

Penjadwalan Kuliah Otomatis Menggunakan Algoritma Genetika Program Studi Ptik Iain Bukittinggi

Zulmahendra¹, Sarwo Derta²

^{1,2} Program Studi Pendidikan Teknik Informatika dan Komputer, IAIN Bukittinggi
zulmahendra27@gmail.com, sarwoderta@iainbukittinggi.ac.id

Abstrak

Penjadwalan kuliah merupakan salah satu kegiatan penting sebelum pelaksanaan perkuliahan dalam setiap semester. Dalam penjadwalan kuliah, terdapat beberapa parameter yang harus ada agar bisa menghasilkan jadwal perkuliahan yang utuh, diantaranya dosen, mata kuliah, pengampu mata kuliah, hari dan jam perkuliahan (waktu), ruangan perkuliahan, serta waktu kesediaan mengajar dosen. Dari parameter-parameter ini akan disusun dan digabungkan menjadi sebuah jadwal utuh untuk perkuliahan mahasiswa dalam setiap semester. Namun dalam praktek penjadwalannya, terdapat beberapa kendala yang membuat proses penjadwalan kurang efisien seperti proses penjadwalan yang masih dilakukan secara manual sehingga rentan terjadinya bentrok jadwal serta bisa memakan waktu dua sampai tiga hari. Penentuan jadwal mengajar dosen yang masih menggunakan pesan teks dengan cara mengirimkan jadwal kesediaan mengajar kepada ketua program. Hal inilah yang melatarbelakangi penulis untuk merancang sebuah sistem penjadwalan kuliah yang mampu menjadwalkan secara otomatis. Sistem ini diharapkan dapat memudahkan proses penjadwalan serta pendistribusian jadwal kepada dosen agar lebih efektif dan efisien.

Kata kunci: Penjadwalan Kuliah, Algoritma Genetika, Fitness, Populasi, Crossover

Abstract

Scheduling courses is one of the important activities before conducting courses in each semester. In scheduling courses, there are several parameters that must be present in order to produce a complete course schedule, including lecturers, courses, days and hours to study (time), rooms, and teaching schedules for lecturers. From these parameters will be arranged and combined into a complete schedule for student courses in each semester. However, in the practice of scheduling, there are several obstacles that make the scheduling process less efficient, such as the scheduling process which is still done manually so that it is prone to schedule conflicts and can take two to three days. Determination of teaching schedules for lecturers who still use text messages by sending a schedule of teaching willingness to the program leader. This is what motivates the writer to design a courses scheduling system that is able to schedule automatically. This system is expected to facilitate the process of scheduling and distributing schedules to lecturers to make it more effective and efficient.

Keywords: Scheduling Courses, Genetic Algorithm, Fitness, Population, Crossover

1. PENDAHULUAN

Institut Agama Islam Negeri (IAIN) Bukittinggi merupakan salah satu instansi pendidikan tinggi yang masih menggunakan cara manual dalam penjadwalan perkuliahan, salah satunya adalah proses penjadwalan di program studi Pendidikan Teknik Informatika dan Komputer (PTIK). Biasanya, ketua program studi akan melakukan pencarian blok-blok atau kolom-kolom mana saja yang masih kosong dengan memperhatikan berbagai komponen seperti dosen, kelas mahasiswa, semester, ruangan dan mata kuliah, kemudian menempatkan komponen pada blok atau kolom tersebut.

Penjadwalan secara manual ini terdapat beberapa kendala yang ditemukan, baik oleh mahasiswa maupun dosen. Pertama adalah waktu yang diperlukan dalam satu kali proses

penjadwalan tergolong cukup lama, bisa memakan waktu dua atau tiga hari. Proses ini bergantung juga terhadap kegiatan sehari-hari orang yang menjadwalkan, apabila terdapat kegiatan lain, tentu akan lebih memakan waktu yang lebih banyak. Kedua adalah, cara penentuan hari mengajar dosen yang kurang sesuai dengan rutinitas dosen tersebut, akibatnya terdapat beberapa dosen yang jadwalnya bentrok dengan rutinitas sehari-harinya. Seperti yang terjadi pada dosen luar biasa, dimana biasanya dosen luar biasa hanya bisa mengajar pada hari Jum'at atau Sabtu, tetapi ada beberapa kasus yang jadwalnya diletakkan pada hari lain. Walaupun awalnya ketua program studi sudah meminta jadwal kesediaan mengajar, namun terkadang terdapat perubahan jadwal kegiatan sehari-hari, sehingga jadwal mengajarpun harus disesuaikan dengan kegiatan sehari-harinya. Hal tersebut tentu membuat ketua program studi harus melakukan penjadwalan ulang yang tentu akan membutuhkan tambahan waktu lagi. Ketiga, rentan terjadinya bentrokan jadwal, dan bentrokan ini akan sulit untuk dideteksi secara manual. Walaupun sistem kampus mempunyai fasilitas untuk mengetahui jadwal yang bentrok, bentrokan jadwal tersebut baru akan terdeteksi apabila jadwal sudah diinput ke sistem, sehingga penjadwalan ulang baru bisa dilakukan setelah jadwal bentrok diketahui dari sistem. Jika terdapat jadwal bentrok, sistem kampus pun hanya mengabaikan bentrokan tersebut tanpa memberikan solusi. Hal ini tentu membuat jadwal perkuliahan harus dijadwalkan kembali secara manual.

Sistem yang dibangun ini nantinya diharapkan akan dapat membantu dalam menjadwalkan perkuliahan secara otomatis pada program studi PTIK IAIN Bukittinggi. Karena dengan adanya sistem ini, maka dapat membuat proses penjadwalan menjadi lebih singkat dan lebih praktis serta mengurangi bahkan mengantisipasi adanya jadwal yang bentrok dalam proses penjadwalan. Selain itu, sistem ini nanti dapat menjadi alat bantu sistem informasi kampus (e-campus) dalam merencanakan proses penjadwalan.

2. LANDASAN TEORI

2.1. Penjadwalan Kuliah

Penjadwalan mata kuliah merupakan penyusunan dan pengaturan jadwal mata kuliah pada slot waktu yang tersedia selama satu minggu disertai dengan pembagian ruang kelasnya. Penjadwalan tersebut harus memperhatikan beberapa hal seperti kapasitas ruang, dosen, mahasiswa, dan jumlah SKS setiap mata kuliah. Masalah penjadwalan meliputi optimasi beberapa kriteria termasuk batasan-batasan seperti kebijakan kurikulum, pemilihan ruang kelas yang sesuai, dan ketersediaan dosen pengajar [1].

2.2. Algoritma Genetika

Algoritma genetika merupakan suatu algoritma pencarian yang berlandaskan atas mekanisme seleksi alam dan genetika seperti dalam dunia biologi. Dalam menyelesaikan masalah optimasi yang kompleks, algoritma genetika sangat tepat untuk digunakan. Sifat algoritma genetika adalah mencari kemungkinan-kemungkinan terbaik dan optimal dari calon solusi yang ada. Cakupan dari solusi-solusi yang layak, yaitu obyek-obyek di antara solusi yang sesuai, dinamakan dengan ruang pencarian. Tiap titik dalam ruang pencarian mempresentasikan suatu solusi yang layak. Tiap solusi yang layak dapat ditandai dengan nilai *fitness*. Kromosom-kromosom yang ada diseleksi menurut nilai *fitness* masing-masing. Kromosom dengan nilai *fitness* yang tinggi mempunyai kemungkinan yang lebih baik untuk bertahan hidup pada generasi berikutnya, tetapi tidak menutup kemungkinan juga bagi kromosom dengan nilai *fitness* yang rendah untuk tetap bertahan hidup. Proses

seleksi tersebut kemudian ditentukan oleh kromosom-kromosom baru (offspring) melalui proses *crossover* dan mutasi dari kromosom yang terpilih (induk) [2].

3. METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini adalah *Research and Development* (RnD) yang merupakan metode penelitian yang digunakan untuk menghasilkan produk tertentu serta menguji validitas dan keefektifan produk tersebut dalam penerapannya [3]. Serta model pengembangan sistem SDLC (*System Development Life Cycle*) versi *waterfall*. Menurut Pressman, dalam model *waterfall* tersirat pendekatan yang sistematis dan berurutan dalam pengembangan sistem. Pengembangan ini melalui tahapan-tahapan perencanaan, pemodelan, konstruksi, penyerahan sistem ke pengguna, dan diakhiri dengan dukungan serta perbaikan (*maintenance*) [4].

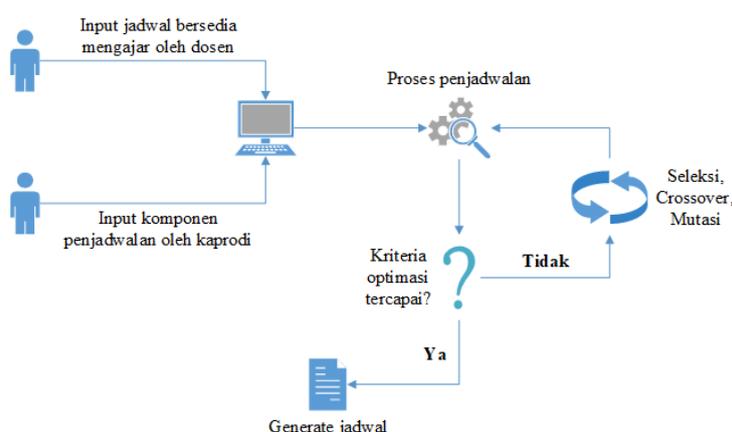
3.1. Analisa Sistem

Sebelum dilakukan perancangan sistem, terlebih dahulu dilakukan analisa terhadap sistem atau cara penjadwalan kuliah yang telah ada sebelumnya. Dari analisa, didapatkan bahwa penjadwalan kuliah yang dilakukan sebelumnya masih dilakukan dengan cara manual dan membutuhkan waktu yang relatif lama. Selain itu, jadwal yang dihasilkan masih memungkinkan terjadinya bentrok apabila tidak dijadwalkan dengan teliti. Terakhir, penentuan jadwal mengajar masih menggunakan pesan teks yang dikirimkan kepada ketua program studi untuk nantinya dijadikan salah satu parameter dalam penjadwalan.

Setelah dilakukan analisa, didapatkan bahwa sistem yang akan dibangun nantinya diharapkan bisa mengatasi beberapa masalah sehingga sistem dapat menjadwalkan perkuliahan secara otomatis. Selain itu sistem juga mampu untuk mengatasi jadwal perkuliahan yang bentrok dan memiliki fasilitas untuk mengelola jadwal kesediaan dosen.

3.2. Desain Sistem

3.2.1. Alur Proses Penjadwalan

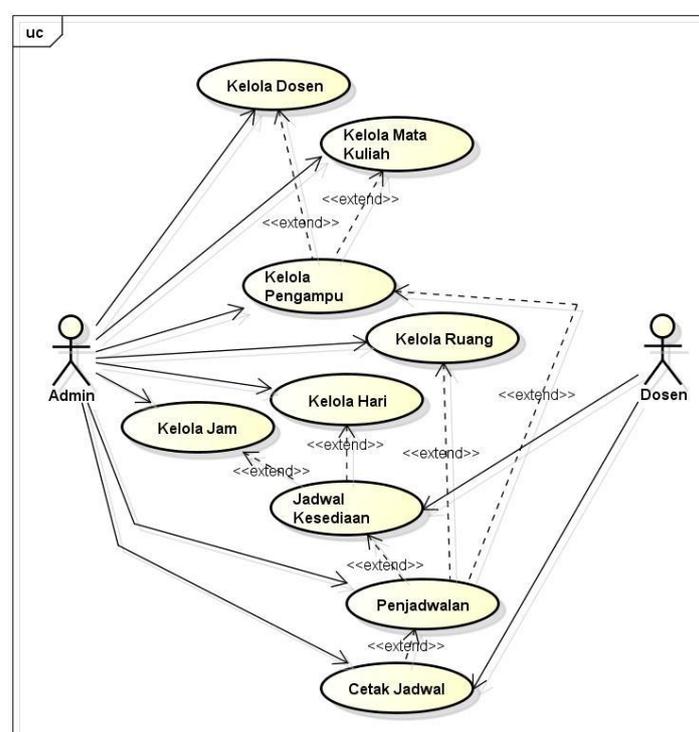


Gambar 1. Alur sistem proses penjadwalan

Ketua program studi yang bertindak sebagai admin akan melakukan proses input data dosen yang akan mengajar, mata kuliah yang ditawarkan, ruangan, dan juga data rentang waktu atau jam yang digunakan dalam perkuliahan. Selain itu, dosen yang akan mengajar

juga akan diberi akses ke sistem untuk menambahkan jadwal kesediaan mengajar. Selain dosen, ketua program studi juga bisa menginputkan jadwal kesediaan mengajar dosen apabila diperlukan. Selanjutnya, ketua program studi masuk ke menu penjadwalan untuk melakukan proses penjadwalan. Di dalam menu penjadwalan, ketua program studi bisa menentukan semester (ganjil atau genap) serta tahun akademik yang akan dilakukan proses penjadwalan kuliah. Saat penjadwalan, akan dicari solusi terbaik dari setiap komponen yang ada, apabila belum ditemukan, maka akan dilakukan proses seleksi, *crossover*, serta mutasi dari komponen yang ada. Selanjutnya proses penjadwalan akan dilanjutkan hingga ditemukan solusi (kriteria optimasi) terbaik. Apabila solusi terbaik didapatkan, maka proses penjadwalan selesai dan akan dijadikan sebagai jadwal yang utuh.

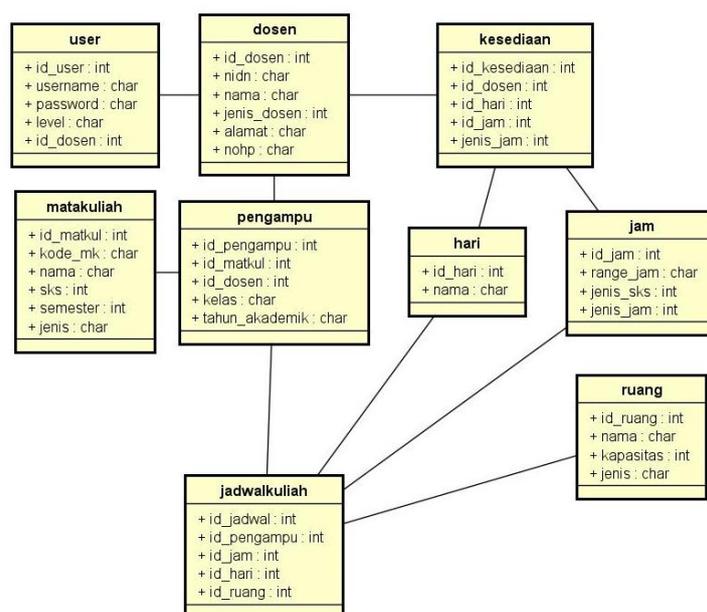
3.2.2. Use Case Diagram



Gambar 2. Use Case Diagram sistem penjadwalan kuliah

Pada gambar 2, terlihat gambaran umum dari aktivitas aktor yang ada pada sistem. Terlihat bahwa terdapat dua aktor yaitu admin dan dosen, admin memiliki hak akses penuh terhadap sistem, yakni dapat mengelola semua data yang ada kecuali data jadwal kesediaan dosen. Sedangkan dosen dapat mengelola jadwal kesediaan mengajar serta mencetak jadwal yang telah selesai dijadwalkan.

3.2.3. Class Diagram



Gambar 3. Class Diagram sistem penjadwalan kuliah

Pada gambar 3, terlihat gambaran struktur *database* yang berupa atribut-atribut dan *actions* yang ada pada sistem penjadwalan kuliah yang dikembangkan. Dari *class diagram* ini juga terlihat bahwa masing-masing tabel mewakili setiap parameter-parameter yang ada pada sistem penjadwalan menggunakan algoritma genetika.

3.3. Pengkodean Program

Proses pengkodean program merupakan proses untuk menuangkan desain yang telah dibuat menjadi bentuk sebuah sistem penjadwalan yang utuh. Proses pengkodean program menggunakan bahasa PHP dengan bantuan *database* MySQL dan menggunakan *framework* Codeigniter. Selain itu, beberapa *tools* yang digunakan yaitu *web server* Apache, *web browser*, serta *text editor* sebagai wadah menulis kode program.

4. HASIL DAN PEMBAHASAN

4.1. Hasil

Bagian ini akan membahas mengenai hasil penelitian yang telah dilakukan. Dalam proses pengembangan sistem penjadwalan kuliah dengan algoritma genetika, terdapat beberapa tahapan yang dimulai dari inisialisasi data hingga proses penetapan individu baru sebagai solusi akhir dari proses algoritma genetika.

4.1.1. Inisialisasi

Sebelum proses ini dilakukan, terdapat beberapa parameter awal yang harus dimasukkan sebagai acuan dalam penjadwalan, diantaranya jumlah populasi, jumlah generasi, probabilitas *crossover*, dan probabilitas mutasi yang ditetapkan oleh user.

Tabel 1. Parameter awal penjadwalan

Parameter awal	Nilai
Jumlah populasi	5
Jumlah generasi	1000

Probabilitas <i>crossover</i>	0,70
Probabilitas mutasi	0,40

Penjadwalan

Semester	Probabilitas CrossOver
<input type="text" value="GANJIL"/>	<input type="text" value="0.70"/>
Tahun Akademik	Probabilitas Mutasi
<input type="text" value="2019-2020"/>	<input type="text" value="0.40"/>
Jumlah Populasi	Jumlah Generasi
<input type="text" value="5"/>	<input type="text" value="1000"/>

Gambar 4. Tampilan input parameter awal oleh user

Proses inisialisasi merupakan proses penempatan kode dari data-data seperti kode dosen, kode mata kuliah dan yang lainnya ke dalam variabel berbentuk larik (*array*) sebagai wadah penyimpanan selama proses penjadwalan dilakukan. Kode yang ditempatkan pada larik ini akan diberikan nilai *index* yang dimulai dari nilai 0 sampai dengan jumlah data dikurangi 1 ($n-1$). Panjang larik sama dengan banyaknya data yang diinisialisasi pada larik, seperti kode dosen (2, 3, 6, 1, 3, 5) dengan *index* kode dosen (0, 1, 2, 3, 4, 5). Sama halnya dengan data mata kuliah dan data lainnya akan diinisialisasi ke dalam bentuk larik untuk mempermudah dalam pemrosesan pada algoritma genetika.

Setelah pengelompokan data selesai dilakukan, selanjutnya akan dibentuk sebuah variabel berbentuk larik lainnya dengan nama individu yang di dalamnya berisi kode pengampu, kode hari, kode jam, kode ruang. Kode data dalam variabel individu ini didapatkan dari pemanggilan bilangan acak dari *index* data dosen, hari, jam, dan ruangan pada proses sebelumnya.

Tabel 2. Variabel individu acak

Individu ke	Isi larik
1	(0, 3, 4, 4)
2	(1, 1, 2, 3)
3	(2, 5, 6, 0)
4	(0, 5, 0, 6)
5	(4, 4, 8, 8)
6	(3, 0, 6, 8)
7	(2, 1, 0, 0)
⋮	⋮
n	(.., .., .., ..)

4.1.2. Nilai *Fitness*

Fitness merupakan nilai yang menentukan apakah suatu individu baik atau tidak. Penentuan baik atau tidaknya berdasarkan perbandingan dengan nilai *fitness* dari individu lain. Untuk menentukan nilai *fitness*, diperlukan rumus:

$$fitness = 1 / (1 + \sum penalti)$$

Penentuan nilai *fitness* ini berdasarkan pada batasan-batasan mutlak (*hard constraint*) yang terdapat dalam proses penjadwalan. Batasan mutlak merupakan batasan yang harus dipatuhi dalam penjadwalan. Solusi yang diajukan tidak boleh melanggar batasan mutlak [5]. Batasan mutlak tersebut diantaranya:

- a. Tidak boleh adanya bentrokan dosen yang sama pada hari dan jam yang sama.
- b. Tidak adanya bentrokan ruang yang sama pada hari dan jam yang sama.
- c. Tidak adanya bentrokan kelas yang sama pada hari dan jam yang sama.
- d. Tidak adanya bentrokan hari dan jam dengan waktu keinginan dosen.

Apabila dalam proses perulangan penjadwalan terdapat pelanggaran batasan mutlak, maka nilai penalti akan bertambah 1 (penalti = penalti sebelumnya + 1). Nilai akhir *fitness* untuk satu siklus perulangan populasi yang sedang dilakukan akan didapatkan apabila perulangan telah selesai.

Fitness populasi ke-1 = 0,1428

Fitness populasi ke-2 = 0,33

Fitness populasi ke-3 = 0,083

Fitness populasi ke-4 = 1

Fitness populasi ke-5 = 0,167

4.1.3. Seleksi

Seleksi merupakan proses untuk melakukan perangkaian atau pengurutan nilai *fitness* dari yang tertinggi ke yang terendah. Setelah diurutkan, maka akan diseleksi untuk mencari individu yang akan menjadi induk dari setiap populasi yang ada. Induk ini nantinya akan dilakukan persilangan dengan induk lainnya agar menghasilkan individu baru yang memiliki nilai *fitness* yang lebih baik. Dari proses sebelumnya, didapatkan bahwa, populasi ke-4 memiliki nilai *fitness* tertinggi, ini berarti populasi pertama memperoleh rangking 4, populasi kedua rangking 2, populasi ketiga rangking 5, populasi keempat rangking 1 dan populasi kelima rangking 3.

Selanjutnya, dilakukan pembangkitan bilangan acak yang disimpan dalam variabel dengan nama target dan sebuah variabel dengan nama cek yang diisi nilai awal 0. Variabel target akan menjadi pembanding bagi variabel cek. Apabila nilai variabel cek lebih kecil dari variabel target, maka nilai cek akan dijumlahkan dengan rangking pada populasi yang sedang dilakukan perulangan.

If cek < target, *Then* cek = cek sebelumnya + rangking

Proses ini akan terus diulang sampai nilai cek lebih besar atau sama dengan nilai target. Apabila telah memenuhi kondisi, maka perulangan saat itu yang akan menjadi induk dari populasi tersebut. Selanjutnya akan memasuki siklus perulangan untuk populasi kedua dan proses akan diulangi seperti sebelumnya sampai perulangan terakhir dari jumlah populasi yang ada.

4.1.4. Persilangan (Crossover)

Proses *crossover* adalah proses menukar gen dari individu yang ada sehingga dapat menghasilkan nilai *fitness* yang lebih baik. Proses *crossover* ini merupakan proses untuk menambah keanekaragaman string dalam satu populasi [6]. Dalam prosesnya, akan

dibangkitkan bilangan acak desimal antara 0 dan 1. Kemudian bilangan ini akan dibandingkan dengan probabilitas *crossover* yang diinputkan oleh user pada awal sebelum proses penjadwalan. Apabila nilai acak ini lebih kecil dari nilai probabilitas *crossover*, maka akan terjadi pertukaran data antara populasi satu dengan populasi setelahnya.

Sebelum terjadinya *crossover*

Individu 1 = (0, 3, 4, 4)

Individu 2 = (1, 1, 2, 3)

Setelah terjadinya *crossover*

Individu 1 = (0, 3, 2, 3)

Individu 2 = (1, 1, 4, 4)

4.1.5. Mutasi

Proses mutasi akan menggantikan nilai gen pada kromosom dengan nilai yang lain. Mutasi ini hampir sama dengan proses *crossover* (persilangan), yaitu pertukaran data dalam setiap populasi. Namun dalam mutasi, data yang ada digantikan dengan nilai acak, bukan digantikan dengan nilai data pada individu lain.

Sebelum terjadinya mutasi

Individu 1 = (0, 3, 4, 4)

Individu 2 = (1, 1, 2, 3)

Setelah terjadinya mutasi

Individu 1 = (1, 3, 4, 2)

Individu 2 = (5, 0, 2, 5)

Setelah dilakukan mutasi, akan ditentukan kembali nilai *fitness* dari setiap populasi yang ada dengan memanggil kembali fungsi menghitung nilai *fitness* pada proses sebelumnya. Nilai *fitness* hasil mutasi ini yang akan menjadi acuan dalam menentukan solusi optimum dalam menentukan jadwal kuliah yang baik.

4.1.6. Individu Baru

Proses ini akan menghasilkan individu baru yang akan menjadi solusi dalam proses penjadwalan. Artinya, individu baru ini merupakan hasil akhir yang akan menjadi jadwal utuh dalam penjadwalan yang dilakukan. Sebelumnya telah ditetapkan oleh pengguna jumlah generasi (iterasi banyaknya proses penjadwalan dilakukan). Misalnya ditetapkan generasi = 1000, artinya akan ada proses perulangan atau iterasi sebanyak 1000 kali proses penjadwalan untuk mencari populasi dengan nilai *fitness* sempurna (nilai *fitness* 1). Populasi dengan nilai *fitness* sempurna ini yang menjadi solusi akhir dan selanjutnya disimpan ke dalam *database* sebagai jadwal utuh.

#	Hari	Jam	Matakuliah	SKS	Semester	Kelas	Dosen	Ruang
1	Senin	07.15-08.55	Pengantar Teknologi Informasi	2	1	C	Gusnita Darmawati, S.Pd, M.Kom	H11
2	Senin	07.15-08.55	Multimedia *	2	3	A	Rina Novita	Laboratorium Multimedia
3	Senin	07.15-08.55	Bahasa Indonesia (Tata Tulis Ilmiah)	2	1	A	Jasmienti, M.Pd	H9
4	Senin	07.15-09.45	Metodologi Penelitian	3	5	C	Radhiatul Husni	H5
5	Senin	08.55-10.35	Pengantar Teknologi Informasi	2	1	E	Gusnita Darmawati, S.Pd, M.Kom	H1
6	Senin	08.55-10.35	Paket Pemrograman	2	1	C	Riri Okra, M.Kom	Laboratorium Pemrograman
7	Senin	09.45-12.15	SDAM (System Design & Analysis Method)	3	5	D	Sarwo Derta, M.Kom	H7

Gambar 5. Jadwal hasil penjadwalan dengan algoritma genetika

4.2. Pembahasan

Sistem penjadwalan kuliah menggunakan algoritma genetika ini dapat dioperasikan baik melalui *web browser* yang ada di *personal computer* (pc) maupun dari *web browser* pada perangkat *Android*. Sistem ini difungsikan untuk menjadwalkan serta mengelola proses penjadwalan perkuliahan pada program studi Pendidikan Teknik Informatika dan Komputer (PTIK) IAIN Bukittinggi.

5. Kesimpulan

Seperti yang diketahui, penjadwalan perkuliahan pada prodi PTIK masih dilakukan secara manual dengan menempatkan komponen-komponen perkuliahan ke dalam Microsoft Excel. Hal seperti ini rentan terjadinya bentrokan jadwal. Dengan adanya sistem ini, ketua prodi ataupun staff prodi tidak perlu lagi menjadwalkan secara manual, hanya perlu menginputkan data dosen, mata kuliah, pengampu, hari, jam, ruangan ke dalam sistem. Selain itu, dosen tidak perlu lagi menentukan jadwal kesediaan mengajar kepada ketua prodi melalui pesan teks, karena pada sistem telah disediakan fitur untuk mengisi jadwal kesediaan mengajar sehingga lebih mudah, efektif dan efisien.

REFERENSI

- [1] D. A. R. Wati and Y. A. Rochman, “Model Penjadwalan Matakuliah Secara Otomatis Berbasis Algoritma Particle Swarm Optimization (PSO),” *J. Rekayasa Sist. Ind.*, vol. 2, no. 1, pp. 22–31, 2013, doi: 10.26593/jrsi.v2i1.333.22-31.
- [2] K. Krisnandi and H. Agung, “Implementasi Algoritma Genetika Untuk Memprediksi Waktu Dan Biaya Pengerjaan Proyek Konstruksi,” *J. Ilm. FIFO*, vol. 9, no. 2, p. 90, 2017, doi: 10.22441/fifo.2017.v9i2.001.
- [3] Hanafi, “Konsep Penelitian R & D Dalam Bidang Pendidikan,” *Saintifika Islam. J. Kaji. Keislam.*, vol. 4, no. 2, pp. 129–150, 2017.
- [4] R. S. Pressman, *Rekayasa Perangkat Lunak Pendekatan Praktisi Buku 1*, 7th ed. Yogyakarta: Andi, 2015.
- [5] R. Hartadi and A. Hidayat, “Perancangan Aplikasi Penjadwalan Mata Kuliah (Studi Kasus : STMIK Provisi Semarang),” *J. Bianglala Inform.*, vol. 4, no. 1, pp. 31–40, 2016.
- [6] N. L. G. P. Suwirmayanti, I. M. Sudarsana, and S. Darmayasa, “Penerapan Algoritma Genetika Untuk Penjadwalan Mata Pelajaran,” *J. Appl. Intell. Syst.*, vol. 1, no. 3, pp. 220–233, 2016.