belum terdapat nilai *fitness* 1, maka selanjutnya dilakukan proses mutasi.

2) Mutasi

Setelah *crossover*, dilakukan mutasi untuk pertukaran gen ke dalam kromosom yang ditentukan secara acak berdasarkan nilai rasio 0,3. Proses mutasi dapat dilihat sebagai berikut:

Sebelum mutasi :

Individu[0]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu[1]:

([001,R01,4,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002

,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu[2]:

([001,R01,11,B01,D003,D005,D002],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

Individu[3]:

([001,R01,9,B01,D005,D002,D003],[002,R01,7,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Setelah mutasi:

Hasil Mutasi Individu[0]:

([001,R01,3,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Hasil Mutasi Individu[1]:

([001,R01,12,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Hasil Mutasi Individu[2]:

([001,R01,8,B01,D002,D003,D005],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B0

2,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

Hasil Mutasi Individu[3]:

([001,R01,7,B01,D003,D005,D002],[002,R01,7,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Selanjutnya didapat populasi baru setelah melalui 1 tahap iterasi algoritma genetika sebagai berikut:

Individu[0]:

([001,R01,3,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu[1]:

([001,R01,12,B01,D005,D002,D003],[002,R01,4,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,6,B05,D003,D004,D005],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Individu[2]:

([001,R01,8,B01,D002,D003,D005],[002,R01,6,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,1,B05,D002,D003,D004],[010,R01,10,B05,D004,D002,D001])

Individu[3]:

([001,R01,7,B01,D003,D005,D002],[002,R01,7,B01,D005,D004,D002],[003,R01,12,B04,D002,D004,D003],[004,R01,3,B05,D005,D003,D004],[005,R01,11,B02,D005,D004,D001],[006,R01,8,B02,D003,D004,D005],[007,R01,9,B04,D004,D002,D001],[008,R01,5,B03,D001,D002,D004],[009,R01,2,B05,D004,D003,D002],[010,R01,7,B05,D002,D001,D004])

Perhitungan *fitness* kembali

**Tabel 10. Fitness akhir iterasi genetika**

Populasi Baru	<i>fitness</i>	
	1	
	$1 + \sum_{i=1}^n \chi^i$	
Individu [0]	1/1+(0)	1
Individu [1]	1/1+(0+1)	0.5
Individu [2]	1/1+(0+1)	0.5
Individu [3]	1/1+(1+1+1)	0.25
Total <i>fitness</i>		2.25

Berdasarkan perhitungan *fitness* diatas dapat diketahui bahwa individu [0] memiliki nilai *fitness* 1 yaitu merupakan solusi untuk menjadwalkan sidang skripsi pada Program Studi Informatika. Susunan solusi dapat dilihat pada tabel 11 dibawah ini:

**Tabel 11. Solusi penjadwalan dengan algoritma genetika**

Nama Mahasiswa	Ketua Peguji	Penguji 1	Penguji 2	Waktu	Ruang
Dwi nopriyani	Helmina, S.Kom., M.S.I	Zulfikri Akbar, S.Kom., M.S.I	Dr. Hetty Rohayani, S.T., M.Kom	Selasa (09.00–10.00)	Ruang 1
Iqbal ardian	Kevin Kurniawansyah, S.Kom., M.Kom	Dr. Hetty Rohayani, S.T., M.Kom	Zulfikri Akbar, S.Kom., M.S.I	Selasa (11.00-12.00)	Ruang 1
Rts. Afrilia indaswari	Kevin Kurniawansyah, S.Kom., M.Kom	Helmina, S.Kom., M.S.I	Dr. Hetty Rohayani, S.T., M.Kom	Rabu (09.00–10.00)	Ruang 1
Muhammad	Heri Santoso, S.Kom.,	Dr. Hetty Rohayani,	Helmina, S.Kom., M.S.I	Rabu (10.00-	Ruang 1

Nama Mahasiswa	Ketua Peguji	Penguji 1	Penguji 2	Waktu	Ruang
Alfareza	M.Kom	S.T., M.Kom		11.00	
Rendi efdiansyah	Zulfikri Akbar, S.Kom., M.S.I	Helmina, S.Kom., M.S.I	Kevin Kurniawansyah, S.Kom., M.Kom	Rabu (11.00-12.00)	Ruang 1
Safira agustina	Dr. Hetty Rohayani, S.T., M.Kom	Heri Santoso, S.Kom., M.Kom	Helmina, S.Kom., M.S.I	Kamis (09.00-10.00)	Ruang 1
Mia fitriyani	Zulfikri Akbar, S.Kom., M.S.I	Helmina, S.Kom., M.S.I	Kevin Kurniawansyah, S.Kom., M.Kom	Kamis (10.00-11.00)	Ruang 1
Muhammad Fauzi	Helmina, S.Kom., M.S.I	Dr. Hetty Rohayani, S.T., M.Kom	Heri Santoso, S.Kom., M.Kom	Kamis (11.00-12.00)	Ruang 1
M. choiril umam	Kevin Kurniawansyah, S.Kom., M.Kom	Helmina, S.Kom., M.S.I	Heri Santoso, S.Kom., M.Kom	Jumat (10.00-11.00)	Ruang 1
Raja aldi M.	Dr. Hetty Rohayani, S.T., M.Kom	Helmina, S.Kom., M.S.I	Zulfikri Akbar, S.Kom., M.S.I	Jumat (11.00-12.00)	Ruang 1

**E. IMPLEMENTASI SISTEM**

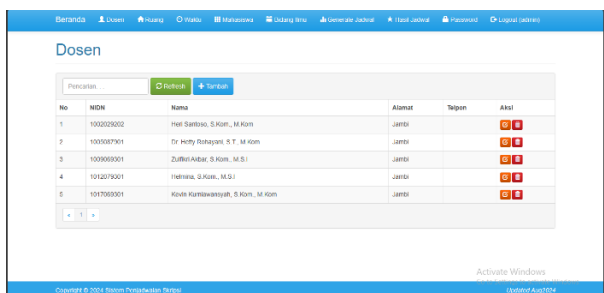
**1. Tampilan halaman login**



**Gambar 7. Tampilan halaman login**

Berdasarkan gambar 7 halaman ini adalah halaman login dimana Ketua Program Studi dapat memasuki sistem dengan menginputkan *username* dan *password* kemudian klik masuk setelah itu Ketua Program Studi dapat melakukan pengelolaan sistem.

**2. Tampilan Halaman Dosen**

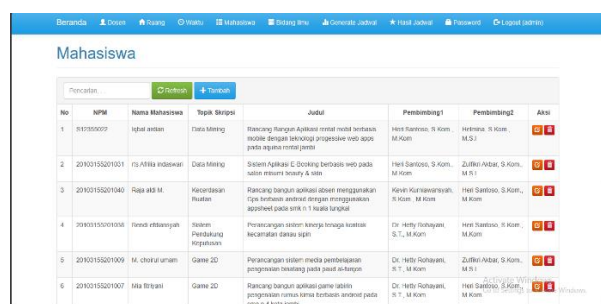


**Gambar 8. Tampilan halaman dosen**

Berdasarkan gambar 8 halaman ini adalah halaman dosen dimana Ketua Program Studi

dapat mengelola data dosen seperti menambah, mengedit, dan menghapus data dosen.

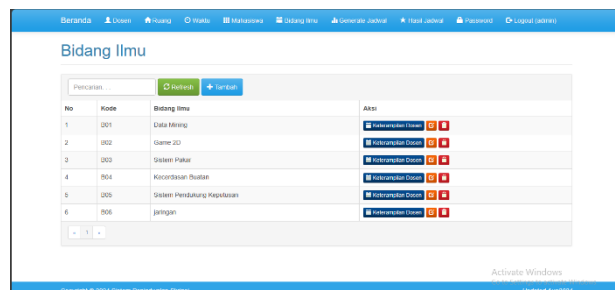
**3. Tampilan Halaman Mahasiswa**



**Gambar 9. Tampilan halaman mahasiswa**

Berdasarkan gambar 9 halaman ini adalah halaman mahasiswa dimana Ketua Program Studi dapat mengelola data mahasiswa seperti menambah, mengedit, dan menghapus data mahasiswa.

**4. Tampilan Halaman Bidang Ilmu**



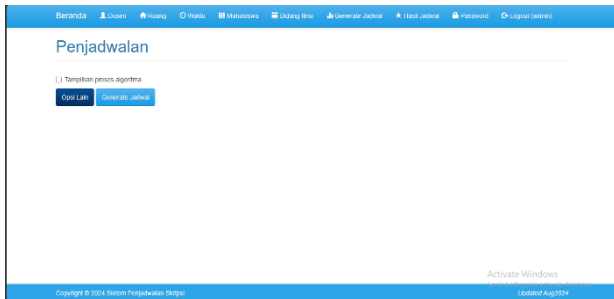
**Gambar 10. Tampilan halaman bidang ilmu**

Berdasarkan gambar 10 halaman ini adalah halaman bidang ilmu dimana Ketua Program



Studi dapat mengelola data bidang ilmu seperti menambah, mengedit, dan menghapus data bidang ilmu.

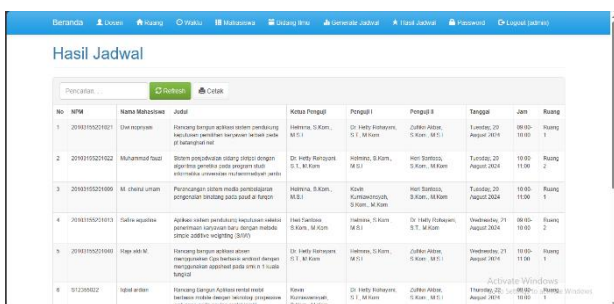
5. Tampilan Halaman Generate Jadwal



Gambar 11. Tampilan halaman generate jadwal

Pada gambar 11 halaman *generate* jadwal merupakan halaman dimana Ketua Program Studi dapat mengelola penjadwalan secara otomatis dengan meng-klik tombol *generate* sebelum itu Ketua Program Studi harus menginputkan banyaknya individu yang dibangkitkan, maksimal generasi yang diharapkan, *crossover rate*, dan mutasi *rate*. Tampilan halaman *generate* jadwal dapat dilihat pada gambar dibawah ini:

6. Tampilan Halaman Hasil Jadwal



Gambar 12. Tampilan halaman hasil jadwal

Berdasarkan gambar 12 halaman ini adalah halaman hasil jadwal dimana proses penjadwalan yang dilakukan Ketua Program Studi sebelumnya didapatkan sebuah hasil jadwal sidang skripsi.

4. SIMPULAN

Berdasarkan hasil penelitian yang telah penulis lakukan maka diambil beberapa kesimpulan, yaitu:

1. Perancangan sistem penjadwalan sidang skripsi dengan Algoritma Genetika pada Program Studi Informatika Universitas Muhammadiyah Jambi dapat memberikan informasi tentang jadwal sidang skripsi mahasiswa.
2. Sistem penjadwalan sidang skripsi dengan Algoritma genetika dapat membantu Ketua Program Studi dalam melakukan penjadwalan sidang skripsi dengan efektif dan efisien.
3. Sistem penjadwalan sidang skripsi ini dapat menentukan kesesuaian keterampilan bidang ilmu dosen penguji utama dengan judul skripsi mahasiswa.

5. SARAN

Dengan laporan penelitian ini, penulis menyarankan sebagai berikut:

1. Penelitian selanjutnya, sistem penjadwalan sidang skripsi dapat dikembangkan dengan menambahkan penjadwalan mata kuliah serta integrasi dengan sistem informasi akademik yang sudah ada.
2. Peneliti selanjutnya dapat mempertimbangkan penggunaan algoritma penjadwalan lainnya untuk membandingkan performa dan efektivitasnya dengan Algoritma Genetika.
3. Untuk mendukung kelancaran dan kinerja sistem, maka harus adanya pemeliharaan terhadap sistem tersebut agar efektivitas sistem dapat terus berjalan dengan baik dan optimal.

DAFTAR PUSTAKA

[1] Z. Akbar, M. Ikhsan, Helmina, R. Dani, and T. Hierdawati, "Peningkatan Keterampilan Pengelolaan Proyek Sistem Informasi Melalui Pelatihan Pembuatan Website Bagi Mahasiswa Universitas Muhammadiyah Jambi," *Jurnal Suara Pengabdian* 45, vol. 2, no. 2, pp. 54–65, 2023.

[2] S. Lena, "Perancangan Dan Implementasi Sistem Manajemen Proposal Skripsi (Studi Kasus Jurusan Manajemen Informatika Politeknik Negeri Sambas)," *Klik - Kumpulan Jurnal Ilmu Komputer*, vol. 8,



- no. 1, p. 73, 2021, doi: 10.20527/klik.v8i1.373.
- [3] F. S. Suwita, “Pengembangan Sistem Informasi Tugas Akhir dan Skripsi (SIMITA) di Universitas Komputer Indonesia (UNIKOM),” *Jurnal Teknologi dan Informasi (JATI)*, vol. 10, pp. 71–82, 2020, doi: 10.34010/jati.v10i1.
- [4] E. S. Nurjanah and A. D. Indriyanti, “Sistem Informasi Rekomendasi Penjadwalan Sidang Skripsi Dengan Metode Depth First Search Pada Jurusan Teknik Informatika Universitas Negeri Surabaya,” *Journal of Emerging Information System and Business Intelligence (JEISBI)*, vol. 2, no. 1, pp. 8–14, 2021, [Online]. Available: <https://ejournal.unesa.ac.id/index.php/JEISBI/article/view/37725%0Ahttps://ejournal.unesa.ac.id>
- [5] D. Pamungkas, “Rancang Bangun Sistem Informasi Penjadwalan Ujian Tugas Akhir (SIJUKIR) berbasis Web,” 2020.
- [6] D. Oktarina and A. Hajjah, “Perancangan Sistem Penjadwalan Seminar Proposal dan Sidang Skripsi dengan Metode Algoritma Genetika,” *JOISIE (Journal Of Information Systems And Informatics Engineering)*, vol. 3, no. 1, p. 32, 2019, doi: 10.35145/joisie.v3i1.421.
- [7] A. M. Dawis *et al.*, *Artificial Intelligence : Konsep Dasar Dan Kajian Praktis*. Tohar Media, 2022.
- [8] M. Julkarnain, “Algoritma Genetika adalah teknik optimasi yang terinspirasi dari proses evolusi alami, menggunakan konsep seleksi,” vol. 1, no. 2, 2023.
- [9] W. F. Mahmudy, A. W. Widodo, and A. Muntasa, *Membangun Sistem Cerdas dengan Kecerdasan Buatan*. Universitas Brawijaya Press, 2023.
- [10] L. Qomariyah, “SISTEM PENJADWALAN UJIAN TAHFIDZ DI PONDOK PESANTREN AL BAROKAH MENGGUNAKAN ALGORITMA GENETIKA,” 2024.
- [11] J. Y. Setiawan, E. Dyah Herwindiati, and T. Sutrisno, “Algoritma Genetika Dengan Roulette Wheel Selection Dan Arithmetic Crossover Untuk Pengelompokan,” *Jurnal Ilmu Komputer dan Sistem Informasi*, pp. 58–64, 2019.
- [12] B. B. Aji, S. B. Bhaskoro, S. Aminah, T. R. Otomasi, and P. M. Bandung, “Sistem penjadwalan sidang tugas akhir menggunakan algoritma genetika,” vol. 7, pp. 27–36, 2021.
- [13] V. N. Wijyaningrum, *Konsep Dan Penerapan Algoritma Genetika*. Banda Aceh, Aceh: Syiah Kuala University Press, 2021.
- [14] I. Robandi, *Artificial Intelligence- Mengupas Rekayasa Kecerdasan Tiruan*. Yogyakarta: Indonesia: ANDI OFFSET, 2019.
- [15] F. N. Hasanah, *Buku Ajar Rekayasa Perangkat Lunak*. 2020. doi: 10.21070/2020/978-623-6833-89-6.