

## **RANCANG BANGUN SISTEM INFORMASI PERGANTIAN (*SHIF*) STAF PADA SWALAYAN DENGAN MENGGUNAKAN ALGORITMA GENETIKA**

**M. Khadafi<sup>1</sup>, Cholish<sup>2</sup>, Bakti V Sundawa<sup>2</sup>**

Teknik Informatika<sup>1</sup>, Teknologi Informasi dan Komputer, Politeknik Negeri Lhokseumawe

Teknik Listrik<sup>2</sup>, Teknik Elektro, Politeknik Negeri Medan

mkhadafi@pnl.ac.id<sup>1</sup>, cholish@polmed.ac.id<sup>2</sup>, baktivsundawa@polmed.ac.id<sup>2</sup>

### **ABSTRAK**

Swalayan merupakan salah satu lapangan kerja saat ini baik di kota besar maupun daerah perdesaan, seiring dengan majunya perkembangan swalayan pada masa sekarang, perlu sebuah sistem penjadwal yang efektif. Merancang dan membangun sebuah sistem dalam rangka mengatur staf. Sistem Informasi pergantian staf swalayan ini dibangun atas dasar kebutuhan dengan menerapkan algoritma genetika. Terkadang staf bekerja di toko yang berbeda dengan perusahaan yang sama perlu mempertimbangkan jarak tempuh staf dalam bekerja. Dari hasil pengujian system menggunakan *blackbox* untuk menguji alhoritma genetikan dapat disimpulkan bahwa dari beberapa aspek bahwa sistem ini dapat digunakan bagi swalayan yang stafnya banyak dant tidak efektif untuk staf yang sedikit.

**Kata Kunci** : Pergantian, Algoritma Genetika, Sistem Informasi

### **PENDAHULUAN**

Pergantian masuk dan keluar staf merupakan sebuah masalah tersendiri apa lagi dilihat dari jarak tempuh tempat tinggal karyawan. Saat ini Pergantian shift masuk pegawai di swalayan masih menggunakan sistem penyusunan jadwal secara maual. Sehingga membutuhkan waktu yang lama dalam proses penyusunan (Febriyana & Mahmudy, 2016).

Salah satu metode yang dapat digunakan dalam membangun sitem dalam rangka meminimalkan permasalahan pergantian staf dengan algoritma genetika. Algoritma Genetika merupakan pendekatan komputasional untuk menyelesaikan masalah yang dimodelkan dengan proses evolusi biologis. Diharapkan dengan digunakannya algoritma genetika akan diperoleh optimasi Pergantian staf yaitu kondisi dimana terjadi kombinasi yang baik sehingga dalam menyusun jadwal pergantian atau shift dapat dilakukan dengan mudah dan akurat. Merancang dan membangun sistem informasi Pergantian staf dengan penerapan algoritma genetika secara sistematis dan terstruktur.

### **TINJAUAN PUSTAKA**

#### **Pengertian Rancang Bangun**

Menurut Stair dan Reynolds (2010), Perancangan sistem adalah fase pengembangan sistem yang mendefinisikan bagaimana sistem informasi akan melakukan perancangan untuk mendapatkan solusi pemecahan masalah. Sedangkan menurut Laudon dan Laudon (2010), Perancangan Sistem merupakan keseluruhan rencana atau model untuk sistem yang terdiri dari semua spesifikasi sistem yang memberikan bentuk dan struktur.

#### **Pengertian Sistem**

Sistem adalah sebuah data atau lebih dan agar mencapai sebuah tujuan itu, maka komponennya harus saling berkaitan, sedangkan informasi yaitu sebuah data diproses untuk mengambil sebuah rekomendasi kebijakan, sehingga jika disimpulkan bahwa sistem informasi yaitu serangkaian data yang saling berInteraksi untuk memproses, menyimpan dan menyediakan informasi (Monalisa et al., 2018). Edhy Sutanta (2011) telah menjelaskan Komponen yang terdapat dalam ERD adalah sebagai berikut : Entitas adalah sebuah objek yang bisa dibedakan dari lainnya yang mampu

diwujudkan dalam *database*. Objek yang paling dasarnya berupa seperti orang, benda, atau hal lain yang kejelasannya harus tersimpan dalam *database*.

### Pengertian Penjadwalan

Penjadwalan adalah pengurutan pembuatan atau pengerjaan produk secara menyeluruh yang dikerjakan pada beberapa buah mesin. Dengan demikian masalah melibatkan pengerjaan sejumlah komponen yang sering disebut dengan istilah *job*. *Job* sendiri masih merupakan komposisi dari sejumlah elemen-elemen dasar yang disebut aktivitas atau operasi. Tiap aktivitas atau operasi ini membutuhkan alokasi sumber daya tertentu selama periode waktu tertentu yang sering disebut dengan waktu proses (R. Ginting, 2009).

### Algoritma Genetika

Algoritma genetika merupakan evaluasi atau perkembangan dunia komputer dalam bidang kecerdasan buatan (*artificial intelligence*). Kemunculan algoritma genetika ini terinspirasi oleh teori Darwin dan teori-teori dalam ilmu biologi, sehingga banyak istilah dan konsep biologi yang digunakan dalam algoritma genetika, karena sesuai dengan namanya, proses-proses yang terjadi dalam algoritma genetika sama dengan apa yang terjadi pada evaluasi biologi (K.C. Laudon and P.L. Jane 2010).

Menurut Kusumadewi (Ulisna Ade Rifai, 2012), Terdapat 8 komponen utama dalam algoritma genetika, yaitu :

#### 1. Teknik Penyandian

Teknik penyandian disini meliputi penyandian gen dari kromosom. Gen merupakan bagian dari kromosom. Satu gen biasanya akan mewakili satu variabel. Variabel adalah suatu sebutan yang dapat diberi nilai angka (kuantitatif) atau nilai mutu (kualitatif). Variabel merupakan pengelompokan secara logis dari dua atau lebih atribut dari objek yang diteliti (T. Sutabri, Konsep Dasar Informasi, 2012). Gen dapat direpresentasikan dalam bentuk : *string bit*, pohon, array bilangan real, daftar aturan, elemen permutasi, elemen program, atau representasi lainnya yang dapat diimplementasikan untuk operator genetika.

#### *String Biner*

0 1 0		1 0 0 1		0 1 1 1
Gen 1		Gen 2		Gen 3

#### 2. Fungsi *Fitness*

Fungsi *Fitness*, alat ukur yang digunakan untuk proses evaluasi kromosom. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas kromosom dalam populasi tersebut [(anjoyo, 2006). Rumusan *fitness* yang digunakan adalah sebagai berikut :

$$F = \frac{\sum \text{Ketersediaan Waktu}}{\text{Minggu}}$$

F = *Fitness*

$\sum$  = Jumlah Keseluruhan

#### 3. Prosedur Inisialisasi

Ukuran populasi tergantung pada masalah yang akan dipecahkan dan jenis operator genetika yang akan diimplementasikan. Setelah ukuran populasi ditentukan, kemudian harus dilakukan inisialisasi terhadap kromosom yang terdapat pada populasi tersebut. Inisialisasi kromosom dilakukan secara acak, namun demikian harus tetap memperhatikan domain solusi dan kendala permasalahan yang ada (sanjoyo, 2006).

#### 4. Pembangkitan Populasi Awal

Populasi awal dibangun secara acak, sedangkan populasi berikutnya merupakan hasil evolusi kromosom-kromosom melalui iterasi yang disebut generasi (Sanjoyo, 2006).

## 5. Fungsi Evaluasi

Dua hal yang harus dilakukan dalam melakukan evaluasi kromosom, yaitu evaluasi fungsi objektif (fungsi tujuan) dan konversi fungsi objektif ke dalam fungsi fitness. Secara umum, fungsi fitness diturunkan dari fungsi objektif dengan nilai yang tidak negatif. Fungsi objektif adalah fungsi yang nilainya akan dioptimalkan. Fungsi objektif bisa bernilai maksimum dan minimum tergantung pada kasusnya. Jika fungsi objektif biaya produksi, maka nilainya dicari yang minimum. Tapi kalau fungsi objektifnya berupa keuntungan, maka nilainya yang maksimum.

## 6. Kriteria optimasi tercapai

Beberapa kriteria berhenti yang sering digunakan antara lain :

- a. Berhenti pada generasi tertentu.
- b. Berhenti setelah beberapa generasi berturut-turut didapatkan nilai *fitness* tertinggi/terendah.
- c. Berhenti bila dalam (n) generasi berikutnya tidak diperoleh nilai *fitness* yang lebih tinggi /rendah.

## 7. Operator genetika

Operator standar yang biasa digunakan dalam algoritma genetika adalah *selection*, *crossover* dan *mutation*. Berikut ini akan dijelaskan masing-masing operator (A. Desiani dan M. Arhami, 2006).

### a. Seleksi

Seleksi bertujuan memberikan kesempatan reproduksi yang lebih besar bagi anggota populasi yang paling fit. Langkah pertama dalam seleksi ini adalah pencarian nilai *fitness*. Masing-masing individu dalam suatu wadah seleksi akan menerima probabilitas reproduksi yang tergantung pada nilai objektif dirinya sendiri terhadap semua individu dalam wadah seleksi tersebut. Suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran performansinya. Fungsi ini dinamakan fungsi *fitness*. Di dalam evolusi alam, individu yang bernilai *fitness* tinggi yang akan bertahan hidup. Sedangkan individu yang bernilai *fitness* rendah akan mati. Dalam algoritma genetika, fungsi *fitness* adalah fungsi objektif dari masalah yang akan dioptimasi. Fungsi ini sebagai ukuran keuntungan yang ingin dimaksimalkan atau sebagai ukuran biaya yang ingin diminimumkan (A. Desiani dan M. Arhami, 2006). Evolusi solusi yang akan mengevaluasi setiap populasi dengan menghitung nilai *fitness* setiap kromosom hingga kriteria berhenti terpenuhi maka akan dibentuk lagi generasi baru dengan mengulangi membentuk generasi baru dengan menggunakan tiga operasi diatas secara berulang-ulang sehingga diperoleh kromosom yang cukup untuk membentuk generasi baru sebagai representasi dari solusi baru.

### b. Rekombinasi (*Crossover*)

*Crossover* (perkawinan silang) bertujuan menambah keanekaragaman string dalam suatu populasi dengan penyilangan antarstring yang diperoleh dari reproduksi sebelumnya.

### c. Mutasi

Mutasi merupakan proses mengubah nilai dari satu atau beberapa gen dalam kromosom. Operasi *crossover* yang dilakukan pada kromosom dengan tujuan untuk memperoleh kromosom-kromosom baru sebagai kandidat solusi pada generasi mendatang dengan *fitness* yang lebih.

## 8. Generasi Terakhir

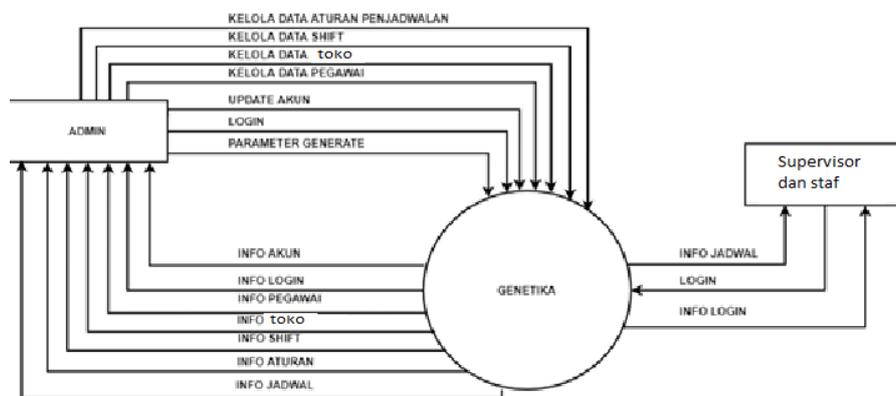
Algoritma genetika adalah algoritma pencarian hasil yang terbaik, yang didasarkan pada perkawinan silang (*crossover*) dan seleksi gen secara alami. Setelah beberapa generasi maka algoritma genetika akan berada pada generasi terakhir dimana menghasilkan kromosom terbaik, yang diharapkan menghasilkan individu baru. Menurut Kusumadewi (Ulisna Ade Rifai, 2011), Algoritma genetika membutuhkan beberapa nilai parameter yang menentukan kinerja program. Parameter yang biasa digunakan pada algoritma genetika adalah :

- a. *Population size*, jumlah individu yang dilibatkan pada setiap generasi.
- b. *Crossover rate* merupakan rasio perbandingan banyaknya *off spring* yang diproduksi pada tiap generasi dengan banyaknya *population size*. *Crossover rate* yang besar membuat area solusi semakin besar dan mengurangi kemungkinan kesalahan perolehan solusi optimum. Tetapi jika *crossover rate* ini terlalu tinggi akan terjadi pembuangan waktu komputasi karena area solusi yang tidak menjanjikan solusi pun akan dieksplorasi.

- c. *Mutation rate*, merupakan representasi terjadinya kromosom baru sebagai akibat mutasi, dari keseluruhan *population size*. Jika *mutation rate* ini terlalu rendah, banyak kemungkinan solusi yang tidak akan dicoba sementara.

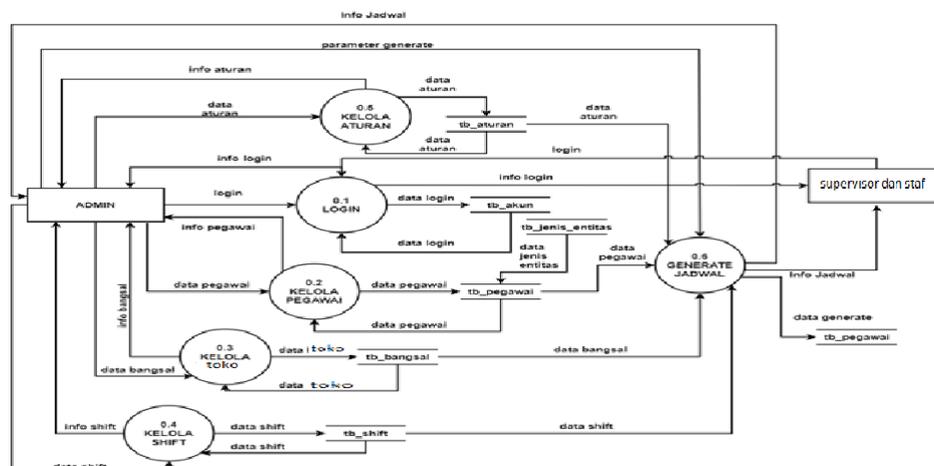
## METODE PENELITIAN

Dalam perancangan sebuah sistem, sangat dibutuhkan analisa dilakukan sebelum memasuki tahap perancangan, yang harus dianalisis adalah kebutuhan yang diperlukan untuk membangun sebuah sistem. Analisis yang dilakukan untuk merancang sistem yaitu; kebutuhan fungsional dan kebutuhan non fungsional. Kebutuhan *fungsional* merupakan sebuah kebutuhan yang berisikan proses-proses yang nantinya akan disediakan oleh sistem bagaimana perilaku sistem pada situasi tertentu dan bagaimana sistem harus bereaksi pada sistem tertentu. Kebutuhan yang dibutuhkan untuk membuat rancangan sistem ini adalah kebutuhan *user*. Perancangan sistem menggunakan metode algoritma genetika memiliki kebutuhan fungsional melakukan *login*, menambah data *user*, data staf, data toko, data jam, data hari, admin dapat membaca data *user*, data toko, data jam, data hari, admin dapat mengubah data *user*, data ruangatau bagian, data jam, data hari, serta dapat menghapus data user, data toko, data jam, data hari, admin dapat melakukan proses *generate* jadwal. Ilustrasinya dapatdi lihat pada diagram di bawah ini :



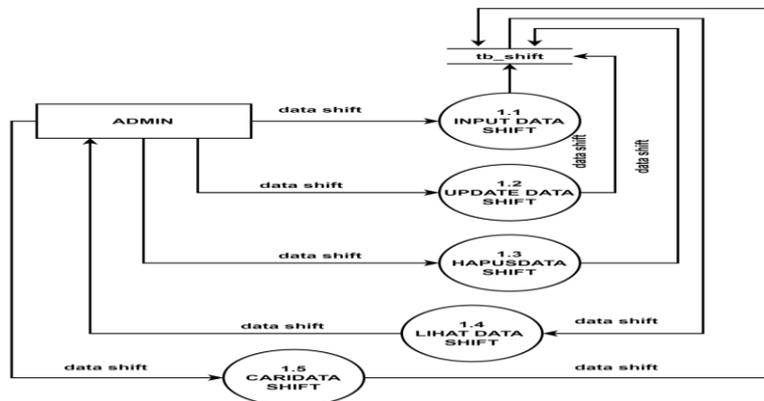
Gambar 1. Diagram Konteks

Gambar 1. menunjukkan bahwa admin dapat login ke sistem dan bertugas untuk mengelola data dan akan diinputkan Supervisor dan staf login dan menerima info pergantian. DFD Level 1 Data Shift dapat dilihat pada gambar di bawah ini.



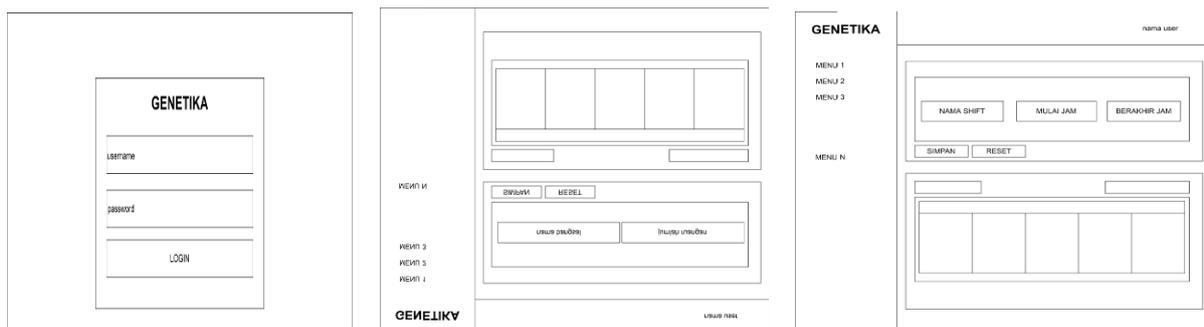
Gambar 2. DFD Level 0

Gambar 2. menunjukkan bahwa admin dapat mengelola sistem secara penuh, baik mengelola pegawai atau staf, toko, shif , aturan dan generate jadwal seperti gambar berikut :



Gambar 3. DFD Level 1 Data Shift

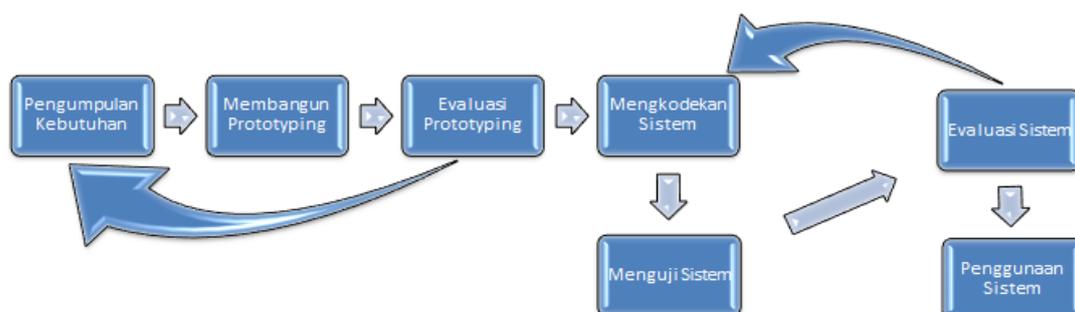
Gambar 3. merupakan Proses data shift ini yang dapat menjalankan hanya entitas admin meliputi input data *shift*, update data, proses dan hapus, Proses mencari data *shift* dari kebutuhan sistem. Rancangan awal dari sistem, meliputi rancangan input berupa login, form input staf supervisor, toko dapat dilihat seperti gambar di bawah ini :



Gambar 4. Menunjukkan dari Sistem

Gambar 4. menunjukkan rancangan dari sistem yang meliputi input, proses data shift ini yang dapat menjalankan hanya entitas admin meliputi input data *shift*, *update* data *shift*, proses, hapus data *Shift*, Proses lihat data *shift*, proses cari data *shift*.

Dalam membangun sistem atau memodelkan sebuah perangkat lunak dibutuhkan beberapa tahapan dalam proses pengembangannya, tahapan inilah yang akan menentukan keberhasilan dari sebuah *software* itu. Tahapan-tahapan dalam model *prototype* seperti gambar berikut :



Gambar 5. Tahapan Metode *Prototype*

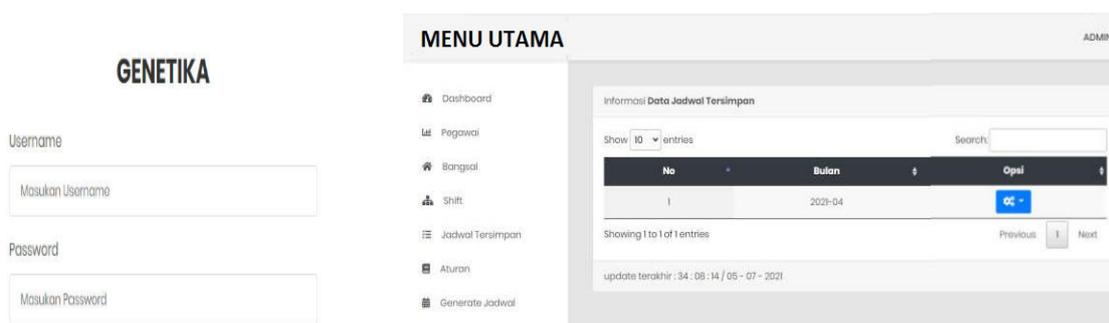
- a. Pengumpulan Kebutuhan  
Pada tahap pengumpulan kebutuhan, pelanggan dan pengembang bersama-sama mendefinisikan format dan kebutuhan keseluruhan perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.
- b. Membangun *prototyping*  
Pada tahap pembangunan *prototyping*, pelanggan dan pembuat sistem bersama-sama membuat format input maupun output yang akan dihasilkan oleh sistem yang dibuat.
- c. Evaluasi *prototyping*  
Selanjutnya, setelah tahap pembangunan *prototyping*, pelanggan dan pengembang bersama-sama mendefinisikan format dan kebutuhan keseluruhan perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.
- d. Mengkodekan Sistem  
Dalam tahap ini *prototyping* yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.
- e. Menguji Sistem  
Pada tahap pengujian sistem, coding yang telah dibuat sebelumnya akan diuji apakah dapat berjalan dengan baik atau masih ada bagian-bagian yang perlu diperbaiki atau masih ada bagian yang belum sesuai dengan keinginan pelanggan.
- f. Evaluasi Sistem  
Evaluasi sistem bukanlah evaluasi *prototyping*, evaluasi sistem adalah mengevaluasi sistem atau perangkat lunak yang sudah jadi apakah sudah sesuai dengan keinginan pelanggan atau belum. Jika belum, maka sistem akan direvisi kembali dan kembali ke tahap 4 dan 5. Jika sistem sudah dikatakan sempurna maka sistem siap dilanjutkan pada tahap selanjutnya.
- g. Menggunakan Sistem  
Tahap ini merupakan tahap akhir dari pembuatan sistem dengan metode *prototyping* model. Pada tahap ini perangkat lunak yang sudah jadi dan sudah lulus uji, siap untuk digunakan oleh pelanggan/user.

## HASIL DAN PEMBAHASAN

Dalam membangun sistem ada beberapa fitur yang menjadi acuan sistem ini hasil yaitu fitur supervisor, staf /karyawan jadwal toko jarak, toko pada proses generate jarak toko menjadi acuan atau kriteria untuk di gunakan dalam rangka mengimplementasikan hasil dari sistem yang dibangun.

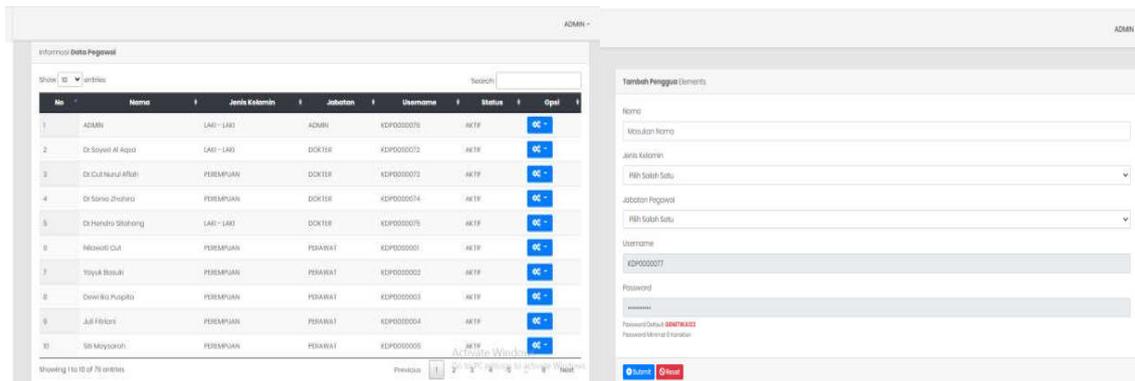
### Implementasi *User Interface*

Implementasi antar muka untuk memberikan akses terhadap manusia dengan sistem yang dapat membuat sistem itu digunakan dengan mudah. Berikut adalah tampilan *User Interface* dari Perancangan Sistem staf Masuk Menggunakan Metode Algoritma Halaman Login, menu utama. Halaman Login adalah halaman yang mengharuskan user untuk melakukan registrasi dengan memasukkan *username* dan *password* agar masuk kedalam beranda utama dari web penjadwalan ini.



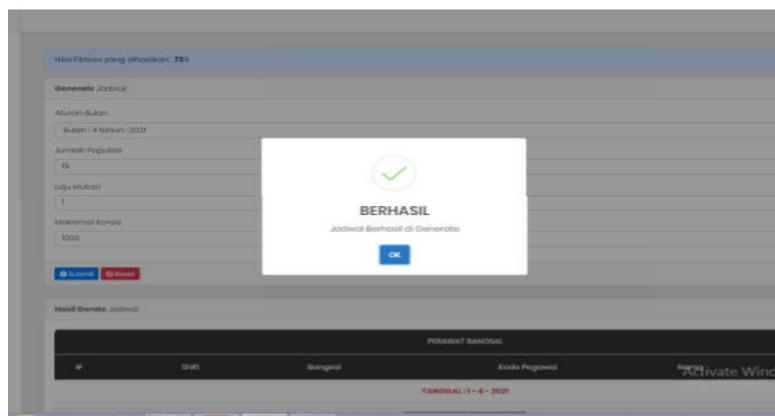
Gambar 6. Halaman Login dan Halaman Menu Utama

Gambar 6. merupakan halaman yang pertama kali muncul untuk masuk ke dalam sistem dengan melakukan login terlebih dahulu. Pada halaman *dashboard admin* terdapat dalam. Untuk tampilan senu input dari sub menu yang dipilih dapat dilihat pada di bawah ini.



Gambar 7. Halaman *form input* dan Hapus

Gambar 7. merupakan halaman untuk melakukan *input* dan hapus dari salah satu *form* yang ada dalam sub sistem. Pada halaman *dashboard admin* terdapat dalam. Untuk tampilan submenu *input* dari sub menu yang dipilih, adapun keberhasilan dari *generate* dapat dilihat pada gambar di bawah ini.



Gambar 8. Keberhasilan dalam Meng-*generate* Menggunakan Algoritma Genetika

### Pengujian *Black Box*

Setiap halaman dilakukan pengujian untuk hasil yang bersesuaian dengan yang dipilih sebagai metode. Pada metode ini dilakukan uji berdasarkan apa yang dilihat dan hanya fokus pada fungsional dan output. Tingkat kesuksesannya sebuah sistem yang telah diimplementasikan, diperlukan pengujian yang akan diuji dapat di lihat pada tabel berikut ini.

Tabel 1. Pengujian *Generate* Jadwal

Deskripsi	Prosedur pengujian	Masukkan	Keluaran yang diharapkan	Hasil	Kesimpulan
Pengujian menu halaman <i>generate</i> jadwal	1. Klik menu <i>generate</i> jadwal 2. masukkan inputan yang diminta pada halaman <i>generate</i> jadwal	Aturan waktu, jumlah populasi, laju mutasi, iterasi	Mucul nama pegawai beserta waktu shift	Muncul halaman nama pegawai beserta waktu shift	Benar

## SIMPULAN

Berdasarkan hasil dan pembahasan pada penelitian yang telah dilakukan, maka dapat disimpulkan bahwa pengujian dilakukan sistem ini memiliki hasil akhir berupa staf mendapatkan jadwal yang bersesuaian shiftnya yang telah *digenerate* secara *random* dengan aturan yang telah ditetapkan efektif terhadap staf yang banyak dan toko dengan lokasi yang berbeda.

Untuk sistem pergantian staf ini dapat mencoba metode lain, atau menggabungkan beberapa metode untuk mendapatkan hasil yang lebih baik lagi.

## UCAPAN TERIMA KASIH

Terimakasih kepada Politeknik Negeri Medan melalui Pusat Penelitian dan Pengabdian kepada Masyarakat (P3M) yang telah mendanai penelitian ini.

## DAFTAR PUSTAKA

- A. Desiani dan M. Arhami (2006), Konsep Kecerdasan Buatan. Yogyakarta : Andi.
- R. Stair and G. Reynolds, Principles of Information Systems (9th edition) (2010). America : Course Technology.
- Sanjoyo, Aplikasi Algoritma Genetika (2006). Jakarta: Erlangga.
- K.C. Laudon and P.L. Jane(2010), Management Information System : Managing the Digital Firm. New Jersey : Prentice-Hall.
- R.Ginting, Penjadwalan Mesin (2009). Yogyakarta :Graha ilmu.
- Suyanto, Algoritma Optimasi Deterministik atau Probabistik (2010). Yogyakarta : Graha Ilmu.
- T. Sutabri, Konsep Dasar Informasi (2012). Yogyakarta : Andi.
- U.A.Rifai, “Pengembangan Aplikasi Penjadwalan Kegiatan dengan Menggunakan Algoritma Genetika (Studi Kasus : Humas Kementerian Agama RI)” (2011), Universitas Islam Negeri Syarif Hidayatullah Jakarta.
- Febriyana, R., & Mahmudy, W. F. (2016). Pergantian Kapal Penyeberangan Menggunakan Algoritma Genetika. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 3(1), 43. <https://doi.org/10.25126/jtiik.201631169>.
- Monalisa, S., Denni Prima Putra, E., Kurnia, F92018), Studi Sistem Informasi, P., Sains dan Teknologi UIN Suska Riau, F., Studi Teknik Informatika, P., Sains dan Teknologi UIN Suska Riau Jl Soebrantas KM, F. H., & Pekanbaru -Riau, P.