

**NASKAH PUBLIKASI**

**SISTEM INFORMASI PENJADWALAN *SHIFT* KERJA  
KARYAWAN MENGGUNAKAN ALGORITMA GENETIKA  
(Studi Kasus Jawa Digital Art Store Yogyakarta)**



Disusun oleh:  
JIHAN MUSTOFA  
5150411230

**PROGRAM STUDI INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN ELEKTRO  
UNIVERSITAS TEKNOLOGI YOGYAKARTA  
2020  
Naskah Publikasi**

**SISTEM INFORMASI PENJADWALAN *SHIFT* KERJA  
KARYAWAN MENGGUNAKAN ALGORITMA GENETIKA  
(Studi Kasus Jawara Digital Art Store Yogyakarta)**

Disusun oleh:  
JIHAN MUSTOFA  
5150411230

Telah disetujui oleh pembimbing

Pembimbing:

Murti Retnowo, S.Kom.,M.Cs.

Tanggal:.....

# SISTEM INFORMASI PENJADWALAN *SHIFT* KERJA KARYAWAN MENGUNAKAN ALGORITMA GENETIKA (Studi Kasus Jawa Digital Art Store Yogyakarta)

**Jihan Mustofa**

*Program Studi Informatika, Fakultas Teknologi Informasi dan Elektro  
Universitas Teknologi Yogyakarta  
Jl. Ringroad Utara Jombor Sleman Yogyakarta  
Email: [jihanmustofa90@gmail.com](mailto:jihanmustofa90@gmail.com)*

## ABSTRAK

Jawara Digital Art Store (JDA Store) adalah sebuah bisnis lokal Yogyakarta Indonesia, yang dimulai pada November 2009, JDA mengalami sebuah masalah dalam menentukan penjadwalan *shift* kerja karyawan dimana masih terdapat banyak tabrakan jadwal dalam penyusunan jadwal *shift* kerja itu sendiri, Tujuan dari Salah satu langkah yang diambil adalah dengan membangun sistem penjadwalan *shift* karyawan yang otomatis dan sesuai kriteria dan meminimalisir adanya benturan jadwal yang sering terjadi ketika penyusunan, Dimana penjadwalan *shift* kerja karyawan merupakan hal yang berpengaruh dalam produktifitas karyawan dalam mencapai sebuah target yang telah ditentukan. Maka dari itu dibutuhkan sebuah sistem penjadwalan *shift* kerja karyawan yang menggunakan Algoritma Genetika, Penggunaan Algoritma Genetika ditujukan agar hasil penjadwalan tersusun secara alami oleh sistem melalui beberapa iterasi perhitungan sehingga dapat membantu karyawan terutama bagian management dalam menyusun jadwal *shift* kerja karyawan secara efektif dan tepat. Dengan menggunakan Algoritma Genetika. Dalam perancangan ini penulis menggunakan bahasa pemrograman PHP dan menggunakan *framework CodeIgniter* dengan DBMS MySQL sebagai database sever, Sublime Text 3 sebagai alat penunjang pemrograman. Setelah dilakukan pengujian didapatkan hasil pada pengujian pertama dengan nilai awal *mutation rate* 0,2 dan batas generasi 5000 menghasilkan nilai *fitness* tertinggi 1623 dengan tingkat kesesuaian 67,12% dengan waktu eksekusi 2 menit 46 detik pada generasi 5000 keatas. Sedangkan pengujian dengan nilai *mutation rate* awal 0.4 dan batas generasi 50000 menghasilkan nilai *fitness* tertinggi 1644 dengan tingkat kesesuaian 67,99% dengan waktu eksekusi 31 menit 25 detik pada generasi 50000 keatas.

Kata Kunci: Sistem, JDA store, Penjadwalan, Algoritma Genetika, *Shift*

## 1. Pendahuluan

### 1.1 Latar Belakang

Perkembangan teknologi informasi saat ini cukup pesat dan dapat dikatakan semakin maju. Banyak pekerjaan manusia mulai digantikan oleh teknologi baik sebagian maupun menyeluruh. Adanya teknologi informasi sangat membantu manusia dalam melakukan pekerjaan yang membutuhkan perhitungan dan ketelitian yang cukup tinggi. Teknologi informasi dapat masuk ke segala bidang termasuk perusahaan. Saat ini perusahaan dituntut agar dapat menggunakan teknologi informasi untuk meningkatkan produktivitas perusahaan. Tenaga kerja merupakan sumber daya manusia yang paling dibutuhkan oleh perusahaan, khususnya manajemen waktu kerja dalam meningkatkan produktifitas karyawan,

Penjadwalan yang baik dapat menentukan produktivitas tenaga kerja dalam melaksanakan pekerjaan, karena dapat menentukan dimana tenaga kerja harus bekerja dan beristirahat atau libur sehingga performa dan kesehatan tenaga kerja tetap terjaga.

Jawara Digital Art (JDA) Store adalah sebuah bisnis lokal Yogyakarta Indonesia, yang dimulai pada November 2009 yang bergerak dibidang *E-commerce* dan memanfaatkan media *facebook* dan *instagram (facebook ads)* sebagai media promosi utama, dimana dalam *facebook ads* terdapat tim-tim yaitu tim advertising sebagai pembuat iklan, tim *packing* dan tim *Customer service* yang mana sebagai perantara order atau melayani pelanggan dengan melalui *Whatsapp*. Jawa Digital Art (JDA) Store sering terjadi

permasalahan penjadwalan *shift* kerja dimana perusahaan selalu beroperasi setiap hari. Salah satu pemanfaatan teknologi informasi yang diterapkan di Jawara Digital Art (JDA) Store adalah dalam penyusunan jadwal *shift* kerja. Meskipun sudah menggunakan bantuan teknologi informasi, tetap saja masih ada kendala dalam proses penyusunan jadwal *shift* kerja. Jawara Digital Art (JDA) Store melakukan penyusunan jadwal menggunakan bantuan aplikasi Microsoft Excel dan masih konvensional.

Penjadwalan tenaga kerja yang dilakukan oleh perusahaan selama ini dianggap kurang tepat dalam memberikan hari libur dan waktu bekerja, karena mempekerjakan tenaga kerjanya selama 7 hari kerja untuk mencapai target yang ditentukan. Penjadwalan karyawan di Jawara Digital Art (JDA) Store diklasifikasikan menjadi 3 shift yaitu shift pagi, shift sore dan shift malam dimana shift pagi dimulai dari jam 08.00-16.00 wib sedangkan shift sore mulai jam 16.00-24.00 wib dan shift malam mulai jam 24.00-08.00. Karyawan yang akan mendapat giliran shift dipilih secara manual dengan melihat apakah karyawan tersebut sudah mendapat giliran shift jadwal atau belum. Jika jumlah karyawan yang akan disusun jadwal sedikit tentu tidak akan masalah, tetapi jika jumlahnya banyak pasti akan menimbulkan kebingungan dalam menentukan shift dan rentan terhadap kesalahan penyusunan oleh petugas (human error). Penyusunan jadwal yang seharusnya dapat dilakukan sekali tetapi dalam prakteknya terkadang dapat dilakukan berkali-kali karena terdapat jadwal yang bentrok. Jika tidak segera dilakukan revisi jadwal maka akan mengakibatkan karyawan yang seharusnya beristirahat harus kembali terkena shift kerja.

Algoritma Genetika (GA) merupakan algoritma yang menggunakan seleksi alami untuk menemukan solusi optimal dan dimanfaatkan dalam optimasi berbagai permasalahan. Pada aplikasinya, GA biasanya digunakan untuk memperoleh solusi optimal seperti penjadwalan ataupun solusi pendekatan dari persoalan optimasi yang mempunyai banyak sekali solusi yang mungkin. Persoalan seperti ini biasanya tidak dapat atau sulit diselesaikan dengan metode-metode eksak. Setelah mengalami evolusi pada beberapa generasi, GA pada umumnya akan mampu memberikan solusi yang baik, dalam hal ini susunan jadwal yang baik. Salah

satu pemanfaatan GA adalah bisa digunakan dalam penjadwalan secara otomatis shift karyawan di Jawara Digital Art (JDA) Store, GA digunakan dalam proses generate otomatis jadwal shift kerja karyawan sesuai dengan constraint (batasan) yang ditentukan dalam jangka waktu tertentu. Beberapa penelitian yang memanfaatkan GA dalam penyusunan jadwal antara lain adalah penelitian (Giting, Clarisa, W., 2017) dengan judul Implementasi Algoritma Genetika Dalam Penjadwalan Shift Kerja Di Call Center Telkomsel.

Berdasarkan beberapa penelitian yang telah dibahas di atas, sudah ada penelitian yang menggunakan GA sebagai metode untuk memecahkan masalah penjadwalan. Akan tetapi dari beberapa penelitian tersebut belum ada yang meneliti untuk penjadwalan shift kerja karyawan yang mengubah menjadi 2 shift. Oleh sebab itu peneliti tertarik untuk melakukan penelitian dengan menggunakan GA untuk mendapatkan susunan jadwal shift yang optimal di Jawara Digital Art (JDA) Store. Tujuan dari penelitian ini adalah agar karyawan khususnya di Jawara Digital Art (JDA) Store menjadi lebih mudah ketika menyusun jadwal shift kerja dalam jangka waktu tertentu sesuai ketentuan atau batasan. Sistem penjadwalan menggunakan GA ini juga dapat meminimalkan terjadinya kesalahan penyusunan jadwal jika dilakukan penyusunan secara konvensional. Penyusunan jadwal shift kerja yang nantinya juga berdampak pada produktivitas perusahaan yang lebih baik.

## 2. Landasan Teori

### 2.1 Pengertian Algoritma Genetika

Menurut Haupt, R. L. dan Haupt, S. E., (2004) Algoritma genetika adalah suatu proses optimasi yang dikembangkan berdasarkan prinsip genetika dan proses seleksi alamiah. Algoritma Genetika banyak dipakai pada aplikasi bisnis, teknik maupun pada bidang keilmuan lainnya. Algoritma ini dimulai dengan kumpulan solusi yang disebut dengan populasi. Solusi-solusi dari sebuah populasi diambil dan digunakan untuk membentuk solusi yang baru. Hal ini dimotivasi dengan harapan bahwa populasi yang dibentuk tersebut akan lebih baik daripada yang lama. Solusi-solusi yang dipilih untuk membentuk solusi-solusi yang baru sesuai dengan fitness mereka masing-masing.

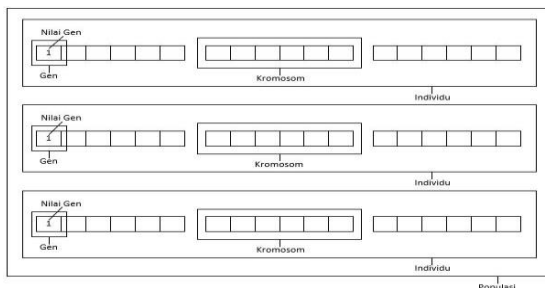
Pada dasarnya, Algoritma Genetika merupakan metode pencarian yang didasarkan pada proses evolusi alamiah Turban, E., (1995), yaitu pada gen-gen dalam kromosomnya. Individu-individu melakukan proses reproduksi untuk melahirkan keturunan. Sifat keturunan dibentuk dari kombinasi sifat kedua induknya atau mewarisi sifat-sifat induknya.

## 2.2 Pengertian Populasi Algoritma Genetika

Populasi dalam algoritma genetika adalah kumpulan kromosom. Kromosom sendiri terbentuk dari banyak gen yang nilainya dapat berupa bilangan numerik, biner, simbol, maupun karakter, tergantung permasalahan yang ingin diselesaikan. Dalam menyusun suatu algoritma genetika menjadi program, maka diperlukan beberapa tahapan proses, yaitu proses pembuatan generasi awal, proses genetika, proses seleksi, dan pengulangan proses. Algoritma genetika dibagi atas beberapa bagian diantaranya, yaitu: gen, kromosom / individu, populasi, fitness, seleksi, crossover, mutasi, generasi, offspring.

## 2.3 Representasi Kromosom

Pada implementasi GA, suatu hal yang pertama kali dipertimbangkan adalah representasi kandidat solusi (kromosom) yang sesuai untuk persoalan yang akan diselesaikan. Representasi yang baik haruslah mampu merepresentasikan semua parameter dan solusi yang mungkin untuk persoalan yang akan diselesaikan. Kumpulan beberapa kromosom menjadi sebuah individu atau lebih luas menjadi satu populasi. Penggambaran satu populasi dapat dilihat pada Gambar 1 .



Gambar 1 Susunan dari Populasi

## 2.4 Pembentukan Generasi Awal

terbentuknya populasi awal secara acak yang terdiri dari individu-individu dengan sifat yang tergantung

Pada umumnya generasi awal pada proses GA berisikan kromosom yang dibangkitkan acak (random). Untuk menghindari diperolehnya solusi yang tidak layak, kromosom generasi awal harus diuji kelayakannya untuk semua batasan (*constraint*). Kromosom yang tidak memiliki salah satu batasan dikatakan sebagai kromosom layak.

## 2.5 Evaluasi Kromosom

Ada banyak metode yang digunakan untuk mengevaluasi suatu kromosom. Namun metode yang paling sering digunakan adalah dengan menghitung nilai fungsi tujuan sebagai *fitness value*. Meskipun pada generasi awal telah dibangkitkan kromosom yang layak, proses crossover dan mutasi sering kali menghasilkan keturunan (*offspring*) yang tidak layak. Oleh karena itu, setiap *offspring* yang dihasilkan *crossover* dan mutasi pada setiap generasi juga harus diuji kelayakannya terhadap fungsi kendala yang ada. Salah satu teknik yang populer untuk mengatasi kromosom yang tidak layak adalah dengan memberikan penalti. Penambahan nilai penalti ini bertujuan agar kromosom tersebut tidak terseleksi ke generasi berikutnya.

## 2.6 Operasional Algoritma Genetika

Struktur dasar dari GA untuk penjadwalan sebagai berikut:

### a. Membangkitkan Generasi Awal

Pada umumnya generasi awal pada proses GA berisikan kromosom yang dibangkitkan acak (random). Untuk menghindari diperolehnya solusi yang tidak layak, kromosom generasi awal harus diuji kelayakannya untuk semua batasan (*constraint*). Kromosom yang tidak memiliki salah satu batasan dikatakan sebagai kromosom layak.

### b. Fungsi *Fitness*

Fungsi *fitness* digunakan untuk proses evaluasi kromosom GA untuk memperoleh kromosom yang diinginkan. Fungsi ini menghitung kualitas dari kromosom untuk mengetahui seberapa baik kromosom yang

dihasilkan. Fungsi fitness pada GA sebagai berikut:

$$f = \text{kromosom} - (1 + 2 + 3)$$

Keterangan:

*Fitness* kromosom = 2418 (total *fitness* jika tidak penalti dalam 1 kromosom)

$$P = \text{pelanggaran} / \text{penalti.}$$

Berdasarkan persamaan di atas nilai *fitness* ditentukan oleh bobot pelanggaran yang biasanya disebut sebagai penalti. Penalti tersebut menunjukkan jumlah pelanggaran kendala pada suatu kromosom. Semakin tinggi nilai *fitness* akan semakin besar kemungkinan kromosom tersebut terpilih ke generasi berikutnya. Nilai penalti akan berbanding terbalik dengan nilai *fitness*, semakin kecil nilai penalti (jumlah pelanggaran) semakin besar nilai *fitness* nya.

### c. Seleksi

Salah satu hal penting pada proses GA adalah bagaimana memilih kromosom yang akan diikuti ke generasi berikutnya. Berdasarkan teori evolusi Darwin, hanya kromosom yang terbaik yang dipilih ke generasi berikutnya. Pada proses GA dikenal beberapa metode yang digunakan untuk memilih kromosom. Metode seleksi tersebut secara umum dapat dikelompokkan menjadi dua kelompok. Kelompok pertama adalah metode seleksi yang dilakukan dengan memilih kromosom berdasarkan nilai *fitness relative* terhadap nilai *fitness* kromosom lain yang ada pada populasi atau sering dikenal dengan metode proposional (*propositional selection methods*). Yang termasuk kelompok ini diantaranya *roulette wheel selection*, *stochastic universal selection* dsb. Kelompok kedua adalah seleksi berdasarkan ranking suatu kromosom pada populasi. Pemberian ranking ini tentu saja didasarkan pada nilai *fitness* kromosom tersebut. Yang termasuk kelompok ini diantaranya *elist selection*, *tournament selection*, *rank selection*, dsb. Untuk lebih jelasnya.

### d. Crossover

Crossover merupakan suatu proses pembentukan kromosom turunan (*offspring*) dengan menggabungkan elemen dari kromosom induk yang terpilih (*parent*). Proses ini dilakukan dalam upaya mendapatkan kromosom

baru dengan solusi yang lebih baik. Ada banyak metode *crossover* yang dibahas pada berbagai literatur. Penggunaan metode *crossover* sangatlah berkaitan dengan metode yang di pilih. Apabila metode representasi nilai atau metode representasi biner digunakan, maka ada beberapa pilihan metode *crossover* diantaranya: *Crossover* satu titik (*one-point crossover*) dan *crossover* dua titik (*two-point crossover*). Tidak semua induk mengalami *crossover*, banyaknya induk yang mengalami *crossover* ditentukan dengan nilai laju *crossover* dan dilakukan secara acak.

### e. Mutasi

Proses mutasi biasanya dilakukan dengan melakukan perubahan terhadap gen pada suatu kromosom. Proses ini bertujuan meningkatkan keragaman kromosom. Proses mutasi berfungsi untuk menghasilkan individu baru yang memiliki sifat baru dengan cara melakukan perubahan pada satu atau lebih gen. Proses mutasi dilakukan setelah proses *crossover* dengan cara menentukan titik mutasi pada kromosom tersebut secara acak, kemudian dengan cara memilih kromosom yang akan dimutasi secara acak pula. Banyaknya kromosom yang akan mengalami mutasi dihitung berdasarkan probabilitas mutasi yang telah di tentukan terlebih dahulu.

## 2.7 Tingkat Kesesuaian

Tingkat kesesuaian jadwal digunakan untuk mengetahui persentase seberapa sesuai jadwal dari hasil generate sistem GA dengan peraturan yang berlaku di Jawara digital Art Store . Tingkat kesesuaian dihitung dari nilai *fitness* yang didapat dengan menggunakan rumus sebagai berikut.

$$\text{Tingkat Kesesuaian} = \frac{\text{Jumlah yang didapat}}{\text{Total}} \times 100 \%$$

## 2.8 Pengertian Penjadwalan

Berdasarkan Kamus Bahasa Indonesia, jadwal merupakan pembagian waktu berdasarkan rencana pengaturan urutan kerja. Jadwal juga didefinisikan sebagai daftar atau tabel kegiatan atau rencana kegiatan dengan pembagian waktu pelaksanaan yang terperinci.



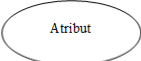

Menurut Ginting (2017), jadwal merupakan sesuatu yang menjelaskan dimana dan kapan orang-orang dan sumber daya berada pada suatu waktu. Sedangkan penjadwalan merupakan proses cara pembuatan menjadwalkan atau memasukkan dalam jadwal, Sedangkan menurut Stevenson & Chuong (2014), yang dialih bahasakan oleh Diana Angelica mengemukakan bahwa “Penjadwalan adalah menetapkan waktu dari penggunaan perlengkapan, fasilitas, dan aktivitas manusia dalam sebuah organisasi”

## 2.9 Entity Relationship Diagram (ERD)

*Entity Relationship Diagram* (ERD) adalah suatu model untuk menjelaskan hubungan antara data dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD berfungsi untuk memodelkan struktur data dan hubungan antar data, untuk menggambarannya digunakan beberapa notasi dan simbol. Ada definisi yang menyebutkan antara lain sebagai berikut:

Menurut Sutanta (2011), pada bukunya yang berjudul *Basis Data Dalam Tinjauan Konseptual* bahwa ERD adalah model data yang dikembangkan berdasarkan objek.

Tabel 1 Notasi dalam ERD





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 sebenarnya yang dimaksud entitas atau relationship dan mewakili atribut dari masing-masing entitas.
4		penghubung antar entitas

## 2.10 Data Flow Diagram (DFD)

*Data Flow Diagram* (DFD) adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, terstruktur dan jelas. Ada definisi yang menyebutkan, antara lain sebagai berikut:

Menurut Jogiyanto (2006), *Data Flow Diagram* (DFD) merupakan diagram yang menggunakan notasi-notasi untuk menggambarkan arus data sistem secara logika. DFD merupakan dokumentasi grafik yang menggunakan simbol penomoran di dalam mengilustrasikan arus data yang saling berhubungan diantara pemrosesan data untuk diubah menjadi informasi

Tabel 2. Data Flow Diagram (DFD)

No.	Gambar	Keterangan
1.		Ekternal Entity (kesatuan luar) atau boundary (batas sistem) yang memisahkan sistem dengan lingkungan luar
2		Data Flow (arus data) mengalir diantara proses, simpanan data dan kesatuan luar.
3		Proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau computer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data keluar dari proses
4.		Data store (simpanan data) merupakan simpanan dari data yang dapat berupa filr atau database

## 3. Metode Penelitian

### 3.1 Obyek Penelitian

Pada penelitian ini, penulis mengambil objek penelitian di *Jawara Digital Art Store*. Data yang digunakan adalah jadwal *shift* kerja karyawan dalam satu bulan. Data tersebut digunakan sebagai sampel guna membuktikan hasil output yang sesuai dengan harapan. Input yang akan digunakan pada sistem yang akan dibangun adalah input nama karyawan yang akan mendapat jadwal *shift* kerja

### 3.2 Metode Pengumpulan Data

Metode penelitian yang dilakukan penulis adalah metode pengumpulan data dengan metode wawancara dengan pihak management yang bertugas dalam pencatatan karyawan di *Jawara Digital Art Store*

### 3.3 Metode Pengembangan Sistem

Pada tahap ini akan dilakukan beberapa proses yaitu analisis sistem, serta desain sistem.

#### 3.3.1 Analisis sistem

Melakukan analisis tentang data-data yang diperoleh digunakan dalam membuat sistem, serta merancang desain dan database dari sistem

yang akan dibangun. Langkah-langkah yang dilakukan penulis pada penelitian ini adalah mendefinisikan masalah dan menentukan solusi yang diinginkan. Setelah itu dari hasil penelitian yang dilakukan di Jawa Digital Art Store dan hasil wawancara diperoleh data-data yang dibutuhkan untuk membuat sistem penjadwalan *shift* kerja karyawan. Dari penelitian ini akan dibuat sebuah sistem penjadwalan *shift* kerja karyawan yang dapat memberikan informasi berupa jadwal yang optimal..

### 3.3.2 Desain sistem

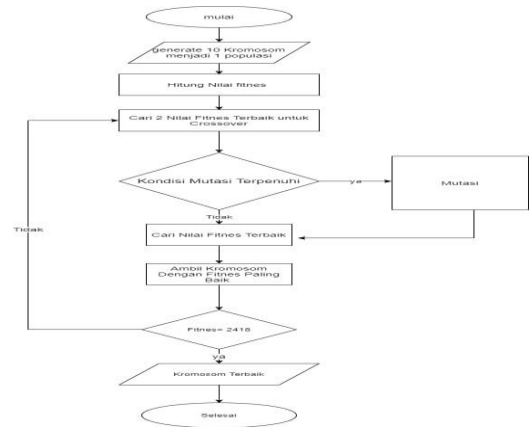
Berdasarkan analisis sistem yang telah dilakukan, dapat dibuat desain rancangan untuk sistem penjadwalan *shift* kerja karyawan Jawa Digital Art Store. Desain sistem yang akan dibangun, meliputi pembuatan desain *input*, desain *output*, desain *interface* dan desain database.

### 3.3.3 Implementasi

Proses implementasi dari perancangan aplikasi yang dilakukan pada bab sebelumnya akan dijelaskan pada bab ini. Implementasi bertujuan untuk menerjemahkan keperluan perangkat lunak ke dalam bentuk sebenarnya yang dimengerti oleh komputer atau dengan kata lain tahap implementasi ini merupakan tahapan lanjutan dari tahap perancangan yang sudah dilakukan. Dalam tahap implementasi ini akan dijelaskan mengenai antarmuka sistem yang berupa tampilan web beserta potongan-potongan script program untuk menampilkan dan mengelola data karyawan beserta proses penjadwalan *shift* kerja

### 3.4 Proses Algoritma Genetika pada penjadwalan

Urutan prosedur di dalam sistem dan apa yang dikerjakan di dalam sistem penjadwalan secara keseluruhan di algoritma genetika ditunjukkan dengan flowchart yang ada pada Gambar 3.1.



Gambar 2 Flowchart Proses GA

## 4. Perhitungan Manual Dan Sistem

### 4.1 Perhitungan Manual

#### 4.1.1 Pembentukan populasi awal.

Misalkan ditentukan jumlah populasi adalah 10 dengan rincian sebagai berikut: Kromosom[1] = [20,41,37,62,54,46,62,57,18,19,47,50,53,75,22,19,31,76,33,6,78,38,20,1,19,35,28,31,63,2,40,65,8,36,57,21,40,73,21,65,21,17,14,44,11,55,64,71,3,52,55,21,26,34,59,12,41,32,59,53,43,58,74,38,23,45,43,40,62,68,24,60,20,15,31,56,20]

Kromosom[2]

= [3,2,40,65,8,36,57,21,40,73,21,65,21,17,14,44,11,55,64,71,3,52,55,21,26,34,59,12,41,32,20,41,37,62,54,46,62,57,18,19,47,50,53,75,22,19,31,76,33,6,78,38,20,1,19,35,28,31,63,59,53,43,58,74,38,23,45,43,40,62,68,24,60,20,15,31,56,10]

Kromosom[3]

= [32,20,41,37,62,54,46,62,57,18,19,47,50,53,75,22,19,31,76,33,6,78,38,20,1,19,35,28,1,3,2,40,65,8,36,57,21,40,73,21,65,21,17,14,44,11,55,64,71,3,52,55,21,26,34,59,12,41,63,59,53,43,58,74,38,23,45,43,40,62,68,24,60,20,15,31,56,9].

Kromosom[4]

= [1,63,59,53,43,58,74,38,23,45,43,40,62,68,24,60,20,15,31,56,31,3,2,40,65,8,36,57,21,40,73,21,65,21,17,14,44,11,55,64,71,3,52,55,21,26,34,59,12,41,32,20,41,37,62,54,46,62,57,18,19,47,50,53,75,22,19,31,76,33,6,78,38,20,1,19,35,28]

Kromosom[5]

= [20,41,37,62,54,46,62,57,18,19,47,50,53,75,22,19,31,76,33,6,78,38,20,1,19,35,28,3,2,40,65,8,36,57,21,40,73,21,65,21,17,14,44,11,55,64,71,3,52,55,21,26,34,59,12,41,32,63,59,53,43,58,74,38,23,45,43,40,62,68,24,60,20,15,31,56,31]

Maksud beberapa kromosom diatas merupakan kumpulan 78 indek pengacakan



#### 4.1.2 Seleksi Kromosom

Hitung nilai fitness sesuai aturan penalti dari setiap kromosom yang telah dibangkitkan:

$$\begin{aligned}
 & \text{fitness}[1] \\
 & = 2418 - 13 \\
 & = 2405 \\
 & \text{fitness}[2] \\
 & = 2418 - 21 \\
 & = 2397 \\
 & \text{fitness}[3] \\
 & = 2418 - 7 \\
 & = 2411 \\
 & \text{fitness}[4] \\
 & = 2418 - 17 \\
 & = 2401 \\
 & \text{fitness}[5] \\
 & = 2418 - 23 \\
 & = 2395
 \end{aligned}$$

#### 4.1.3 Proses Crossover

Pada proses crossover dalam penelitian ini menggunakan 2 induk terbaik yang dipilih berdasarkan nilai fitnessnya. Jika ada nilai fitness yang sama maka diambil salah satu kromosom. Dari populasi hasil seleksi di atas induk yang dipilih adalah:

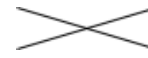
$$\text{kromosom}[1] \text{ fitness}[1] = 2411$$

$$\text{kromosom}[3] \text{ fitness}[3] = 2405$$

Misalkan posisi cut-point crossover didapat dari hasil random nilai [1-jumlah\_gen] mendapatkan nilai, maka crossover akan dilakukan dari gen ke 57 sampai ke gen terakhir dalam kromsom

$$\begin{aligned}
 & \text{Kromosom}[1] \quad \quad \quad = \\
 & = [32,20,41,37,62,54,46,62,57,18,19,47,50,53,7 \\
 & 5,22,19,31,76,33,6,78,38,20,1,19,35,28,1,3,2,4 \\
 & 0,65,8,36,57,21,40,73,21,65,21,17,14,44,11,55
 \end{aligned}$$

,64,71,3,52,55,21,26,34,59,12,41,63,59,53,43,  
58,74,38,23,45,43,40,62,68,24,60,20,15,31,56,  
9]



$$\begin{aligned}
 & \text{Kromosom}[3] \quad \quad \quad = \\
 & = [20,41,37,62,54,46,62,57,18,19,47,50,53,75,2 \\
 & 2,19,31,76,33,6,78,38,20,1,19,35,28,31,63,2,4 \\
 & 0,65,8,36,57,21,40,73,21,65,21,17,14,44,11,55 \\
 & ,64,71,3,52,55,21,26,34,59,12,41,32,59,53,43, \\
 & 58,74,38,23,45,43,40,62,68,24,60,20,15,31,56, \\
 & 20]
 \end{aligned}$$

#### 4.1.4 Proses Mutasi

Misalkan nilai mutation rate (MR) yang ditentukan adalah 0,2, kemungkinan sebanyak 20% dari populasi akan termutasi. Prosesnya adalah dengan membangkitkan nilai acak R dalam range 0-1. Jika  $R[1] < MR$  maka pilih kromosom 1 untuk dimutasi. Misal didapat R setiap kromosom

$$\begin{aligned}
 & R[1] = 0,103 \\
 & R[2] = 0,231 \\
 & R[3] = 0,642 \\
 & R[4] = 0,394 \\
 & R[5] = 0,117
 \end{aligned}$$

#### 4.2 Perhitungan Sistem

Pengujian pada penelitian ini dilakukan sesuai paramater GA yang digunakan yaitu ukuran populasi, jumlah karyawan perhari, jumlah hari, *mutation rate*, dan banyak generasi. Masing-masing parameter diuji dengan beberapa variasi nilai untuk mendapatkan kombinasi jadwal yang paling optimal. Semua pengujian dilakukan mengacu pada hasil nilai *fitness* terbaik yang dihasilkan dan tingkat kesesuaian yang dicapai.

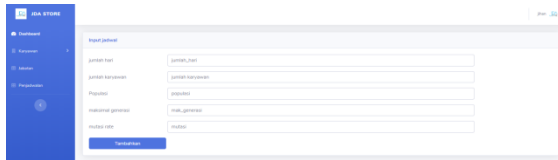
Pada sistem yang dibuat, untuk memudahkan menganalisis hasil dari generate proses GA, peneliti membuat semacam *bot* / sistem pengecekan. *Bot* ini berfungsi mengecek jadwal hasil generate GA sudah sesuai atau masih ada benturan. Kemudian untuk memudahkan mengetahui gen mana yang bertabrakan.

Tabel 3 Tabel Warna Penalty

penalti	Warna gen jadwal
Benturan jadwal Shift karyawan yang sama dalam satu shift	merah
Benturan jadwal shift karyawan dalam satu hari	kuning
Ada karyawan wanita yang mendapat shift di malam hari	Merah muda
Benturan jadwal shift karyawan yang sudah masuk shift malam masih mendapat jadwal keesok harinya shift pagi	biru

#### 4.2.1 Halaman Penjadwalan Shift Kerja

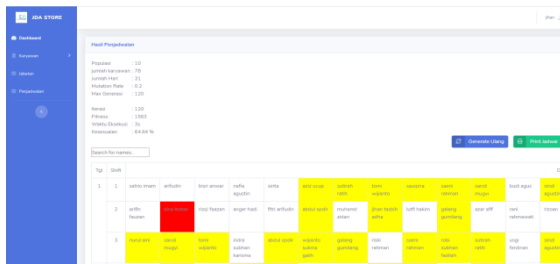
merupakan halaman yang menampilkan form untuk mengisi parameter penjadwalan yang akan di generate secara otomatis menggunakan Algoritma Genetika. Dimana nantinya memasukan parameter jumlah hari dalam 1 bulan jadwal yang mau di buat, jumlah karyawan, populasi, maksimal generasi dan mutasi rate.



Gambar 3 Halaman Penjadwalan Shift kerja

#### 4.2.2 Halaman Hasil Penjadwalan

Gambar 4 merupakan halaman untuk menampilkan hasil generate proses penjadwalan menggunakan GA. Ditampilkan hasil nilai fitness, banyaknya iterasi yang diperlukan, lamanya waktu generate, serta hasil jadwal yang optimal.



Gambar 4 Warna Gen Jadwal Benturan

#### 4.2.3 Hasil Pengujian Pemilihan Bibit Populasi Terbaik

Pengujian pertama yaitu memilih bibit populasi yang paling baik agar menghasilkan solusi jadwal yang sesuai atau dapat dikatakan paling optimal. Pengujian ini dilakukan dengan memilih bibit populasi yang memiliki nilai *fitness* dan tingkat kesesuaian terbaik untuk dijadikan calon populasi pada proses GA selanjutnya. Peneliti menggunakan ukuran populasi sebanyak 15 populasi, jumlah orang yang dibutuhkan dalam satu hari 78 orang, *mutation rate* 0.2 dengan generasi sebanyak 5000 dan *mutation rate* 0.4 generasi sebanyak 50000. Pengujian tersebut dilakukan sebanyak 10 kali untuk mendapatkan satu bibit populasi dengan nilai *fitness* dan tingkat kesesuaian terbaik.

Tabel 4 Hasil pengujian bibit dengan *mutation rate* 0.2 dan 5000 generasi

Pengujian	<i>Fitness</i>	Kesesuaian (%)	Waktu Eksekusi
1	1606	66,42	6m.16s
2	1559	64,47	3m.5s
3	1579	65,3	3m.8s
4	1575	65,14	3m.2s
5	1576	65,18	6m.1s
6	1584	65,51	2m.52s
7	1588	65,67	2m.47s
8	1595	65,96	6m.30s
9	1601	66,21	6m.0s
10	1623	67,12	2m.46s

Berdasarkan hasil pengujian pada Tabel 5.2, pengujian ke 10 menghasilkan nilai paling tinggi dari 10 pengujian yang dilakukan dengan *fitness* sebesar 1623 dan tingkat kesesuaian 67,12%. Dari hasil nilai tersebut, populasi pada pengujian ke 10 menjadi bibit yang paling baik untuk calon populasi pada proses pengujian selanjutnya

Tabel 5 Hasil pengujian bibit dengan *mutation rate* 0.4 dan 50000 generasi

<i>Mutation Rate</i>	<i>Fitness</i>	Kesesuaian (%)	Waktu Eksekusi
0,2	1602	66,25	5m.52s
0,4	1639	67,78	6m.30s
0,6	1625	67,2	15m.8s
0,8	1611	66,63	8m.9s

Pengujian	<i>Fitness</i>	Kesesuaian (%)	Waktu Eksekusi
1	1563	64,64	30m.47s
2	1625	67,2	30m.54s
3	1564	64,68	30m.27s
4	1579	65,3	35m.23s
5	1600	66,17	30m.56s
6	1571	64,97	31m49s
7	1644	67,99	31m.25s
8	1607	66,46	29m.56s
9	1613	66,71	30m.39s
10	1601	66,21	29m.41s

Berdasarkan hasil pengujian pada Tabel 6, pengujian ke 7 menghasilkan nilai paling tinggi dari 10 pengujian yang dilakukan dengan *fitness* sebesar 1644 dan tingkat kesesuaian 67,99 %. Dari hasil nilai tersebut, populasi pada pengujian ke 7 menjadi bibit yang paling baik untuk calon populasi pada proses pengujian selanjutnya.

#### 4.2.4 Hasil Pengujian Mutation Rate

Pengujian kedua yaitu menguji beberapa nilai *mutation rate* terhadap nilai *fitness* yang dihasilkan serta tingkat kesesuaian yang dicapai. Pengujian ini dilakukan untuk mengetahui *mutation rate* terbaik yang dapat menghasilkan solusi jadwal yang optimal. Nilai generasi digunakan sebesar 2000 dan 15000 generasi dan jumlah populasi sebesar 15 populasi. Populasi yang dibunakan sesuai dengan hasil pengujian sebelumnya dengan harapan dapat menghasilkan solusi jadwal yang lebih optimal. Nilai *mutation rate* yang diuji mulai dari 0.2., 0.4, 0.6 dan 0,8

Tabel 6 Hasil pengujian *mutation rate* dengan 2000 generasi

Mutation Rate	Fitness	Kesesuaian (%)	Waktu Eksekusi
0,2	1649	68,2	2m.35s
0,4	1607	66,46	1m.18ss
0,6	1629	67,37	1m.29s
0,8	1611	66,63	1m.39s

Pengujian pada Tabel 5.4, *mutation rate* dengan nilai 0.2 dapat menghasilkan nilai paling tinggi daripada nilai *mutation rate* lainnya yaitu dengan hasil *fitness* 1649 dan tingkat kesesuaian 68,2 %. Berdasarkan hasil tersebut, nilai *mutation rate* 0.2 menjadi acuan untuk parameter pengujian berikutnya agar mendapat hasil yang lebih optimal lagi.

Tabel 7 Hasil pengujian *mutation rate* dengan 10000 generasi

Pengujian pada Tabel 5.5, tidak ada peningkatan nilai *fitness* dan tingkat kesesuaian meskipun *mutation rate* telah diubah mulai dari 0,2 hingga 0,8. Meskipun tidak ada peningkatan nilai *fitness*, dari hasil ini peneliti mencoba untuk mengambil nilai *mutation rate* tertinggi yang ada yaitu 0.4 untuk diuji di penelitian selanjutnya.

#### 4.2.5 Hasil Pengujian Generasi

Pengujian terakhir yaitu menguji beberapa generasi terhadap nilai *fitness* yang dihasilkan dan tingkat kesesuaian yang dicapai. Pengujian ini dilakukan untuk mengetahui seberapa banyak generasi yang diperlukan agar mencapai hasil jadwal yang optimal. Populasi yang digunakan berjumlah 15 populasi hasil dari pengujian pemilihan bibit populasi dan nilai *mutation rate* yang digunakan sebesar 0,2 untuk bibit populasi dengan awal 2000 generasi dan *mutation rate* 0,4 untuk bibit populasi dengan awal 10000 generasi. Jumlah generasi yang diuji mulai dari 100 sampai 100000 generas.

Tabel 8 Hasil pengujian generasi dengan *mutation rate* 0.2

generasi	Fitness	Kesesuaian (%)	Waktu Eksekusi
100	1606	66,42	3s
500	1628	67,33	39s
700	1603	66,29	52s
1000	1596	66	1m.9s
1500	1614	66,75	1m.4s
1700	1638	67,74	2m5s
10000	1602	66,25	5m.52s
15000	1638	67,74	18m.21s
17000	1613	66,67	21m.0s
100000	1611	66,63	55m.2s

Berdasarkan pengujian Tabel 5.6, dari 100 hingga 100000 generasi tidak terjadi peningkatan nilai *fitness* dan tingkat kesesuaian. Tapi ada dua generasi yang kesesuaian dan *fitness* yang sama yaitu 1638 di generasi 1700 dan 15000 dengan kesesuaian 67,75 akan tetapi beda pada waktu eksekusi. Pengujian generasi terus dilakukan sampai

100000 generasi dengan nilai *fitness* 1611 dan tingkat kesesuaian 66,63%.

Tabel 9 Hasil pengujian generasi dengan *mutation rate* 0.4

generasi	Fitness	Kesesuaian (%)	Waktu Eksekusi
100	1615	66,79	4s
500	1620	67	38s
700	1593	65,88	56s
1000	1611	66,63	1m.20s
1500	1607	66,46	2m.1s
1700	1614	66,75	2m19s
10000	1639	67,78	6m.30s
15000	1638	67,74	18m.21s
17000	1613	67,33	10m.54s
100000	1635	67,62	1h 2m.42s

Berdasarkan pengujian Tabel 5.7, dari 100 hingga 100000 generasi nilai *fitness* dan tingkat kesesuaian masih naik turun . pada table pengujian tersebut terjadi peningkatan paling menonjol di generasi 10000 dengan nilai *fitness* 1639 dan tingkat kesesuaian mencapai 67,78% dengan waktu eksekusi 6m.30s. Pengujian terus dilakukan sampai pada generasi 100000. Pengujian dilakukan hingga 100000 generasi dengan nilai *fitness* 1635 dan tingkat kesesuaian 67,62%.

## 5. Penutup

### 5.1 Kesimpulan

Kesimpulan dari implementasi algoritma genetika dalam penjadwalan *shift* kerja karyawan di JDA Store adalah sebagai berikut:

1. Algoritma genetika terbukti dapat digunakan untuk menyelesaikan permasalahan penjadwalan *shift* kerja karyawan di JDA store.
2. Hasil jadwal dari generate sistem yang menggunakan algoritma genetika cukup sesuai dengan peraturan yang ada di JDA store dan dibuktikan dengan hasil pengujian yang telah peneliti lakukan.
3. Pada penelitian ini algoritma genetika menggunakan representasi kromosom dengan panjang 2418 gen yang mewakili jadwal dalam satu bulan atau 31 hari. Populasi yang digunakan berjumlah 15 populasi diambil dari hasil pengujian selama 10 kali. Metode seleksi yang digunakan yaitu *Tournament Selection*, metode *crossover* menggunakan *one-cut-point* dan metode mutasi menggunakan penyisipan. Dari ketiga metode tersebut, peneliti hanya melakukan variasi pada nilai *mutation rate* mulai dari rentan 0.2, 0.4, 0.6 dan 0,8 serta batas generasi mulai dari rentan 100 – 100000 generasi. nilai *crossover rate* tidak divariasikan dalam penelitian ini karena sudah pasti

kromosom yang dikawin silangkan merupakan 2 kromosom terbaik dalam setiap generasi.

4. Berdasarkan hasil penelitian, pada pengujian pertama dengan nilai awal *mutation rate* 0,2 dan batas generasi 5000 menghasilkan nilai *fitness* tertinggi 1623 dengan tingkat kesesuaian 67,12%. Sedangkan pengujian dengan nilai *mutation rate* awal 0.4 dan batas generasi 50000 menghasilkan nilai *fitness* tertinggi 1644 dengan tingkat kesesuaian 67,99% pada generasi 500000 keatas.
5. Pemilihan bibit populasi awal mempengaruhi kualitas populasi ketika dilakukan pengujian dengan variasi nilai *mutation rate* dan batas generasi.

Semakin tinggi *mutation rate* belum tentu hasil yang didapatkan optimal. Sementara semakin tinggi batas generasi, hasil nilai *fitness* dan tingkat kesesuaian bisa semakin baik. Tetapi hal tersebut mengakibatkan proses generate jadwal memakan waktu eksekusi yang lama

## 5.2 Saran

Beberapa saran yang dapat untuk pengembangan sistem yang telah dibuat diantaranya adalah:

1. Pada penelitian penjadwalan menggunakan GA selanjutnya, untuk mendapatkan hasil yang semakin optimal perlu dilakukan lebih banyak variasi lagi pada metode seleksi, metode *crossover* dan metode mutasi yang digunakan. Ketiga metode tersebut juga dikombinasikan dengan banyaknya populasi serta batas maksimal generasi dapat divariasikan lagi untuk mencapai hasil optimal yang diinginkan.
2. Untuk mendapatkan hasil jadwal yang optimal, waktu eksekusi program cukup memakan waktu yang lama. Maka dari itu perlu dipertimbangkan hardware / komputer yang digunakan untuk melakukan generate jadwal.
3. Peneliti memberi alternatif solusi dengan cara mengosongkan gen jadwal yang masih terjadi benturan agar petugas yang membagi jadwal *shift* tetap dapat mengisi nama karyawan yang

masih berbenturan jadwalnya dengan nama karyawan lain.

## Daftar Pustaka

- [1] Damayanti, Putri, M., Dan Fauzi, A. (2017), *Implementasi Algoritma Genetika Untuk Penjadwalan Customer Service ( Studi Kasus : Biro Perjalanan Kangaroo )*, *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya, Malang*.
- [2] Giting, clarisa, W. (2017), *Implementasi Algoritma Genetika Dalam Penjadwalan Shift Kerja Di Call Center Telkomsel Medan, UNIVERSITAS SUMATERA UTARA*, 1–70.
- [3] Haupt, R.L. dan Haupt, S.E. (2004), *practical genetic algorithms John Wiley & Sons, Inc. Canada*.
- [4] Nugraha, Dudo, Dan Saud (2017), *Sistem Penjadwalan Perkuliahan M Enggunakan Algoritma Genetika ( Studi Kasus Pada Jurusan Teknologi Inform Asi Fakultas Teknik Universitas Tadulako )*, *Palu, Sulawesi tengah*.
- [5] Priambodo, M.A., Nhita, Dan Aditsania, A. (2016), *penjadwalan Mata Kuliah Menggunakan Metode Hybrid Algoritma Genetika Dan, EProceedings of Engineering, University Telkom Bandung*.
- [6] Satriyanto, E. (2017), *Penjadwalan Satpam Jaga Dengan Algoritma Genetika, , 1(2)*, 81–87.
- [7] Sianturi, A.L. (2012), *Optimasi Penjadwalan Karyawan Pengawas Pembangunan Kapal Dengan Menggunakan Algoritma Gentika, Universitas Indonesia, Depok*.
- [8] Suseno dan Dhuha, E. (2017), *Penjadwalan Tenaga Kerja Untuk Tiga Shift Kerja Dengan Pengembangan Metode Algoritma Tibrewala, Philippe Dan Browne, Teknologi, Universitas Teknologi Yogyakarta, Yogyakarta*.
- [9] Turban, E. (1995). *Decision support and expert systems: management support systems (4 ed.)*. Prentice Hall.
- [10] Susandi, D., Dan Milana, L. (2015). *Perancangan dan Pembuatan Aplikasi*

*Penyusunan Jadwal Kerja Dinas Jaga Perawat IGD Menggunakan Algoritma TPB. Metris, 16, 29–34. <https://doi.org/1411-3287>*

- [11] Ilmi, R. R., Mahmudy, W. F., Dan Ratnawati, D. E. (2015). *Optimasi Penjadwalan Perawat Menggunakan Algoritma Genetika. Repository Jurnal Mahasiswa PTIIK Universitas Brawijaya, Surabaya.*