

PENERAPAN ALGORITMA GENETIKA UNTUK PENJADWALAN SIDANG SKRIPSI PADA SISTEM INFORMASI PELAYANAN SKRIPSI INFORMATIKA UNVERSTIAS TEKNOLOGI SUMBAWA

Wiki Nasmansyah¹, M. Julkarnain^{*1}

¹Informatika, Rakayasa Sistem, Universitas Teknologi Sumbawa, Indonesia

m.julkarnain@uts.ac.id

ABSTRAK

Algoritma Genetika adalah teknik optimasi yang terinspirasi dari proses evolusi alami, menggunakan konsep seleksi, *crossover*, dan mutasi untuk mencari solusi optimal pada permasalahan optimasi penjadwalan. Metode RAD (*Rapid Application Development*) digunakan dalam pengembangan sistem untuk memastikan pengembangan yang cepat, efisien, dan adaptif. Dalam penelitian ini, Algoritma Genetika digunakan untuk mencari jadwal sidang skripsi yang efisien berdasarkan ketersediaan dosen penguji dan preferensi waktu mahasiswa. Proses Algoritma Genetika bekerja seperti evolusi alami, di mana individu-individu terbaik dipilih untuk berkembang biak melalui operasi *crossover* dan mutasi, sehingga menghasilkan generasi baru yang lebih baik secara iteratif. Metode RAD memungkinkan pengembangan sistem yang cepat dan adaptif melalui iterasi berulang antara pengembangan dan evaluasi. Dengan pendekatan ini, penelitian ini bertujuan untuk meningkatkan efisiensi dan kualitas penjadwalan sidang skripsi, mengoptimalkan pemanfaatan sumber daya, serta meminimalkan konflik jadwal. Hasil penelitian menunjukkan bahwa penerapan Algoritma Genetika dengan metode RAD berhasil menghasilkan jadwal sidang skripsi yang lebih optimal dan efisien, sehingga dapat membantu program studi Informatika Universitas Teknologi Sumbawa dalam melakukan penjadwalan skripsi yang relative cepat dan akurat.

Kata kunci: Penerapan, Algoritma Genetika, Penjadwalan, *Rapid Application Development*

ABSTRACT

Genetic Algorithm is an optimization technique inspired by natural evolution processes, utilizing the concepts of selection, crossover, and mutation to find optimal solutions for combinatorial optimization problems such as scheduling. RAD (Rapid Application Development) method is employed in the system development to ensure fast, efficient, and adaptive progress. In this study, Genetic Algorithm is applied to efficiently schedule thesis defense sessions based on the availability of examiners and students' time preferences. The Genetic Algorithm process mimics natural evolution, where the fittest individuals are selected to reproduce through crossover and mutation operations, leading to the generation of improved solutions iteratively. RAD method enables rapid and adaptive system development through iterative cycles of development and evaluation. With this approach, the research aims to enhance the efficiency and quality of thesis defense scheduling, optimizing resource utilization, and minimizing scheduling conflicts. The results demonstrate that implementing Genetic Algorithm with RAD method successfully generates more optimal and efficient thesis defense schedules, thereby assisting the Informatics program at Universitas Teknologi Sumbawa in conducting relatively quick and accurate scheduling of thesis defense sessions.

Keywords: *Deployment, Genetic Algorithm, Scheduling, Rapid Application Development*

PENDAHULUAN

Penjadwalan seminar proposal skripsi dan sidang tugas akhir skripsi merupakan proses penting dalam menyelesaikan proses perkuliahan pada suatu perguruan tinggi, baik perguruan tinggi negeri maupun swasta. Pentingnya penjadwalan yang efektif, efisien, dapat menghindari pemborosan waktu proses penjadwalan. Terdapat beberapa faktor yang mempengaruhi

penjadwalan, seperti ketersediaan jadwal dosen yang akan menjadi penguji, ketersediaan ruangan untuk seminar proposal skripsi dan sidang skripsi, ketersediaan peserta seminar dan sidang skripsi, waktu yang tersedia untuk pelaksanaan seminar dan sidang skripsi, dan jumlah mahasiswa yang mengambil seminar proposal dan sidang skripsi. Semua faktor ini harus dipertimbangkan secara matang dalam proses penjadwalan agar tercipta penjadwalan yang ideal.

Berdasarkan hasil pengamatan dan wawancara awal penulis dengan sekretaris program studi pada program studi Informatika Universitas Teknologi Sumbawa, penjadwalan sidang tugas akhir yang digunakan masih dilakukan dengan cara konvensional yaitu melakukan penjadwalan menggunakan *Microsoft excel*, dan untuk sistem informasi pelayanan skripsi yang telah dibuat belum di implementasikan, karena pada fitur proses penjadwalan belum dibuatkan secara otomatis sehingga menyulitkan sekretaris program studi dalam mengatur proses penjadwalan sidang akhir skripsi yang ideal (M.Julkarnain, 2023). Hal ini karena pihak program studi harus melakukan penginputan di menu admin sistem secara manual untuk memeriksa ketersediaan Dosen Penguji dan ruang kelas.

Dari latarbelakang permasalahan diatas penulis memiliki sebuah solusi dalam menyelesaikan permasalahan tersebut dengan membuat sistem yang dapat membantu dalam proses penjadwalan sidang skripsi yang otomatis yaitu dengan menerapkan algoritma genetika sehingga penjadwalan dapat dilakukan dengan waktu yang relative cepat.

Berdasarkan studi literature yang memiliki kemiripan dengan penelitian penulis sebagai pendukung penelitian ini, ditemukan penelitian yang memiliki relevansi dengan penelitian yang dilakukan oleh penulis. Penelitian pertama oleh (Ardiansyah & Junianto, 2022) yang berjudul “Penerapan Algoritma Genetika Untuk Penjadwalan Sidang dan Seminar IIB Darmajaya”. Hasil dari penelitian ini adalah dapat menerapkan Algoritma Genetika pada sistem informasi penjadwalan seminar proposal dan sidang skripsi yang telah dibuat dengan mengoptimalkan algoritma genetika sehingga meminimalisir masalah penjadwalan yang berbenturan dengan baik. Penelitian yang kedua dilakukan oleh (Oktarina & Hajjah, 2019) yang berjudul “Sistem Penjadwalan Seminar Proposal Dan Sidang Skripsi Dengan Metode Algoritma Genetika”. Hasil dari penelitian ini bawah sistem penjadwalan seminar proposal dan sidang skripsi dapat mempercepat proses penjadwalan dan meminimalisir keterlambatan penjadwalan. Penelitian ketiga dilakukan oleh (Bhaskoro et al., 2021) yang berjudul “Sistem Penjadwalan Sidang Tugas

Akhir menggunakan Algoritma Genetika” Hasil dari penelitian ini dapat membangun sistem penjadwalan yang otomatis menggunakan metode pengembangan sistem *Waterfall* serta menjadi solusi dalam menyelesaikan permasalahan penjadwalan. Penelitian keempat dilakukan oleh (Institut & Iib, n.d.) yang berjudul “Optimasi Algoritma Genetika dalam Perancangan Sistem Informasi Penjadwalan Seminar dan Sidang Skripsi Mahasiswa Institut Informatika dan Bisnis (IIB) Darmajaya”. Penelitian ini dalam proses pengembangan sistemnya menggunakan metode Prototype. Hasil penelitian ini dapat mempercepat proses penjadwalan dan meminimisir terjadinya bentrokan jadwal sidang akhir skripsi. Penelitian Ke lima dilakukan oleh (Khader et al., 2018) yang berjudul “Penjadwalan Matakuliah Menggunakan Algoritma *Greedy* (Studi Kasus Penjadwalan Semeseter Ganji 2017-2018 Informatika Itenas)”. Hasil Penelitian ini dapat menerapkan algoritma *Greedy* untuk membantu proses penjadwalan matakuliah yang efisien. Penelitian keenam yang dilakukan oleh (Salmiati et al., 2023) yang berjudul “Penerapan Algoritma Genetika Untuk Penjadwalan Penggunaan Laboratorium Komputer pada Fakultas Teknik Universitas Jabal Ghafur Berbasis Web”. Penelitian ini menggunakan algoritma genetika dalam penjadwalan dan metode pengembangan menggunakan *Waterfall*. Penelitian ke tujuh yang dilakukan oleh (Sumantri & Laluma, 2022) yang berjudul “Penerapan Algoritma Genetika Untuk Optimasi Penjadwalan Akademik Di Universitas Sangga Buana”. Penelitian ini berfokus dalam penerapan algoritma genetika untuk proses optimasi dalam penjadwalan akademik yang otomatis serta tidak mengalami bentrok. Penelitian ke delapan oleh (Qashlim & Assiddiq, 2016) yang berjudul “Penerapan algoritma genetika untuk sistem penjadwalan kuliah. Jurnal Ilmiah Ilmu Komputer Fakultas Ilmu Komputer Universitas Al Asyariah Mandar”. Penelitian ini berfokus dalam optimasi penjadwalan metakuliah dengan menerapkan algoritma genetika berbasis web.

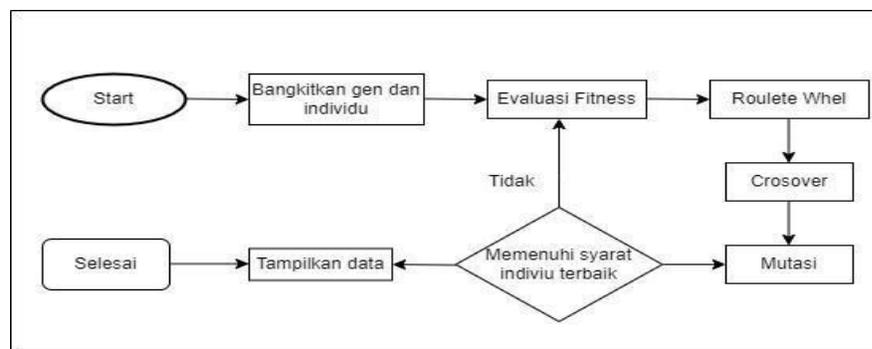
Dari ke delapan literature penelitian yang relevan dengan penelitian penulis diatas memiliki perbedaan pada penelitian penulis, baik dari segi tempat penelitian, metodologi penelitian yang digunakan, metode pengembangan sistem serta batasan pada penerapan algoritma yang digunakan oleh penulis memiliki perbedaan. Berikut beberapa teori pendukung yang digunakan dalam penelitian ini:

Penerapan Menurut Ahmadi, David C.E. Lisapaly dalam bukunya yang membahas tentang efektivitas pembelajaran daring di masa pandemi Covid-19, penerapan dapat diartikan sebagai tindakan untuk mengimplementasikan sesuatu. Namun, beberapa ahli juga menggambarkan

penerapan sebagai praktek atau tindakan yang menggunakan teori, metode, atau elemen lainnya untuk mencapai tujuan yang telah direncanakan dan disusun sebelumnya, serta memenuhi kepentingan kelompok atau golongan yang terlibat (Lisapaly et al., 2022).

Algoritma Genetika merupakan suatu metode *heuristik* yang terinspirasi dari proses genetika dalam teori evolusi Darwin, terutama melalui konsep seleksi alam. Metode optimasi ini awalnya dikembangkan oleh John Holland pada tahun 1960-an, dan kemudian menjadi populer di kalangan para ahli melalui karya-karya David Goldberg, seorang murid Holland, pada tahun 1980-an. Dalam algoritma genetika, proses pencarian atau seleksi solusi terjadi saat individu dipilih untuk bertahan hidup selama proses evolusi (Ardiansyah & Junianto, 2022).

Secara umum, algoritma genetika dapat dijelaskan sebagai serangkaian langkah-langkah yang direpresentasikan dalam bentuk diagram, seperti yang ditunjukkan pada gambar di bawah ini.



Gambar 2.1 Alur Genetika

Langkah awal sebelum menerapkan Algoritma Genetika adalah memilih fungsi fitness yang akan menjadi fokus dari proses optimisasi. Semakin tinggi nilai fitness yang berhasil dicapai, semakin baik sistem yang dihasilkan. Metode heuristik digunakan untuk menentukan fungsi fitness yang sesuai. Algoritma genetika terdiri dari tiga operasi utama, yaitu reproduksi, persilangan (*crossover*), dan mutasi. Keseluruhan struktur algoritma genetika dapat dijelaskan sebagai berikut:

- a. Bangkitkan populasi awal secara acak.
- b. Proses terbentuknya generasi baru dilakukan dengan mengulang tiga operasi inti yaitu seleksi, *crossover*, dan mutasi secara berulang-ulang. Tujuannya adalah untuk mendapatkan kromosom-kromosom yang sesuai dan dapat membentuk generasi baru sebagai solusi baru yang direpresentasikan.

- c. Proses evolusi solusi melibatkan penilaian setiap populasi melalui perhitungan nilai kecocokan (*fitness*) pada setiap kromosom sampai batas akhir tercapai.

Dalam penerapannya algoritma genetika memiliki tahapan-tahapan di antaranya sebagai berikut:

1. Membangun Generasi Awal

Tahap awal dalam algoritma ini melibatkan pembentukan sejumlah populasi awal yang akan digunakan untuk mencari solusi optimal. Dalam konteks tugas akhir ini, populasi awal dibuat dengan menghasilkan bilangan acak dalam rentang yang telah ditetapkan sebelumnya (Adriana., 2015).

2. Fungsi *Fitness*

Tugas utama fungsi *fitness* adalah mengevaluasi kualitas kromosom dengan tujuan untuk mencapai kromosom yang diinginkan. Melalui proses evaluasi ini, fungsi *fitness* dapat membedakan tingkat keunggulan dari setiap kromosom, sehingga mampu menentukan sejauh mana kromosom tersebut efektif dalam mencapai tujuan yang telah ditetapkan sebelumnya (Teknik & Musamus, 2018).

$$Fitness = \frac{1}{Penalty}$$

Sebagai komponen dari fungsi *fitness*, nilai *penalty* dihitung sesuai dengan yang dijelaskan dalam persamaan di atas. *Penalty* ini mengindikasikan jumlah pelanggaran kendala yang terjadi pada kromosom tertentu. Semakin rendah nilai *penalty* (artinya semakin sedikit pelanggaran), semakin tinggi nilai *fitness*. Dengan demikian, fungsi *fitness* dapat dinyatakan dalam bentuk persamaan sebagai berikut:

Keterangan:

Bp = Bobot pelanggaran

Np = Indikator pelanggaran

3. Seleksi

Sebagai komponen dalam fungsi *fitness*, nilai *penalty* dihasilkan sesuai dengan persamaan yang telah dijelaskan sebelumnya. *Penalty* tersebut mencerminkan jumlah pelanggaran kendala yang terjadi pada kromosom tertentu. Semakin kecil nilai *penalty* (yang menunjukkan sedikitnya pelanggaran yang terjadi), semakin tinggi nilai *fitness* yang diperoleh. Oleh karena itu, fungsi *fitness* dapat diungkapkan dalam bentuk persamaan (Afandi & Setyaningsih, 2019)

4. Seleksi *Roulette*

Metode seleksi *Roulette* adalah suatu teknik seleksi individu yang bertujuan untuk menjaga keberagaman populasi dengan mempertimbangkan nilai *fitness* individu. Metode ini mengambil inspirasi dari prinsip roda *roulette*, di mana setiap individu ditempatkan pada lingkaran roda *roulette* secara proporsional sesuai dengan nilai *fitness* mereka (Afandi & Setyaningsih, 2019).

5. *Crossover*

Untuk menghasilkan keturunan yang lebih optimal, digunakan teknik kawin silang (*crossover*) yang melibatkan penggabungan dua kromosom induk. Namun, pada populasi yang sangat kecil, terkadang terjadi situasi di mana kromosom memiliki gen yang sama dengan kromosom lainnya. Untuk mengatasi masalah ini, diterapkan aturan di mana kawin silang hanya terjadi pada kromosom yang memiliki nilai acak di dalam rentang $[0,1]$ yang lebih kecil dari probabilitas *crossover* (pc). Kromosom-kromosom yang memiliki nilai acak lebih besar dari pc dianggap perlu diperbaiki dan digunakan sebagai induk dalam proses kawin silang untuk menghasilkan keturunan yang lebih baik (Suryanto, 2007).

6. *Mutation*

Mutasi merupakan proses yang mengubah nilai-gen dalam suatu kromosom. Terdapat beberapa teknik mutasi yang dapat dilakukan, seperti mengubah nilai-gen dari 0 menjadi 1 atau sebaliknya, memindahkan posisi-gen ke lokasi lain, atau memodifikasi gen dengan batasan tertentu sesuai dengan representasi individu. Tujuan dari mutasi adalah untuk meningkatkan perbedaan antara kromosom-kromosom dalam populasi sehingga mencakup seluruh ruang solusi yang mungkin. (Suryanto, 2007)

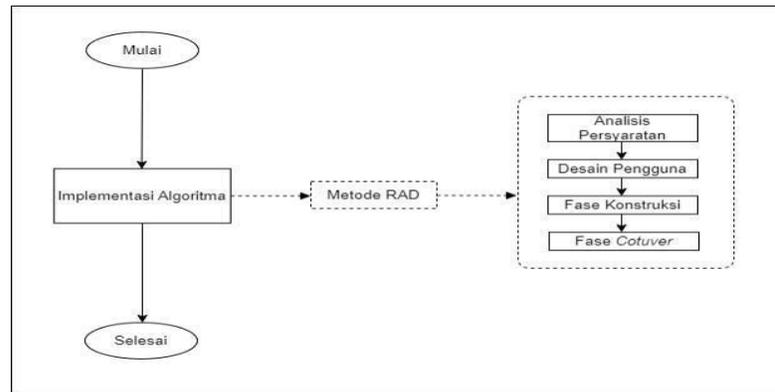
Penjadwalan Menurut pendapat Pinedo (2016) dalam buku (Suradi, 2023), Penjadwalan merupakan proses pengambilan keputusan dalam mengalokasikan sumber daya atau mesin untuk mengeksekusi serangkaian tugas dalam jangka waktu tertentu. Tujuannya adalah untuk mencapai optimalisasi satu atau lebih tujuan yang ingin dicapai, dan hal ini umumnya diterapkan oleh perusahaan atau jasa. Dalam penjadwalan, upaya dilakukan untuk memaksimalkan efisiensi, produktivitas, atau keuntungan dengan memanfaatkan sumber daya yang ada secara optimal.

RAD (*Rapid Application Development*) adalah metode yang mengadopsi pendekatan iteratif dan inkremental dalam pengembangan perangkat lunak. Fokus utamanya adalah pada pengelolaan waktu dan biaya yang efektif sesuai dengan kebutuhan yang ada (Ndaumanu et al., 2022) Metode RAD digunakan untuk membangun aplikasi dengan cepat dan efisien, serta melibatkan pelanggan dan pengembang secara aktif dalam seluruh proses pengembangan aplikasi

Sistem Informasi Dalam pendapat Sri Mulyani yang dikutip dari dalam karya Romney dan Steinbart, sistem dapat diartikan sebagai sebuah entitas yang terdiri dari berbagai komponen yang saling terkait dan bekerja secara sinergis untuk mencapai tujuan yang telah ditetapkan. Dalam konteks ini, sistem merupakan suatu kesatuan yang terdiri dari elemen-elemen yang berinteraksi secara harmonis guna mencapai tujuan tertentu (Mulyani, 2017).

METODOLOGI

Dalam penelitian ini, peneliti menerapkan dua pendekatan, yaitu pengumpulan data dan pengembangan perangkat lunak. Penelitian ini digolongkan sebagai penelitian kualitatif. Berikut adalah langkah-langkah yang diikuti dalam proses pengumpulan data tersebut



Gambar 2. Alur Penelitian

Fase Perencanaan (*Requirement Planning Phase*)

Berikut adalah hasil dari proses pengumpulan data yang diperoleh dalam penelitian ini:

Obesrvasi

Langkah awal dalam pengumpulan data adalah melalui metode observasi, di mana peneliti mengamati secara langsung dan mencatat proses sistem yang sedang berlangsung dalam penjadwalan seminar proposal dan sidang skripsi di Program Studi Informatika Universitas Teknologi Sumbawa.

Wawancara

Langkah selanjutnya dalam pengumpulan data adalah melakukan wawancara dengan Bapak M. Julkarnain M.Sc, yang merupakan sekretaris Program Studi Informatika dan dipilih sebagai narasumber karena memiliki kewenangan dalam penjadwalan seminar dan sidang skripsi. Wawancara ini dilakukan untuk memperoleh pemahaman yang lebih mendalam mengenai alur proses sistem yang sedang berjalan saat ini.

Dokumentasi

Dokumentasi berperan penting dalam mengumpulkan data dalam bentuk informasi, catatan, dan berkas terkait jadwal seminar proposal dan sidang akhir. Tujuan dari dokumentasi ini

adalah memastikan bahwa fitur-fitur yang disediakan sesuai dengan kebutuhan dan tujuan yang diinginkan.

Studi Literatur

Metode studi literatur melibatkan pengumpulan informasi dan data yang relevan dengan topik atau masalah yang sedang diteliti. Sumber informasi yang digunakan meliputi buku, literatur, jurnal ilmiah, tesis, dan juga internet. Dalam penelitian ini, penulis melakukan pencarian dan pengumpulan referensi dari berbagai sumber, seperti buku, e-book, skripsi terdahulu, dan artikel yang dapat ditemukan di internet, yang terkait dengan topik penelitian yang sedang dilakukan. Hal ini bertujuan untuk memberikan dukungan atau penjelasan yang tepat pada penelitian yang dilakukan.

Fase Desain Pengguna (*User Design Phase*)

Proses desain merupakan langkah awal bagi peneliti untuk mengembangkan sistem, termasuk menentukan fitur-fitur yang akan disediakan dan merancang tampilan halaman yang akan digunakan. Dalam tahap ini, peneliti akan merancang desain sistem dan melakukan perbaikan jika diperlukan, agar sesuai dengan kebutuhan pengguna. Peran pengguna sangat penting dalam pembuatan aplikasi ini karena masukan dari pengguna akan membantu peneliti membangun aplikasi yang sesuai dengan harapan mereka. Peneliti akan bertanya tentang fitur dan desain aplikasi kepada pengguna, untuk mendapatkan tanggapan dan masukan yang akan digunakan untuk meningkatkan desain tersebut. Proses ini akan dilakukan secara berulang sampai fitur dan tampilan yang diinginkan oleh pengguna sesuai dengan harapan mereka..

Tahap Konstruksi (*Construction Phase*)

Pada tahap konstruksi atau implementasi, dilakukan proses pembuatan kode program untuk mengimplementasikan sistem yang telah direncanakan sebelumnya. Dalam konteks aplikasi ini, kode program dibuat menggunakan Visual Studio Code sebagai Integrated Development Environment (IDE), dengan menggunakan bahasa pemrograman PHP, HTML, dan CSS. Selain itu, desain database juga dilakukan menggunakan MySQL. Tahap *Cotuver*

Tahap penelitian melibatkan pengujian aplikasi menggunakan metode User Experience Test untuk memastikan optimalitas fungsi implementasi algoritma genetika serta tampilan aplikasi yang dibuat sehingga sesuai dengan harapan yang diinginkan oleh pengguna aplikasi dalam hal ini Program Studi Informatika Universitas Teknologi Sumbawa.

Kebutuhan Data

Dalam penerapan Algoritma genetika pada sistem penjadwalan sidang akhir skripsi dibutuhkan beberapa sampel data diantaranya Data Dosen, Data Judul Skripsi dalam hal ini Mahasiswa, Data Waktu, Jam Dan Ruangan untuk sebagai sampel data yang digunakan. Berikut Tabel kebutuhan Datanya.

Tabel 3.1 Kebutuhan Data

| No | Data | Jumlah Data |
|----|-----------|-----------------|
| 1 | Dosen | 12 |
| 2 | Mahasiswa | 16 |
| 3 | Jam | 1,2 jam persesi |
| 4 | Hari | 5 |
| 5 | Ruangan | 1 |

Dari data sampel diatas digunakan dalam proses perhitungan algoritma genetika untuk optimasi penjadwalan sidang akhir skripsi yang otomatis.

PEMBAHASAN

Tahapan Algoritma Genetika

Adapun tahapan Algoritma Genetika yang akan dijabarkan untuk mendaptan solusi yang optimal dalam proses penjadwalan sebagai berikut:

Inisialisasi Populasi

Inisialisasi Populasi adalah tahapan awal dalam proses pembangkitan populasi secara acak dalam proses algoritma genetika.

Tabel 4.7 Inisialisasi Populasi

| Populasi 1 | | | | | | | |
|------------|---------|-------|-------|------------|-----------|-----------|---------------|
| No | Hari | Jam | Kelas | Pembimbing | Penguji 1 | Penguji 2 | Judul Skripsi |
| 1 | Senin | 9:30 | A | Dosen 1 | Dosen 3 | Dosen 5 | Judul 1 |
| 2 | Selasa | 10:30 | A | Dosen 2 | Dosen 4 | Dosen 6 | Judul 2 |
| 3 | Rabu | 9:00 | A | Dosen 3 | Dosen 5 | Dosen 7 | Judul 3 |
| 4 | Kamis | 10:00 | A | Dosen 4 | Dosen 6 | Dosen 8 | Judul 4 |
| 5 | Juma'at | 11:00 | A | Dosen 5 | Dosen 1 | Dosen 2 | Judul 5 |
| 6 | Senin | 9:30 | A | Dosen 6 | Dosen 7 | Dosen 9 | Judul 6 |
| 7 | Selasa | 10:30 | A | Dosen 7 | Dosen 10 | Dosen 11 | Judul 7 |
| 8 | Rabu | 11:30 | A | Dosen 8 | Dosen 12 | Dosen 9 | Judul 8 |

| | | | | | | | |
|----|---------|-------|---|----------|----------|----------|----------|
| 9 | Kamis | 9:00 | A | Dosen 9 | Dosen 2 | Dosen 3 | Judul 9 |
| 10 | Juma'at | 10:00 | A | Dosen 10 | Dosen 4 | Dosen 5 | Judul 10 |
| 11 | Senin | 11:00 | A | Dosen 11 | Dosen 9 | Dosen 8 | Judul 11 |
| 12 | Selasa | 13:00 | A | Dosen 12 | Dosen 10 | Dosen 9 | Judul 12 |
| 13 | Rabu | 11:30 | A | Dosen 3 | Dosen 2 | Dosen 4 | Judul 13 |
| 14 | Kamis | 9:30 | A | Dosen 5 | Dosen 1 | Dosen6 | Judul 14 |
| 15 | Juma'at | 10:00 | A | Dosen 6 | Dosen 3 | Dosen 8 | Judul 15 |
| 16 | n..300 | 11:30 | A | Dosen 1 | Dosen 4 | Dosen 11 | Judul 16 |

| Populasi..n | | | | | | | |
|-------------|---------|-------|-------|------------|-----------|-----------|---------------|
| No | Hari | Jam | Kelas | Pembimbing | Penguji 1 | Penguji 2 | Judul Skripsi |
| 1 | Senin | 9:30 | A | Dosen 1 | Dosen 3 | Dosen 5 | Judul 1 |
| 2 | Selasa | 10:30 | A | Dosen 2 | Dosen 4 | Dosen 6 | Judul 2 |
| 3 | Rabu | 9:00 | A | Dosen 3 | Dosen 5 | Dosen 7 | Judul 3 |
| 4 | Kamis | 10:00 | A | Dosen 4 | Dosen 6 | Dosen 8 | Judul 4 |
| 5 | Juma'at | 11:00 | A | Dosen 5 | Dosen 1 | Dosen 2 | Judul 5 |
| 6 | Senin | 9:30 | A | Dosen 6 | Dosen 7 | Dosen 9 | Judul 6 |
| 7 | Selasa | 10:30 | A | Dosen 7 | Dosen 10 | Dosen 11 | Judul 7 |
| 8 | Rabu | 11:30 | A | Dosen 8 | Dosen 12 | Dosen 9 | Judul 8 |
| 9 | Kamis | 9:00 | A | Dosen 9 | Dosen 2 | Dosen 3 | Judul 9 |
| 10 | Juma'at | 10:00 | A | Dosen 10 | Dosen 4 | Dosen 5 | Judul 10 |
| 11 | Senin | 11:00 | A | Dosen 11 | Dosen 9 | Dosen 8 | Judul 11 |
| 12 | Selasa | 13:00 | A | Dosen 12 | Dosen 10 | Dosen 9 | Judul 12 |
| No | Hari | Jam | Kelas | Pembimbing | Penguji 1 | Penguji 2 | Judul Skripsi |
| 13 | Rabu | 11:30 | A | Dosen 3 | Dosen 2 | Dosen 4 | Judul 13 |
| 14 | Kamis | 9:30 | A | Dosen 5 | Dosen 1 | Dosen6 | Judul 14 |
| 15 | Juma'at | 10:00 | A | Dosen 6 | Dosen 3 | Dosen 8 | Judul 15 |
| 16 | n..300 | 11:30 | A | Dosen 1 | Dosen 4 | Dosen 11 | Judul 16 |

Menghitung nilai *fitness*

Langkah berikutnya menghitung nilai *fitness*, untuk mendapatkan parent maka nilai *fitness* yang diambil adalah nilai *fitness* tertinggi. Sebelum menghitung nilai *fitness* kita perlu menentukan nilai *penalty* dari setiap populasi. Berikut Nilai *Penalty* yang didapatkan.

Tabel 4.8 Nilai *Penalty*

| Population | <i>Penalty</i> |
|------------|----------------|
| 1 | 23 |

| | |
|----|----|
| 2 | 17 |
| 3 | 13 |
| 4 | 14 |
| 5 | 20 |
| 6 | 24 |
| 7 | 16 |
| 8 | 15 |
| 9 | 8 |
| 10 | 22 |
| 11 | 18 |
| 12 | 10 |
| 13 | 6 |
| 14 | 17 |
| 15 | 15 |
| 16 | 28 |

Setelah menentukan nilai *penalty* dari setiap populasi yang ada. Nilai *penalty* didapatkan dari pengecekan bentrokan jadwal atau Nilai *penalty* dalam algoritma genetika umumnya didapatkan berdasarkan pelanggaran terhadap batasan atau tujuan tertentu. Berikut rumus untuk mendapatkan nilai *fitness*.

Seleksi Pertama

$$fitness = \frac{1}{1 + \sum_{i=1}^n x_i}$$

$$fitness = \frac{1}{1 + (x_1 + x_2 + x_3 + \dots + x_n)}$$

$$\frac{1}{1 + (2 + 3 + 1 + \dots + 5)} = 0,04166666666667$$

Gambar 4.43 Perhitungan Nilai *Fitness*

Berikut hasil perhitungan secara lengkapnya secara berulang

Tabel 4.9 Hasil Nilai *Fitness*

| Population | <i>Fitness</i> |
|-------------------|-----------------------|
| 1 | 0.041666667 |
| 2 | 0.055555556 |
| 3 | 0.071428571 |
| 4 | 0.066666667 |
| 5 | 0.047619048 |
| 6 | 0.04 |

| | |
|--------------|--------------------|
| 7 | 0.058823529 |
| 8 | 0.0625 |
| 9 | 0.111111111 |
| 10 | 0.043478261 |
| 11 | 0.052631579 |
| 12 | 0.090909091 |
| 13 | 0.142857143 |
| 14 | 0.055555556 |
| 15 | 0.0625 |
| 16 | 0.034482759 |
| Total | 1.037785536 |

Seleksi Kedua Dengan *Roulette Wheel* Hasil seleksi sebelumnya akan diseleksi kembali dengan sistem *roulette wheel* yang menggunakan rumus. Berikut hasilnya:

Seleksi Kedua Dengan Roulette Wheel

$$p_1 = \frac{1}{\sum_{i=1}^n f_i}$$

$$p_1 = \frac{0,041666666666667}{0,041666666666667 + \dots + 0,03448275862069}$$

$$p_1 = 0.0401495927747084$$

Gambar 4.44 Perhitungan *Roulette Whell*

Seleksi sebelumnya akan diperbaharui dengan menggunakan metode *Roulette Whell* yang menerapkan rumus kedua diatas .Dalam lanjutan ini, ditampilkan hasilnya sebagai berikut hasil perhitungan lengkapnya.

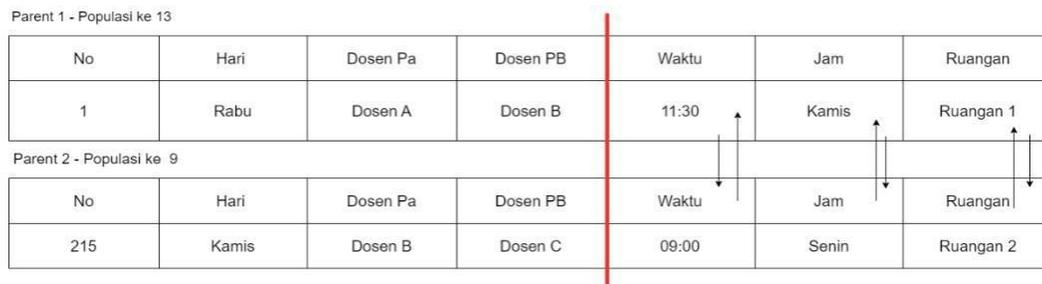
Tabel 4.10 Hasil Perhitungan *Roulette Whell*

| <i>Population</i> | <i>Fitness Percentage</i> |
|-------------------|---------------------------|
| 1 | 0.04014959277470840 |
| 2 | 0.0535327903662779 |
| 3 | 0.0688278733280706 |
| 4 | 0.0642393484395333 |
| 5 | 0.0458852488853811 |
| 6 | 0.0385436090637198 |
| 7 | 0.0566817780348823 |

| | |
|----|--------------------|
| 8 | 0.0602243891620622 |
| 9 | 0.1070655807325540 |
| 10 | 0.0418952272431735 |
| 11 | 0.0507152750838414 |
| 12 | 0.0875991115084541 |
| 13 | 0.1376557466561390 |
| 14 | 0.0535327903662779 |
| 15 | 0.0602243891620622 |
| 16 | 0.0332272491928622 |

Reproduksi Crossover

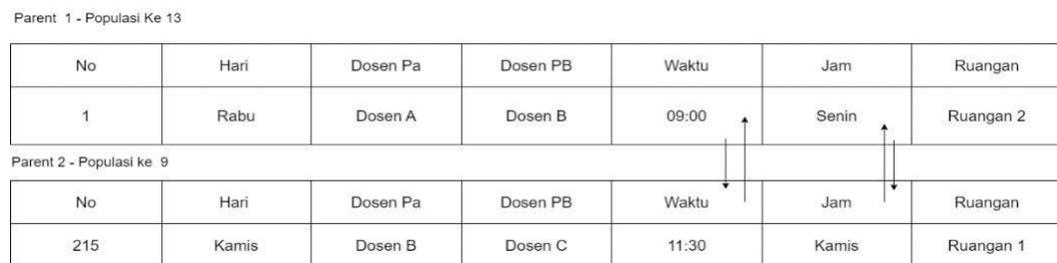
Dua nilai fitness tertinggi dari populasi diatas yang telah melalui seleksi roulette wheel dipilih untuk dilakukan reproduksi menggunakan operator crossover dan mutasi lanjutan.



.Gambar 4.45 Reproduksi crossover

Reproduksi – Mutasi

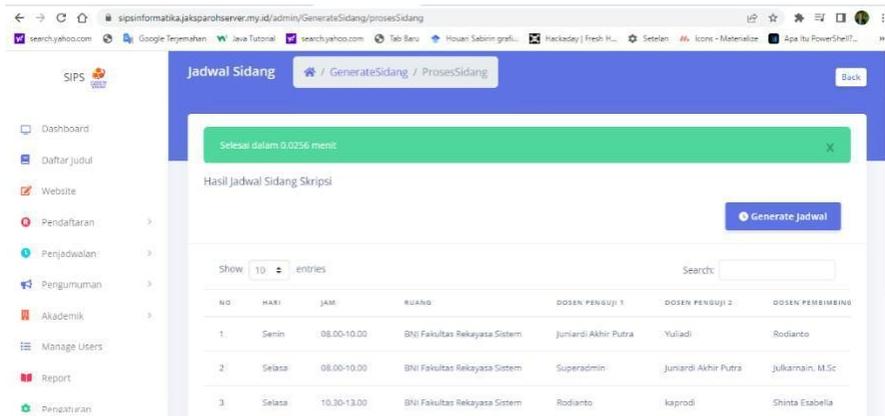
Setelah crossover, dilakukan mutasi untuk pertukaran *genotipe* waktu dan tanggal antara populasi 13 dan populasi 9, seperti berikut:



Gambar 4.46 Reproduksi Mutasi

Setelah melakukan reproduksi, akan terjadi generasi baru yang memiliki potensi sebagai solusi baru dalam penjadwalan. Proses ini akan terus berlanjut hingga mencapai nilai *penalty* nol, sehingga menghasilkan jadwal tanpa konflik.

Halaman Hasil *Generate* Jadwal Sidang Skripsi



Gambar 4.58 Halaman Hasil *Generate* Jadwal Sidang Skripsi

Halaman Hasil *Generate* Jadwal Sidang Akhir Skripsi adalah hasil generate dari data-data yang telah diinputkan oleh sekretaris program studi untuk menghasilkan data penjadwalan yang baik dan akurat serta efisien untuk dapat digunakan sebagai data penjadwalan yang akan di route kemenu pengumuman yang dapat dilihat oleh Mahasiswa, Dosen Pembimbing dan Dosen Penguji.

KESIMPULAN

Berdasarkan analisis dan Penerapan Algoritma genetika sebelumnya, telah berhasil membangun sistem Penjadwalan otomatis Seminar dan Sidang Tugas Akhir yang menerapkan algoritma genetika Pada Sistem Informasi Pelayanan Skripsi Informatika Universitas Teknologi Sumbawa. Pengembangan Sistem menggunakan PHP dengan *Framework CodeIgniter* dan database MariaDB. Penelitian ini bertujuan untuk mempermudah proses penjadwalan seminar dan sidang tugas akhir yang otomatis sehingga memberikan solusi optimal dalam melakukan penjadwalan Seminar dan Sidang akhir Skripsi yang efektif dan efisien di Program Studi Informatika Universitas Teknologi Sumbawa. Metode penelitian yang digunakan adalah metode kualitatif dan pengembangan perangkat lunak menggunakan metode *Rapid Application Development* (RAD).. Hasil penelitian ini diharapkan dapat membantu memberikan solusi yang optimal dalam proses penjadwalan seminar dan sidang tugas akhir yang otomatis di Program Studi Informatika, Universitas Teknologi Sumbawa.

DAFTAR PUSTAKA

Adriana, F., & Fadly, R. L. (2015). Algoritma Genetika dan Penerapannya.

afandi, yusuf, & Setyaningsih, W. (2019). Sistem Pejadwalan Kuliah Meggunakan Metode Algoritma Genetika pada Program Magister Fakultas Ekonomi dan Bisnis. RAINSTEK :

- Jurnal Terapan Sains & Teknologi, 1(1), 40–47. <https://doi.org/10.21067/jtst.v1i1.3069>
- Ardiansyah, H., & Junianto, M. B. S. (2022). Penerapan Algoritma Genetika untuk Penjadwalan Mata Pelajaran. *Jurnal Media Informatika Budidarma*, 6(1), 329-336.
- Bhaskoro, S. B., Aji, B. B., & Aminah, S. (2021). Sistem Penjadwalan Sidang Tugas Akhir menggunakan Algoritma Genetika. *JTT (Jurnal Teknologi Terapan)*, 7(1), 27-36. Institut, M., & Iib, B. (n.d.). *Teknika 16 (01): 133 – 140*. 16(01), 133–140.
- Khader, Y. M., Nurhasanah, Y. I., & Kartika, A. D. (2018). Penjadwalan Matakuliah Menggunakan Algoritma Greedy (Studi Kasus Penjadwalan Semester Ganjil 2017-2018 Informatika Itenas). *Jurnal Ilmiah Teknologi Infomasi Terapan*, 4(3).
- Lisapaly, D. C. (2022). Efektivitas Penerapan Pembelajaran Daring di Tengah Badai COVID-19. *Media Sains Indonesia*.
- Mulyani, S. (2017). *Metode Analisis dan Perancangan Sistem. In Abdi Sistematika*.
- Ndaumanu, R. I., Guterres, J. A. D., Dewi, R., Suarezsaga, F., Susanti, W., Thayf, M. S. S., & others. (2022). *Tahapan-Tahapan Rekayasa Perangkat Lunak*. *Media Sains Indonesia*. <https://books.google.co.id/books?id=qoWeEAAAQBAJ>
- Oktarina, D., & Hajjah, A. (2019). Perancangan Sistem Penjadwalan Seminar Proposal dan Sidang Skripsi dengan Metode Algoritma Genetika. *JOISIE (Journal Of Information Systems And Informatics Engineering)*, 3(1), 32-40.
- Qashlim, A., & Assiddiq, M. (2016). Penerapan algoritma genetika untuk sistem penjadwalan kuliah. *Jurnal Ilmiah Ilmu Komputer Fakultas Ilmu Komputer Universitas Al Asyariah Mandar*, 2(1), 1-6.
- Salmiati, S., Qadriah, L., & Rizal, M. (2023). PENERAPAN ALGORITMA GENETIKA UNTUK PENJADWALAN PENGGUNAAN LABORATORIUM KOMPUTER PADA FAKULTAS TEKNIK UNIVERSITAS JABAL GHAFUR BERBASIS WEB. *Jurnal Real Riset*, 5(1), 197-202.
- Sumantri, G. W., & Laluma, R. H. (2022, November). PENERAPAN ALGORITMA GENETIKA UNTUK OPTIMASI PENJADWALAN AKADEMIK DI UNIVERSITAS SANGGA BUANA. In *Prosiding Seminar Sosial Politik, Bisnis, Akuntansi dan Teknik (Vol. 4, pp. 158-166)*.
- Sistem Produksi. (2023). (n.p.): TOHAR MEDIA.
- Suryanto. (2007). *Algoritma Genetika dalam Matlab*.

Suwarjono, S., & Susanto, S. (2018). Sistem Penjadwalan Perkuliahan Pada Universitas Musamus Menggunakan Algoritma Genetika Berbasis Web. *Musamus Journal Of Research Information and Communication Technology*, 1(1), 12-17