



PENERAPAN ALGORITMA GENETIKA MUTUAL SELECTION UNTUK PENJADWALAN SHIFT KERJA KARYAWAN PADA PT. MIFA BERSAUDARA

M. Iqbal Maulana

Program Studi Teknik Informatika, Fakultas Teknik,
Universitas Islam Sumatera Utara

Email: Iqbal.muhammad199940@gmail.com

Abstrak

Penelitian ini dilaksanakan dengan menggunakan algoritma genetika mutual selection untuk menyelesaikan permasalahan penjadwalan kerja karyawan di PT Mifa Bersaudara. Perusahaan ini merupakan salah satu entitas anak PT Media Djaya Bersama yang didirikan pada tahun 2002 dan bergerak dalam bidang pertambangan baru bara di wilayah Aceh Barat, Aceh. Adapun tujuan dari penelitian ini adalah untuk mendapatkan penjadwalan kerja karyawan yang terintegrasi dan merata. Dimana tidak adalagi kesenjangan dalam jumlah libur, jarak libur antar karyawan, dan urutan shift malam dan pagi yang bertabrakan. Algoritma genetika merepresentasikan kandidat solusi penjadwalan ke dalam kromosom-kromosom secara acak, lalu dievaluasi menggunakan fungsi *fitness* dan dilakukan seleksi menggunakan roulette wheel. Kemudian dilakukan *crossover one-cut-point* dan mutasi. Berdasarkan hasil pengujian sistem informasi penjadwalan dengan menggunakan metode algoritma genetika yang dibuat, didapatlan hasil penjadwalan yang maksimal dimana tidak ada lagi jadwal yang bertabrakan dan jumlah libur serta jarak libur semua karyawan merata.

Kata kunci: Penjadwalan Karyawan, Algoritma Genetika, Mutual Selection

Abstract

This research was conducted using a genetic alhorithm with mutual selection to solve the employee work scheduling problem at PT. MIFA Bersaudara. The company is a subsidiary of PT. Media Djaya Bersama established in 2002 and engaged in coal mining operations in West Aceh, Aceh. The aim of this study is to produce an integrated and equitable spacing between rest days among employees, and overlapiing shifts between night and morning duties. The genetic algorithm represents candidate scheduling solutions as chromosomes generated randomly, which are then evaluated using a fiitness crossover and mutation are then applied. Based on testing results of the scheduling information system using the genetic algorithm method, the generated schedyle prove optimal-showing no overlapping shifts, and equal distributiom of leave days and rest intervals among all employees.

Keywords: Employee Scheduling, Genetic Algorithm, Mutual Selection

Article History

Received: April 2025

Reviewed: April 2025

Published: April 2025

Plagiarism Checker No 631

Prefix DOI : Prefix DOI :

10.8734/Kohesi.v1i2.365

Copyright : Author

Publish by : Kohesi



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)



PENDAHULUAN

Latar Belakang

PT Mifa Bersaudara (MIFA) merupakan perusahaan pertambangan batubara yang beroperasi di wilayah Aceh Barat. Dalam operasionalnya yang berlangsung setiap hari, perusahaan menghadapi kendala dalam hal penjadwalan shift kerja, seperti ketimpangan distribusi libur, ketidakaturan rotasi shift, dan penggunaan metode penjadwalan yang masih konvensional. Meskipun telah menggunakan perangkat lunak seperti Microsoft Excel, metode tersebut belum optimal dalam menyelesaikan kompleksitas masalah penjadwalan.

Algoritma Genetika (GA) merupakan metode berbasis seleksi alami yang dikenal efektif dalam menyelesaikan persoalan optimasi dengan banyak kemungkinan solusi. *GA mutual selection* memungkinkan evolusi solusi melalui iterasi dan seleksi individu terbaik berdasarkan fungsi kebugaran (*fitness function*). Oleh karena itu, pendekatan ini relevan untuk menyusun jadwal shift kerja yang optimal dengan mempertimbangkan batasan-batasan tertentu.

Penelitian serupa sebelumnya telah dilakukan oleh Ginting dan Clarisa (2017) dalam konteks penjadwalan shift call center dan oleh Puspaningrum et al. (2013) dalam penjadwalan mata kuliah di perguruan tinggi. Namun, penelitian mengenai penerapan *GA mutual selection* pada kasus penjadwalan shift kerja karyawan dengan perubahan dari tiga menjadi dua shift di sektor pertambangan masih terbatas. Maka dari itu, penelitian ini berfokus pada penerapan GA untuk menyusun jadwal shift yang optimal pada divisi ICT PT Mifa Bersaudara.

Rumusan Masalah

Penelitian ini dirancang untuk menjawab dua pertanyaan utama:

1. Bagaimana algoritma genetika *mutual selection* dapat digunakan untuk membagi penjadwalan shift kerja karyawan dari tiga shift menjadi dua shift?
2. Bagaimana efektivitas sistem penjadwalan terintegrasi menggunakan algoritma genetika *mutual selection*?

Batasan Masalah

Agar penelitian lebih terarah, ruang lingkup dibatasi sebagai berikut:

- Fokus pada karyawan wanita di divisi ICT PT Mifa Bersaudara dengan sistem kerja tiga shift.
- Data diperoleh langsung dari PT Mifa Bersaudara berupa data karyawan, shift kerja, dan hasil wawancara.
- Pengembangan sistem dilakukan menggunakan bahasa pemrograman PHP dan basis data MySQL.

Tujuan Penelitian

Tujuan dari penelitian ini adalah:

1. Mengoptimalkan perencanaan jadwal kerja agar distribusi jam kerja lebih efisien dan seimbang tanpa konflik penjadwalan.
2. Menjamin distribusi jatah libur yang adil dan proporsional sesuai kebutuhan operasional perusahaan.

Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

1. Meningkatkan produktivitas perusahaan melalui sistem penjadwalan kerja yang terintegrasi dan otomatis.
2. Meminimalkan kesalahan dalam penyusunan jadwal dibandingkan metode konvensional.



KAJIAN PUSTAKA

Algoritma Genetika (AG) Mutual Selection

Algoritma Genetika Mutual Selection adalah metode optimasi berbasis prinsip-prinsip genetika dan seleksi alam. Algoritma ini banyak diterapkan dalam berbagai bidang seperti bisnis, teknik, dan ilmu pengetahuan. Prosesnya dimulai dengan sekelompok solusi yang disebut populasi. Solusi-solusi ini dipilih untuk menghasilkan generasi baru yang diharapkan memiliki kualitas lebih baik dibandingkan generasi sebelumnya. Seleksi didasarkan pada nilai kecocokan (*fitness*) dari masing-masing solusi (Haupt et al., 2004).

Algoritma genetika pertama kali diperkenalkan oleh John Holland pada tahun 1975 di Universitas Michigan, Amerika Serikat. Konsep ini kemudian dikembangkan lebih lanjut oleh David Goldberg, yang memperkenalkan teori skema. Holland memperkenalkan konsep kromosom sebagai representasi alternatif solusi dalam bentuk bit string (angka 0 dan 1) yang disebut gen. Kromosom-kromosom tersebut mengalami proses *crossover* (pertukaran materi genetik) dan *mutasi* (perubahan acak pada gen) untuk membentuk solusi baru (Pramana Henriyan et al., 2019).

Kromosom

Kromosom adalah elemen dasar dalam Algoritma Genetika. Permasalahan yang hendak diselesaikan harus terlebih dahulu direpresentasikan dalam bentuk kromosom agar proses seperti seleksi, *crossover*, dan *mutasi* dapat dilakukan (Adnyana, 2018). Kumpulan kromosom membentuk individu, dan kumpulan individu membentuk populasi. Umumnya, generasi pertama dibentuk secara acak dan setiap kromosom dievaluasi untuk memastikan kelayakannya terhadap batasan yang berlaku. Kromosom yang memenuhi semua batasan disebut kromosom layak (Mustofa, 2020).

Fungsi *Fitness*

Fungsi *fitness* digunakan untuk mengukur kualitas suatu kromosom dan menentukan apakah kromosom tersebut layak dipertahankan atau dieliminasi. Fungsi ini juga menunjukkan seberapa dekat solusi terhadap tujuan optimasi. Semakin tinggi nilai *fitness*, maka solusi dianggap semakin baik.

Fungsi *fitness* yang digunakan dalam penelitian ini adalah:

$$f_{fitness}(x) = f(x) + \sum_{i=1}^m \max(0, g_i(x)) + \sum_{j=1}^p |h_j(x)|$$

Di mana:

- $f(x)$ adalah fungsi objektif yang ingin dioptimasi (misalnya, biaya atau keuntungan),
- $g_i(x)$ adalah batasan ketidaksetaraan yang harus dipenuhi, yaitu $g_i(x) \leq 0$
- $h_i(x)$ adalah batasan persamaan yang harus tepat sama dengan nol, yaitu $h_i(x)=0$
- m adalah jumlah batasan ketidaksetaraan, dan
- p adalah jumlah batasan persamaan.

Jika suatu solusi melanggar batasan, maka penalti diterapkan untuk meningkatkan nilai *fitness*-nya, sehingga solusi tidak valid akan cenderung dieliminasi (Reddy & Kumar, 2006).

Seleksi: Metode *Roulette Wheel*

Metode *Roulette Wheel* adalah salah satu teknik seleksi dalam algoritma genetika yang bersifat proporsional terhadap nilai *fitness*. Individu dengan nilai *fitness* lebih tinggi memiliki peluang lebih besar untuk dipilih, namun individu dengan nilai *fitness* rendah tetap memiliki peluang untuk terpilih (Pranata, 2019).

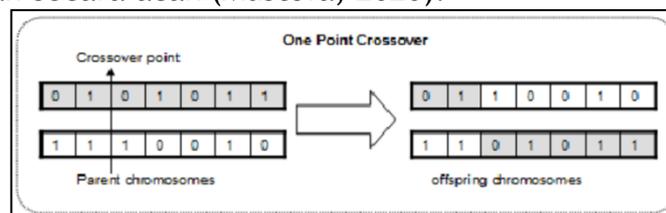
Probabilitas pemilihan individu x_i dihitung menggunakan rumus:

$$P(x_i) = \frac{f(x_i)}{F_{total}}$$

Di mana F_{total} adalah total nilai *fitness* seluruh populasi. Angka acak antara 0 hingga 1 kemudian dihasilkan, dan individu yang segmen rodanya mencakup angka tersebut akan terpilih (Taha et al., 2024).

Crossover

Crossover adalah proses pembentukan individu baru (offspring) dari kombinasi dua individu induk terpilih. Tujuannya adalah untuk memperoleh kromosom baru yang potensial menghasilkan solusi yang lebih baik. Beberapa metode *crossover* yang umum digunakan antara lain *one-point crossover* dan *two-point crossover*, tergantung pada representasi kromosom yang digunakan. Tidak semua individu mengalami proses *crossover*; pelaksanaannya ditentukan oleh laju *crossover* dan dilakukan secara acak (Mustofa, 2020).



Mutasi

Mutasi adalah proses perubahan satu atau lebih gen dalam kromosom dengan tujuan meningkatkan keragaman populasi. Mutasi bisa dilakukan secara otomatis oleh sistem atau secara manual oleh pengguna untuk memperbaiki solusi yang dihasilkan. Proses ini penting untuk menghindari terjebaknya algoritma pada solusi lokal (Puspaningrum et al., 2013).

Langkah-langkah mutasi:

1. Tentukan jumlah individu yang akan mengalami mutasi berdasarkan probabilitas mutasi (P_m).
2. Bangkitkan bilangan acak antara 0 dan 1. Jika nilai acak lebih kecil dari P_m , maka gen dimutasi.
3. Tentukan posisi gen yang dimutasi dengan bilangan acak antara 1 hingga panjang gen.
4. Gantilah gen dengan nilai acak baru antara 0 sampai batas tertentu.
5. Evaluasi kembali kromosom hasil mutasi terhadap batasan yang berlaku.
6. Ulangi proses hingga semua individu selesai dimutasi (Taha et al., 2024).

Penjadwalan

Penjadwalan adalah proses pengaturan waktu untuk pelaksanaan aktivitas atau penggunaan sumber daya secara efisien. Menurut Kamus Besar Bahasa Indonesia, jadwal adalah pembagian waktu berdasarkan rencana kerja. Dalam konteks manajemen, penjadwalan merupakan kegiatan mengatur kapan dan di mana aktivitas serta sumber daya digunakan, agar kegiatan berjalan efektif dan efisien (Ginting & Clarisa, 2017; Stevenson & Chuong, 2014). Penjadwalan membantu mengurangi beban kerja yang berlebihan dan meningkatkan kepuasan kerja.

Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah representasi grafis yang digunakan untuk merancang struktur data dalam sistem informasi. ERD memperlihatkan hubungan antar entitas serta atribut yang dimilikinya. Model ini pertama kali diperkenalkan oleh Peter Chen pada tahun 1976. Menurut Hoffer et al. (2016), ERD sangat penting dalam tahap analisis dan desain basis



data karena membantu memvisualisasikan struktur logis dari sistem dan mempermudah komunikasi antara tim pengembang dan pengguna.

No.	Gambar	Keterangan
1.		Sesuatu objek data yang ada di dalam sistem, nyata maupun abstrak dimana data tersimpan atau dimana terdapat data
2.		Hubungan alamiah yang terjadi antar entitas. Umumnya diberi nama dengan kata kerja dasar
3.		Menjelaskan apa yang sebenarnya yang dimaksud entitas atau relasian dan mewakili atribut dari masing-masing entitas.
4.		Penghubung antar entitas

Class Diagram

Class Diagram adalah salah satu komponen dalam pemodelan sistem berorientasi objek. Diagram ini menampilkan hubungan antar kelas dan detail dari setiap kelas, seperti atribut dan metode. Kelas biasanya digambarkan dalam bentuk persegi panjang yang dibagi menjadi tiga bagian: nama kelas, atribut, dan metode. *Class Diagram* identik dengan entitas dalam database dan menjadi bagian inti dalam proses perancangan sistem berorientasi objek (Ramdany, 2024).

No.	Gambar	Nama	Keterangan
1		Generalization	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>)
2		Nary Association	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
3		Class	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama
4		Collaboration	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		Realization	Operasi yang benar-benar dilakukan oleh suatu objek
6		Dependency	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri yang mempengaruhi elemen yang tergantung padanya
7		Association	Apa yang menghubungkan antara objek satu dengan objek lainnya

Use Case Diagram (UCD)

Use Case Diagram adalah alat untuk merepresentasikan interaksi antara sistem dengan aktor eksternal, seperti pengguna atau sistem lain. *Use Case* berfungsi untuk mendokumentasikan fungsi-fungsi sistem dari perspektif pengguna, baik untuk sistem yang sedang berjalan maupun sistem yang sedang dirancang. Setiap *Use Case* menggambarkan satu fungsi spesifik yang diakses oleh aktor (Santoso & Migunani, 2021)

Simbol	Keterangan
	Aktor: Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case
	Use Case: Abstraksi dan interaksi antara sistem dan aktor
	Association: Abstraksi dan interaksi antara sistem dengan use case
	Generalisasi: Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya
	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi

METODE

Tempat dan Waktu Penelitian

Penelitian ini dilaksanakan di **PT. Mifa Bersaudara**, yang berlokasi di Peunaga Cut Ujong, Kecamatan Meureubo, Kabupaten Aceh Barat. Waktu pelaksanaan penelitian berlangsung selama 4 bulan, terhitung mulai dari tanggal **1 April hingga 30 Juli 2024**.

Alat dan Data Penelitian

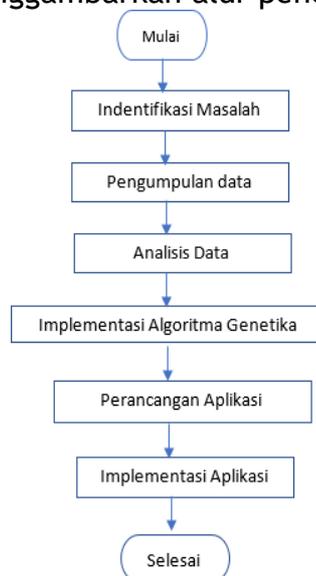
Penelitian ini menggunakan satu unit **laptop** dengan spesifikasi sebagai berikut: prosesor **Intel® Celeron® CPU N3350 @1.10GHz**, RAM **4 GB**, HDD **1 TB**, dan sistem operasi **Windows**. Perangkat lunak yang digunakan antara lain:

- **XAMPP**
- **PHP**
- **MySQL**

Data yang digunakan adalah **roster kerja** dalam format **.xlsx**, yang diperoleh dari **PT. Mifa Bersaudara** melalui proses wawancara dengan pihak **HRD**.

Skema Kerja

Berikut ini adalah bagan yang menggambarkan alur penelitian yang digunakan.



a) Identifikasi Masalah

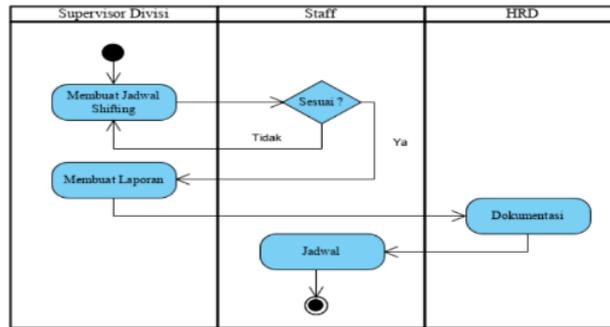
Identifikasi masalah dilakukan melalui wawancara dengan pihak manajemen perusahaan untuk memperoleh pemahaman mendalam terhadap permasalahan yang sedang dihadapi, khususnya dalam hal penjadwalan kerja. Setelah wawancara dilakukan, peneliti melanjutkan dengan studi literatur dari jurnal, artikel, dan buku ilmiah yang relevan. Sumber pustaka diperoleh melalui pencarian daring dan perpustakaan digital seperti **UISU** dan **iPusnas**. Proses ini bertujuan untuk memperkaya dasar teori dan mendukung kerangka analisis dalam penelitian.

b) Pengumpulan Data

Data diperoleh melalui wawancara langsung dengan pihak HRD PT. Mifa Bersaudara. Beberapa aspek yang dikaji melalui wawancara meliputi:

1. Jumlah karyawan dan sistem pergantian shift.
 2. Keadaan jadwal dan frekuensi terjadinya konflik atau tabrakan jadwal.
 3. Teknik penyusunan jadwal, apakah menggunakan sistem manual atau perangkat lunak.
- Data ini menjadi dasar untuk analisis lebih lanjut dalam menyusun solusi penjadwalan.

c) Analisis Sistem Berjalan



Analisis sistem menunjukkan bahwa penyusunan jadwal shift kerja melibatkan tiga pihak utama:

- Supervisor setiap divisi
- Staf operasional
- HRD

Supervisor bertanggung jawab atas pemenuhan kebutuhan operasional melalui jadwal, staf melaksanakan tugas sesuai jadwal, dan HRD mengatur sumber daya manusia serta memastikan kebijakan penjadwalan diterapkan dengan baik.

d) Implementasi Algoritma Genetika



Implementasi algoritma genetika dimulai dari inisialisasi populasi awal yang terbentuk secara acak. Setiap individu dinilai menggunakan fungsi *fitness*, lalu melalui proses seleksi, individu dengan nilai terbaik dipilih untuk reproduksi melalui *crossover* dan mutasi. Proses berlanjut secara iteratif hingga ditemukan solusi terbaik. Pengujian dilakukan menggunakan metode black-box, yakni menguji fungsionalitas sistem tanpa melihat detail kodenya. Uji performa dilakukan untuk mengevaluasi efektivitas parameter algoritma dalam menghasilkan jadwal kerja yang optimal.

e) Perancangan Aplikasi

Aplikasi dibangun dengan menggunakan bahasa pemrograman PHP dan basis data MySQL, serta dirancang dengan antarmuka GUI (Graphical User Interface) yang memudahkan pengguna, baik dari kalangan umum maupun HRD.

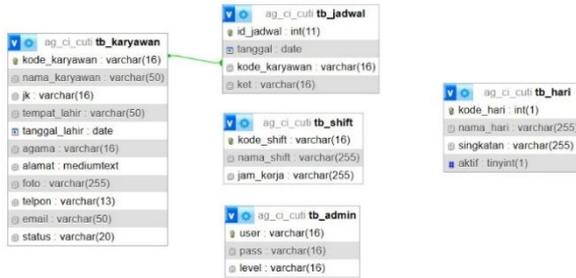
Tujuan aplikasi:

- Membantu penyusunan jadwal kerja secara efisien
- Mencegah bentrok jadwal antar karyawan
- Memudahkan HRD dalam manajemen jadwal kerja

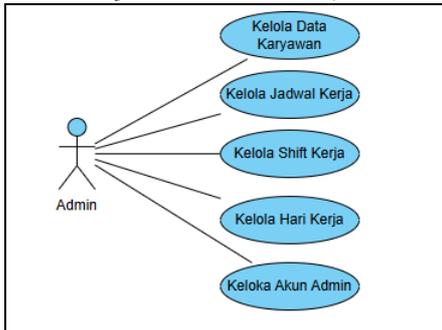


Perancangan UML dilakukan dengan tiga diagram utama:

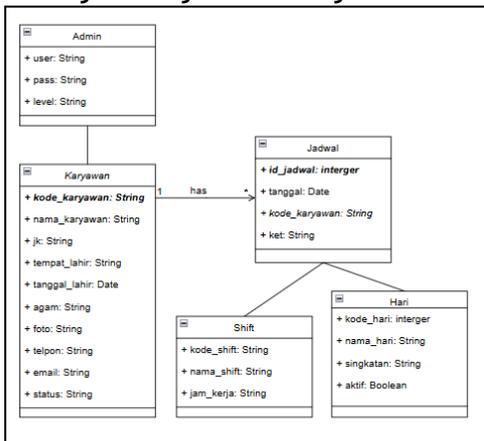
- **Entity Relationship Diagram (ERD):** Memodelkan relasi antar entitas seperti *tb_karyawan*, *tb_jadwal*, *tb_shift*, *tb_hari*, dan *tb_admin*.



- **Use Case Diagram:** Menjelaskan peran Admin (pengelola data karyawan, jadwal, shift, hari kerja, akun admin) dan Karyawan (melihat jadwal kerja).



- **Class Diagram:** Menjelaskan struktur objek seperti Karyawan, Jadwal, Shift, Hari, dan Admin. Setiap kelas memiliki atribut dan relasi antar objek yang menggambarkan sistem manajemen jadwal kerja secara terstruktur.



HASIL DAN PEMBAHASAN

Implementasi Algoritma Genetika

Implementasi algoritma genetika dimulai dengan pemetaan data penjadwalan yang diambil dari file Excel berisi informasi karyawan pada divisi Information Communication Technology (ICT). Data ini terdiri dari tiga kolom utama, yaitu *Nama Karyawan*, *Tanggal*, dan *Shift*, yang dikodekan sebagai *P* (shift pagi), *S* (shift siang), *M* (shift malam), dan *O* (libur).



mifo		PT MIFA BERSAUDARA																															
		SHIFT KERJA GENERAL AFFAIR/ICT																															
NO.	NAMA	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30		
1	Ade Pijana	S	M	M	O	P	P	S	S	M	M	O	P	P	S	S	M	M	O	P	P	S	S	M	M	O	P	P	S	S	M	M	O
2	Agung Dwi Satrio	S	P	O	O	P	M	S	P	O	O	P	M	S	P	O	O	P	M	S	P	O	O	P	M	S	P	O	O	P	M	S	P
3	Dwi Sulistyio Nugroho	M	O	P	P	S	M	S	M	O	P	P	S	M	S	M	O	P	P	S	M	S	M	O	P	P	S	M	S	M	O	P	
4	Lukha Fajar	O	O	P	P	S	M	S	O	O	P	P	S	M	S	O	O	P	P	S	M	S	O	O	P	P	S	M	S	O	O	P	
5	Nini Ratihani	O	S	S	O	O	S	S	O	O	S	S	O	O	S	S	O	O	S	S	O	O	S	S	O	O	S	S	O	O	S	S	O
6	Rinal Putra	P	O	O	S	S	O	O	P	O	O	S	S	O	O	P	O	O	S	S	O	O	P	O	O	S	S	O	O	P	O	O	S
7	Santi Lestiana	M	S	O	M	S	S	O	M	S	O	M	S	S	O	M	S	O	M	S	S	O	M	S	O	M	S	S	O	M	S	O	M

Ket:
 P : Pagi
 S : Siang
 M : Malam
 O : Libur

Yang Menjabat,
 PT. MIFA BERSAUDARA
 Adh Rifa'ul
 GM Operation

1. Representasi Kromosom

Satu kromosom dalam algoritma genetika mewakili satu solusi penjadwalan lengkap untuk seorang karyawan selama periode tertentu, misalnya satu minggu. Jumlah kromosom awal ditentukan berdasarkan input pengguna—misalnya, 5 kromosom. Setiap kromosom terdiri atas sejumlah gen yang dipilih secara acak dari pilihan shift yang tersedia. Dalam konteks ini, penjadwalan mencakup hari Senin hingga Minggu, dengan waktu kerja selama 24 jam (dimulai pukul 08:00 dan berakhir pukul 08:00 keesokan harinya). Kromosom disimpan dalam array, misalnya Kromosom = [0, 1, 2, off, 0, 1, 0], di mana 0 mewakili shift pagi, 1 shift siang, 2 shift malam, dan off menunjukkan libur. Proses pembangkitan kromosom ini diulang untuk seluruh populasi awal, sehingga tiap kromosom memiliki kombinasi gen yang berbeda-beda.

```

Kromosom[0]: 0,off,0,2,off,0,2,1,1,0,off,2,1,1,1,0,0,off,1,1,1,1,0,0,0,2,2,off,1,0,0
Kromosom[1]: 0,off,0,off,1,1,1,1,2,2,off,0,2,2,1,1,0,off,0,0,1,0,2,2,1,0,0,off,1,2,0
Kromosom[2]: off,1,0,0,2,off,1,0,off,2,1,0,1,0,1,0,1,off,off,0,0,1,2,0,2,2,1,0,0,0,0
Kromosom[3]: 0,0,1,2,1,1,2,1,off,1,1,2,0,1,0,0,0,off,1,0,off,1,0,off,0,2,2,off,1,0,0
Kromosom[4]: 0,off,0,2,1,1,1,0,0,0,off,2,1,1,1,off,0,0,1,0,1,0,off,2,1,2,2,off,0,0,1
    
```

2. Perhitungan Nilai Fitness

Setelah kromosom terbentuk, tahap selanjutnya adalah menghitung *fitness* untuk masing-masing individu. Nilai *fitness* digunakan untuk mengevaluasi kualitas solusi berdasarkan beberapa parameter penilaian. Sebagai contoh, untuk Kromosom[0], dilakukan perhitungan *fitness* dengan mempertimbangkan tiga parameter evaluasi yaitu, P1 = 0, yang menunjukkan bahwa jadwal shift karyawan tersebut berurutan atau tidak ada ketidaksesuaian pada urutan jadwal. P2 = 2, yang berarti ada masalah dalam jarak libur yang lebih kecil dari 4 hari atau lebih besar dari 7 hari. P3 = 0, yang berarti jumlah hari libur selain 5 hari adalah sesuai.

Maka perhitungan nilai *fitness*:

$$Fitness[0] = \frac{1}{1 + 0 + 2 + 0} = 0.333333$$

Proses ini diulang untuk setiap kromosom dalam populasi. Semakin tinggi nilai *fitness*, semakin besar kemungkinan kromosom tersebut dipilih dalam proses seleksi.

<i>Fitness</i> [0]	1/(1+0+2+0)	0.33333333
<i>Fitness</i> [1]	1/(1+1+2+0)	0.25
<i>Fitness</i> [2]	1/(1+1+4+0)	0.16666667
<i>Fitness</i> [3]	1/(1+1+4+0)	0.16666667
<i>Fitness</i> [4]	1/(1+0+1+0)	0.5
Total Fitness		0.14666667



3. Seleksi

Proses seleksi bertujuan memilih kromosom dengan nilai *fitness* terbaik sebagai induk (parent). Semakin tinggi *fitness*, semakin besar peluang untuk dipilih. Probabilitas seleksi dihitung dari pembagian nilai *fitness* masing-masing individu dengan total nilai *fitness* populasi. Setelah menghitung *probabilitas kumulatif* (PK), proses seleksi dilakukan menggunakan metode *Roulette Wheel*. Dengan membangkitkan bilangan acak R dalam rentang 0-1, pemilihan induk dilakukan berdasarkan kriteria:

- Jika $R[n] < PK[1]$, maka kromosom pertama terpilih
- Jika $PK[n-1] < R[n] < PK[n]$, maka kromosom ke-n terpilih

Individu yang terpilih kemudian digunakan untuk membentuk populasi baru yang diharapkan lebih baik.

```
Kromosom[0]: 0,off,0,2,off,0,2,1,1,0,off,2,1,1,1,0,0,off,1,1,1,1,0,0,0,2,2,off,1,0,0
Kromosom[1]: 0,off,0,2,1,1,1,0,0,0,off,2,1,1,1,off,0,0,1,0,1,0,off,2,1,2,2,off,0,0,1
Kromosom[2]: 0,off,0,2,1,1,1,0,0,0,off,2,1,1,1,off,0,0,1,0,1,0,off,2,1,2,2,off,0,0,1
Kromosom[3]: off,1,0,0,2,off,1,0,off,2,1,0,1,0,1,0,1,off,off,0,0,1,2,0,2,2,1,0,0,0,0
Kromosom[4]: 0,off,0,2,1,1,1,0,0,0,off,2,1,1,1,off,0,0,1,0,1,0,off,2,1,2,2,off,0,0,1
```

4. Crossover

Pada tahap *crossover*, kromosom induk dipertukarkan untuk menghasilkan keturunan dengan kombinasi gen dari kedua induknya menggunakan metode *one-point crossover*. Titik pemotongan dipilih secara acak, dan gen setelah titik tersebut dipertukarkan antara induk. Proses ini dipengaruhi oleh *crossover rate* (CR), di mana bilangan acak digunakan untuk memilih kromosom yang akan disilangkan. Jika nilai acak lebih kecil atau sama dengan CR, *crossover* dilakukan. Titik potong ditentukan secara acak berdasarkan jumlah gen, dan hasilnya adalah gabungan gen dari kedua kromosom, dengan ketentuan gen off tetap diambil dari kromosom pertama. Setelah *crossover*, kromosom yang terlibat akan mendapatkan gen baru dari induknya, sementara yang tidak terlibat tetap mempertahankan gen aslinya.

```
Kromosom[0]: 0,off,0,2,off,0,2,1,1,0,off,2,1,1,1,0,0,off,1,1,1,1,0,0,0,2,2,off,1,0,0
Kromosom[1]: 0,off,0,2,1,1,1,0,0,0,off,2,1,1,1,off,0,0,1,0,1,0,off,2,1,2,2,off,0,0,1
Kromosom[2]: 0,off,0,2,1,1,1,0,0,0,off,2,1,1,1,off,0,0,1,0,1,0,off,2,1,2,2,off,0,0,1
Kromosom[3]: off,1,0,0,2,off,1,0,off,2,1,0,1,0,1,0,1,off,off,0,0,1,2,0,2,2,1,0,0,0,0
Kromosom[4]: 0,off,0,2,1,1,1,0,0,0,off,2,1,1,1,off,0,0,1,0,1,0,off,2,1,2,2,off,0,0,1
```

5. Mutasi

Mutasi dalam kasus ini adalah penggantian gen pada kromosom yang sama, tanpa mengubah jumlah gen yang "off." Jumlah gen yang dimutasi ditentukan oleh *Mutation Rate* (MR). Sebagai contoh, dengan MR 25%, jika total gen adalah 155, maka $25\% \times 155 = 39$ gen akan dimutasi. Proses mutasi berhenti jika jumlah mutasi mencapai nilai yang ditentukan atau jika nilai *fitness* = 1 sudah tercapai sebelum jumlah mutasi maksimal tercapai.



```
Kromosom[0]: 0,off,1,2,off,0,2,1,1,0,off,2,1,0,1,0,0,off,1,1,1,1,0,0,0,2,2,off,1,0,0
Kromosom[1]: 0,off,0,2,1,1,1,0,off,0,0,2,1,1,1,off,0,0,1,0,1,0,off,2,1,2,2,off,0,0,1
Kromosom[2]: 0,off,0,2,1,1,1,0,0,0,off,2,1,2,1,off,0,0,1,0,1,0,off,2,1,1,2,off,0,0,1
Kromosom[3]: off,1,0,0,2,off,1,0,off,0,1,0,1,0,1,0,1,off,off,0,0,1,2,0,2,2,1,0,2,0,0
Kromosom[4]: 0,off,0,2,1,1,1,0,0,1,off,2,1,1,1,off,0,0,1,0,0,0,off,2,1,2,2,off,0,0,1
```

Penerapan Aplikasi

1. Tampilan Karyawan

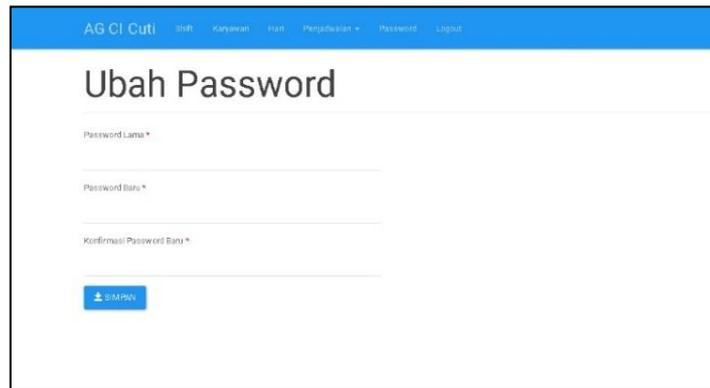
- Halaman Login akan muncul pertama kali. Pengguna wajib memasukkan username dan password yang valid. Setelah berhasil login, pengguna diarahkan ke Halaman Beranda.



- Halaman Jadwal: Menampilkan jadwal shift masing-masing karyawan, memudahkan pengelolaan dan pengawasan jadwal kerja.

No	Karyawan	Tanggal	Kot
1	Dwi Sulistyani	2025-10-01	L
2	Dwi Sulistyani	2025-10-02	L
3	Dwi Sulistyani	2025-10-03	PH
4	Dwi Sulistyani	2025-10-04	PH
5	Dwi Sulistyani	2025-10-05	PH
6	Dwi Sulistyani	2025-10-06	PH
7	Dwi Sulistyani	2025-10-07	PH
8	Dwi Sulistyani	2025-10-08	L
9	Dwi Sulistyani	2025-10-09	L
10	Dwi Sulistyani	2025-10-10	C
11	Dwi Sulistyani	2025-10-11	PH
12	Dwi Sulistyani	2025-10-12	PH
13	Dwi Sulistyani	2025-10-13	PH
14	Dwi Sulistyani	2025-10-14	PH
15	Dwi Sulistyani	2025-10-15	L
16	Dwi Sulistyani	2025-10-16	L
17	Dwi Sulistyani	2025-10-17	PH
18	Dwi Sulistyani	2025-10-18	PH
19	Dwi Sulistyani	2025-10-19	PH
20	Dwi Sulistyani	2025-10-20	C
21	Dwi Sulistyani	2025-10-21	PH
22	Dwi Sulistyani	2025-10-22	L
23	Dwi Sulistyani	2025-10-23	L
24	Dwi Sulistyani	2025-10-24	PH
25	Dwi Sulistyani	2025-10-25	PH

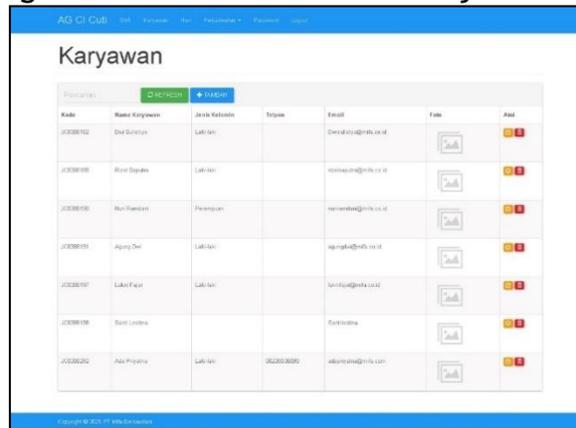
- **Halaman Ubah Password:** Memberikan fasilitas keamanan agar karyawan dapat mengganti kata sandi mereka sendiri.



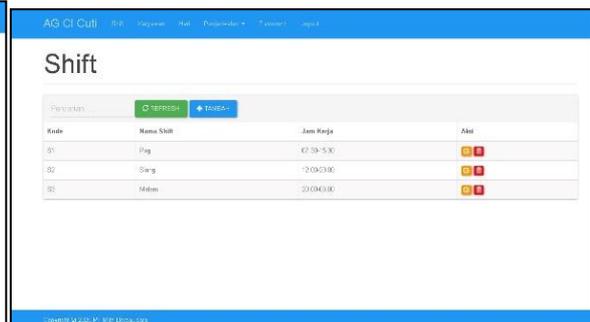
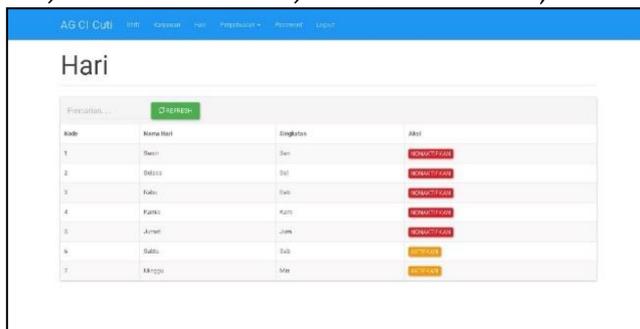
2. Tampilan Admin

Admin memiliki hak akses yang lebih luas. Setelah login, admin diarahkan ke dashboard utama dan dapat melakukan hal-hal berikut:

- **Halaman Data Karyawan:** Menyimpan, mengedit, dan menambah data karyawan. Kolom aksi digunakan untuk mengaktifkan atau menonaktifkan jadwal kerja karyawan.



- **Halaman Hari dan Shift:** Admin dapat memilih hari dan menentukan shift kerja (S1: 07.30-15.30, S2: 12.00-20.00, S3: 20.00-06.00).





- **Halaman Generate Jadwal:** Digunakan untuk menghasilkan jadwal akhir dalam bentuk file PDF yang dapat dibagikan atau disimpan untuk keperluan dokumentasi.

Jadwal Shift dan Cuti Bulan Februari 2025

No	Kode	Nama	1 Sab	2 Min	3 Sen	4 Sel	5 Rab	6 Kam	7 Jum	8 Sab	9 Min	10 Sen	11 Sel	12 Rab	13 Kam	14 Jum	15 Sab	16 Min	17 Sen	18 Sel	19 Rab	20 Kam	21 Jum	22 Sab	23 Min	24 Sen	25 Sel	26 Rab	27 Kam	28 Jum	
1	JC8388182	Dwi Sulistiyo	S1	C	S2	S2	S2	S2	S2	S2	S3	S3																			
2	JC8388188	Rizal Saputra	S2	C	S3	S3	S3	S3	S3	S3	S1	S1																			
3	JC8388190	Nuri Ramdani	S3	C	S1	S1	S1	S1	S1	S1	S2	S2																			
4	JC8388191	Agung Dwi	S1	C	S2	S2	S2	S2	S2	S2	S3	S3																			
5	JC8388197	Lukni Fajar	S2	C	S3	S3	S3	S3	S3	S3	S1	S1																			
6	JC8388198	Santi Lestina	C	S3	S3	S3	S3	S3	S3	C	S1	S1	S1	S1	S1	S1	S2	S2													
7	JC8388292	Ade Priyatna	S1	C	S2	S2	S2	S2	S2	S2	S3	S3																			

Keterangan

1. L : Libur
2. C : Cuti
3. S1: Pagi (07:30-15:30)
4. S2: Siang (12:00-20:00)
5. S3: Malam (20:00-06:00)

KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa:

1. Algoritma genetika efektif menyelesaikan masalah penjadwalan kompleks secara efisien, menghasilkan jadwal yang adil dan optimal dengan mempertimbangkan preferensi karyawan serta kebutuhan operasional.
2. Algoritma ini mampu meminimalkan konflik shift dan memastikan distribusi kerja merata, sehingga pembagian tugas lebih seimbang.
3. Sistem mendukung transparansi, mempercepat pembaruan jadwal, dan mengurangi kesalahan, serta memberikan kenyamanan bagi karyawan dan kelancaran operasional perusahaan.

DAFTAR PUSTAKA

Adnyana, I. M. B. (2018). Implementasi Algoritma Genetika untuk Penjadwalan Asisten Dosen di STIKOM Bali. *Jurnal Sistem Dan Informatika*, 12(2), 166-173.

Ginting, & Clarisa, W. (2017). Implementasi Algoritma Genetika Dalam Penjadwalan Shift Kerja Di Call Center Telkomsel Medan. In *Repository Institusi USU*.

Haupt, R. L., Haupt, S. E., & Wiley, A. J. (2004). *Algorithms Second Edition*.

Jeffrey A. Hoffer, Venkataraman, R., & Venkataraman, R. (2016). Modern Database Management 12th Edition. In *Pearson* (Vol. 4, Issue 1).

Mustofa, J. (2020). *Sistem Informasi Penjadwalan Shift Kerja Karyawan Menggunakan Algoritma Genetika (Studi Kasus Jawara Digital Art Store Yogyakarta)*. 1-13.

Pramana Henriyan, A., Hananto, B., Muhammad Ardi, F., Lenggana, H., Ulum, P., & Fitriati. (2019). Implementasi Algoritma Genetika pada Sistem Informasi Penjadwalan Matakuliah. *Semnati*, 283-287.

Pranata, H. (2019). *SISTEM PENJADWALAN SHIFT KERJA ANGGOTA KEPOLISIAN MENGGUNAKAN ALGORITMA GENETIKA (Studi Kasus: Kepolisian Sektor Magelang Tengah)*.



<http://eprints.uty.ac.id/4082/>

- Puspaningrum, W. A., Djunaidy, A., & Vinarti, R. A. (2013). Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika di Jurusan Sistem Informasi ITS. *Jurnal Teknik Pomits*, 2(1), 127-131.
- Ramdany, S. (2024). Penerapan UML *Class Diagram* dalam Perancangan Sistem Informasi Perpustakaan Berbasis Web. *Journal of Industrial and Engineering System*, 5(1). <https://doi.org/10.31599/2e9afp31>
- Reddy, M. J., & Kumar, D. N. (2006). Multi-Objective Optimization using Evolutionary Algorithms. *Water Resources Management*, 20(6), 861-878.
- Santoso, J. T., & Migunani. (2021). *Sistem Berorientasi Obyek dengan UML*.
- Stevenson, W. J., & Chuong, S. C. (2014). *Manajemen Operasi Perspektif Asia (Operations Management An Asian Perspective) 2, E9*. Jakarta Salemba Empat.
- Taha, Z. Y., Abdullah, A. A., & Rashid, T. A. (2024). *Optimizing Feature Selection with Genetic Algorithms : A Review of Methods and Applications*.
- Tiwari, R., & Vidyarthi, H. (2018). Intellectual capital and corporate performance: a case of Indian banks. *Journal of Accounting in Emerging Economies*, 8(1), 84-105. <https://doi.org/10.1108/JAEE-07-2016-0067>