

BAB 3

ANALISIS DAN PERANCANGAN SISTEM

3.1 ANALISIS SISTEM

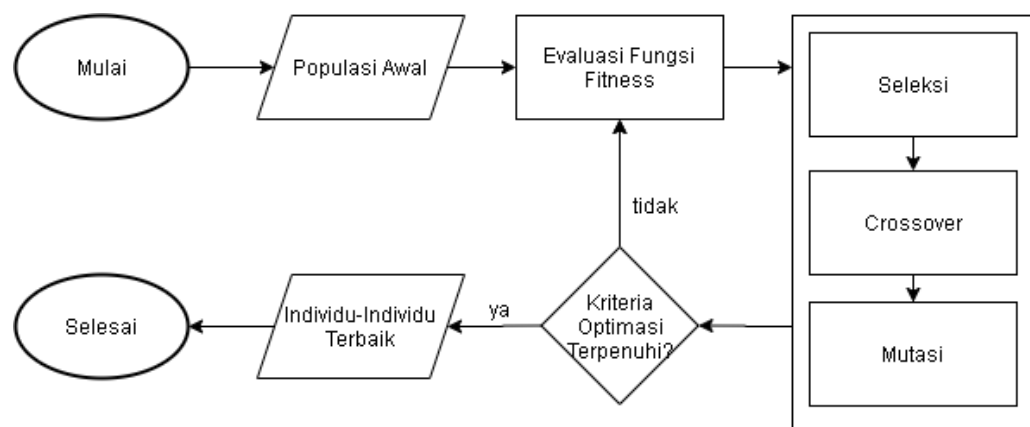
Penjadwalan ruangan di Gedung Dakwah Muhammadiyah Cabang Dukun (GDMCD) dikelola oleh staf administrasi melalui dua cara, yaitu pengajuan langsung ke kantor administrasi atau melalui pesan WhatsApp resmi. Setiap pengajuan harus menyertakan informasi berupa waktu pelaksanaan, jenis kegiatan, jumlah peserta, dan identitas lembaga pengguna. Staf administrasi akan memproses permohonan tersebut dengan memeriksa ketersediaan ruangan sesuai dengan waktu yang diajukan. Setelah permohonan diterima, staf administrasi mengevaluasi ketersediaan ruangan dengan memeriksa jadwal yang dicatat menggunakan buku catatan dan papan pengumuman.

Permasalahan yang terjadi diawali dengan pencatatan yang tidak konsisten dalam mengatur penjadwalan, seperti adanya permohonan yang telah disetujui melalui WhatsApp namun belum diperbarui dalam catatan utama. Setiap permintaan penggunaan ruangan harus mempertimbangkan berbagai faktor seperti waktu, jenis kegiatan, jumlah peserta, dan identitas lembaga peminjam. Proses ini sering menimbulkan bentrokan jadwal akibat ketidakkonsistenan prosedur administratif dan kendala komunikasi. Selain itu, penggunaan buku catatan dan papan pengumuman untuk mencatat jadwal menyulitkan pemantauan ketersediaan ruangan. Masalah utama dalam penjadwalan GDMCD yaitu belum adanya sistem penjadwalan yang mampu menentukan jadwal yang optimal dalam penggunaan ruangan, untuk memastikan efisiensi jadwal bagi pengguna. Karena jadwal yang disusun secara tidak efisien dapat menimbulkan konflik dan tumpang tindih antar pengguna. Dari permasalahan tersebut peneliti membuat Sistem Penjadwalan Ruangan dengan menggunakan metode Algoritma Genetika.

3.2 HASIL ANALISIS SISTEM

Penelitian ini menggunakan Algoritma Genetika untuk menyelesaikan masalah penjadwalan dalam proses penggunaan ruangan di Gedung Dakwah Muhammadiyah Cabang Dukun (GDMCD). Algoritma ini dipilih karena

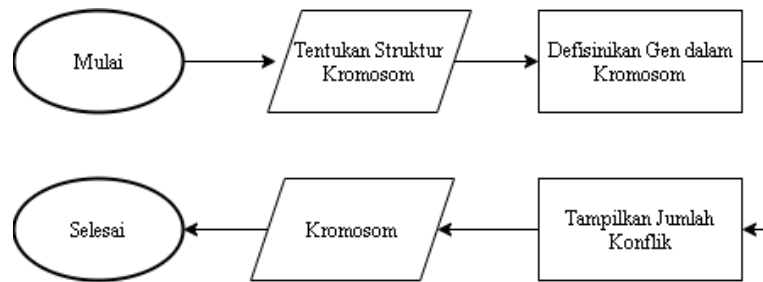
kemampuannya dalam meningkatkan efisiensi proses penjadwalan. Penerapan Algoritma Genetika dalam menentukan jadwal penggunaan ruangan di GDMCD diharapkan dapat memberikan kontribusi yang signifikan terhadap optimalisasi pemanfaatan ruangan. Keputusan menggunakan Algoritma Genetika didasarkan pada kemudahan penerapannya serta hasil optimal yang dapat dicapai. Dalam penelitian ini, dikembangkan sebuah sistem penjadwalan ruangan di GDMCD dengan menggunakan metode Algoritma Genetika.



Gambar 3.1 Diagram Alir Algoritma Genetika

Pada Gambar 3.1 menunjukan bagian yang dikurung kotak dalam diagram menunjukkan tiga operator utama dalam algoritma genetika. Seleksi adalah proses memilih individu terbaik berdasarkan nilai *Fitness* untuk dipertahankan atau direproduksi. Selanjutnya, *Crossover* atau persilangan dilakukan dengan menggabungkan sebagian gen dari dua individu induk untuk menghasilkan individu baru (keturunan). Terakhir, mutasi dilakukan dengan mengubah gen secara acak pada individu baru untuk menjaga keragaman solusi dan mencegah konvergensi prematur. Ketiga operator ini bekerja secara berurutan dalam setiap generasi untuk mencari solusi optimal.

a. Populasi Awal

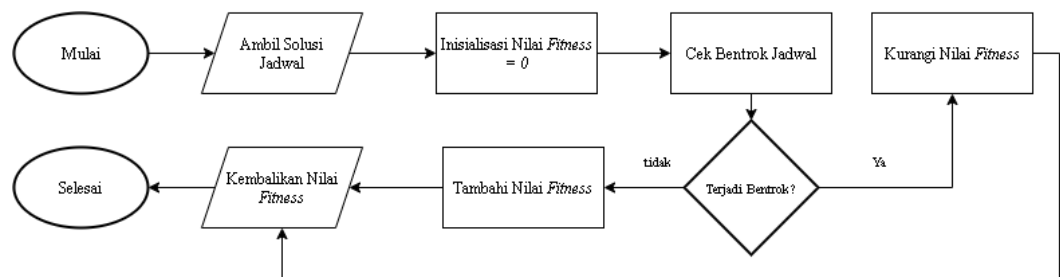


Gambar 3.2 Diagram Populasi Awal

Proses awal dari algoritma genetika dapat dilihat pada gambar 3.2. Diagram ini merupakan proses pembentukan populasi awal. Penjelasan langkah-langkahnya adalah sebagai berikut:

1. Menentukan struktur kromosom sesuai dengan permasalahan.
2. Mendefinisikan gen dalam kromosom untuk merepresentasikan elemen jadwal.
3. Memasukkan data pengguna, ruangan, dan waktu sebagai dasar pembentukan jadwal.
4. Membuat kromosom awal secara acak untuk membentuk populasi awal.

b. Fungsi *Fitness*

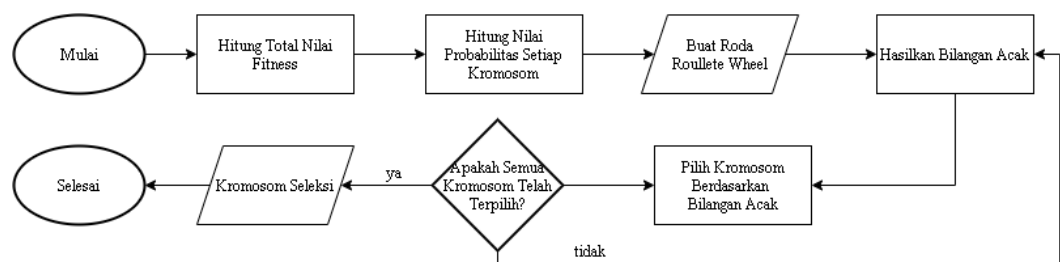


Gambar 3.3 Diagram Alir *Fitness*

Proses evaluasi *fitness* dapat dilihat pada gambar 3.3. Diagram ini menjelaskan langkah-langkah yang dilakukan untuk menghitung nilai *fitness* setiap individu dalam populasi. Tahapan prosesnya adalah sebagai berikut:

1. Mengambil solusi jadwal yang akan dievaluasi.
2. Menginisialisasi nilai *Fitness* dengan 0.
3. Memeriksa bentrok jadwal dalam solusi tersebut.
4. Mengurangi nilai *Fitness* jika terjadi bentrok.
5. Menambahkan nilai *Fitness* jika tidak terjadi bentrok.
6. Mengembalikan nilai *Fitness* akhir sebagai hasil evaluasi.

c. Seleksi

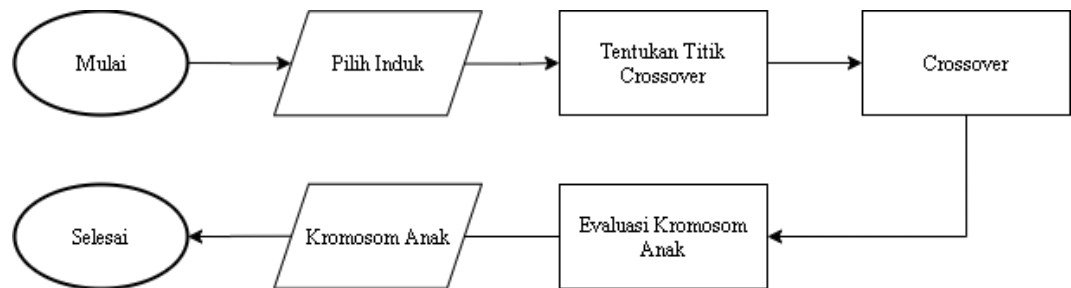


Gambar 3.4 Diagram Alir Seleksi

Proses seleksi dapat dilihat pada gambar 3.4. Diagram ini menunjukkan bagaimana proses seleksi dilakukan dalam algoritma genetika untuk memilih kromosom terbaik berdasarkan nilai *Fitness*. Penjelasan langkah-langkahnya adalah sebagai berikut:

1. Menghitung total nilai *Fitness* dari semua kromosom.
2. Menghitung probabilitas seleksi untuk setiap kromosom berdasarkan nilai *Fitness*.
3. Membuat roda *roulette* sebagai mekanisme seleksi berbasis *probabilitas*.
4. Menghasilkan bilangan acak untuk memilih kromosom.
5. Memilih kromosom berdasarkan bilangan acak hingga semua kromosom terpilih.
6. Menyimpan kromosom hasil seleksi untuk tahap mutasi berikutnya.

d. Crossover

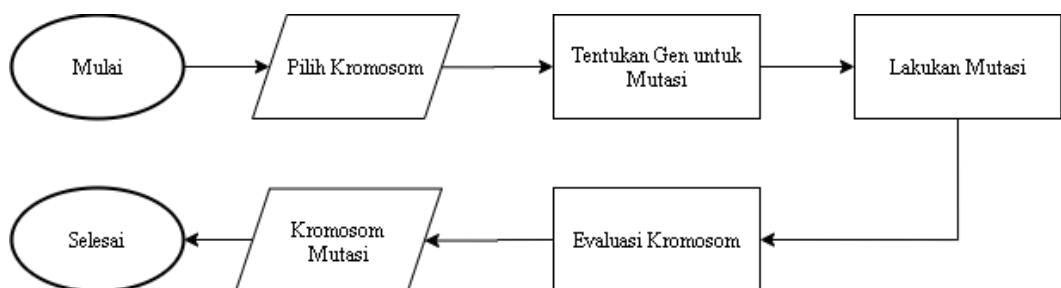


Gambar 3.5 Diagram Alir *Crossover*

Proses *Crossover* dapat dilihat pada gambar 3.5. Diagram ini menjelaskan proses pembuatan kromosom baru (anak) dari dua kromosom induk. Penjelasan langkah-langkahnya adalah sebagai berikut:

1. Memilih dua kromosom induk untuk proses persilangan.
2. Menentukan titik *Crossover* sebagai tempat penggabungan elemen kromosom.
3. Melakukan *Crossover* untuk menghasilkan kromosom anak.
4. Mengevaluasi kromosom anak untuk memastikan validitasnya.
5. Menyimpan kromosom anak ke dalam populasi baru.

e. Mutasi



Gambar 3.6 Diagram Alir Mutasi

Proses mutasi dapat dilihat pada gambar 3.6, Diagram ini menjelaskan perubahan kecil pada kromosom untuk menjaga keberagaman populasi. Penjelasan langkah-langkahnya adalah sebagai berikut:

1. Memilih kromosom dari populasi untuk proses mutasi.

2. Menentukan gen dalam kromosom yang akan dimutasi.
3. Melakukan perubahan pada gen sesuai mekanisme mutasi.
4. Mengevaluasi kromosom hasil mutasi untuk memastikan validitasnya.
5. Menyimpan kromosom hasil mutasi ke dalam populasi.
6. Menyimpan kromosom anak ke dalam populasi baru.

Berdasarkan analisis sistem yang dilakukan, asumsi-asumsi yang digunakan dalam proses penjadwalan penggunaan ruangan di GDMCD adalah sebagai berikut:

Tabel 3.1 Data Lembaga

TABEL LEMBAGA			
NO	Id_lembaga	NAMA LEMBAGA	JUMLAH PESERTA
1	MTsMTQ	MTs Maarif Tanwirul Qulub	90
2	MTsMH	MTs Mathlabul Huda	250
3	MTsM10	MTs Muhammadiyah 10	190
4	MTsIU	MTs Ihyaul Ulum	220
5	MTsTA	MTs Tarbiyatul Athfal	130
...
10	MTsM1	MTs Muhammadiyah 1	220

Nama-nama lembaga yang terlibat dalam penelitian, beserta jumlah peserta dari masing-masing lembaga, dapat dilihat pada tabel 3.1. Data ini digunakan untuk mengelompokkan peserta berdasarkan lembaga asalnya. Pengelompokan ini menjadi bagian dari variabel GEN 1 dalam proses algoritma genetika.

Tabel 3.2 Data Kegiatan

TABEL KEGIATAN		
NO	id_kegiatan	KEGIATAN
1	kg-01	Pusat Dakwah
2	kg-02	Pendidikan
3	kg-03	Sosial
4	kg-04	Rapat dan Koordinasi
5	kg-05	Umum/Non Lembaga

Nama-nama kegiatan yang terlibat dalam penelitian, dapat dilihat pada tabel 3.2. Data ini digunakan untuk mengelompokkan kegiatan berdasarkan kategori atau jenisnya. Pengelompokan ini menjadi bagian dari variabel GEN 1 dalam proses algoritma genetika.

Tabel 3.3 Data Hari

TABEL HARI		
NO	id_hari	HARI
1	h1	SENIN
2	h2	SELASA
3	h3	RABU
4	h4	KAMIS
5	h5	JUM'AT

Informasi mengenai hari-hari kegiatan yang terlibat dalam penelitian, dapat dilihat pada tabel 3.2. Data ini digunakan untuk mengelompokkan waktu berdasarkan hari, yang menjadi bagian dari variabel GEN 3 dalam proses algoritma genetika.

Tabel 3.4 Data Jam

TABEL JAM		
NO	id_jam	JAM
1	1	08:00 - 12:00
2	2	13:00 - 17:00

Informasi mengenai dua sesi waktu dalam satu hari, yang digunakan dalam penelitian, dapat dilihat pada tabel 3.4. Data ini digunakan untuk mengelompokkan waktu ke dalam sesi tertentu, yang juga menjadi bagian dari variabel GEN 3 dalam proses algoritma genetika.

Tabel 3.5 Data Ruangan

TABEL RUANGAN			
NO	KODE RUANGAN	RUANGAN	KAPASITAS
1	r1	Lantai 1 - Ruang 1	50
2	r2	Lantai 1 - Ruang 2	100

3	r3	Lantai 1 - Ruang 3	150
4	r4	Lantai 1 - Ruang 4	250
5	r5	Lantai 1 - Ruang 5	400

Daftar ruangan yang tersedia untuk kegiatan, lengkap dengan informasi kapasitas dapat dilihat pada tabel 3.5. Data ini menjadi bagian dari variabel GEN 2 dalam proses algoritma genetika.

Tabel 3.6 Data Kelompok Pengguna

TABEL PENGGUNA (GEN 1)			
NO	KEGIATAN	LEMBAGA	PESERTA
1	kg-01	MTsMTQ	90
2	kg-02	MTsMH	250
3	kg-03	MTsM10	190
4	kg-04	MTsIU	220
5	kg-05	MTsTA	130
...
10	kg-05	MTsMI	220

Pada tabel 3.6, dapat dilihat terdapat satu entitas pengguna yang terdiri atas lembaga dengan jumlah peserta spesifik yang terlibat dalam kegiatan tertentu. Variabel ini menjadi bagian dari pengelompokan GEN 1 dalam proses algoritma genetika.

Tabel 3.7 Data Kelompok Ruangan

TABEL RUANGAN (GEN 2)		
NO	RUANGAN	KAPASITAS
1	r1	50
2	r2	100
3	r3	150
4	r4	250
5	r5	400

Daftar ruangan yang tersedia untuk kegiatan, dengan informasi tambahan kapasitas dan kode identifikasi ruangan dapat dilihat pada tabel 3.7.

Variabel ini tetap menjadi bagian penting dalam pengelompokan GEN 2 dalam proses algoritma genetika.

Tabel 3.8 Data Kelompok Waktu

TABEL WAKTU (GEN 3)		
NO	HARI	JAM
1	SENIN	1
2	SENIN	2
3	SELASA	1
4	SELASA	2
5	RABU	1
6	RABU	2
7	KAMIS	1
8	KAMIS	2
9	JUM'AT	1
10	JUM'AT	2

Kombinasi hari dan sesi waktu yang tersedia untuk pelaksanaan kegiatan dapat dilihat pada tabel 3.8. Data ini menjadi bagian dari variabel GEN 3 dalam proses algoritma genetika.

Tabel 3.9 Matriks Populasi Awal

Populasi Awal dan Hasil Matrik					
Kromosom	idv 1	idv 2	idv 3	idv 4	idv 5
k1	[1, 3, 8]	[1, 5, 5]	[1, 5, 7]	[1, 4, 2]	[1, 3, 7]
k2	[2, 2, 4]	[2, 4, 10]	[2, 3, 6]	[2, 2, 5]	[2, 1, 4]
k3	[3, 4, 2]	[3, 4, 3]	[3, 4, 4]	[3, 1, 1]	[3, 4, 3]
k4	[4, 5, 3]	[4, 3, 6]	[4, 2, 3]	[4, 2, 3]	[4, 2, 2]
k5	[5, 4, 2]	[5, 5, 2]	[5, 4, 7]	[5, 4, 4]	[5, 3, 1]
k6	[6, 2, 3]	[6, 3, 2]	[6, 2, 5]	[6, 2, 7]	[6, 2, 10]
k7	[7, 5, 6]	[7, 3, 1]	[7, 3, 9]	[7, 5, 10]	[7, 3, 3]
k8	[8, 3, 4]	[8, 3, 5]	[8, 1, 4]	[8, 3, 4]	[8, 4, 1]
k9	[9, 3, 1]	[9, 2, 6]	[9, 4, 10]	[9, 2, 1]	[9, 5, 2]
k10	[10, 1, 9]	[10, 4, 8]	[10, 1, 1]	[10, 1, 1]	[10, 3, 4]

Matriks populasi awal dalam algoritma genetika ini terdiri dari 10 kromosom (k1-k10) dan 5 individu (idv 1-5) seperti pada tabel 3.9, dimana

setiap sel berisi kromosom dengan format [GEN1, GEN2, GEN3] yang merepresentasikan pengguna_id (1-10), ruangan_id (1-5), dan waktu_id (1-10). Pembangkitan nilai dilakukan secara acak dengan GEN1 tetap berurutan 1-10 untuk setiap kromosom, sementara GEN2 dan GEN3 dibatasi sesuai jumlah ruangan dan slot waktu yang tersedia. Sebagai contoh, kromosom k1 pada idv 1 [1,3,8] menunjukkan pengguna 1 mendapat ruangan 3 pada waktu 8, dan kromosom k4 pada idv 3 [4,2,3] berarti pengguna 4 mendapat ruangan 2 pada waktu 3. Total terdapat 50 kombinasi jadwal yang menyediakan beragam solusi awal untuk proses evolusi, membentuk basis untuk operasi genetika selanjutnya, dan memungkinkan eksplorasi berbagai kombinasi jadwal yang akan diproses melalui tahapan algoritma genetika (seleksi, crossover, mutasi) untuk menghasilkan solusi penjadwalan yang optimal.

3.3 REPRESENTASI MODEL

Algoritma genetika terdiri dari empat proses utama. Pertama, inisialisasi awal digunakan menemukan solusi awal untuk masalah. Kemudian, proses reproduksi yang mencakup *mutation* dan *Crossover*, menghasilkan keturunan baru. Selanjutnya, evaluasi dilakukan untuk mengukur nilai kecocokan setiap Kromosom dalam populasi. Terakhir, seleksi digunakan untuk memilih Kromosom baru berdasarkan nilai kecocokan, dengan memprioritaskan mereka yang memiliki nilai kecocokan yang paling tinggi.

a. Inisialisasi Parameter Algoritma Genetika

Langkah awal algoritme genetika, kita menentukan parameter-parameter seperti jumlah kromosom, jumlah individu, jumlah generasi, *crossover rate*, dan mutasi *rate*. Parameter yang digunakan dalam penelitian ini untuk menyelesaikan masalah dengan algoritma genetika adalah:

- Jumlah Kromosom : 10
- Jumlah Individu : 5
- Jumlah Generasi : 1
- *Crossover Rate* : 0,75

- *Mutation Rate* : 0,20

b. Mendefinisikan Gen

Gen dalam konteks ini merupakan gabungan dari beberapa variabel yang merepresentasikan penggunaan ruangan yang diajukan melalui sistem, di mana Gen 1, Gen 2, dan Gen 3 memiliki peran tersendiri. Representasi kelompok gen berdasarkan **Tabel 3.6, Tabel 3.7, dan Tabel 3.8** adalah sebagai berikut:

Gen 1 = Kelompok Pengguna [Kegiatan, Lembaga, dan Jumlah Peserta]
 Gen 2 = Kelompok Ruangan [Ruangan dan Kapasitas]
 Gen 3 = Kelompok Waktu [Hari dan Jam]

Pada satu kromosom, terdapat tiga gen yang terdiri dari 1 hingga 3. Gen-gen ini membentuk suatu urutan jadwal yang tidak teratur. Proses optimasi dilakukan untuk menemukan susunan jadwal yang paling optimal.

c. Membangkitkan Populasi Awal

Pada penelitian ini, diasumsikan bahwa setiap populasi awal akan terdiri dari 10 kromosom yang berbeda. Pada tahap ini, populasi awal dibuat dengan mengacak gen2 dan gen3 untuk menciptakan variasi pada populasi awal. Agar mempermudah proses implementasi algoritma genetika, peneliti menggunakan data dengan skala kecil sebagai acuan. Berikut adalah tabel yang menampilkan proses inisialisasi populasi:

Tabel 3.10 Individu Ke-1 Pembentukan Populasi Awal

Kro	Gen 1			Gen 2		Gen 3	
	Nama Kegiatan	Lembaga	Jumlah Peserta	Ruangan	Kapasitas	Hari	Jam
1	KG-01	MTsMTQ	90	r3	150	KAMIS	2
2	KG-02	MTsMH	250	r2	100	SELASA	2
3	KG-03	MTsM10	190	r4	200	SENIN	2
4	KG-04	MTsIU	220	r5	400	SELASA	1
5	KG-05	MTsTA	130	r4	200	SENIN	2

Kro	Gen 1			Gen 2		Gen 3	
	Nama Kegiatan	Lembaga	Jumlah Peserta	Ruangan	Kapasitas	Hari	Jam
6	KG-01	MTsNR	30	r2	100	SELASA	1
7	KG-02	MTsNH	90	r5	400	RABU	2
8	KG-03	MTsYKUISG	80	r3	150	SELASA	2
9	KG-04	MTsAK1	40	r3	150	SENIN	1
10	KG-05	MTsM1	220	r1	50	JUM'AT	1

Keterangan Pinalti: Lembaga Menggunakan Ruangan yang Tidak Sesuai Kapasitasnya dan Ruang Konflik

Tabel 3.11 Individu Ke-2 Pembentukan Populasi Awal

Kro	Gen 1			Gen 2		Gen 3	
	Nama Kegiatan	Lembaga	Jumlah Peserta	Ruangan	Kapasitas	Hari	Jam
1	KG-01	MTsMTQ	90	r5	400	RABU	1
2	KG-02	MTsMH	250	r4	250	JUM'AT	2
3	KG-03	MTsM10	190	r4	200	SELASA	1
4	KG-04	MTsIU	220	r3	150	RABU	2
5	KG-05	MTsTA	130	r5	400	SENIN	2
6	KG-01	MTsNR	30	r3	150	SENIN	2
7	KG-02	MTsNH	90	r3	150	SENIN	1
8	KG-03	MTsYKUISG	80	r3	150	RABU	1
9	KG-04	MTsAK1	40	r2	100	RABU	2
10	KG-05	MTsM1	220	r4	200	KAMIS	2

Keterangan Pinalti: Lembaga Menggunakan Ruangan yang Tidak Sesuai Kapasitasnya

Tabel 3.12 Individu Ke-3 Pembentukan Populasi Awal

Kro	Gen 1			Gen 2		Gen 3	
	Nama Kegiatan	Lembaga	Jumlah Peserta	Ruangan	Kapasitas	Hari	Jam
1	KG-01	MTsMTQ	90	r5	400	KAMIS	1
2	KG-02	MTsMH	250	r3	150	RABU	2
3	KG-03	MTsM10	190	r4	200	SELASA	2
4	KG-04	MTsIU	220	r2	100	SELASA	1
5	KG-05	MTsTA	130	r4	200	KAMIS	1
6	KG-01	MTsNR	30	r2	100	RABU	1
7	KG-02	MTsNH	90	r3	150	JUM'AT	1
8	KG-03	MTsYKUISG	80	r1	50	SELASA	2
9	KG-04	MTsAK1	40	r4	200	JUM'AT	2

Kro	Gen 1			Gen 2		Gen 3	
	Nama Kegiatan	Lembaga	Jumlah Peserta	Ruangan	Kapasitas	Hari	Jam
10	KG-05	MTsM1	220	r1	50	SENIN	1

Keterangan Pinalti: Lembaga Menggunakan Ruangan yang Tidak Sesuai Kapasitasnya

Tabel 3.13 Individu Ke-4 Pembangkitan Populasi Awal

Kro	Gen 1			Gen 2		Gen 3	
	Nama Kegiatan	Lembaga	Jumlah Peserta	Ruangan	Kapasitas	Hari	Jam
1	KG-01	MTsMTQ	90	r4	200	SENIN	2
2	KG-02	MTsMH	250	r2	100	RABU	1
3	KG-03	MTsM10	190	r1	50	SENIN	1
4	KG-04	MTsIU	220	r2	100	SELASA	1
5	KG-05	MTsTA	130	r4	200	SELASA	2
6	KG-01	MTsNR	30	r2	100	KAMIS	1
7	KG-02	MTsNH	90	r5	400	JUM'AT	2
8	KG-03	MTsYKUISG	80	r3	150	SELASA	2
9	KG-04	MTsAK1	40	r2	100	SENIN	1
10	KG-05	MTsM1	220	r1	50	SENIN	1

Keterangan Pinalti: Lembaga Menggunakan Ruangan yang Tidak Sesuai Kapasitasnya dan Ruang Konflik

Tabel 3.14 Individu Ke-5 Pembangkitan Populasi Awal

Kro 5	Gen 1			Gen 2		Gen 3	
	Nama Kegiatan	Lembaga	Jumlah Peserta	Ruangan	Kapasitas	Hari	Jam
1	KG-01	MTsMTQ	90	r3	150	KAMIS	1
2	KG-02	MTsMH	250	r1	50	SELASA	2
3	KG-03	MTsM10	190	r4	250	SELASA	1
4	KG-04	MTsIU	220	r2	100	SENIN	2
5	KG-05	MTsTA	130	r3	150	SENIN	1
6	KG-01	MTsNR	30	r2	100	JUM'AT	2
7	KG-02	MTsNH	90	r3	150	SELASA	1
8	KG-03	MTsYKUISG	80	r4	250	SENIN	1
9	KG-04	MTsAK1	40	r5	400	SENIN	2
10	KG-05	MTsM1	220	r3	150	SELASA	2

Keterangan Pinalti: Lembaga Menggunakan Ruangan yang Tidak Sesuai Kapasitasnya

d. Menentukan Nilai *Fitness*

Dalam penelitian ini, nilai *Fitness* dihitung berdasarkan tingkat bentrok atau konflik pada setiap kromosom dalam individu. Bentrok dihitung dengan mempertimbangkan batasan yang telah ditentukan sebelumnya pada bagian 3.2. Jika batasan-batasan tersebut terpenuhi, maka nilai konflik pada individu akan ditambah 1. Setelah melakukan pengecekan terhadap pelanggaran pada setiap gen dalam kromosom, hasil pelanggaran tersebut akan digunakan untuk menghitung nilai *Fitness* guna mencari tau seberapa optimal solusi yang dihasilkan dari individu yang ada. Perhitungan nilai *Fitness* dapat dihitung dengan persamaan 2.1 berikut:

$$F = \frac{1}{1+(KR+KK)}$$

Keterangan:

$F = Fitness$

KR = Konflik Ruang

KK = Konflik Kapasitas

Berikut perhitungan nilai *Fitness* berdasarkan konflik yang telah didapatkan pada populasi awal:

1	$F \text{ Individu 1} = \frac{1}{1 + (1 + 2)} =$	0.250
2	$F \text{ Individu 2} = \frac{1}{1 + (0 + 2)} =$	0.333
3	$F \text{ Individu 3} = \frac{1}{1 + (0 + 4)} =$	0.200
4	$F \text{ Individu 4} = \frac{1}{1 + (1 + 4)} =$	0.167
5	$F \text{ Individu 5} = \frac{1}{1 + (0 + 3)} =$	0.250
<i>Total Nilai fitness =</i>		1.200

e. Proses Seleksi

Metode seleksi yang digunakan pada penelitian ini adalah metode seleksi *roulette wheel*. Pada metode ini, setiap individu akan mendapatkan

proporsi dalam lingkaran berdasarkan nilai *Fitness*-nya (Elva 2019). Proses seleksi pada *roulette wheel* dibagi menjadi beberapa proses yaitu:

a. Menghitung Nilai Probabilitas

Nilai *Fitness* dari setiap individu akan dibagi dengan total nilai *Fitness* yang telah dihitung sebelumnya. Proses perhitungan nilai probabilitas ini dapat dilihat pada rumus persamaan 2.2 berikut:

Tabel 3.15 Hitung Nilai Probabilitas

Kromosom	Probabilitas	Hasil Probabilitas
1	0.250/1.200	0.208
2	0.333/1.200	0.278
3	0.200/1.200	0.167
4	0.167/1.200	0.139
5	0.250/1.200	0.208

Pada tabel 3.15 Menjelaskan nilai probabilitas dihitung dengan membagi nilai *Fitness* setiap kromosom dengan total nilai *Fitness*. Hasil *probabilitas* ini akan digunakan dalam proses seleksi *roulette-wheel* selanjutnya.

b. Menghitung Nilai Komulatif

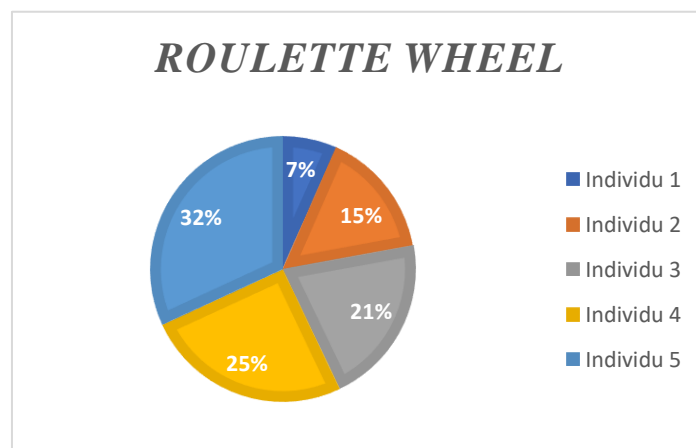
Pada langkah ini, hasil nilai probabilitas dari setiap individu akan dijumlahkan dengan nilai inisialisasi pada penjumlahan kumulatif sebelumnya.

Tabel 3.16 Hitung Nilai Komulatif

Kromosom	Kumulatif	Hasil Komulatif
1	0+0.208	0.208
2	0.208+0.278	0.486
3	0.486+0.167	0.653
4	0.653+0.139	0.792
5	0.792+0.208	1.000

Pada tabel 3.16 dilakukan perhitungan kumulatif dengan menjumlahkan nilai *probabilitas* dari individu sebelumnya. Misalnya, individu 1 memiliki nilai *probabilitas* 0.208, maka nilai kumulatifnya menjadi $0 + 0.208 = 0.208$. Kemudian, Individu 2 memiliki nilai probabilitas 0.288, sehingga nilai kumulatifnya menjadi $0.208 + 0.278 = 0,486$. Proses ini dilakukan secara berkelanjutan hingga mencapai individu terakhir, di mana nilai kumulatifnya mencapai 1.

Berdasarkan nilai kumulatif yang telah dihasilkan, dapat dilakukan pemetaan untuk setiap individu. Setelah melakukan pemetaan untuk individu, langkah selanjutnya adalah melakukan proses seleksi individu. Langkah pertama dalam proses seleksi adalah menghasilkan bilangan acak antara 0 hingga 1 sebanyak jumlah individu pada populasi. Diasumsikan, nilai acak yang dihasilkan adalah 0,49, 0,26, 0,50, 0,96 dan 0,79.



Gambar 3.7 *Probabilitas Kumulatif dalam Roulette wheel*

Pada gambar 3.7 dapat dilihat bahwa individu yang memiliki nilai 0,49 adalah individu 3, dengan interval nilai antara 0,48 hingga 0,65. Oleh karena itu, individu 3 akan menempati posisi individu 1. Selanjutnya, individu yang memiliki nilai 0,26 adalah individu 2 dengan *interval* nilai antara 0,28 hingga 0,48. Maka, individu 2 akan menempati posisi individu 2, proses ini akan terus berlangsung hingga semua bilangan acak yang dibangkitkan telah diletakan pada posisinya masing-masing. Berikut susunan populasi baru hasil dari proses seleksi:

Tabel 3.17 Populasi Baru Hasil Seleksi

Hasil <i>Roulette wheel</i>		
NO	KETERANGAN	HASIL
1	individu [1]	individu [3]
2	individu [2]	individu [2]
3	individu [3]	individu [3]
4	individu [4]	individu [5]
5	individu [5]	individu [5]

Hasil seleksi populasi dengan menggunakan metode *Roulette Wheel* dapat dilihat pada tabel 3.17 ini menampilkan hasil seleksi populasi menggunakan metode *Roulette wheel*. Setiap individu dalam populasi awal dinilai berdasarkan nilai *Fitness*, dan beberapa individu dipilih untuk membentuk populasi baru berdasarkan probabilitas seleksi.

Tabel 3.18 Populasi Baru dengan Induk

Hasil Seleksi					
Kromosom	idv 1	idv 2	idv 3	idv 4	idv 5
k1	[1, 5, 7]	[1, 5, 5]	[1, 5, 7]	[1, 3, 7]	[1, 3, 7]
k2	[2, 3, 6]	[2, 4, 10]	[2, 3, 6]	[2, 1, 4]	[2, 1, 4]
k3	[3, 4, 4]	[3, 4, 3]	[3, 4, 4]	[3, 4, 3]	[3, 4, 3]
k4	[4, 2, 3]	[4, 3, 6]	[4, 2, 3]	[4, 2, 2]	[4, 2, 2]
k5	[5, 4, 7]	[5, 5, 2]	[5, 4, 7]	[5, 3, 1]	[5, 3, 1]
k6	[6, 2, 5]	[6, 3, 2]	[6, 2, 5]	[6, 2, 10]	[6, 2, 10]
k7	[7, 3, 9]	[7, 3, 1]	[7, 3, 9]	[7, 3, 3]	[7, 3, 3]
k8	[8, 1, 4]	[8, 3, 5]	[8, 1, 4]	[8, 4, 1]	[8, 4, 1]
k9	[9, 4, 10]	[9, 2, 6]	[9, 4, 10]	[9, 5, 2]	[9, 5, 2]
k10	[10, 1, 1]	[10, 4, 8]	[10, 1, 1]	[10, 3, 4]	[10, 3, 4]

Berdasarkan hasil pada tabel 3.18, individu 1 dan 4 tidak terpilih dalam proses seleksi karena tidak masuk ke dalam bilangan acak yang dihasilkan oleh metode *roulette wheel*. Populasi baru hasil seleksi ini kemudian akan digunakan sebagai input pada tahap *Crossover* untuk menghasilkan solusi yang lebih optimal.

c. Proses Crossover

Langkah awal dalam proses *Crossover* adalah menentukan tingkat nilai probabilitas *Crossover*. Dalam kasus ini, nilai *Crossover rate* ialah 0,75. Setelah nilai *Crossover rate* ditentukan langkah selanjutnya ialah menentukan bilangan acak sesuai dengan jumlah individu yang ada. Berikut bilangan acak yang telah di bangkitkan secara acak 0,08, 0,27, 0,76, 0,55, dan 0,62.

Proses *Crossover* hanya dilakukan jika nilai bilangan acak yang dihasilkan individu lebih kecil daripada nilai *Crossover rate* yang telah ditetapkan. Berdasarkan penyesuaian antara nilai *Crossover rate* dan bilangan acak, didapatkan bahwa kromosom yang akan mengalami proses *Crossover* adalah kromosom 1,2,4,5.

Setelah mendapatkan individu yang akan melakukan proses *Crossover* langkah selanjutnya ialah menentukan titik potong pada setiap proses *Crossover* secara acak. Bilangan acak yang digunakan untuk menentukan titik potong adalah 1 sampai dengan jumlah panjang kromosom dikurangi 1, berdasarkan aturan tersebut didapatkan *range* atau jarak yang akan digunakan ialah 1 sampai dengan 9.

Kemudian bilangan acak tersebut akan dibangkitkan sebanyak tingkat *Crossover* yang akan dilakukan. Dalam hal ini bilangan acak yang digunakan untuk pemotongan ialah 5,8,2,4. Berikut adalah proses *Crossover*:

Tabel 3.19 Proses *Crossover* 1

Proses <i>Crossover</i> 1			
Kromosom	idv 1	idv 2	<i>offspring</i> 1
k1	[1, 5, 7]	[1, 5, 5]	[1, 5, 7]
k2	[2, 3, 6]	[2, 4, 10]	[2, 3, 6]
k3	[3, 4, 4]	[3, 4, 3]	[3, 4, 4]
k4	[4, 2, 3]	[4, 3, 6]	[4, 2, 3]
k5	[5, 4, 7]	[5, 5, 2]	[5, 4, 7]
k6	[6, 2, 5]	[6, 3, 2]	[6, 3, 2]
k7	[7, 3, 9]	[6, 3, 2]	[6, 3, 2]
k8	[8, 1, 4]	[8, 3, 5]	[8, 3, 5]

k9	[9, 4, 10]	[9, 2, 6]	[9, 2, 6]
k10	[10, 1, 1]	[10, 4, 8]	[10, 4, 8]

Tabel 3.20 Proses *Crossover 2*

Proses <i>Crossover 2</i>			
Kromosom	idv 2	idv 4	offspring 2
k1	[1, 5, 5]	[1, 3, 7]	[1, 5, 5]
k2	[2, 4, 10]	[2, 1, 4]	[2, 4, 10]
k3	[3, 4, 3]	[3, 4, 3]	[3, 4, 3]
k4	[4, 3, 6]	[4, 2, 2]	[4, 3, 6]
k5	[5, 5, 2]	[5, 3, 1]	[5, 5, 2]
k6	[6, 3, 2]	[6, 2, 10]	[6, 3, 2]
k7	[7, 3, 1]	[7, 3, 3]	[7, 3, 1]
k8	[8, 3, 5]	[8, 4, 1]	[8, 3, 5]
k9	[9, 2, 6]	[9, 5, 2]	[9, 5, 2]
k10	[10, 4, 8]	[10, 3, 4]	[10, 3, 4]

Tabel 3.21 Proses *Crossover 3*

Proses <i>Crossover 3</i>			
Kromosom	idv 4	idv 5	offspring 3
k1	[1, 3, 7]	[1, 3, 7]	[1, 3, 7]
k2	[2, 1, 4]	[2, 1, 4]	[2, 1, 4]
k3	[3, 4, 3]	[3, 4, 3]	[3, 4, 3]
k4	[4, 2, 2]	[4, 2, 2]	[4, 2, 2]
k5	[5, 3, 1]	[5, 3, 1]	[5, 3, 1]
k6	[6, 2, 10]	[6, 2, 10]	[6, 2, 10]
k7	[7, 3, 3]	[7, 3, 3]	[7, 3, 3]
k8	[8, 4, 1]	[8, 4, 1]	[8, 4, 1]
k9	[9, 5, 2]	[9, 5, 2]	[9, 5, 2]
k10	[10, 3, 4]	[10, 3, 4]	[10, 3, 4]

Tabel 3.22 Proses *Crossover 4*

Proses <i>Crossover 4</i>			
Kromosom	idv 5	idv 1	offspring 4
k1	[1, 3, 7]	[1, 5, 7]	[1, 3, 7]
k2	[2, 1, 4]	[2, 3, 6]	[2, 1, 4]
k3	[3, 4, 3]	[3, 4, 4]	[3, 4, 3]
k4	[4, 2, 2]	[4, 2, 3]	[4, 2, 2]

k5	[5, 4, 7]	[5, 4, 7]	[5, 4, 7]
k6	[5, 3, 1]	[6, 2, 5]	[6, 2, 5]
k7	[7, 3, 3]	[7, 3, 9]	[7, 3, 9]
k8	[8, 4, 1]	[8, 1, 4]	[8, 1, 4]
k9	[9, 5, 2]	[9, 4, 10]	[9, 4, 10]
k10	[10, 3, 4]	[10, 1, 1]	[10, 1, 1]

Dapat dilihat pada tabel 3.19, proses *Crossover* 1 dilakukan dengan mengawinkan dua individu induk menggunakan titik potong yang telah ditentukan untuk menghasilkan individu baru. Pada tabel tersebut, individu 1 yang terdiri dari urutan [1, 5, 7], [2, 3, 6], [3, 4, 4], [4, 2, 3], dan [5, 4, 7] (ditandai dengan warna hijau) dijadikan induk 1, sementara individu 2 yang terdiri dari urutan [6, 3, 2], [6, 3, 2], [8, 3, 5], [9, 2, 6], dan [10, 4, 8] (ditandai dengan warna *orange*) dijadikan induk 2. Proses *Crossover* dilakukan dengan menukar kromosom pada titik potong yang telah ditentukan, menghasilkan individu anak yang terdiri dari gabungan kromosom dari kedua induk, seperti yang ditunjukkan dalam hasil kawin silang: [1, 5, 7], [2, 3, 6], [3, 4, 4], [4, 2, 3], [5, 4, 7], [6, 3, 2], [6, 3, 2], [8, 3, 5], [9, 2, 6], dan [10, 4, 8]. Proses *Crossover* ini berlanjut hingga semua individu induk menjalani kawin silang dan menghasilkan individu anak. Setelah itu, nilai fitness dari individu anak dihitung untuk mengevaluasi apakah solusi yang dihasilkan dari proses *Crossover* memberikan dampak positif pada populasi yang ada dan memenuhi kriteria penjadwalan. Berikut perhitungan nilai *Fitness* individu anak:

$$1 \quad F \text{ individu anak 1} = \frac{1}{1 + (0 + 3)} = 0.250$$

$$2 \quad F \text{ individu anak 2} = \frac{1}{1 + (0 + 2)} = 0.333$$

$$3 \quad F \text{ individu anak 3} = \frac{1}{1 + (2 + 2)} = 0.200$$

$$4 \quad F \text{ individu anak 4} = \frac{1}{1 + (0 + 3)} = 0.250$$

$$\text{Total Nilai fitness} = 1.033$$

Berdasarkan perhitungan nilai *Fitness*, dihasilkan nilai *Fitness* tertinggi adalah 0,333, dari hasil ini populasi yang dihasilkan melalui melalui proses *Crossover* belum dapat dikatakan optimal. Oleh Karena itu, diperlukan langkah selanjutnya yaitu proses mutasi.

d. Mutasi

Mutasi bertujuan untuk mengubah gen gen yang ada dalam kromosom, dengan potensi menghasilkan peningkatan atau penurunan nilai *Fitness* pada individu. Pada proses mutasi, beberapa gen dipilih secara acak untuk diubah, terutama pada gen gen kelompok ruangan dan waktu. Tingkat *probabilitas* mutasi yang digunakan dalam penelitian ini diasumsikan sebesar 20%. Untuk menentukan posisi gen yang akan dimutasi, jumlah total gen dalam populasi dihitung dengan mengalikan jumlah kromosom dalam setiap individu dalam populasi. Dalam kasus ini, terdapat 5 individu yang terdiri dari 10 kromosom, sehingga total kromosom dalam populasi adalah 50. Setelah mengetahui total kromosom dalam populasi, langkah selanjutnya adalah mengalikan jumlah total kromosom dengan tingkat *probabilitas* mutasi. Dalam hal ini, tingkat mutasi adalah 20% atau 0,20. Sehingga, hasil perkalian tersebut adalah $0,20 \times 50 = 10$. Langkah berikutnya adalah memilih bilangan acak sebanyak 10 kali dari rentang 1 hingga 50. Contoh bilangan acak yang didapatkan adalah 2, 45, 38, 5, 46, 16, 10, 32, 11, dan 26. Proses penukaran gen dapat dilihat pada tabel 3.23 dan 3.24.

Tabel 3.23 Sebelum Mutasi

Sebelum Mutasi					
Kromosom	anak 1	anak 2	induk 3	anak 4	anak 5
k1	[1, 5, 7]	[1, 5, 5]	[1, 5, 7]	[1, 3, 7]	[1, 3, 7]
k2	[2, 3, 6]	[2, 3, 6]	[2, 4, 10]	[2, 1, 4]	[2, 1, 4]
k3	[3, 4, 4]	[3, 4, 4]	[3, 4, 3]	[3, 4, 3]	[3, 4, 3]
k4	[4, 2, 3]	[4, 2, 3]	[4, 3, 6]	[4, 2, 2]	[4, 2, 2]
k5	[5, 4, 7]	[5, 4, 7]	[5, 5, 2]	[5, 3, 1]	[5, 4, 7]
k6	[6, 2, 5]	[6, 3, 2]	[6, 3, 2]	[6, 2, 10]	[6, 2, 5]
k7	[7, 3, 9]	[7, 3, 1]	[7, 3, 1]	[7, 3, 3]	[7, 3, 9]

k8	[8, 1, 4]	[8, 3, 5]	[8, 3, 5]	[8, 4, 1]	[8, 4, 1]
k9	[9, 4, 10]	[9, 2, 6]	[9, 5, 2]	[9, 5, 2]	[9, 4, 10]
k10	[10, 1, 1]	[10, 4, 8]	[10, 3, 4]	[10, 3, 4]	[10, 1, 1]

Pada tabel 3.23 menunjukkan populasi sebelum proses mutasi. Gen-gen yang akan mengalami mutasi telah ditandai dengan warna oranye berdasarkan bilangan acak yang dihasilkan. Adapun bilangan acak yang digunakan untuk proses mutasi pada gen adalah [3,9], [2,3], [3,9], [5,10], [3,7], [1,2], [2,2], [3,2], [2,6], dan [5,4].

Tabel 3.24 Sesudah Mutasi

Sesudah Mutasi					
Kromosom	anak 1	anak 2	induk 3	anak 4	anak 5
k1	[1, 5, 7]	[2, 3, 6]	[1, 5, 7]	[1, 3, 7]	[1, 3, 7]
k2	[2, 3, 9]	[2, 3, 6]	[2, 1, 2]	[2, 1, 4]	[2, 1, 4]
k3	[3, 4, 4]	[4, 2, 3]	[3, 4, 3]	[3, 4, 3]	[3, 4, 3]
k4	[4, 2, 3]	[4, 2, 3]	[4, 3, 6]	[4, 2, 2]	[4, 2, 2]
k5	[5, 2, 3]	[5, 4, 7]	[5, 5, 2]	[5, 3, 1]	[5, 2, 6]
k6	[6, 2, 5]	[6, 3, 7]	[6, 2, 2]	[6, 2, 10]	[6, 5, 4]
k7	[7, 3, 9]	[7, 3, 1]	[7, 3, 1]	[7, 3, 3]	[7, 3, 9]
k8	[8, 1, 4]	[8, 3, 5]	[8, 3, 5]	[8, 3, 2]	[8, 4, 1]
k9	[9, 4, 10]	[9, 2, 6]	[9, 5, 2]	[9, 5, 2]	[9, 4, 10]
k10	[10, 3, 9]	[10, 4, 8]	[10, 3, 4]	[10, 3, 4]	[10, 1, 1]

Berdasarkan tabel 3.24, terlihat bahwa setiap individu masih memiliki konflik. Pada anak 1 terdapat 5 konflik, anak 2 memiliki 2 konflik kapasitas; induk 3 memiliki 5 konflik kapasitas, anak 4 memiliki 3 konflik kapasitas, dan anak 5 memiliki 5 konflik kapasitas. Langkah selanjutnya adalah menghitung nilai *Fitness* untuk semua individu dalam populasi. Proses perhitungan nilai *Fitness* dapat dijelaskan melalui persamaan berikut:

$$1 \quad F \text{ individu 1} = \frac{1}{1 + (2 + 4)} = 0.143$$

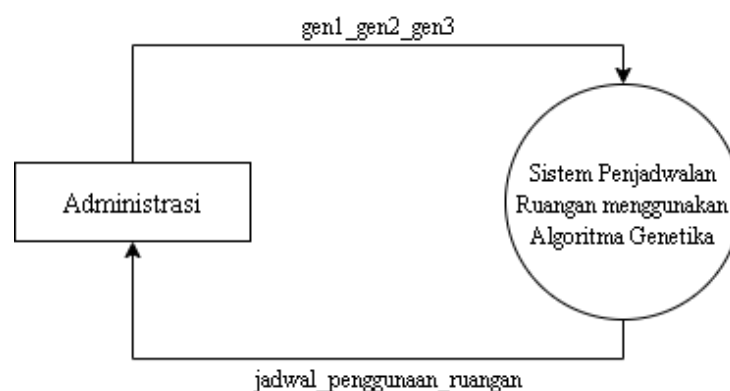
2	$F \text{ individu } 2 = \frac{1}{1 + (0 + 2)} =$	0.333
3	$F \text{ individu } 3 = \frac{1}{1 + (1 + 3)} =$	0.200
4	$F \text{ individu } 4 = \frac{1}{1 + (0 + 3)} =$	0.250
5	$F \text{ individu } 5 = \frac{1}{1 + (2 + 3)} =$	0.167
	<i>Total Nilai fitness =</i>	1.093

Menurut hasil perhitungan nilai *Fitness* di atas, masih terjadi bentrokan dalam proses penjadwalan. Oleh karena itu, diperlukan seleksi ulang atau pengulangan proses yang sama hingga diperoleh nilai *Fitness* terbaik, yaitu 1, yang menunjukkan bahwa solusi optimal telah ditemukan tanpa adanya bentrokan.

3.4 Perancangan Sistem

3.4.1 Diagram Konteks

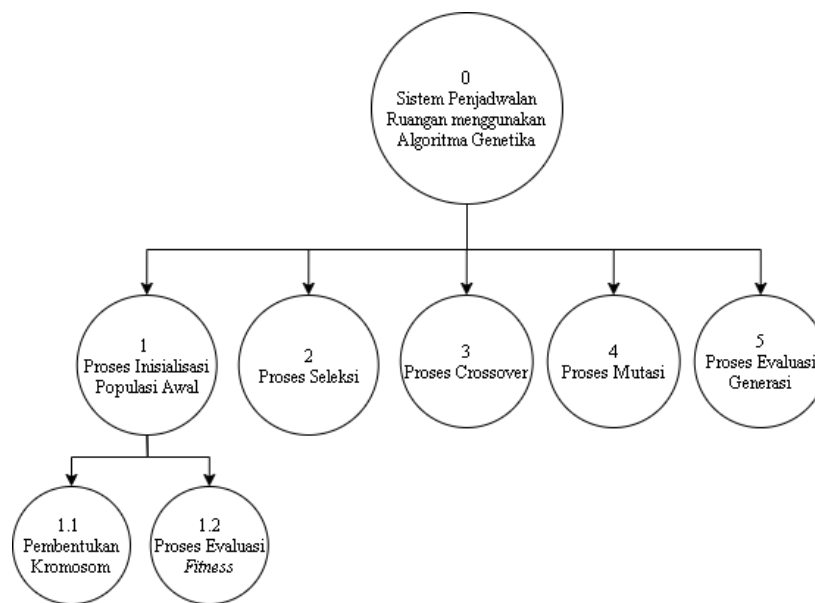
Diagram ini menunjukkan entitas di luar atau terlibat dalam sistem, termasuk aktivitas dari data yang menghubungkan entitas sistem.



Gambar 3.8 Diagram Konteks Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika

Diagram konteks sistem penjadwalan ruangan dapat dilihat pada gambar 3.8. Diagram ini menggambarkan di mana Administrasi berperan sebagai entitas utama dalam pengelolaan sistem tersebut. Proses dimulai dengan Administrasi memasukkan input berupa data gen1, gen2, dan gen3. Melalui proses analisis dalam sistem, dihasilkan keluaran berupa Jadwal Penggunaan Ruangan.

3.4.2 Diagram Jenjang Sistem



Gambar 3.9 Diagram Berjenjang Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika

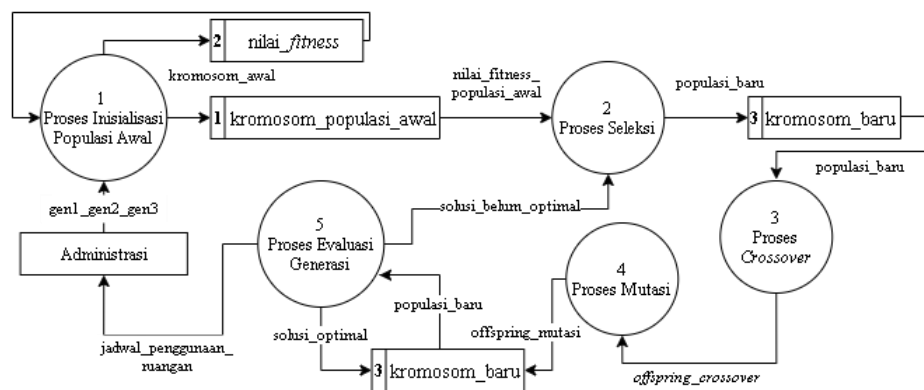
Pada Gambar 3.9 menggambarkan diagram berjenjang sistem yang terdiri dari:

1. Top level : Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika.
2. Level 1 : Hasil dari pembagian proses aplikasi sistem menjadi beberapa sub proses yang meliputi Inisialisasi Populasi Awal, Proses Seleksi, Proses *Crossover*, Proses Mutasi dan Proses Evaluasi Generasi.

3. Level 2 : Bagian dari proses yang ada pada Level 1 Inisialisasi populasi mencakup langkah-langkah yaitu pembentukan kromosom dan menghitung nilai *Fitness*.

3.4.3 Data Flow Diagram (DFD) Sistem

3.4.3.1 DFD Level 1 Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika



Gambar 3.10 DFD Level 1 Sistem Penjadwalan Ruangan di Gedung Dakwah Muhammadiyah Cabang Dukun

Aktivitas DFD level 1 pada Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika terdapat beberapa proses antara lain yaitu :

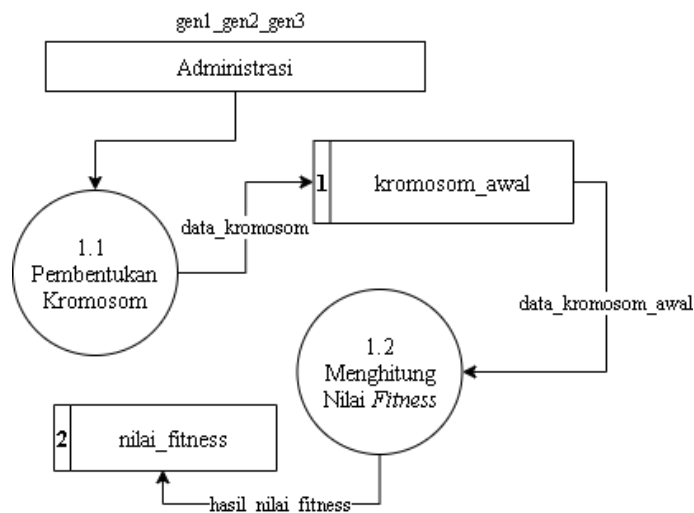
1. Pada Proses 1: Inisialisasi Populasi Awal, dilakukan pembentukan populasi awal berupa kumpulan kromosom yang merepresentasikan individu. Data input berupa gen1_gen2_gen3 digunakan untuk membentuk kromosom awal dengan pendekatan randomisasi. Kromosom yang dihasilkan disimpan dalam *database* 1:kromosom_populasi_awal untuk digunakan dalam proses berikutnya.
2. Pada Proses 2: Seleksi, bertujuan untuk memilih individu dengan kualitas terbaik berdasarkan nilai *fitness*.

Kromosom yang tersimpan dalam Database 1 diambil dan dievaluasi menggunakan fungsi *fitness* yang dirancang untuk mengukur sejauh mana solusi memenuhi kriteria optimalitas. Seleksi dilakukan dengan menggunakan metode *roulette wheel selection*, di mana individu dengan nilai *fitness* lebih tinggi memiliki peluang lebih besar untuk dipilih. Namun, peluang seleksi juga diberikan kepada kromosom dengan *fitness* lebih rendah untuk menjaga keragaman populasi. Kromosom terpilih kemudian disimpan dalam Database 3: kromosom_baru untuk diolah pada tahap selanjutnya.

3. Pada Proses 3: *Crossover*, individu hasil seleksi dikombinasikan menggunakan teknik *One-point crossover*, di mana segmen genetik dari dua individu induk ditukar pada titik potong tertentu. Tujuan utama dari proses ini adalah untuk menghasilkan solusi baru dan berpotensi lebih optimal, yang menghasilkan keturunan yang disebut sebagai *offspring_crossover*, yang selanjutnya disimpan dalam *database*: 3 kromosom_baru.
4. Pada Proses 4: mutasi, bertujuan untuk meningkatkan keragaman genetik dalam populasi. Individu hasil *crossover* dimodifikasi dengan mengubah nilai gen tertentu secara acak untuk menciptakan solusi baru yang belum ada dalam populasi sebelumnya. Proses ini penting untuk mencegah algoritma terjebak pada solusi sub optimal (*local optimum*). Untuk menjaga keseimbangan antara solusi baru dan solusi yang ada. Hasil mutasi, yang disebut *offspring_mutasi*, disimpan kembali ke *database* 3: kromosom_baru.
5. Pada Proses 5: Evaluasi Generasi, Kromosom yang telah mengalami proses mutasi dievaluasi kembali untuk

menentukan apakah solusi yang dihasilkan sudah mencapai solusi optimal. Evaluasi ini dilakukan dengan membandingkan nilai *fitness* setiap individu dengan batas optimal yang telah ditentukan. Jika ditemukan solusi optimal, data tersebut disimpan dalam *database* 3: kromosom_baru dan diteruskan ke proses Administrasi untuk digunakan sebagai jadwal final. Jika solusi belum optimal, populasi baru dikembalikan ke proses 3: seleksi untuk menjalani iterasi berikutnya hingga mencapai solusi terbaik.

3.4.3.2 DFD Level 2 Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika



Gambar 3.11 DFD Level 2 Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika

Aktivitas DFD level 2 pada Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika terdapat beberapa proses antara lain yaitu :

1. Proses 1.1: Pembentukan Kromosom

Pada tahap ini, data Gen 1, Gen 2, dan Gen 3 digunakan untuk membentuk kromosom awal yang merepresentasikan

solusi penjadwalan. Kromosom ini kemudian disimpan dalam *database* 1: kromosom_awal.

2. Proses 1.2: Menghitung Nilai *Fitness*

Kromosom yang terbentuk selanjutnya dihitung nilai *fitness*-nya untuk menilai kualitas solusi berdasarkan kriteria penjadwalan. Hasil perhitungan nilai *fitness* disimpan dalam *database* 2: nilai_ *fitness* .

3.5 Perancangan Basis Data

Basis data (*Database*) adalah kumpulan dari data yang berhubungan antara satu dengan yang lainnya, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Adapun perancangan database mengenai sistem informasi ini diperlukan beberapa tabel sebagai berikut:

3.5.1 Tabel User

Tabel user menyimpan data pengguna dan memungkinkan pengguna mengakses aplikasi.

Tabel 3.25 Tabel User

#	Name	Type	Length	Key
1	id	int	15	PK
2	nama	varchar	155	-
3	username	varchar	155	-
4	password	password	155	-
5	created_at	timestamp	-	-
6	update_at	timestamp	-	-

3.5.2 Tabel Pengguna (GEN 1)

Tabel pengguna adalah data Identitas yang digunakan untuk melakukan proses penggunaan ruangan.

Tabel 3.26 Tabel Lembaga

#	Name	Type	Length	Key
1	id	int	15	PK
2	kode_lembaga	varchar	155	FK
3	ktam	int	155	-
4	nama_lembaga	varchar	155	-

5	email	Varchar	155	-
6	no_telp	Int	155	-
7	alamat	text	155	-
8	created_at	timestamp	-	-
9	update_at	timestamp	-	-

Tabel 3.27 Tabel Kegiatan

#	Name	Type	Length	Key
1	id	int	15	PK
2	id_kegiatan	varchar	155	FK
3	kegiatan	varchar	155	-
4	jumlah_peserta	int	155	-
5	created_at	timestamp	-	-
6	update_at	timestamp	-	-

3.5.3 Tabel Ruangan (GEN 2)

Tabel ruangan menyimpan data ruangan yang akan digunakan dalam proses penggunaan.

Tabel 3.28 Tabel Ruangan

#	Name	Type	Length	Key
1	id	int	15	PK
2	id_ruangan	varchar	155	FK
3	nama_ruangan	varchar	155	-
4	kapasitas	password	155	-
5	created_at	timestamp	-	-
6	update_at	timestamp	-	-

3.5.4 Tabel Waktu (GEN 3)

Tabel waktu adalah data untuk menentukan jadwal penggunaan ruangan.

Tabel 3.29 Tabel Hari

#	Name	Type	Length	Key
1	id	int	15	PK
2	id_hari	varchar	155	FK
3	nama_hari	varchar	155	-
5	created_at	timestamp	-	-
6	update_at	timestamp	-	-

Tabel 3.30 Tabel Jam

#	Name	Type	Length	Key
1	id	int	15	PK
2	id_jam	varchar	155	FK
3	jam	date	155	-
5	created_at	timestamp	-	-
6	update_at	timestamp	-	-

3.5.5 Tabel Kromosom Populasi Awal

Tabel kromosom populasi awal merupakan tabel yang menampung dari setiap tabel yang dibutuhkan untuk melakukan proses penjadwalan dengan menggunakan algoritma genetika.

Tabel 3.31 Tabel kromosom_populasi_awal

#	Name	Type	Length	Key
1	id	bigint	155	PK
2	lembaga_id	bigint	155	-
3	kegiatan_id	bigint	155	-
4	Jumlah_peserta_id	bigint	155	-
5	ruangan_id	bigint	155	-
6	kapasitas_id	bigint	155	-
7	hari_id	int	155	-
8	sesi_id	int	155	-
9	fitness	bigint	-	FK
10	created_at	timestamp	-	-
11	update_at	timestamp	-	-

3.5.6 Tabel Kromosom Baru

Tabel kromosom baru merupakan tabel yang menyimpan hasil perubahan dari setiap proses, yang dibutuhkan untuk menemukan penjadwalan optimal menggunakan algoritma genetika.

Tabel 3.32 Tabel kromosom_baru

#	Name	Type	Length	Key
1	id	bigint	155	PK
2	lembaga_id	bigint	155	-
3	kegiatan_id	bigint	155	-
4	Jumlah_peserta_id	bigint	155	-
5	ruangan_id	bigint	155	-
6	kapasitas_id	bigint	155	-

7	hari_id	int	155	-
8	sesi_id	int	155	-
9	fitness	bigint	155	FK
10	offspring_crossover	Int	155	-
11	offspring_mutasi	Int	155	-
12	generasi	int	155	-
13	created_at	timestamp	-	-
14	update_at	timestamp	-	-

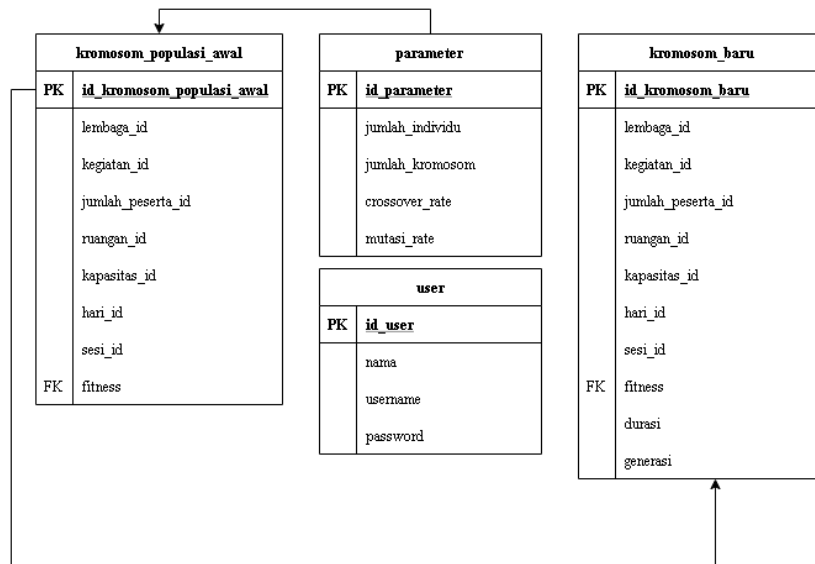
3.5.7 Tabel Parameter

Tabel parameter adalah tabel yang berisi input proses perhitungan Algoritma Genetika.

Tabel 3.33 Tabel Parameter

#	Name	Type	Length	Key
1	id	int	155	PK
2	Jumlah_gen	int	155	-
3	Jumlah_populasi	int	155	-
4	Crossover_rate	int	155	-
5	Mutasi_rate	int	155	-
6	created_at	timestamp	-	-
7	update_at	timestamp	-	-

Sistem ini memiliki tabel *user* yang terhubung dengan tabel kromosom_populasi_awal, parameter, dan kromosom_baru. Relasi antara user dan kedua tabel kromosom adalah *one-to-many*, yang berarti satu pengguna dapat memiliki banyak kromosom. Selain itu, relasi antara parameter dengan kedua tabel kromosom juga *one-to-many*, yang menunjukkan bahwa satu set parameter dapat mempengaruhi banyak kromosom, seperti yang digambarkan pada gambar 3.12.

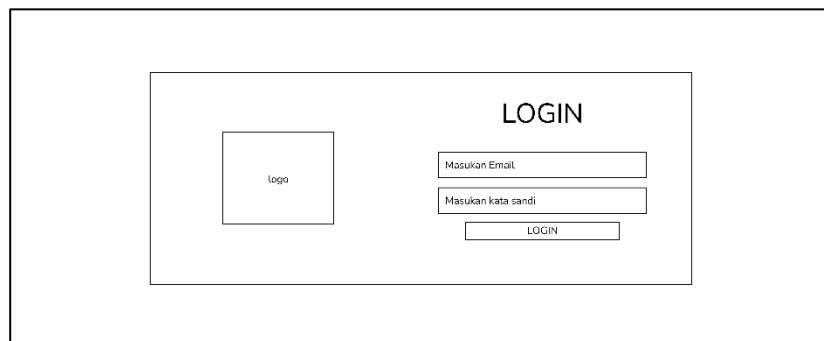


Gambar 3.12 ERD Sistem Penjadwalan Ruangan menggunakan Algoritma Genetika

3.6 Perancangan Antarmuka Sistem

3.6.1 Halaman Login

Halaman login digunakan oleh administrasi yang ingin menjadwalkan ruangan. Administrasi harus memasukkan *username* dan *password* yang benar. Jika informasi yang dimasukkan tidak sesuai dengan data yang tersimpan, sistem akan mengeluarkan peringatan yang menunjukkan bahwa *username* dan *password* yang diberikan salah.



Gambar 3.13 Halaman Login

3.6.2 Halaman Dashboard

Halaman dashboard berfungsi sebagai antarmuka utama yang disajikan kepada administrasi setelah berhasil masuk ke sistem. Halaman ini mencakup informasi terkait data ruangan, aktivitas, dan detail relevan lainnya. Selain itu, halaman dashboard memberikan informasi jadwal penggunaan ruangan saat ini.

GDM ADMIN	DASHBOARD				nama user profile	
	DASHBOARD ADMIN Data Pengguna GEN 1 Data Ruangan GEN 2 Data Waktu GEN 3 Penjadwalan Generate Jadwal Lihat Jadwal	TOTAL USER	TOTAL RUANGAN	TOTAL KEGIATAN	TOTAL LEMBAGA	
	PENGATURAN	JADWAL PENGGUNAAN RUANGAN				

Gambar 3.14 Halaman Dashboard

3.6.3 Halaman Input Pengguna

Suatu halaman akan dibuat untuk penyertaan data pengguna, yang akan digunakan dalam proses penjadwalan.

GDM ADMIN	PROSES JADWAL					NAMA USER	
	DASHBOARD ADMIN Data Pengguna GEN 1 Data Ruangan GEN 2 Data Waktu GEN 3 Penjadwalan Generate Jadwal Lihat Jadwal	PROSES INPUT PEMINJAMAN RUANGAN					TAMBAH PEMINJAM
		NO	NAMA LEMBAGA	NAMA KEGIATAN	JUMLAH PESERTA	AKSI	
PENGATURAN		1	MTs Maarif Tanwirul Qulub	Pusat Dakwah	90	EDIT	DELETE
		2	MTs Mathlabul Huda	Pendidikan	250	EDIT	DELETE
		3	MTs Muhammadiyah 10	Sosial	190	EDIT	DELETE
		4	MTs Ihyaul Ulum	Rapat dan Koordinasi	220	EDIT	DELETE
		5	MTs Tarbiyatul Athfal	Umum	130	EDIT	DELETE
		6	MTs Nur Rahmat	Seminar	30	TERIMA	TOLAK

Gambar 3.15 Halaman Input Pengguna

3.6.4 Halaman Proses Algoritma Genetika

Halaman Proses Algoritma Genetika merupakan formulir yang digunakan untuk memproses data yang telah diinputkan sebelumnya dengan

menggunakan parameter-parameter algoritma genetika, yaitu jumlah gen, banyaknya kromosom, serta *probabilitas Crossover* dan Mutasi. Rancangan *form* parameter algoritma genetika untuk pemrosesan jadwal dapat dilihat pada gambar 3.16.

GDM ADMIN DASHBOARD ADMIN Data Pengguna GEN 1 Data Ruangan GEN 2 Data Waktu GEN 3 Penjadwalan Generate Jadwal Lihat Jadwal PENGATURAN	JADWAL				NAMA USER	
	JUMLAH GENERASI	JUMLAH INDIVIDU	CROSSOVER RATE	MUTASI RATE		
	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>		
	<input type="button" value="GENERATE JADWAL"/>					
	PROSES 1 POPULASI AWAL					
	PROSES 2 SELEKSI					
	PROSES 3 CROSSOVER					
	PROSES 4 MUTASI					
	PROSES 5 EVALUASI GENERASI					

Gambar 3.16 Halaman Proses Algoritma Genetika Admin

Form ini digunakan untuk menampilkan hasil akhir dari proses penjadwalan, yaitu jadwal penggunaan yang telah dihasilkan. Untuk melihat hasil akhir tersebut, pengguna dapat mengklik tombol "Generate Jadwal" dan hasilnya akan ditampilkan. Form ini dapat dilihat pada Gambar 3.17.

GDM ADMIN DASHBOARD ADMIN Data Pengguna GEN 1 Data Ruangan GEN 2 Data Waktu GEN 3 Penjadwalan Generate Jadwal Lihat Jadwal PENGATURAN	JADWAL							NAMA USER
	NO	NAMA LEMBAGA	NAMA KEGIATAN	JUMLAH PESERTA	RUANGAN	KAPASITAS	HARI	JAM
	1	MTs Maarif Tanwirul Qulub	Pusat Dakwah	90	Lantai 1 - Ruang 3	150	SENIN	13:00 - 17:00
	2	MTs Mathlabul Huda	Pendidikan	250	Lantai 1 - Ruang 3	150	SENIN	08:00 - 12:00
	3	MTs Muhammadiyah 10	Sosial	190	Lantai 1 - Ruang 4	250	SELASA	13:00 - 17:00
	4	MTs Ihyaul Ulum	Rapat dan Koordinasi	220	Lantai 1 - Ruang 5	400	SELASA	08:00 - 12:00
	5	MTs Tarbiyatul Athfal	Umum	130	Lantai 1 - Ruang 3	400	RABU	08:00 - 12:00
	6	MTs Nur Rahmat	Seminar	30	Lantai 1 - Ruang 2	150	RABU	13:00 - 17:00

Gambar 3.17 Halaman Hasil Penjadwalan

3.7 Spesifikasi Pengembangan Sistem

1. Kebutuhan Perangkat Keras

Perangkat keras adalah alat yang digunakan untuk menunjang dalam pembuatan sistem. Dalam pembuatan sistem ini perangkat keras yang digunakan yaitu laptop dengan spesifikasi:

- a. *Processor Intel Core I5*
- b. *RAM 16 GB*
- c. *SSD 512 GB*
- d. *HDD 1024 GB*
- e. *Monitor 14"*
- f. *Mouse*
- g. *Keyboard*

2. Kebutuhan Perangkat Lunak

Perangkat lunak adalah program atau aplikasi yang digunakan untuk membangun sistem. Perangkat lunak yang dibutuhkan dalam pembuatan sistem ini adalah:

- a. *Windows 11*
- b. *Visual Studio Code*
- c. *Web server Apache*
- d. *Bahasa PHP Framework Laravel versi 11*
- e. *Bootstrap*
- f. *Database server MySQL*

3.8 Skenario Pengujian Sistem

Adapun perencanaan pengujian dilakukan sebagai berikut:

1. Data yang digunakan:
 - a. Pengguna yang terdiri atas:
 - Lembaga
 - Kegiatan
 - Jumlah Peserta
 - b. Ruangan terdiri atas ruangan dan kapasitas ruangan
 - c. Waktu terdiri atas hari dan sesi
2. Melakukan pengujian terhadap

- 5 Data
- 10 Data
- 15 Data

Dengan melakukan 5 kali generate tiap pengujian

Tabel 3.34 Tabel Sekenario Pengujian

Jumlah Data	Pengujian	Jumlah Generasi	Crossover Rate	Mutas Rate	Waktu
5	1				
	2				
	3				
	4				
	5				
10	1				
	2				
	3				
	4				
	5				
15	1				
	2				
	3				
	4				
	5				

3. Hasil tiap pengujian ditampilkan seperti pada tabel 3.35.

Tabel 3.35 Hasil Pengujian

Lembaga	Kegiatan	Jumlah Peserta	Ruangan	Kapasitas	Hari	Sesi