

Implementasi Algoritma Genetika Proses Mutasi Differential Evolution Pada Sistem Penjadwalan Mata Pelajaran

Meiliya Cahya Yustina, Ichwan Nul Ichsan*, Galura Muhammad Suranegara

Kampus Daerah Purwakarta, Sistem Telekomunikasi, Universitas Pendidikan Indonesia, Purwakarta, Indonesia

Email: ¹meiliyachya@upi.edu, ^{2,*}ichwannul.ichsan90@upi.edu, ³galurams@upi.edu

Email Penulis Korespondensi: ichwannul.ichsan90@upi.edu

Abstrak—Pembuatan jadwal di SMAN 2 Purwakarta memerlukan upaya ekstra karena banyaknya kelas dan guru. Sistem penjadwalan saat ini menggunakan dua software berbeda yaitu Asc Timetables dan Microsoft Excel. Penggunaan kedua software tersebut tidak efisien dan rawan kesalahan. Kedua software tidak dapat saling terhubung sehingga pembuatan jadwal harus terpisah dan memerlukan banyak waktu dalam sinkronisasi berdampak pada kualitas pembelajaran yang diperoleh siswa. Untuk mengatasi masalah ini diperlukan sebuah sistem penjadwalan terpadu yang dapat bekerja secara otomatis dan efisien dalam satu sistem. Penelitian ini bertujuan untuk membuat sebuah sistem penjadwalan yang dibuat secara otomatis dengan metode optimasi. Salah satu metode optimasi yaitu Algoritma Genetika dan Differential Evolution. Terminologi yang digunakan yaitu populasi awal, hitung fitness, seleksi, crossover, mutasi dan iterasi. Sehingga implementasi sistem pada penelitian ini menggunakan algoritma genetika dengan proses mutasi differential evolution. Metode yang digunakan yaitu menggunakan Software Development Life Cycle (SDLC) dengan model pengembangan waterfall. Hasil penerapan dari algoritma genetika dengan proses mutasi differential evolution menunjukkan bahwa pembuatan jadwal memenuhi batasan agar hasil lebih optimal. Batasan tersebut yaitu tidak boleh terjadi benturan jadwal antar guru dalam jam dan hari yang sama. Berdasarkan hasil pengujian menggunakan SUS sistem mendapatkan skor dari perhitungan SUS yaitu 69,1. Dari skor tersebut mendapatkan bobot skor SUS Grade Scale “B” dan Adjective Rating “Good”. Sehingga memerlukan pengembangan lebih lanjut pada sistem. Namun, dengan kategori tersebut sistem layak digunakan jika berpatokan pada pengujian SUS.

Kata Kunci: Penjadwalan; Algoritma Genetika; Differential Evolution; Mata Pelajaran; Optimasi

Abstract—Making schedules at SMAN 2 Purwakarta requires extra effort due to the large number of classes and teachers. The current scheduling system uses two different software namely Asc Timetables and Microsoft Excel. The use of both software is inefficient and error-prone. The two software cannot be connected to each other so that the creation of schedules must be separate and requires a lot of time in synchronization which has an impact on the quality of learning obtained by students. To overcome this problem, an integrated scheduling system is needed that can work automatically and efficiently in one system. This research aims to create a scheduling system that is made automatically with optimization methods. One of the optimization methods is Genetic Algorithm and Differential Evolution. The terminology used is the initial population, calculate fitness, selection, crossover, mutation and iteration. So that the implementation of the system in this study uses a genetic algorithm with a differential evolution mutation process. The method used is using the Software Development Life Cycle (SDLC) with a waterfall development model. The results of the application of the genetic algorithm with the differential evolution mutation process show that the schedule generation meets the limitations for more optimal results. The limitation is that there should be no schedule clashes between teachers in the same hours and days. Based on the test results using SUS, the system gets a score from the SUS calculation which is 69.1. From this score get the weight of the SUS Grade Scale score “B” and Adjective Rating “Good”. So that it requires further development on the system. However, with this category the system is feasible to use if based on SUS testing.

Keywords: Scheduling; Genetic Algorithm; Differential Evolution; Subject; Optimization

1. PENDAHULUAN

Menurut berbagai pakar pendidikan, pembelajaran memiliki makna yang lebih konstruktif yaitu mengupayakan agar peserta didik dapat belajar, terdorong untuk belajar, ingin belajar, dan tertarik untuk terus belajar [1]. Sehingga, menekankan pada peserta didik aktif dalam belajar dan memberikan stimulus untuk mencapai hasil yang maksimal dalam proses pembelajaran. Proses tersebut menggunakan pendekatan pembelajaran [2]. Pendekatan kepada peserta didik agar pembelajaran berjalan secara optimal dapat menggunakan metode pembelajaran. Metode pembelajaran merupakan cara atau jalan yang digunakan oleh pendidik dalam proses pembelajaran. Salah satu cara untuk mendukung proses pembelajaran adalah dengan adanya jadwal mata pelajaran [3].

Mata pelajaran merupakan kumpulan instrumen pembelajaran yang berisi materi. Materi dirancang untuk membantu siswa mencapai tujuan pembelajaran. Materi pembelajaran digunakan oleh guru dalam menyampaikan materi kepada siswa. Penjadwalan mata pelajaran sangat penting untuk kegiatan belajar mengajar agar dapat berkesinambungan dengan mekanisme pelaksanaan di dalam ruang kelas. Tujuan kegiatan tersebut adalah untuk memfasilitasi, memperbaiki, dan meningkatkan standar pendidikan. Penjadwalan mata pelajaran umumnya mendukung kegiatan akademik dengan meningkatkan disiplin antara para guru dan siswa [4].

Membuat, merencanakan, dan menyusun rencana adalah tugas umum yang dilakukan setiap sekolah di awal tahun ajaran baru. Diperlukan upaya ekstra saat penyusunan jadwal. Jika kurang teliti maka jadwal akan berbenturan. Akibatnya akan terjadi tumpang tindih jadwal antara guru dan siswa [5]. Dengan banyaknya kelas dan guru di sekolah, tak dapat dipungkiri bahwa pengoptimalan diperlukan untuk membantu penjadwalan. Institusi pendidikan sering kali menghadapi kendala dalam hal penjadwalan akademik [6]. Kendala penjadwalan akademik dapat disebabkan karena beberapa sekolah masih melakukan penjadwalan secara manual sehingga kurang efisien. Penjadwalan ini merupakan masalah optimasi yang dapat diselesaikan dengan menggunakan sebuah metode [7]. Optimasi adalah suatu prosedur tujuannya untuk

mencapai hasil optimal dan memaksimalkan efektivitas. Optimasi mengacu pada peningkatan produk yang sudah ada atau mendesain sesuatu menjadi efisien [8].

Penjadwalan mata pelajaran di sekolah membutuhkan waktu, usaha dan ketelitian. Perancangan jadwal kursus perlu mempertimbangkan aturan penjadwalan serta faktor-faktor yang mempengaruhi seperti guru, kelas, waktu, mata pelajaran dan batasan [9]. Batasan-batasan memiliki peran penting dalam menciptakan jadwal yang efektif. Untuk mencegah benturan jadwal memerlukan batasan yang berhubungan antar komponen dalam alokasi jadwal [10]. Semakin banyak kendala yang dihadapi maka semakin sulit alokasi jadwal [11]. Teknik optimasi dilakukan untuk menyelesaikan masalah tersebut. Batasan-batasan memiliki peran penting dalam menciptakan jadwal yang efektif. Untuk mencegah benturan jadwal memerlukan batasan pada hubungan antar komponen untuk mencegah benturan jadwal [12].

Adapun batasan lain yang digunakan dalam sistem yaitu tidak mengulang jadwal yang sama dalam satu hari dan waktu. Pihak sekolah bertanggung jawab untuk penyusunan jadwal mata pelajaran [13]. Namun, penyusunan jadwal membutuhkan waktu yang cukup lama. Hal tersebut dikarenakan pihak sekolah menggunakan metode lama ataupun perangkat lunak yang digunakan lebih dari satu sehingga tidak efisien [14]. Penulis mencoba untuk menerapkan Algoritma Genetika dengan proses mutasi *Differential Evolution* pada penjadwalan mata pelajaran di sekolah untuk mengatasi tantangan ini.

Dalam ilmu komputer, algoritma genetika merupakan sebuah metode yang prosesnya melibatkan evolusi biologis yaitu seleksi alam dan genetika. Kategori algoritma evolusioner yang dikenal sebagai algoritma genetika menggunakan metode termasuk populasi awal, seleksi, *crossover* dan mutasi yang dipengaruhi oleh biologi evolusioner [15]. Algoritma genetika lebih disukai daripada metode tradisional. Hal ini dikarenakan kemampuan untuk menghasilkan optimasi berskala besar pada ruang parameter yang kompleks dengan kecepatan yang lebih cepat dan lebih efisien [16]. Masing-masing populasi dimodifikasi dengan cara pengulangan menggunakan algoritma genetika. Hasil akan memperoleh sejumlah penyelesaian yang ideal dan optimal [17]. Algoritma genetika terbukti efektif dalam menyelesaikan benturan jadwal mata pelajaran sesuai penelitian terdahulu. Parameter algoritma genetika berupa populasi awal, nilai *fitness*, seleksi, *crossover*, mutasi dan iterasi. Jumlah populasi dengan hasil yang terbaik berdasarkan nilai *fitness* akan menghentikan proses iterasi [18]. Metode optimasi lain yang mirip dengan Algoritma Genetika yaitu *Differential Evolution*. *Differential Evolution* (DE) merupakan sebuah metode optimasi berfungsi untuk penyelesaian masalah secara kompleks dan terarah [19]. Parameter *differential evolution* sama seperti algoritma genetika, namun prosesnya bermula dari mutasi, *crossover* dan seleksi. Mutasi pada *differential evolution* terdapat faktor skala (F) yang menjadi pengontrol besarnya perubahan dari proses ini. Individu baru terbentuk dari kombinasi 3 individu secara acak yang berasal dari populasi sebelumnya.

Berdasarkan penelitian yang dilakukan oleh Pangestu dkk. pada tahun 2023, didapatkan hasil jika penelitian terkait penjadwalan mata pelajaran algoritma genetika di SD Kreatif Muhammadiyah 2 Bontang. Dalam pengujian menunjukkan bahwa algoritma genetika berhasil menyelesaikan kasus penjadwalan. Pengujian menggunakan teknik sensitivitas dengan tingkat nilai *fitness* rata-rata 0,5 pada skenario 1 dan skenario 2 nilai *fitness* 1 [20]. Dalam penelitian Lukman dkk. pada tahun 2021 membahas mengenai perbandingan implementasi algoritma genetika dan *differential evolution* pada proses penjadwalan. Penelitian tersebut membandingkan nilai *fitness* dan kecepatan eksekusi dari kedua algoritma dan hasil analisis menyebutkan bahwa *differential evolution* memiliki nilai *output* yang lebih baik dibandingkan algoritma genetika [8]. Penelitian oleh F. Mone dan J. E. Simarmata pada tahun 2021 menghasilkan pembuatan jadwal kuliah berupa *web* menggunakan algoritma genetika di Universitas Timor secara efisien. Pengujian menggunakan *black-box testing* dan hasil dari *running* sebanyak 30 kali pada 51 dosen (51 kromosom) secara berurut-turut yaitu 25,86 menit [12]. Pada penelitian lain yang menggunakan algoritma genetika pada kasus penjadwalan sekolah dilakukan oleh P. Puspitasari dan M. A. I. Pakereng di SMP Negeri 2 Wonorego pada tahun 2023. Hasil penelitian tersebut dapat membuat jadwal yang optimal dengan nilai *fitness* sebesar 1 sehingga guru tidak memiliki jam dalam waktu bersamaan [21]. Dari keberhasilan penelitian sebelumnya oleh karena itu penulis akan menggunakan Algoritma Genetika dengan proses mutasi *Differential Evolution* sebagai optimasi penyelesaian masalah.

Menurut kepala kurikulum SMAN 2 Purwakarta, penyusunan jadwal mata pelajaran di sekolah dilakukan dengan dua metode menggunakan aplikasi aSc TimeTables dan Microsoft Excel. Namun, hasil penjadwalan yang dibuat oleh tim kurikulum masih kurang ideal karena sering terjadi benturan jadwal guru jika hanya menggunakan satu *software*. Untuk mengatasi masalah ini, tim kurikulum menggunakan *conditional formatting* di Microsoft Excel sebagai penanda untuk menghindari benturan. Namun proses menggunakan dua *software* memakan waktu, membutuhkan ketelitian tinggi dan berdampak pada kualitas pembelajaran yang diperoleh siswa. Selain itu kedua *software* tersebut berasal dari pihak luar, sehingga penulis mengembangkan sistem penjadwalan khusus untuk SMAN 2 Purwakarta.

Data yang digunakan untuk menyusun jadwal mencakup data dari mata pelajaran, kelas, guru, jam atau slot mengajar dan hari yang disusun. Adapun tujuan dari penelitian untuk menyusun sebuah jadwal mata pelajaran dengan memenuhi kriteria batasan yang telah ditetapkan. Dengan begitu akan diperoleh jadwal pelajaran yang lebih optimal, efektif dan efisien. Berdasarkan latar belakang yang sudah diuraikan, penelitian ini mencoba memberikan solusi berupa sistem penjadwalan mata pelajaran sekolah dengan studi kasus di SMA Negeri 2 Purwakarta menggunakan algoritma genetika dengan proses mutasi dari *differential evolution*. Penelitian ini diharapkan dapat membantu unit kurikulum SMAN 2 Purwakarta dalam penyusunan jadwal mata pelajaran di sekolah dengan sebuah sistem yang lebih mudah dan efisien.

2. METODOLOGI PENELITIAN

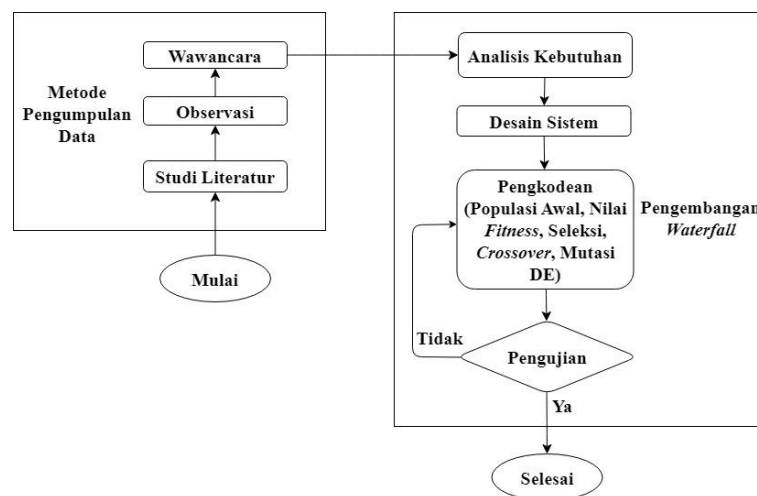
2.1 Model Pengembangan Sistem Waterfall

Penelitian ini menggunakan metode adalah metode untuk mengembangkan sistem informasi. SDLC menjadi langkah awal untuk membuat kerangka kerja dalam pengendalian dan perencanaan sistem informasi. Dengan model pengembangan SDLC salah satunya yaitu *waterfall*. Pada penelitian ini menggunakan *waterfall* yang prosesnya dilakukan secara berurutan sebagai model pengembangan sistem [22].

Model air terjun (*waterfall*) adalah salah satu metodologi pengembangan yang paling banyak digunakan. Pendekatan pengembangan ini mendorong penulis untuk menentukan apa yang harus dicapai oleh sistem. Capaian sistem untuk mengumpulkan dan mendeskripsikan kebutuhan sistem sebelum dibangun. Metodologi ini membantu menguraikan pengembangan yang kompleks menjadi tahapan-tahapan logis berupa analisis, desain, pengkodean dan pengujian. Dari setiap tahapan akan menghasilkan produk akhir yang siap digunakan. Untuk menjamin bahwa sistem tersebut dapat digunakan, setiap tahapan akan membutuhkan validasi, masukan, dan kriteria yang harus dipenuhi [22].

2.2 Alur Penelitian

Alur penelitian yang akan dilakukan secara garis besar dapat dilihat pada gambar 1.



Gambar 1. Alur Penelitian

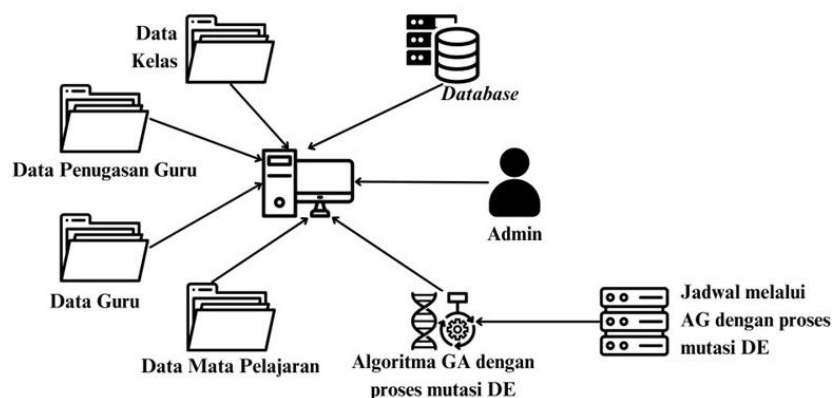
Penjelasan lebih lanjut mengenai diagram alir mengenai alur penelitian akan dibahas secara merinci sebagai berikut.

2.2.1 Metode Pengumpulan Data

Metode pengumpulan data dan informasi dilakukan dalam perancangan sistem yaitu studi literatur, observasi dan wawancara. Studi literatur dengan mengumpulkan data sesuai dengan topik yang relevan. Observasi dilakukan dengan meninjau secara langsung terhadap objek yang akan diteliti. Wawancara pada penelitian ini dilakukan secara langsung oleh kepala kurikulum SMAN 2 Purwakarta yang menentukan persebaran jadwal mengajar guru.

2.2.2 Analisis Kebutuhan

Gambaran untuk mengetahui apa saja yang dibutuhkan oleh sistem dengan visualisasi arsitektur sistem yang terlihat pada gambar 2.

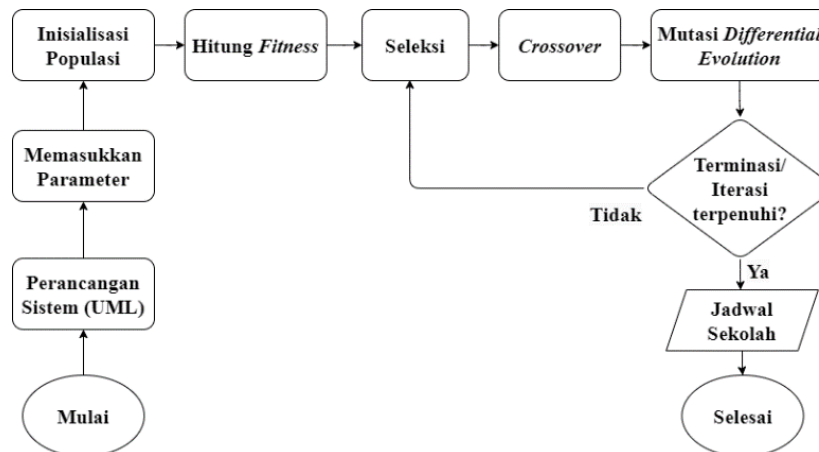


Gambar 2. Arsitektur sistem

Admin berperan penting dalam penggunaan sistem. Sistem diakses oleh admin dalam membuat jadwal. Pembuatan jadwal berkaitan dengan data kelas, data guru, data mata pelajaran dan data penugasan guru. Data penjadwalan diaplikasikan menggunakan algoritma genetika dengan proses mutasi *differential evolution* sehingga dalam sistem menghasilkan jadwal yang dibuat secara otomatis. Jadwal yang sudah dibuat dimasukkan serta disimpan dalam *database*.

2.2.3 Desain Sistem

Diagram alir untuk memperoleh gambaran awal mengenai perancangan sistem penjadwalan mata pelajaran menggunakan Algoritma Genetika proses mutasi *Differential Evolution* dapat dilihat pada gambar 3.



Gambar 3. Diagram alir algoritma genetika dengan proses mutasi *differential evolution*

Visualisasi dari algoritma genetika dengan proses mutasi *differential evolution* terlihat pada gambar 3. Dari diagram alir tersebut dapat diuraikan sebagai berikut:

- UML dirancang untuk penggambaran untuk pengembangan sebuah sistem.
- Parameter penelitian yang digunakan seperti data guru, kelas, mata pelajaran dan penugasan yang ditetapkan oleh admin.
- Inisialisasi populasi merupakan tahap utama dari proses algoritma genetika. Dengan membangkitkan individu-individu agar mendapatkan hasil yang terbaik secara acak. Dalam penelitian ini mengasilkan menentukan waktu dan ruang untuk aktivitas pembelajaran guru dan siswa. Dalam satu individu berisikan hari, jam mengajar, kelas, dan kode guru (termasuk mata pelajaran yang diampu oleh guru terkait).
- Hitung *Fitness* mengevaluasi nilai setiap individu dari jumlah benturan guru. Parameter nilai *fitness* yaitu tidak boleh terjadi benturan jadwal guru dalam jam yang sama. Rumus menghitung *fitness* pada (1).

$$F = \frac{1}{1+(\text{jumlah benturan guru})} \quad (1)$$

- Seleksi dapat menghitung nilai probabilitas (2) dari setiap individu menggunakan metode *roulette-wheel*. Nilai probabilitas akan ditambahkan untuk mendapatkan nilai kumulatif (3). Perhitungan nilai kumulatif akan bertambah terus menerus hingga seluruh individu sudah terhitung. Dari perhitungan nilai probabilitas dan kumulatif ditentukan *roulette-wheel* sesuai dengan jumlah individu. Penempatan kromosom yang memiliki nilai *fitness* lebih besar berada dipotongan lingkaran *roulette-wheel* lebih besar, dibandingkan dengan kromosom dengan nilai *fitness* rendah.

$$\text{Probabilitas} = \frac{\text{nilai fitness}}{\text{total nilai fitness}} \quad (2)$$

$$\text{Kumulatif} = \text{gabungan nilai probabilitas individu} \quad (3)$$

- Crossover* dengan menukarkan kromosom antara dua buah individu yang berdasarkan *crossover rate*. Pada penelitian ini *crossover rate* yang digunakan yaitu 0.9. Nilai *crossover rate* sudah dimasukkan pada proses pengkodean. Titik potong diperlukan untuk menukarkan kromosom. Titik potong yang digunakan dalam penelitian adalah *one point crossover*. *One point crossover* merupakan metode kawin silang dengan membagi dua bagian dengan satu titik potong. Hal yang dilakukan pada metode *crossover* ini yaitu membangkitkan bilangan acak antara 1 sampai dengan n (panjang kromosom dikurangi 1). Penempatan titik potong dipilih secara acak oleh sistem.
- Mutasi dengan proses *Differential Evolution* yaitu membuat individu baru dari individu sebelumnya dengan menggunakan rumus yang berbeda dari Algoritma Genetika (secara acak). Jika rumus mutasi pada *Differential Evolution* V merupakan mutasi baru, F merupakan faktor skala sebesar 0,5 (nilai tersebut sudah dimasukkan pada proses pengkodean) dan X_{r1}, X_{r2}, X_{r3} merupakan kromosom yang sedang dimutasi. Secara sistematis rumus mutasi *Differential Evolution* dituliskan sebagai berikut (4):

$$V = X_{r1} + F \times (X_{r2} - X_{r3}) \quad (4)$$

- h. Terminasi yaitu dimana proses jika iterasi telah terpenuhi maka seluruh populasi telah diproses dan mencapai jadwal yang terbaik sesuai dengan nilai *fitness*. Proses ini juga berpengaruh terhadap masukkan dari jumlah populasi dan maksimal generasi pada sistem. Seluruh masukkan akan diproses hingga selesai dan akan dipilih menurut sistem untuk jadwal yang terbaik.

2.2.4 Pengkodean

Pengkodean dilakukan dengan bahasa pemrograman PHP. Bahasa PHP didukung dengan *framework* Laravel, HTML, CSS yang terkoneksi dengan *database* dan dijalankan memanfaatkan Laragon sebagai *web server* lokal. Pengkodean dilakukan pada Visual Studio Code dengan konfigurasi dari Laravel. Proses Algoritma Genetika dengan mutasi *Differential Evolution* yang telah dikembangkan pada tahap desain sistem diubah menjadi kode optimasi sehingga menghasilkan jadwal secara otomatis.

2.2.5 Pengujian System Usability Scale (SUS)

Kuesioner yang telah divalidasi oleh para ahli sebagai instrumen penelitian menggunakan variabel kegunaan (*usability*). Instrumen pengukuran yang dapat digunakan untuk mengukur kegunaan sistem adalah *System Usability Scale* (SUS). Pengujian SUS memiliki kelebihan mudah dipelajari. Pada penelitian yang dilakukan oleh Isnaeni Rachmawati dan Resad Setyadi menyebutkan bahwa ukuran sampel yang digunakan pada SUS dapat menggunakan ukuran sampel berukuran sangat kecil yaitu minimal 2 responden. Daftar pertanyaan yang diujikan berjumlah 10 tetap memberikan hasil yang terbaik [23].

3. HASIL DAN PEMBAHASAN

3.1 Istilah Algoritma Genetika

Implementasi komponen algoritma genetika dalam penelitian ini, dengan proses mutasi menggunakan *differential evolution* dapat dinyatakan secara singkat dalam istilah pada tabel 1.

Tabel 1. Istilah algoritma genetika

Istilah	Pengertian
Populasi	Jumlah jadwal acak yang dihasilkan oleh kuantitas tertentu. Jadwal 1 dan 2 disebut sebagai satu unit populasi.
Individu	satu rangkaian jadwal mata pelajaran.
Kromosom	Satu kelas dalam satu minggu, gen dalam sebuah jadwal akan memengaruhi panjang kromosom.
Gen	Setiap pertemuan guru pada sebuah kelas.
Iterasi	Angka yang merepresentasikan laju pembangkitan dan perulangan.
Crossover	Proses kawin silang, dimana dua kromosom dipasangkan dengan pasangannya setelah memotong matriks tertentu.
Mutasi	Pembentukan individu baru dengan kombinasi dari 3 vektor secara acak berasal dari populasi sebelumnya.

3.2 Rancangan Kebutuhan

Observasi dan wawancara dilakukan secara langsung di SMAN 2 Purwakarta. Wawancara dilakukan kepada kepala kurikulum yang mengatakan bahwa pembuatan jadwal pelajaran sudah menggunakan aplikasi bernama ASc TimeTables namun masih melakukan pemeriksaan secara manual menggunakan *conditional formatting* di Microsoft Excel untuk meminimalkan benturan jadwal guru.

Pihak sekolah terbuka untuk mencoba sistem penjadwalan yang baru namun belum paham mengenai Algoritma Genetika. Hasil penjadwalan yang dibuat oleh tim kurikulum masih kurang ideal karena sering terjadi benturan jadwal guru jika hanya menggunakan satu *software* saja. Sehingga penulis melakukan penelitian untuk membuat sistem penjadwalan mata pelajaran khusus SMAN 2 Purwakarta yang menggunakan Algoritma Genetika dengan proses mutasi *Differential Evolution*. Dalam perancangan sistem membutuhkan perangkat keras dan lunak dalam konfigurasi dan pengembangan sistem. Perangkat keras yang digunakan pada penelitian terdapat pada tabel 2.

Tabel 2. Spesifikasi perangkat keras

Spesifikasi
Laptop bermerek HP
Model 14s-cf2xxx
Prosesor Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz (8 CPUs) ~2.1GHz
RAM 8 GB

Pada tabel 2 mencantumkan perangkat keras yang digunakan pada penelitian ini. Terdapat merek laptop yaitu HP, model laptop 14s-cf2xxx, prosesor pada perangkat yaitu Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz (8 CPUs)

~2.1GHz dan RAM sebanyak 8 GB yang digunakan. Sementara itu, perangkat lunak yang digunakan pada penelitian ini dapat terlihat pada tabel 3.

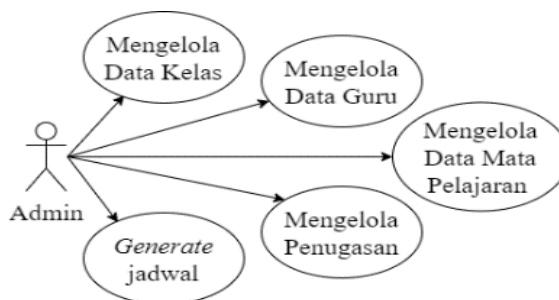
Tabel 3. Kebutuhan perangkat lunak

Perangkat Lunak	Kegunaan
Windows 10	Sistem operasi
Microsoft Edge	Web browser
Laragon versi 6.0	Web server
MySQL versi 8.0.30	Database
Visual Studio Code	Editor pengkodean

Pada tabel 3 mencantumkan kebutuhan dari perangkat lunak dan kegunaannya yang digunakan pada penelitian ini. Kegunaan untuk sistem operasi menggunakan Windows 10. Microsoft Edge digunakan sebagai *web browser*. Laragon menggunakan versi 6.0 sebagai *web server*. MySQL dengan versi 8.0.30 digunakan untuk *database*. Terakhir, Visual Studi Code digunakan untuk editor pengkodean.

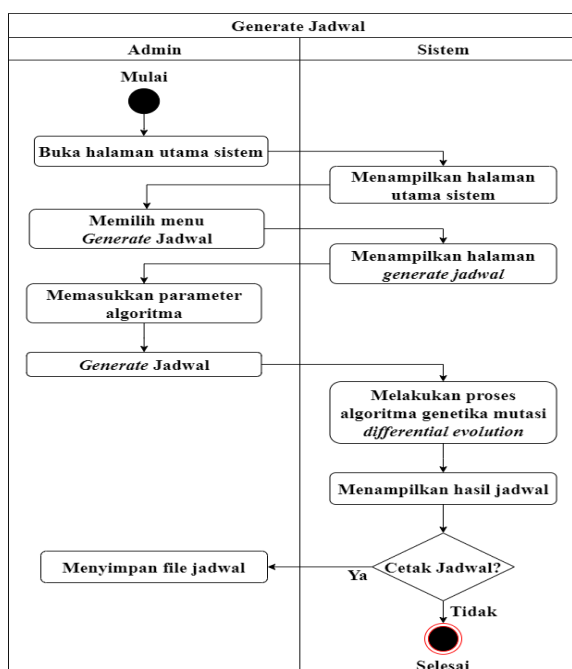
3.3 Hasil Desain Sistem

Kerangka acuan proses perancangan sistem dalam penelitian menggunakan UML (*Unified Modelling Language*). Proses perancangan sistem diuraikan dengan *use case diagram* dan *activity diagram*. *Use case diagram* bagian dari UML. Fungsinya untuk mengilustrasikan perilaku sistem dari sudut pandang luar sistem dan hubungan admin terkait dengan sistem yang sedang dibuat. *Use case diagram* dari sistem yang akan dibuat dapat dilihat pada gambar 4.



Gambar 4. Use Case Diagram

Admin akan melakukan semua tugas yang ada dalam sistem seperti mengelola data kelas, data guru, data mata pelajaran, penugasan dan *generate* jadwal. Pengembangan dari *Use Case Diagram* menciptakan *Activity Diagram*. *Activity Diagram* berbentuk visual yang mengilustrasikan deretan proses aktivitas sebuah sistem. Berikut prosedur dari *activity diagram* dari sistem pembuatan jadwal mata pelajaran menggunakan Algoritma Genetika proses mutasi *Differential Evolution* di SMAN 2 Purwakarta terlihat pada gambar 5.



Gambar 5. Activity diagram generate jadwal

Dalam gambar 5 menggambarkan bahwa admin memiliki kendali seluruh proses pada sistem. Setiap proses yang dilakukan oleh admin akan berpengaruh pada sistem. Admin dapat memulai dengan membuka pada halaman utama sistem dan sistem akan menampilkan halaman utama. Setelah itu admin akan membuat jadwal dengan memilih menu *generate* jadwal dan sistem akan menampilkannya. Jika sudah admin memasukkan parameter dari algoritma genetika yaitu jumlah populasi dan maksimal generasi sesuai yang diinginkan. Admin dapat memencet tombol *generate* dan sistem akan membuat jadwal sesuai dengan proses algoritma genetika dan proses mutasinya menggunakan *differential evolution*. Setelah selesai sistem akan menampilkan jadwal yang terbaik. Admin dapat mencetak jadwal atau tidak dan proses pembuatan jadwal selesai.

3.4 Proses Perhitungan Algoritma Genetika Mutasi *Differential Evolution*

Seluruh tahapan dibuat dalam bentuk pemisalan agar proses perhitungan lebih mudah dipahami. Proses perhitungan algoritma genetika dengan proses mutasi *differential evolution* dapat diuraikan sebagai berikut.

a. Inisialisasi Populasi

Dalam penelitian ini, solusi penjadwalan menghasilkan penentuan waktu dan ruang untuk aktivitas pembelajaran guru. Contoh tabel 4, tabel 5, dan tabel 6 adalah mewakili dalam satu individu. Angka yang terdapat di dalam kelas merupakan kode guru yang sudah mewakili nama guru dan mata pelajaran.

Tabel 4. Individu 1

Hari	Jam Ke	Waktu	Kelas		
			X1	X2	X3
Senin	1	07.00 – 08.00	5	1	5
	2	08.00 – 09.00	5	1	5
	3	09.00 – 10.00	1	3	2
Selasa	1	07.00 – 08.00	2	2	4

Tabel 5. Individu 2

Hari	Jam Ke	Waktu	Kelas		
			X1	X2	X3
Senin	1	07.00 – 08.00	5	6	4
	2	08.00 – 09.00	5	6	4
	3	09.00 – 10.00	1	3	3
Selasa	1	07.00 – 08.00	6	2	1

Tabel 6. Individu 3

Hari	Jam Ke	Waktu	Kelas		
			X1	X2	X3
Senin	1	07.00 – 08.00	4	5	1
	2	08.00 – 09.00	4	5	1
	3	09.00 – 10.00	2	3	4
Selasa	1	07.00 – 08.00	5	4	5

Tabel 4 mewakili dari individu 1, tabel 5 mewakili individu 2 dan tabel 6 mewakili individu 3. Pada setiap individu membangkitkan jadwal yang berisikan hari, jumlah jam mengajar, waktu mengajar, kelas dan kode guru.

b. Hitung *Fitness*

Pada individu 1 memiliki jumlah benturan guru sebanyak 2 kali, individu 2 sebanyak 1 kali, individu 3 sebanyak 1 kali maka hasil perhitungan dari nilai *fitness* masing-masing individu sesuai dengan persamaan (1) yaitu:

$$\text{Individu 1: } F = 1/1+2 = 0,333$$

$$\text{Individu 2: } F = 1/1+1 = 0,5$$

$$\text{Individu 3: } F = 1/1+1 = 0,5$$

$$\text{Total nilai fitness} = 0,333 + 0,5 + 0,5 = 1,333$$

c. Seleksi

Memproses nilai *fitness* dengan menghitung nilai probabilitas dan kumulatif setiap individu. Menghitung nilai probabilitas sesuai dengan persamaan 2:

$$\text{Individu 1} = 0,333/1,333 = 0,2$$

$$\text{Individu 2} = 0,5/1,333 = 0,3$$

$$\text{Individu 3} = 0,5/1,333 = 0,3$$

Setelah nilai probabilitas sudah dihitung maka langkah selanjutnya yaitu menghitung nilai kumulatif sesuai dengan persamaan 3

$$\text{Kumulatif 1} = \text{Probabilitas 1} = 0,2$$

$$\text{Kumulatif 2} = \text{Probabilitas 1} + \text{Probabilitas 2} = 0,2 + 0,3 = 0,5$$

Kumulatif 3 = Probabilitas 1 + Probabilitas 2 + Probabilitas 3 = 0,2 + 0,3 + 0,3 = 0,7

Dalam metode *roulette-wheel* dari perhitungan proses seleksi nilai individu 2 dan 3 akan dipilih pada proses selanjutnya.

d. *Crossover*

Individu 2 dan 3 akan di-*crossover*, kromosom yang akan dipilih yaitu secara acak sebagai berikut:

Sebelum *crossover*

Individu 2 = Kromosom X2 = 6 6 | 3 2

Individu 3 = Kromosom X3 = 1 1 | 4 5

Menerapkan *one point crossover* secara acak. Jika yang terpilih 2 maka akan seperti berikut:

Sesudah *crossover*

Individu 2 = Kromosom X2 = 6 6 4 5

Individu 3 = Kromosom X3 = 1 1 3 2

Jika hasil proses *crossover* berbentuk tabel dapat dilihat pada tabel 7 dan 8.

Tabel 7. Hasil *Crossover* Individu 2

Hari	Jam Ke	Waktu	Kelas		
			X1	X2	X3
Senin	1	07.00 – 08.00	5	6	4
	2	08.00 – 09.00	5	6	4
	3	09.00 – 10.00	1	4	3
Selasa	1	07.00 – 08.00	6	5	1

Tabel 8. Hasil *Crossover* Individu 3

Hari	Jam Ke	Waktu	Kelas		
			X1	X2	X3
Senin	1	07.00 – 08.00	4	5	1
	2	08.00 – 09.00	4	5	1
	3	09.00 – 10.00	2	3	3
Selasa	1	07.00 – 08.00	3	4	2

Tabel 7 dan tabel 8 merupakan hasil dari proses *crossover*. Hasil dari populasi baru pada individu 3 masih belum memenuhi nilai *fitness*. Sehingga dilanjutkan pada tahap selanjutnya.

e. *Mutasi Differential Evolution*

Contoh perhitungan mutasi dari rumus menggunakan individu 3 sebagai berikut:

Kromosom X1 = 4 4 2 3 (Target vektor)

Kromosom X2 = 5 5 3 4 (Vektor acak 1)

Kromosom X3 = 1 1 3 2 (Vektor acak 2)

Maka jika dimasukkan ke dalam persamaan 4 yaitu sebagai berikut:

$$V = (4,4,2,3) + 0.5 \times (5,5,3,4 - 1,1,3,2)$$

$$V = (4,4,2,3) + 0.5 \times (4,4,0,2)$$

$$V = (4,4,2,3) + (2,2,0,1)$$

$$V = (6,6,2,4)$$

Populasi baru dari kromosom X1 dengan menggunakan vektor acak X2 dan X3 dari individu 3 adalah V = (6,6,2,4). Individu baru dari populasi mutasi akan bertambah setiap iterasi dilakukan.

f. Iterasi

Proses iterasi berpengaruh terhadap masukkan dari jumlah populasi dan maksimal generasi pada sistem. Seluruh masukkan akan diproses hingga selesai dan akan dipilih menurut sistem untuk jadwal yang terbaik.

3.5 Implementasi Sistem

Implementasi dari pengkodean bahasa pemrograman yang sudah dirancang sebelumnya akan diuraikan pada tahap ini. Tampilan antarmuka sistem yaitu sebagai berikut.

a. Halaman Awal

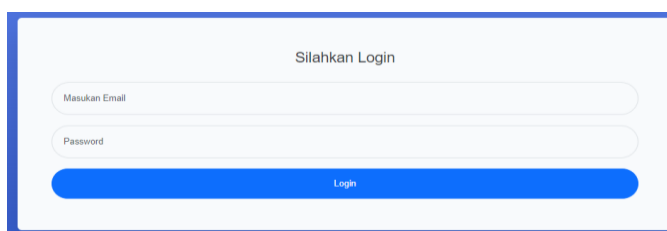
Halaman ini berisikan tampilan awal dari sistem. Jika ingin masuk ke halaman utama dapat memencet tombol “*Login*” pada kanan atas halaman terlihat pada gambar 6.



Gambar 6. Halaman awal

b. Halaman *Login*

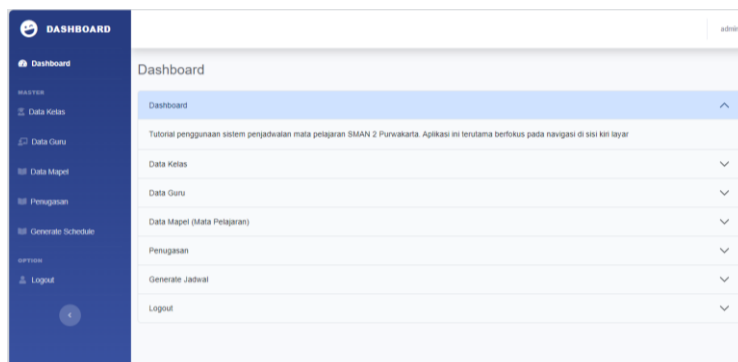
Pengguna dapat memasukkan email dan *password* pada kolom yang terlihat pada halaman *Login*. Email dan *password* sesuai dengan data yang sudah tersimpan dalam sistem terlihat pada gambar 7.



Gambar 7. Halaman *login*

c. Halaman Utama Sistem

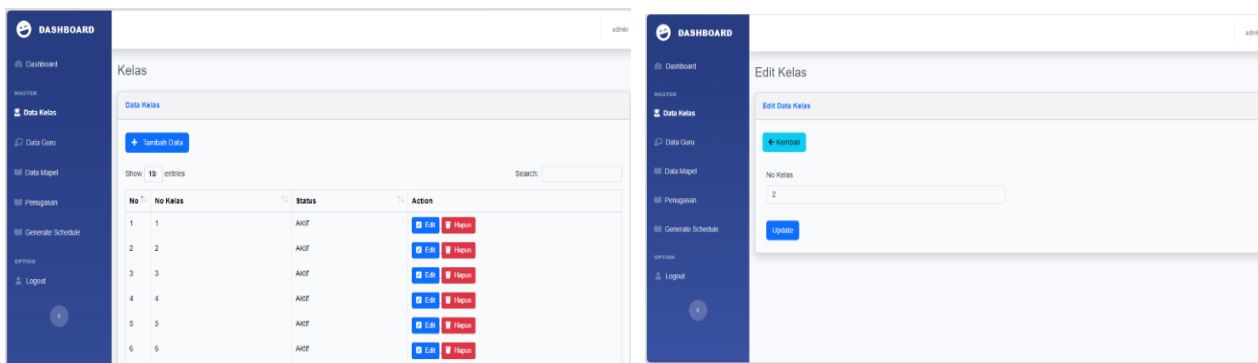
Pada tampilan *dashboard* menampilkan tata cara penggunaan sistem penjadwalan. Sistem tersebut berfokus pada navigasi sisi kiri halaman terlihat pada gambar 8.



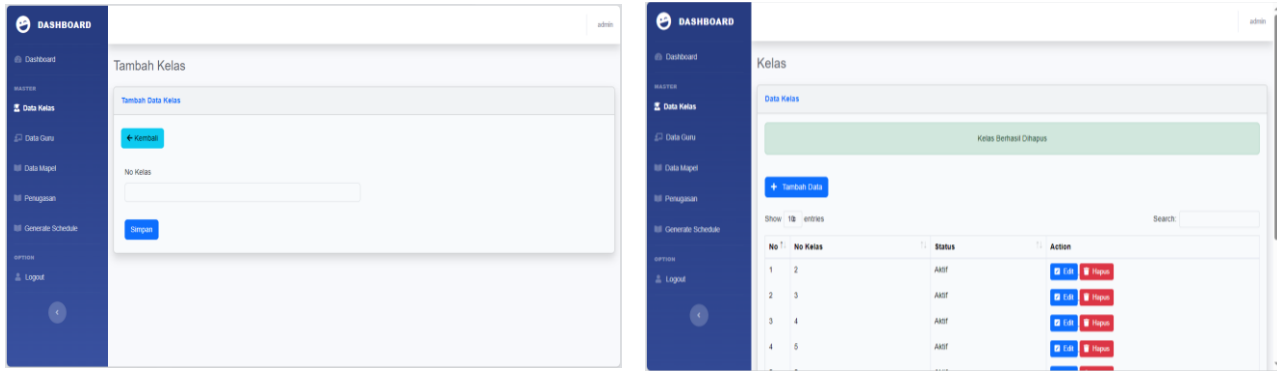
Gambar 8. Halaman *dashboard*

d. Halaman Data Kelas

Pada halaman ini menampilkan kelas-kelas mana saja yang akan dimasukkan dalam pembuatan jadwal. Pengguna dapat menambahkan, mengedit, dan menghapus kelas. Data yang wajib diisi dalam halaman ini yaitu nomor kelas terlihat pada gambar 9 dan gambar 10.



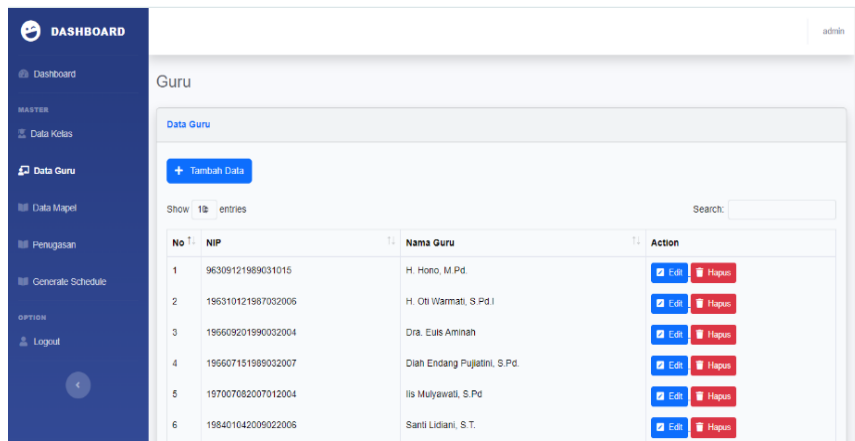
Gambar 9. Halaman data kelas (1)



Gambar 10. Halaman data kelas (2)

e. Halaman Data Guru

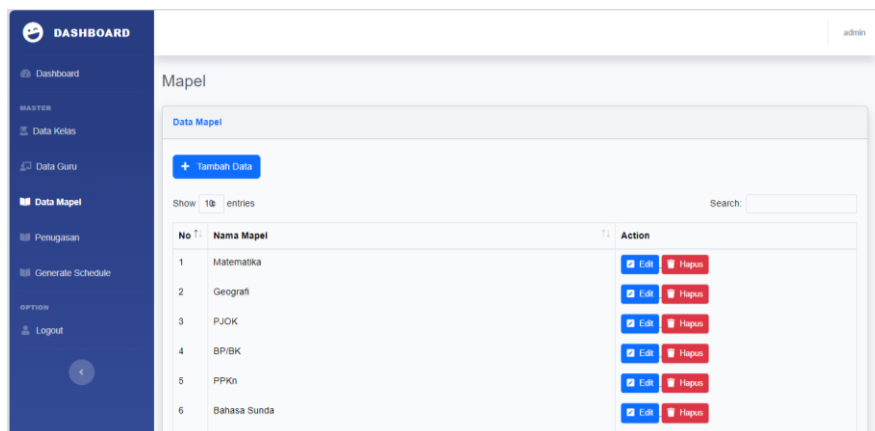
Pada halaman ini menampilkan guru mana saja yang akan dimasukkan dalam pembuatan jadwal. Dalam halaman data guru dapat menambahkan guru dengan memencet tombol “Tambah Data”. Pada data guru kolom yang wajib diisi yaitu Nama Guru, NIP, dan Jenis Kelamin. Jika ingin menghapus data guru dapat memencet tombol “Hapus”. Mengubah data dengan memencet tombol “Edit” pada halaman data guru terlihat pada gambar 11.



Gambar 11. Halaman data guru

f. Halaman Data Mapel

Pada halaman ini menampilkan mata pelajaran (mapel) mana saja yang akan dimasukkan dalam pembuatan jadwal. Dalam halaman data mapel dapat menambahkan mata pelajaran dengan memencet tombol “Tambah Data”. Pada data mapel kolom yang wajib diisi yaitu Nama Mapel. Jika ingin menghapus data mapel dapat memencet tombol “Hapus”. Dapat mengubah data dengan memencet tombol “Edit” pada halaman data mapel terlihat pada gambar 12.

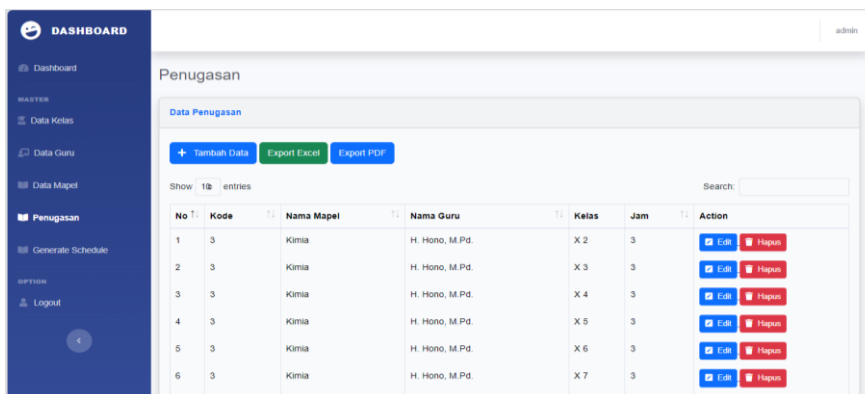


Gambar 12. Halaman data mapel

g. Halaman Penugasan

Pada halaman ini menampilkan penugasan guru mana saja yang akan dimasukkan dalam pembuatan jadwal. Dalam halaman penugasan dapat menambahkan penugasan guru dengan memencet tombol “Tambah Data”. Dalam menambahkan penugasan kolom yang wajib diisi yaitu Kode Guru, Jam Mengajar, Kelas, Guru/Pengajar dan Nama Mapel. Jika ingin menghapus data penugasan dapat memencet tombol “Hapus”. Dapat mengubah data dengan memencet

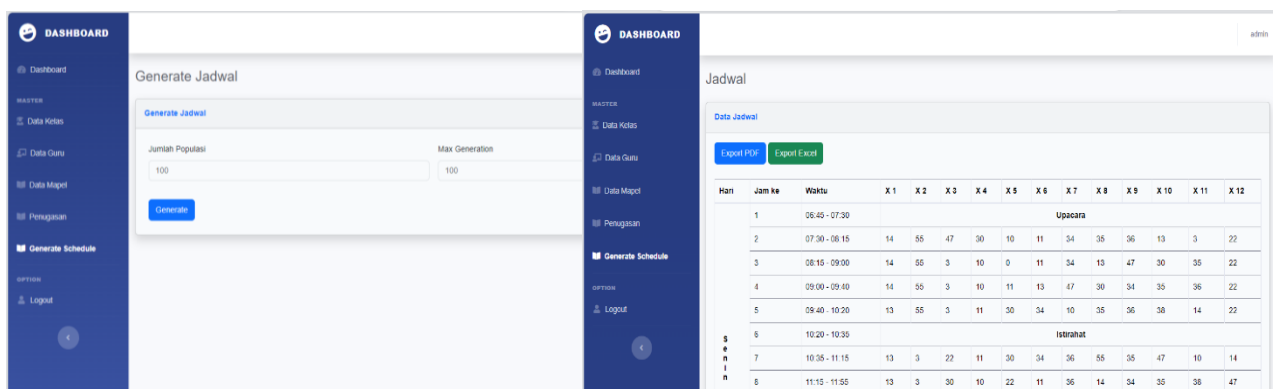
tombol “Edit” pada halaman data penugasan. Pengguna dapat mengunduh penugasan berupa PDF pada “Export PDF” atau Microsoft Excel “Export Excel” terlihat pada gambar 13.



Gambar 13. Halaman penugasan guru

h. Halaman *Generate Jadwal*

Pada halaman *generate* jadwal masukkan jumlah populasi dan maksimal generasi dalam rentan 1 hingga 100. Masukkan tersebut akan diolah oleh sistem maka hasilnya akan menampilkan data jadwal dari senin hingga jumat. Jumlah masukkan akan mempengaruhi waktu pembuatan jadwal. Semakin tinggi angka, semakin lama waktu yang dibutuhkan. Pengguna dapat mengunduh jadwal berupa PDF atau Microsoft Excel terlihat pada gambar 14.



Gambar 14. Halaman *generate* jadwal

i. Hasil *Generate Jadwal* PDF

Hasil dari *generate* jadwal yang sudah diunduh berupa PDF terlihat pada gambar 15.

Hari	Jam ke	Waktu	X 1	X 2	X 3	X 4	X 5	X 6	X 7	X 8	X 9	X 10	X 11	X 12
	1	06:45 - 07:30							Upacara					
	2	07:30 - 08:15	14	55	47	30	10	11	34	35	36	13	3	22
	3	08:15 - 09:00	14	55	3	10	0	11	34	13	47	30	35	22
	4	09:00 - 09:40	14	55	3	10	11	13	47	30	34	35	36	22
	5	09:40 - 10:20	13	55	3	11	30	34	10	35	36	38	14	22
	6	10:20 - 10:35							Istirahat					
Senin	7	10:35 - 11:15	13	3	22	11	30	34	36	55	35	47	10	14
	8	11:15 - 11:55	13	3	30	10	22	11	36	14	34	35	38	47
	9	11:55 - 12:25	47	3	14	13	22	30	36	44	10	34	0	11
	10	12:25 - 13:05							Istirahat					
	11	13:05 - 13:45	10	47	13	3	22	14	11	44	34	30	35	36
	12	13:45 - 14:25	3	10	34	22	35	36	13	44	30	14	11	38
	13	14:25 - 15:05	3	35	11	14	10	47	30	34	36	38	22	13

Gambar 15. Hasil *generate* jadwal

j. Hasil *Generate Jadwal* Excel

Hasil dari *generate* jadwal yang sudah diunduh berupa Ms. Excel terlihat pada gambar 16.

Hari	Jam ke	Waktu	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10	X11	X12
Senin	1	06:45 - 07:30	Upacara											
	2	07:30 - 08:15	14	55	47	30	10	11	34	35	36	13	3	22
	3	08:15 - 09:00	14	55	3	10	0	11	34	13	47	30	35	22
	4	09:00 - 09:40	14	55	3	10	11	13	47	30	34	35	36	22
	5	09:40 - 10:20	13	55	3	11	30	34	10	35	36	38	14	22
	6	10:20 - 10:35	Istirahat											
	7	10:35 - 11:15	13	3	22	11	30	34	36	55	35	47	10	14
	8	11:15 - 11:55	13	3	30	10	22	11	36	14	34	35	38	47
	9	11:55 - 12:25	47	3	14	13	22	30	36	44	10	34	0	11
	10	12:25 - 13:05	Istirahat											
	11	13:05 - 13:45	10	47	13	3	22	14	11	44	34	30	35	36
	12	13:45 - 14:25	3	10	34	22	35	36	13	44	30	14	11	38
	13	14:25 - 15:05	3	35	11	14	10	47	30	34	36	38	22	13
	1	06:45 - 07:15	Pembiasaan											
	2	07:15 - 08:00	0	11	47	14	13	10	22	30	34	3	35	36
	3	08:00 - 08:45	11	14	22	30	10	34	35	13	36	3	38	39
	4	08:45 - 09:25	22	11	10	14	3	34	35	13	36	38	39	30
	5	09:25 - 10:05	30	38	11	13	3	58	10	14	22	34	35	36
	6	10:05 - 10:20	Istirahat											

Gambar 16. Hasil generate jadwal

3.6 Pengujian SUS

Pertanyaan dari pengujian SUS terlihat pada tabel 9.

Tabel 9. Butir pertanyaan SUS

No.	Pertanyaan SUS
1.	Saya berpikir akan menggunakan sistem ini lagi
2.	Saya merasa sistem ini rumit untuk digunakan
3.	Saya merasa sistem ini mudah digunakan
4.	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini
5.	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya
6.	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini)
7.	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat
8.	Saya merasa sistem ini membingungkan
9.	Saya merasa tidak ada hambatan dalam menggunakan sistem ini
10.	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini

Skor dari setiap pertanyaan memiliki 5 tingkat jawaban yang disediakan oleh SUS dapat dilihat pada tabel 10.

Tabel 10. Skor pertanyaan SUS

Jawaban	Pertanyaan SUS
Sangat Tidak Setuju	1
Tidak Setuju	2
Netral	3
Setuju	4
Sangat Setuju	5

Pedoman untuk menentukan skor kuesioner SUS yaitu sebagai berikut:

- Kurangi satu skor pengguna untuk pertanyaan ganjil,
- Kurangi lima skor pengguna untuk pertanyaan genap, dan
- Kalikan jumlah dari skor pengguna dengan angka 2,5.

Untuk menghitung skor rata-rata dengan mencari skor SUS dari setiap responden yaitu menjumlahkan semua skor pengguna dan membaginya dengan jumlah total responden. Skor SUS dihitung dengan menggunakan rumus berikut (5).

$$\text{Skor rata-rata} = \frac{\text{Jumlah Skor SUS}}{\text{Jumlah Responden}} \quad (5)$$

Jika setelah skor dihitung maka dapat disimpulkan dari parameter penilaian yaitu *Grade Scale* dan *Adjective Rating*. Untuk lebih lanjut mengenai hasil skor yang dicapai dapat dilihat pada tabel 11.

Tabel 11. Bobot skor SUS

SUS Score	Grade Scale	Adjective Rating
> 80.3	A	Excellent

68 – 80.3	B	Good
68	C	Okay
51-68	D	Poor
< 51	E	Awful

Sampel pengujian menggunakan data kelas 10 tahun ajaran 2023/2024 semester ganjil. Alasan data tersebut dipilih sebab di SMAN 2 Purwakarta untuk pertama kalinya menggunakan Kurikulum Merdeka. Oleh karena itu, untuk kedepannya sistem ini akan digunakan untuk penjadwalan di SMAN 2 Purwakarta. Data pengujian menggunakan 12 kelas, 17 mata pelajaran dan 25 guru. Setiap guru memiliki kode masing-masing yang dibuat pada halaman penugasan. Sehingga dalam kode tersebut sudah mewakili mata pelajaran yang diampu oleh guru terkait. Pada penelitian ini menggunakan 3 responden. Responden yang dipilih berkaitan dengan sistem penjadwalan yang dibuat. Responden tersebut yaitu 1 kepala kurikulum dan 2 staf kurikulum sekolah. Berikut adalah akumulasi hasil pengujian SUS yang merupakan data asli sesuai dengan pengisian kuesioner oleh *user* sebelum diolah menggunakan perhitungan SUS dapat dilihat pada tabel 12.

Tabel 12. Data asli kuesioner responden

No.	Skor Asli									
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
R1	3	2	4	4	3	3	4	2	3	4
R2	4	2	4	2	4	2	4	2	4	3
R3	4	1	4	2	5	2	5	2	4	3

Data asli yang sesuai dengan pengisian kuesioner tersebut akan diolah dengan perhitungan SUS. R1, R2, dan R3 merupakan responden dan P1-P10 merupakan pertanyaan. Berdasarkan pedoman SUS pertanyaan ganjil akan dikurangi 1 maka sebagai contoh R1 dari pertanyaan 1 data asli bernilai 3 maka hasil perhitungan SUS menjadi 2. Langkah selanjutnya kurangi 5 skor pengguna untuk pertanyaan genap sebagai contoh R2 dari pertanyaan 2 data asli bernilai 2 maka hasil perhitungan SUS menjadi 3. Seluruh perhitungan SUS setiap pertanyaan akan dijumlahkan maka hasil dari responden 1 totalnya berjumlah 22, responden 2 berjumlah 29 dan responden 3 berjumlah 32. Setelah itu, kalikan jumlah dari skor setiap responden dengan angka 2,5. Terakhir, untuk mengetahui skor rata-rata dapat menjumlahkan keseluruhan nilai total dari setiap responden sesuai dengan persamaan 5.

$$\text{Skor rata-rata} = \frac{55 + 72,5 + 80}{3} = 69,1$$

Akumulasi perhitungan dari data dapat dilihat pada tabel 13.

Tabel 13. Hasil perhitungan SUS

No.	Skor Perhitungan SUS										Total	Nilai (Total × 2,5)
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10		
R1	2	3	3	1	2	2	3	3	2	1	22	55
R2	3	3	3	3	3	3	3	3	3	2	29	72,5
R3	3	4	3	3	4	3	4	3	3	2	32	80
Skor rata-rata (hasil akhir)												69,1

3.7 Pembahasan

Penelitian dilakukan sesuai dengan metodologi penelitian yang sudah dipaparkan sebelumnya. Hasil dari penelitian ini yaitu sebuah sistem penjadwalan mata pelajaran menggunakan algoritma genetika dengan proses mutasi *differential evolution*. Proses algoritma genetika dimulai dengan membangkitkan populasi awal secara acak yang berisikan kumpulan dari individu-individu. Setiap individu tersebut dihitung nilai *fitness* dan seleksi yang mana saja individu yang dapat lanjut ke proses selanjutnya. Individu yang terpilih akan dikawin silang (*crossover*) pada kromosom secara acak dan proses terakhir mutasi gen dari kromosom tertentu sehingga menghasilkan individu baru. Jika individu sudah memenuhi nilai *fitness* yang baik maka proses selesai, namun jika individu belum memenuhi nilai *fitness* maka proses akan terus berulang hingga mencapai nilai *fitness* terbaik atau memenuhi batasan yang diberikan. Batasan dari sistem penjadwalan ini yaitu tidak boleh terjadi benturan jadwal antar guru dalam jam dan hari yang sama.

Sistem berhasil memenuhi batasan tersebut sehingga jadwal yang terbaik akan dipilih dan ditampilkan setelah *generate* jadwal. Namun kelemahan dari algoritma genetika sendiri hasil jadwal sangat mengacak dan proses *generate* jadwal membutuhkan waktu yang cukup lama sehingga diharapkan pada penelitian selanjutnya memperbaiki mengenai proses *generate* ini. Selain itu, jumlah jam mengajar masih belum sinkron pada halaman penugasan dengan hasil *generate* jadwal, perlu perbaikan mengenai hal tersebut. Berdasarkan data yang telah diolah, sistem mendapatkan skor dari perhitungan SUS yaitu 69,1. Dari skor tersebut mendapatkan bobot skor SUS *Grade Scale* "B" dan *Adjective Rating* "Good". Sehingga memerlukan pengembangan lebih lanjut pada sistem. Namun, dengan kategori tersebut sistem layak digunakan jika berpatokan pada pengujian SUS. Keputusan penggunaan sistem ini diserahkan sepenuhnya kepada pihak sekolah.

4. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan dapat disimpulkan bahwa sistem penjadwalan mata pelajaran menggunakan algoritma genetika proses mutasi *differential evolution* yaitu sistem yang dirancang mempunyai 4 tahapan yaitu analisis kebutuhan (menganalisis kebutuhan sistem serta perangkat lunak dan keras), desain sistem (UML digunakan dalam rancangan desain sistem), pengkodean (bahasa pemrograman PHP dengan perhitungan dari algoritma genetika dan mutasi *differential evolution*) dan pengujian (menggunakan SUS). Uji sistem pada sisi kegunaan yang menggunakan SUS. Responden yang terlibat yaitu kepala kurikulum dan staf kurikulum sekolah yang menjadi admin. Hasil pengujian SUS pada penelitian ini yaitu 69,1 dan memiliki rentan kelayakan *good* serta dapat digunakan dengan pertimbangan oleh pihak sekolah. Dengan adanya sistem ini, sekolah tidak perlu menggunakan 2 buah *software* yang berbeda. Sehingga, pembuatan penjadwalan lebih efektif dan efisien. Untuk saran penelitian selanjutnya dapat membuat tampilan agar lebih menarik dan setiap guru memiliki persebaran mengajar kelas mana saja yang akan diajar oleh guru tersebut dalam satu minggu dan data tersebut dapat diunduh pada sistem secara langsung. Sehingga guru tidak mencari kodenya dan menandai penugasan kelas mana saja yang akan diajar. Proses *generate* jadwal memerlukan perbaikan agar tidak memakan waktu yang cukup lama.

REFERENCES

- [1] N. Wahyono, A. Subhan Akbar, dan J. Minardi, "Sistem Penjadwalan Mengajar di SMA NU Al Ma'ruf Kudus Menggunakan Algoritma Genetika," *J Inf Sys Comp*, vol. 2, no. 1, hlm. 1–14, Jul 2022, doi: 10.34001/jister.v1i2.267.
- [2] I. Efendi, M. Prawitasari, dan H. Susanto, "Implementasi Penilaian Pembelajaran Pada Kurikulum 2013 Mata Pelajaran Sejarah," *Prabayaksa*, vol. 1, no. 1, hlm. 21, Mar 2021, doi: 10.20527/prb.v1i1.3081.
- [3] T. K. Jeriko, D. F. Racma, C. E. Widjayanti, dan A. A. Setyawan, "Penerapan Algoritma Genetika Dalam Sistem Informasi Penjadwalan Mata Kuliah Berbasis Website Pada Stikom Yos Sudarso Purwokerto," *Jurnal Muara Sains, Teknologi, Kedokteran dan Ilmu Kesehatan*, vol. 6, no. 1, hlm. 101–118, 2022, doi: <https://doi.org/10.24912/jmstkk.v6i1.17262>.
- [4] T. J. Ahyana dan Y. Jumaryadi, "Perancangan Sistem Informasi Penjadwalan Mengajar Menggunakan Metode Algoritma Genetika (Studi Kasus: SMK Satria Jakarta)," *Ensiklopedia of Journal*, vol. 1, no. 2, hlm. 112–116, 2019, [Daring]. Tersedia pada: <http://jurnal.ensiklopediaku.org>
- [5] S. N. S. Sidabutar, M. Amin, dan A. Putri, "Penjadwalan Operasi Mesin Produksi Dengan Metode CDS (Campbell Dudek Smith) di PT Tjokro Bersaudara Balikpapanindo," *PROTON (Jurnal Ilmu-Ilmu Teknik Mesin)*, vol. 11, no. 2, hlm. 53–61, 2019, doi: <https://doi.org/10.31328/jp.v1i1i2.1234>.
- [6] S. N. Sari, R. Kaban, A. Khaliq, dan A. Andari, "Sistem Penjadwalan Mata Pelajaran Sekolah Menggunakan Metode *Hybrid Artificial Bee Colony* (HABC)," *JNASTEK*, vol. 2, no. 1, hlm. 20–32, Feb 2022, doi: 10.61306/jnastek.v2i1.21.
- [7] Y. Makaborang, "Evaluasi Implementasi Kurikulum 2013 Mata Pelajaran Biologi di SMA Negeri," *Kelola*, vol. 6, no. 2, hlm. 130–145, Des 2019, doi: 10.24246/j.jk.2019.v6.i2.p130-145.
- [8] N. Lukman, M. Irfan, A. Nugraha, dan J. Jumadi, "Comparison Between Genetic Algorithm with Differential Evolution in Study Scheduling," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 1098, no. 3, hlm. 032082, Mar 2021, doi: 10.1088/1757-899X/1098/3/032082.
- [9] M. Furqan, R. Ananda, dan Armansyah, "Algoritma Genetika Untuk Perancangan Aplikasi Penjadwalan Mata Pelajaran," *J-SAKTI (Jurnal Sains Komputer & Informatika)*, vol. 6, no. 2, hlm. 591–600, 2022, doi: <http://dx.doi.org/10.30645/j-sakti.v6i2.476>.
- [10] P. A. Rizki, Y. Hendriyani, D. Novaliendry, dan K. Budayawan, "Rancang Bangun Aplikasi Penjadwalan Mata Pelajaran SMK Muhammadiyah 1 Pekanbaru Berbasis Web Menggunakan Algoritma Genetika," *Fakultas Ilmu Pendidikan Universitas Pahlawan Tuanku Tambusai*, vol. 7, no. 3, hlm. 24787–24797, 2023, doi: <https://doi.org/10.31004/jptam.v7i3.10541>.
- [11] D. Oktarina dan A. Hajjah, "Perancangan Sistem Penjadwalan Seminar Proposal dan Sidang Skripsi dengan Metode Algoritma Genetika," *JOISIE*, vol. 3, no. 1, hlm. 32, Jul 2019, doi: 10.35145/joisie.v3i1.421.
- [12] F. Mone dan J. E. Simarmata, "Aplikasi Algoritma Genetika Dalam Penjadwalan Mata Kuliah," *BAREKENG: J. Il. Mat. & Ter.*, vol. 15, no. 4, hlm. 615–628, Des 2021, doi: 10.30598/barekengvol15iss4pp615-628.
- [13] M. K. Fajarlestari dan I. B. Suban, "Kombinasi *Crossover* dan Mutasi Terbaik pada Algoritma Genetika dalam Penjadwalan Mata Kuliah," *tc*, vol. 22, no. 4, hlm. 843–853, Nov 2023, doi: 10.33633/tc.v22i4.9298.
- [14] A. Yusron Mubarak dan U. Chotijah, "Penerapan Algoritma Genetika Untuk Mencari Optimasi Kombinasi Jalur Terpendek Dalam Kasus Travelling Salesman Problem," *j. teknologi terpadu*, vol. 7, no. 2, hlm. 77–82, 2021, doi: 10.54914/jtt.v7i2.424.
- [15] D. Wahyuningsih dan E. Helmud, "Penerapan Algoritma Genetika Untuk Optimasi Penjadwalan pada MTS Negeri 1 Pangkalpinang," *SISFOKOM*, vol. 9, no. 3, hlm. 435–441, Des 2020, doi: 10.32736/sisfokom.v9i3.994.
- [16] Eka Yulia Sari, Dina Yulina H, dan Titik Rahmawati, "Pemodelan Sistem Optimasi Penjadwalan Mata Kuliah Dengan Algoritma Genetika," *teknimedia*, vol. 4, no. 1, hlm. 70–78, Jun 2023, doi: 10.46764/teknimedia.v4i1.97.
- [17] Y. Elva, "Sistem Penjadwalan Mata Pelajaran Menggunakan Algoritma Genetika," *JurTI*, vol. 3, no. 1, hlm. 49, Jul 2019, doi: 10.36294/jurti.v3i1.687.
- [18] A. Andriyadi, D. Yuliawati, dan S. Saleh, "Penerapan Algoritma Genetika Untuk Penjadwalan Sidang dan Seminar IIB Darmajaya," *Seminar Nasional Hasil Penelitian dan Pengabdian Masyarakat 2022*, vol. 1, hlm. 22–31, 2022.
- [19] Aswani, V. V. Praveen, dan S. Thangavelu, "Performance Analysis of Variants of Differential Evolution on Multi-Objective Optimization Problems," *Indian Journal of Science and Technology*, vol. 8, no. 17, hlm. 1–6, Agu 2015, doi: 10.17485/ijst/2015/v8i17/65727.
- [20] L. A. Pangestu, S. H. Suryawan, dan A. J. Latipah, "Penerapan Algoritma Genetika Dalam Penjadwalan Mata Pelajaran," *Jurnal Penelitian Teknik Informatika, Manajemen Informatika dan Sistem Informasi*, vol. 10, no. 2, hlm. 194–205, Okt 2023, doi: 10.31294/inf.v10i2.16701.

- [21] P. Puspitasari dan M. A. I. Pakereng, "Implementasi Algoritma Genetika Untuk Penjadwalan Sekolah (Studi Kasus: SMP Negeri 2 Wonoregore)," *Jurnal Sains Komputer & Informatika (J-SAKTI)*, vol. 7, no. 1, hlm. 369–382, 2023, doi: <http://dx.doi.org/10.30645/j-sakti.v7i1.600>.
- [22] A. A. Wahid, "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi," *Jurnal Ilmu-ilmu Informatika dan Manajemen STMIK*, hlm. 1–5, 2020.
- [23] I. Rachmawati dan R. Setyadi, "Evaluasi Usability Pada Sistem Website Absensi Menggunakan Metode SUS," *josh*, vol. 4, no. 2, hlm. 551–561, Jan 2023, doi: 10.47065/josh.v4i2.2868.