

Algoritma Genetika pada Sistem Penjadwalan *Shift* Kerja Perusahaan Berbasiskan Permintaan

Nina Aulia Ramadhani¹

Intan Dzikria²

¹Universitas 17 Agustus 1945 Surabaya
1461900183@surel.untag-sby.ac.id

Abstrak, Penjadwalan merupakan proses perencanaan dan penentuan waktu operasional sebagai bagian dari proses pekerjaan secara keseluruhan dalam kurun waktu tertentu. Penjadwalan *shift* kerja pada perusahaan menjadi hal yang penting untuk diperhatikan, mengingat penjadwalan *shift* bertujuan untuk mengatur jam kerja karyawan agar dapat meningkatkan produktivitas karyawan sehingga memberi dampak positif baik bagi karyawan maupun perusahaan, tanpa menimbulkan kelelahan yang berarti. Penelitian ini bertujuan untuk meneliti penerapan algoritma genetika pada pengembangan sistem penjadwalan *shift* kerja perusahaan. Pada sistem penjadwalan yang dikembangkan oleh peneliti, tetap melibatkan karyawan suatu perusahaan untuk memperoleh jadwal *shift* kerja yang ter-generate secara otomatis, sehingga hak karyawan untuk memilih hari liburnya sesuai jumlah libur yang ditentukan dapat terpenuhi. Dengan adanya penelitian ini, diharapkan dapat bermanfaat dalam memperoleh variasi model penjadwalan *shift* kerja karyawan suatu perusahaan, meningkatkan efisiensi dan efektivitas waktu dalam penyusunan jadwal *shift* kerja karyawan, serta mengurangi probabilitas jadwal yang bentrok dalam proses penyusunan jadwal *shift* kerja. Algoritma genetika merupakan Teknik untuk menemukan solusi optimal dari permasalahan yang mempunyai banyak solusi. Teknik ini akan melakukan pencarian dari beberapa solusi yang diperoleh sampai mendapatkan solusi terbaik sesuai dengan kriteria yang telah ditentukan atau yang disebut sebagai fungsi fitness. Dapat disimpulkan penerapan algoritma genetika dalam pengembangan sistem penjadwalan perusahaan ini berpengaruh dalam proses perolehan solusi terbaik, sehingga dapat dijadikan acuan sebagai sistem untuk menyusun jadwal *shift* pada perusahaan.

1. Pendahuluan

Permasalahan yang sering dihadapi oleh perusahaan seringkali muncul dalam optimasi bidang sumber daya manusia. Jalan keluar yang dapat ditempuh oleh perusahaan untuk mengatasi masalah tersebut adalah dengan membuat suatu system penjadwalan kerja untuk karyawan mereka. Sumber daya manusia yang produktif mampu memberikan dapat positif bagi perusahaan. Tidak hanya terkait dengan sumber daya manusianya saja, penjadwalan juga akan mempengaruhi produktivias manajemen perusahaan. [1]

Pada beberapa kasus, seperti perusahaan yang beroperasi 24/7, memilih untuk menerapkan sistem *shift* kerja pada karyawannya agar dapat terhindar dari jam kerja yang berlebih (overtime), sehingga *shift* kerja dirasa mampu menjadi strategi yang tepat untuk meningkatkan

produktivitas perusahaan secara maksimal [2]. Namun pada praktiknya, seringkali proses penyusunan hingga penjadwalan *shift* kerja masih berjalan secara kurang efektif dan efisien. Tidak jarang, hasil dari proses penjadwalan *shift kerja* malah menimbulkan suatu permasalahan baru seperti kesehatan para karyawan yang menurun, akibat pemberian waktu kerja yang berlebihan, dan penjadwalan *shift* kerja yang kurang tepat, sehingga memforsir tenaga dan memakan waktu istirahat yang dibutuhkan oleh karyawan [2].

Oleh karena itu, diperlukan suatu penjadwalan *shift* karyawan yang menggunakan system komputerisasi untuk meningkatkan efisiensi dan efektivitas, serta meminimalisir terjadinya kesalahan saat proses penyusunan jadwal. Salah satu cara yang bisa diterapkan adalah menggunakan algoritma genetika. Algoritma genetika merupakan cabang dari algoritma evolusi yang terkenal sebagai sebuah solusi permasalahan pengoptimalan yang tidak mudah dipecahkan pada berbagai bidang, biologi, sosiologi, ekonomi, fisika, dan sebagainya [3]. Algoritma genetika sebelumnya telah diteliti untuk dimanfaatkan pada penjadwalan *shift* kerja di *call center* Telkomsel [4], kemudian untuk optimasi penjadwalan perawat [5], dan untuk penjadwalan *shift* kerja anggota kepolisian sektor Magelang Tengah [6].

Meskipun telah banyak digunakan dan mampu memberikan hasil penjadwalan yang optimal, namun pada penelitian terdahulu masih belum melakukan pengembangan dari algoritma genetika untuk penjadwalan *shift* kerja yang disertai dengan keterlibatan karyawan seperti melakukan permintaan pribadi (*request*) tanggal hari libur yang dikehendaki, agar meningkatkan fleksibilitas dan kenyamanan karyawan.

Penjadwalan *shift* kerja perusahaan dilakukan untuk mengatur jam kerja karyawan agar lebih terstruktur dan proses bisnis dapat berjalan dengan baik. Dengan adanya *shift* kerja, kegiatan perusahaan akan dapat terus berjalan karena sekelompok karyawan akan mulai bekerja ketika kelompok sebelumnya telah selesai bekerja [7].

Tujuan penelitian ini adalah menerapkan algoritma genetika pada penjadwalan *shift* kerja perusahaan sehingga mampu meningkatkan optimasi penjadwalan berdasarkan batasan-batasan yang diberikan oleh perusahaan. Hasil penelitian ini diharapkan dapat memecahkan masalah tersebut dengan melakukan pengembangan system penjadwalan *shift* kerja pada perusahaan dengan menggunakan algoritma genetika agar proses penyusunan jadwal *shift* kerja menjadi lebih cepat, efektif, efisien, dan optimal, sesuai dengan kehendak karyawan.

2. Tinjauan Pustaka

2.1. Penjadwalan Shift Kerja

Penjadwalan dapat diartikan sebagai suatu kegiatan alokasi sumber daya dengan memiliki kendala (batasan) yang diberikan kepada suatu objek seperti di ruang-waktu, sedemikian rupa untuk memenuhi sedekat mungkin tujuan yang diinginkan [8].

Pada penjadwalan *shift* kerja sekelompok karyawan akan mulai bekerja ketika kelompok sebelumnya telah selesai bekerja [7]. Proses penjadwalan *shift* kerja perusahaan saat ini dilakukan secara manual dan seringkali menyulitkan sumber daya manusianya. Karena akan membutuhkan waktu lebih banyak untuk penyusunan jadwalnya, dan tidak menutup kemungkinan dari hasil proses tersebut, terjadi bentrokan pada jadwal *shift* karyawan. Permasalahan lain yang mungkin terjadi pada proses penjadwalan *shift* kerja perusahaan adalah, pembagian jumlah libur dan jarak libur karyawan tidak dapat merata, serta terjadi kesulitan dalam menentukan urutan *shift* kerja [1].

2.3 Algoritma Genetika

Metode algoritma genetika adalah salah satu algoritma yang meniru mekanisme dari genetika alam yang dikembangkan untuk mencari solusi bagi permasalahan seperti penjadwalan. Permasalahan dengan model matematika yang kompleks atau bahkan sulit dibangun dapat diselesaikan menggunakan algoritma genetika [3].

Algoritma genetika memiliki beberapa komponen utama dalam proses perhitungannya, yaitu generasi awal, representasi kromosom, *crossover*, mutasi, nilai *fitness*, seleksi, dan kondisi berhenti atau *termination condition* [2]. Generasi awal adalah menentukan sejumlah populasi awal yang berisikan bilangan secara acak dan menyesuaikan permasalahan yang akan diselesaikan [2]. Menurut Widodo yang dikutip dari [2] representasi kromosom dalam algoritma genetika memiliki beberapa jenis, yaitu representasi biner, representasi permutasi, representasi bilangan riil, representasi bilangan bulat, representasi menggunakan simbol-simbol (seperti A, B, C), dan lain-lain. Perhitungan penalti digunakan untuk mengurangi kesalahan karena menyalahi aturan, yang dapat terjadi selama proses perhitungan algoritme genetika [2]. Perhitungan penalti dilakukan dengan cara menghitung berapa banyak pelanggaran yang dilakukan pada setiap kromosom, dan juga memberikan bobot pada tiap jenis pelanggaran [2]. Perhitungan penalti ini akan berpengaruh kepada nilai *fitness*, dimana nilai *fitness* yang tinggi memiliki kemungkinan yang tinggi untuk terpilih ke generasi berikutnya [2].

Menurut Desiani dan Arhami yang dikutip dari [2], nilai *fitness* adalah alat ukur sebuah kromosom untuk mengetahui seberapa baik kinerja sebuah kromosom. Kromosom yang baik merupakan kromosom yang memiliki nilai *fitness* yang tinggi dan kromosom yang seperti ini memiliki kemungkinan untuk terpilih ke generasi berikutnya. *Crossover* merupakan proses kawin silang dengan cara memilih dua kromosom dari kromosom induk yang dipilih secara acak, kemudian hasil dari kawin silang tersebut menghasilkan kromosom anak atau biasa disebut dengan *offspring*, untuk menambah keragaman individu yang terdapat pada sebuah populasi. Mutasi adalah proses mengubah nilai gen di dalam sebuah kromosom, baik satu atau lebih gen [2]. Mutasi berfungsi untuk memberikan kromosom baru dan akan mendapatkan nilai *fitness* dari kromosom baru tersebut, sehingga mendapat keragaman kromosom dan nantinya akan menemukan kromosom dengan nilai *fitness* yang lebih baik dan dapat digunakan pada generasi berikutnya dan menghasilkan solusi yang optimal [2]. Gen yang diubah pada suatu kromosom tersebut bersifat acak, tidak ditentukan gen mana yang akan ditukar [2]. Seleksi adalah tahap terakhir dari proses algoritma, yaitu menentukan kromosom mana yang layak untuk lanjut ke proses perhitungan pada generasi berikutnya [2].

Pada algoritme genetika, setelah melalui semua prosesnya dan menghasilkan *offspring*, nilai *fitness* tersebut akan dilihat apakah sudah mencapai nilai yang optimal atau belum, apabila nilai *fitness* sudah mencapai nilai yang optimal, maka proses akan berhenti dan menjadikan kromosom dengan nilai *fitness* optimal tersebut menjadi solusi permasalahan, jika belum, maka proses seleksi akan terus berjalan sampai memenuhi kriteria kondisi berhenti [2]. Menurut [9] yang dikutip dari [2] kriteria-kriteria kondisi berhenti tersebut adalah sebagai berikut :

1. Iterasi akan berhenti sampai pada generasi tertentu yang telah ditentukan (*maximum generation*),
2. Iterasi akan berhenti jika nilai *fitness* yang dihasilkan pada beberapa iterasi yang berurutan menghasilkan nilai *fitness* yang sama atau tidak berubah,
3. Iterasi akan berhenti jika nilai *fitness* yang dihasilkan pada iterasi-iterasi berikutnya tidak menghasilkan nilai *fitness* yang lebih baik

Struktur dasar dari GA untuk penjadwalan ditunjukkan pada pseudocode berikut [6] :

Pseudocode	Algoritma
Genetika	
populasi <= Ukuran Populasi	
maxGen <= Maksimal Generasi	
MR <= Mutation Rate	
bangkitkanGenerasiAwal (jmlHari)	
evaluasiKromosom(fitness function())	
while (g < maxGen)	
seleksiKromosom(fitness)	
random [1,n]	
if (new fitness > old fitness) then	
kromosom dipilih	
proses crossover ()	
if (new fitness > old fitness) then	
posisi_x = random [1, jml_gen]	
crossover[posisi_x]	
proses mutasi (MR)	
while (k < jumlahGen) do	
r = random bilangan [0,1]	
if (r < MR) then	
posisi_a = rand [1, jml_gen]	
posisi_b = rand [1, jml_gen]	
mutasi_gen[posisi_a, posisi_b]	69
k = k +1	
g = g+1	

3. Metodologi Penelitian

3.1. Teknik Pengumpulan Data

Teknik pengumpulan data yang digunakan oleh peneliti dalam penelitian ini menggunakan metode observasi dan studi literatur. Metode observasi merupakan metode pengumpulan data dengan cara melakukan pengamatan pada berbagai aspek yang dibutuhkan dalam proses penelitian, meliputi data serta model penjadwalan saat ini agar kemudian dapat dikembangkan ke depannya oleh peneliti. Pada penelitian ini, observasi dilakukan peneliti secara langsung pada salah satu perusahaan telekomunikasi di kota Surabaya. Data yang dikumpulkan berupa data nama karyawan, data shift, dan data hari.

Peneliti melakukan studi literatur dengan cara melakukan studi mengenai algoritma genetika, serta penjadwalan karyawan melalui literatur seperti jurnal, dan sumber ilmiah lain yang berkaitan dengan topik penelitian.

3.2 Metode Pengembangan Perangkat Lunak

Model pengembangan perangkat lunak air terjun (*Waterfall*) digunakan dalam penelitian ini menyediakan pendekatan siklus hidup pengembangan perangkat lunak secara terurut (*sequential*) atau sering juga disebut model sekuensial linier [10]. Menurut Pambudi, Waluyo dan Fatich [11] model *Waterfall* memiliki beberapa tahapan dalam pengembangan perangkat lunak yang dilakukan dalam penelitian.

Tahap pertama merupakan analisis masalah yang ditemukan pada proses bisnis yang ada saat ini, analisis kebutuhan, dan analisis sistem. Hal-hal yang dibutuhkan dalam mengembangkan sistem penjadwalan shift kerja perusahaan dikumpulkan dengan metode observasi dan studi literatur. Tahap desain adalah tahap kedua untuk membangun arsitektur sistem secara keseluruhan. Desain perangkat lunak melibatkan pengidentifikasian dan penggambaran abstraksi sistem perangkat lunak mendasar dan hubungannya. Tahap ketiga merupakan penulisan kode program untuk mewujudkan desain perangkat lunak, dengan menulis kode program sesuai dengan desain interface yang telah dibuat sebelumnya, menggunakan bahasa pemrograman PHP dan database MySQL. Setelah perancangan sistem selesai dilakukan, tahapan selanjutnya adalah melakukan pengujian untuk memastikan apakah sistem yang telah dirancang tersebut telah sesuai dengan kebutuhan. Tahap terakhir yaitu pemeliharaan melibatkan beberapa hal seperti koreksi kesalahan yang tidak ditemukan pada tahap siklus hidup, meningkatkan implementasi unit sistem dan meningkatkan layanan sistem saat persyaratan dan kebutuhan baru ditemukan.

3.3 Pengujian Perangkat Lunak

Setelah perancangan sistem selesai dilakukan, tahapan selanjutnya adalah melakukan pengujian untuk memastikan apakah sistem yang telah dirancang tersebut telah sesuai dengan kebutuhan atau belum. Pengujian sistem yang digunakan dalam pengembangan sistem penjadwalan shift kerja perusahaan adalah *blackbox testing*. *Blackbox testing*, menguji apakah kebutuhan dan persyaratan fungsional perangkat lunak telah sesuai. Proses yang dilakukan dalam *blackbox testing* adalah membuat himpunan kondisi masukan (input), kemudian diuji seluruh syarat fungsional dalam sistem penjadwalan *shift* kerja perusahaan tersebut [11].

4. Hasil dan Pembahasan

4.1 Analisis Kebutuhan

Kebutuhan perangkat lunak penjadwalan *shift* kerja perusahaan dianalisis di dalam penelitian ini untuk mengetahui kebutuhan fungsional dan non fungsional sistem. Kebutuhan fungsional merupakan kebutuhan akan fasilitas yang dibutuhkan oleh sistem, serta aktifitas apa saja yang dilakukan oleh sistem secara umum [12]. Syarat kebutuhan fungsional antara lain berisi tentang aktifitas-aktifitas yang harus dilakukan dalam sistem, berdasar pada prosedur serta fungsi-fungsi bisnis, serta didokumentasikan dalam model [13].

Tabel 1 menunjukkan kebutuhan fungsional system penjadwalan *shift* kerja perusahaan dengan keterangan aktor yang diberikan hak akses atas kebutuhan fungsional tersebut. Terdapat tiga aktor utama dalam sistem yaitu admin, *team leader* (TL), dan pegawai..

Tabel 1. Tabel Kebutuhan Fungsional

CODE	FUNCTIONAL REQUIREMENT	AKTOR
F-001	<i>Login</i>	Admin, TL, Pegawai
F-001-a	Autentikasi <i>Login</i>	Admin, TL, Pegawai
F-001-b	Pemulihan <i>Password</i>	Admin, TL, Pegawai
F-001-c	Ganti <i>Password</i>	Admin, TL, Pegawai
F-002	Mengelola Data Divisi	Admin
F-002-a	Menambah Data Divisi	Admin
F-002-b	Melihat Data Divisi	Admin
F-002-c	Mengubah Data Divisi	Admin

F-002-d	Menghapus Data Divisi	Admin
F-003	Mengelola Data <i>Jobdesc</i>	Admin
F-003-a	Menambah Data <i>Jobdesc</i>	Admin
F-003-b	Melihat Data <i>Jobdesc</i>	Admin
F-003-c	Mengubah Data <i>Jobdesc</i>	Admin
F-003-d	Menghapus Data <i>Jobdesc</i>	Admin
F-004	Pengelolaan Data <i>Team Leader</i>	Admin
F-004-a	Menambah Data <i>Team Leader</i>	Admin
F-004-b	Melihat Data <i>Team Leader</i>	Admin
F-004-c	Mengubah Data <i>Team Leader</i>	Admin
F-004-d	Menghapus Data <i>Team Leader</i>	Admin
F-005	Pengelolaan Data Pegawai	Admin, TL
F-005-a	Menambah Data Pegawai	Admin, TL
F-005-b	Melihat Data Pegawai	Admin, TL
F-005-c	Mengubah Data Pegawai	Admin, TL
F-005-d	Menghapus Data Pegawai	Admin, TL
F-006	Pengelolaan Jadwal	TL, Pegawai
F-006-a	Mulai Sesi <i>Request</i> Jadwal	TL
F-006-b	<i>Input Request</i> Jadwal	Pegawai
F-006-c	Menyetujui <i>Request</i> Jadwal	TL
F-006-d	Mengirim Notifikasi Persetujuan <i>Request</i> Jadwal	Sistem
F-006-e	Menerima Notifikasi Persetujuan <i>Request</i> Jadwal	Pegawai
F-006-f	Menolak <i>Request</i> Jadwal	TL
F-006-g	Mengirim Notifikasi Penolakan <i>Request</i> Jadwal	System
F-006-h	Menerima Notifikasi Penolakan <i>Request</i> Jadwal	Pegawai
F-006-i	Menutup Sesi <i>Request</i> Jadwal	TL
F-006-j	Mengatur parameter penjadwalan	TL
F-006-k	<i>Generate</i> Proses Penjadwalan	TL
F-006-l	<i>Generate</i> Jadwal <i>Shift</i> Berdasar Algoritma Genetika	Sistem
F-006-m	Menampilkan Jadwal	TL, Pegawai
F-006-n	Mengubah Jadwal	TL
F-007	<i>Logout</i>	Admin, TL, Pegawai

Kebutuhan non fungsional sistem ditinjau dari aspek keandalan, ketersediaan, keamanan, pemeliharaan, dan kinerja. Dalam aspek keandalan, sistem dapat berjalan sesuai kebutuhan pengguna, dengan catatan pengguna terkoneksi dengan jaringan internet untuk menghasilkan fitur dan data yang optimal. Aspek ketersediaan menunjukkan kebutuhan pengguna terkait penggunaan 24 jam selama 7 hari. Sistem juga harus mampu menjamin keamanan data pengguna dengan menerapkan *hashing*. Pada aspek pemeliharaan, sistem membutuhkan berbagai dukungan perangkat lunak untuk mempermudah pemeliharaan yang akan dilakukan. Sistem juga diharapkan mampu untuk bekerja secara *real time* untuk memaksimalkan fitur notifikasi verifikasi *request* jadwal.

4.2 Rancangan Sistem

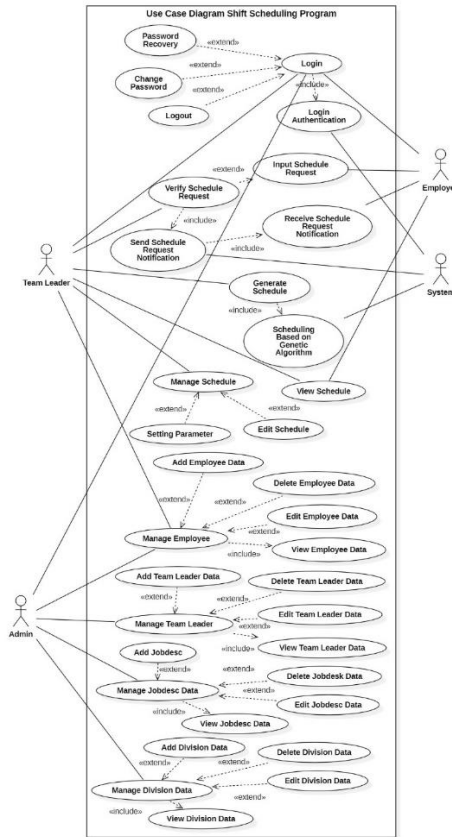
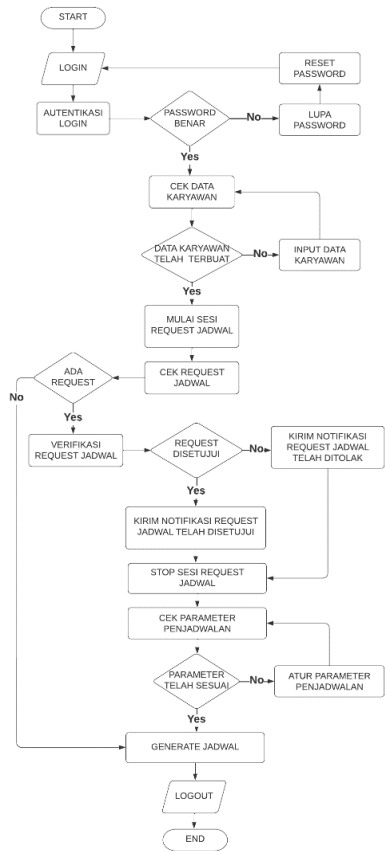
Gambar 1a menunjukkan proses penjadwalan *shift* kerja yang dilakukan oleh *team leader*, dimulai dari *team leader login* ke system, manajemen *schedule request*, hingga melakukan *generate schedule*. Proses diawali dari *team leader* melakukan *login* ke sistem, setelah berhasil *login*, *team leader* dapat melakukan pengecekan data karyawan, jika data karyawan sudah

dibuat, *team leader* dapat melanjutkan proses *schedule request* dengan cara memulai sesi *schedule request*. Selanjutnya *team leader* melakukan proses verifikasi *schedule request*, setelah seluruh *schedule request* selesai dilakukan verifikasi, *team leader* dapat melanjutkan ke tahap terakhir yaitu melakukan *generate schedule*.

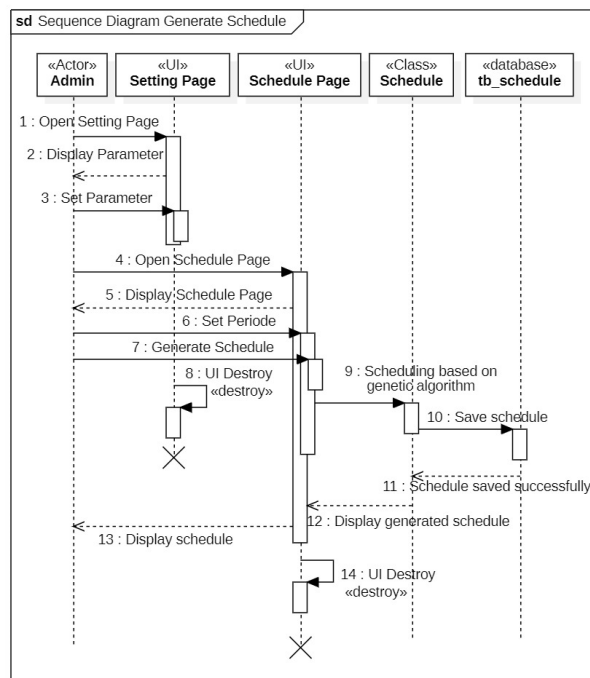
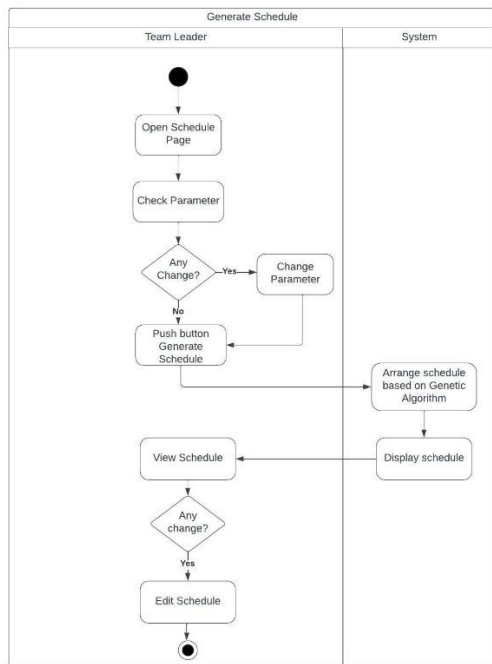
Gambar 1b menunjukkan *use case diagram shift scheduling system*. *Use case diagram* menjabarkan empat aktor pengguna dari aplikasi, mulai dari admin, *team leader*, karyawan, dan sistem itu sendiri. Aktor *team leader* memiliki beberapa kasus pengguna utama, antara lain melakukan *login*, melakukan verifikasi *schedule request* yang diajukan oleh karyawan, melakukan *generate schedule*, serta pengelolaan data karyawan, Aktor *employee* (karyawan) disini memiliki kasus pengguna utama, yaitu melakukan *input schedule request* yang kemudian akan diverifikasi oleh aktor *team leader*. Aktor admin, memiliki beberapa kasus pengguna, antara lain, melakukan pengelolaan (*manage*) data karyawan, data *team leader*, data divisi, dan data *jobdesc*.

Penelitian ini juga merancang *activity diagram* yang menggambarkan aliran fungsionalitas dalam suatu sistem informasi, dan mendefinisikan dimana *work flow* dimulai, dan *work flow* akan berhenti, aktifitas apa saja yang terjadi selama *work flow*, dan bagaimana urutan kejadian aktifitas tersebut [14]. Gambar 2a menunjukkan alur proses *generate schedule* yang dapat dilakukan oleh *team leader*, dimulai dari aktifitas pengecekan parameter apakah sudah sesuai atau belum, jika belum sesuai maka *team leader* dapat melakukan *setting* parameter terlebih dahulu, kemudian aktifitas *generate schedule* agar sistem dapat melakukan penjadwalan dengan penerapan algoritma genetika, setelah proses penjadwalan selesai, maka sistem akan menampilkan jadwal hasil *generate*, setelah jadwal disimpan, *team leader* dapat melakukan edit jadwal apabila ada perubahan, sehingga *team leader* dapat menampilkan jadwal *shift* kerja karyawan sesuai kebutuhan.

Sequence diagram merupakan alat yang bekerja berlandaskan kepada objek di dalam pengembangan sistem untuk menampilkan interaksi antar objek [14]. *Sequence diagram* menggambarkan interaksi antar objek di dalam dan di sekitar sistem, berupa pesan (*message*) yang disusun dalam suatu urutan waktu berdasarkan urutan kejadian yang dilakukan oleh seorang aktor dalam menjalankan sistem [15]. Pada Gambar 2b menunjukkan *sequence diagram generate schedule* yang mana proses di dalamnya berdasar pada *activity diagram*.

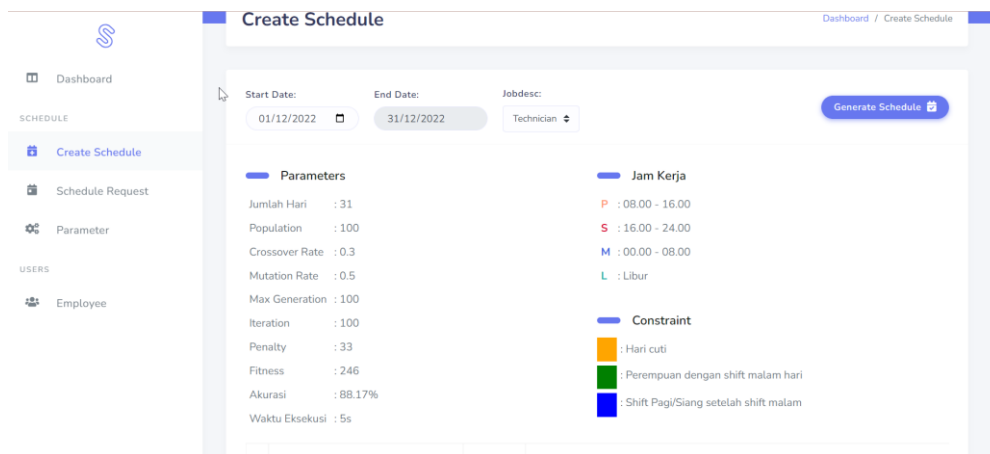


Gambar 1a. (kiri) Alur Proses Penjadwalan (*Generate Schedule*) Team Leader
 Gambar 1b. Use Case Diagram *Shift Scheduling System*



Gambar 2a. (kiri) Activity Diagram *Generate Schedule*
 Gambar 2b. (kanan) Sequence Diagram *Generate Schedule*

User Interface



Gambar 3a. Tampilan Halaman *Generate Schedule*

Gambar 3a merupakan tampilan halaman *generate schedule*, dimana terdapat beberapa keterangan parameter serta indicator dalam penjadwalan. Parameter yang ditampilkan antara lain total hari yang di-*generate*, jumlah populasi, *crossover rate*, *mutation rate*, *max generation*, *iteration*, *penalty*, *fitness*, akurasi, serta waktu eksekusi. Parameter yang tertera dalam halaman *generate schedule* tersebut bermanfaat bagi team leader sebagai informasi performa sistem dalam proses penjadwalan *shift* kerja.

Pada halaman *generate schedule*, juga terdapat indicator yang memberikan keterangan pada hasil penjadwalan, sehingga memudahkan *team leader* selaku pengguna dalam menggunakan system. Indikator yang ditampilkan antara lain pembagian jam kerja *shift* yakni P (Pagi), S (Siang), M (Malam), dan L (Libur). Selain itu, juga terdapat indicator warna yang menandai *constraint*, dimana warna kuning artinya dikenai *constraint* pengajuan hari libur, warna hijau artinya dikenai *constraint* perempuan dengan *shift* malam hari, dan warna biru artinya dikenai *constraint shift pagi / siang setelah shift malam*.

No	Nama	Gender	01	02	03	04	05	06	07	08	09	10	11	12	13	1.
1	RYCCO PERMANA SAPUTRA	Male	L	P	S	L	M	L	S	M	M	L	L	P	P	S
2	MIFTACHUL CHUSNUL MANAF	Male	L	P	L	M	L	M	M	S	M	L	P	S	M	L
3	M NUR MUBAROK	Male	M	L	P	M	M	L	L	P	P	M	L	S	L	P
4	WISNU HERNINDYA NOVIANTORO	Male	M	L	P	S	L	S	P	M	L	M	S	M	M	L
5	FATKHUROKHMAN	Male	S	S	P	M	L	P	M	L	S	P	L	P	P	P
6	ANDRIANTO FIRMANSYAH	Male	P	M	S	S	M	L	L	P	S	M	M	L	S	M
7	MUCHAMAD ARIS SETIAWAN	Male	S	M	M	L	P	S	P	L	P	S	P	L	S	M
8	MUHAMMAD NOVRIZAL IRSAN	Male	L	P	M	S	L	S	M	S	L	S	S	L	S	S
9	MOCHAMMAD JEFRI	Male	P	S	S	L	S	M	P	L	S	P	M	M	L	S
10	MOCHAMAD HANIFAN	Male	M	L	L	P	S	P	S	L	P	S	S	M	M	L
11	HUSNI DWI DHUHAWAN AL MUBAROK	Male	P	S	M	L	S	M	S	M	L	P	L	S	L	P
12	YUDIVAN ILHAM SANTOSO	Male	S	M	L	P	P	P	L	S	M	L	M	L	P	M
PRESENT			9	9	9	8	8	8	9	9	8	9	9	8	8	9
PERCENT (%)			75%	75%	75%	67%	67%	75%	75%	67%	75%	75%	67%	67%	75%	71%

Gambar 3b. Tampilan Halaman *Generate Schedule*

Gambar 3b merupakan tampilan halaman *generate schedule* setelah dihasilkan jadwal *shift* hasil proses penjadwalan menggunakan algoritma genetika. Dapat dilihat pada gambar tersebut, terdapat nama pegawai, gender, tanggal, serta *shift* pegawai yang telah dijadwalkan. Pada

beberapa kolom *shift* terdapat warna yang merupakan *constraint* pada hasil penjadwalan tersebut. Untuk mengatasi adanya *constraint*, pengguna dapat melakukan edit jadwal, setelah jadwal disimpan.

4.3 Pengujian Algoritma Genetika

Tabel 2. Tabel Pengujian Algoritma Genetika

No	Population	Mutation Rate	Crossover Rate	Max Generation	Fitness	Accuration	Time Execution
1	100	0,5	0,5	100	245	87,81%	3s
2	100	0,3	0,5	100	248	88,89%	3s
3	100	0,1	0,5	100	251	89,96%	2s
4	100	0,5	0,3	100	252	90,32%	3s
5	100	0,5	0,1	100	251	89,96%	3s
6	100	0,5	0,3	50	249	89,25%	2s
7	100	0,5	0,3	200	246	88,17%	6s
8	200	0,5	0,3	100	247	88,53%	8s

Dari hasil pengujian algoritma genetika di atas, dicapai tingkat akurasi tertinggi pada angka 90,32%, dalam waktu eksekusi 3 detik, dengan jumlah populasi 100, *mutation rate* 0,5, *crossover rate* 0,3, dan *max generation* 100. Selain itu, dari hasil pengujian dapat kita ketahui bahwa, jumlah populasi dan *max generation* berpengaruh terhadap lamanya waktu eksekusi.

5. Kesimpulan dan Saran

Penelitian ini melakukan penerapan algoritma genetika pada sistem penjadwalan *shift* kerja perusahaan. Hasil penelitian ini menunjukkan bahwa algoritma genetika terbukti mampu bekerja secara optimal pada pengembangan system penjadwalan *shift* kerja perusahaan, baik dari segi *time execution* maupun hasil penjadwalan *shift*. Pengembangan system penjadwalan *shift* kerja perusahaan mampu memecahkan permasalahan yang terdapat dalam kehidupan nyata, sehingga fleksibilitas karyawan dalam mengatur hari libur mereka semakin meningkat. Hal ini mampu meningkatkan kesejahteraan dan produktivitas karyawan. Meskipun dikatakan optimal, tingkat kesesuaian antara hasil penjadwalan dengan harapan pengguna belum bisa mencapai angka 100%. Penentuan parameter seperti populasi, *crossover rate*, *mutation rate*, dan *max generation* berpengaruh terhadap hasil penjadwalan dan waktu eksekusi (*time execution*).

Penelitian ini masih memiliki beberapa kekurangan. Hasil dari *schedule request* masih banyak dikenai *constraint* dan tidak sepenuhnya terealisasi setelah dilakukan *generate schedule*. Sehingga sebaiknya dilakukan pengembangan dan modifikasi lagi sehingga *schedule request* yang sudah terverifikasi dapat langsung terealisasi tanpa melakukan proses edit untuk memperbaiki *schedule request*. Agar hasil penjadwalan lebih optimal, sebaiknya penentuan parameter untuk proses seleksi, mutasi, dan *crossover* dibuat lebih bervariasi lagi. Agar waktu eksekusi program tidak memakan waktu yang lama, maka perlu didukung dengan hardware yang mumpuni.

Hasil penelitian ini dapat diterapkan secara praktis pada proses bisnis penjadwalan *shift* kerja pada perusahaan yang menerapkan jam kerja 24/7

Daftar Pustaka

- [1] Z. Fatkhurrohman and Y. Ardian, "Sistem Informasi Penjadwalan Shift Kerja Karyawan Menggunakan Metode Algoritma Genetika," *Semin. Nas. FST*, vol. 1, pp. 475–483, 2018.
- [2] S. A. Darmawan, "Optimasi Penjadwalan Mesin Dan Shift Karyawan Menggunakan Algoritma Genetika (Studi Kasus Pada Pt . Petro Jordan Abadi)," 2018.
- [3] W. F. Mahmudy, "Algoritma Evolusi," *Progr. Teknol. Inf. dan Ilmu Komputer, Univ. Brawijaya, Malang*, no. September, pp. 1–101, 2013.
- [4] W. C. Ginting, "Implementasi Algoritma Genetika Dalam Penjadwalan Shift Kerja di Call Center Telkomsel Medan," vol. 1, no. 3, pp. 82–91, 2017.
- [5] R. R. Ilmi, W. F. Mahmudy, and D. E. Ratnawati, "Optimasi Penjadwalan Perawat Menggunakan Algoritma Genetika," *Univ. Brawijaya*, vol. 5, no. 13, pp. 1–8, 2015, [Online]. Available: wayanfm@ub.ac.id
- [6] H. Pranata, "SISTEM PENJADWALAN SHIFT KERJA ANGGOTA KEPOLISIAN MENGGUNAKAN ALGORITMA GENETIKA (Studi Kasus: Kepolisian Sektor Magelang Tengah)," 2019, [Online]. Available: <http://eprints.uty.ac.id/4082/>
- [7] Suseno and E. Dhuha, "Penjadwalan Tenaga Kerja untuk Tiga Shift Kerja dengan Pengembangan Metode Algoritma," *Semin. Nas. Tek. Ind.*, vol. 1, no. 1, pp. 13–14, 2017.
- [8] E. Suhartono, "Optimasi Penjadwalan Mata Kuliah dengan Algoritma Genetika (Studi Kasus di AMIK JTC Semarang)," *Infokam*, vol. 2, pp. 132–146, 2015.
- [9] Y. Arkeman, K. B. Serminar, and H. Gunawan, "Algoritma Genetika Teori dan Aplikasi," *Graha Ilmu*. p. 119, 2014.
- [10] A. Rinaldi and A. A. Rismayadi, "Optimasi Penjadwalan Proyek Dengan Metode Algoritma Genetika," *eProsiding Tek. Inform.*, vol. 3, no. 1, pp. 107–120, 2022.
- [11] A. P. Pambudi, A. Waluyo, and E. V. L. N. Fatich, "Perancangan Sistem Penjadwalan Perkuliahan Berbasis Website Menggunakan Algoritma Genetika," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 8, no. 3, pp. 1133–1146, 2021, doi: 10.35957/jatisi.v8i3.1051.
- [12] F. E. Nugroho, "Perancangan Sistem Informasi Penjualan Online Studi Kasus Tokoku," *CCIT J.*, vol. 11, no. 1, pp. 102–114, 2018, doi: 10.33050/ccit.v11i1.563.
- [13] L. Setiyani and E. Tjandra, "Analisis kebutuhan fungsional aplikasi penanganan keluhan mahasiswa studi kasus:stmik rosma karawang," vol. 02, 2021.
- [14] S. Adi and D. M. Kristin, "STRUKTURISASI ENTITY RELATIONSHIP DIAGRAM DAN DATA FLOW DIAGRAM BERBASIS BUSINESS EVENT-DRIVEN," vol. 5, no. 9, pp. 26–34, 2014.
- [15] B. Fitriani, T. Angraini, Y. Hadi, and G. Putra, "Pemodelan Use Case Diagram Sistem Informasi Inventaris Laboratorium Teknik Mesin," pp. 626–631, 2018.