



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB III

METODOLOGI PENELITIAN DAN PERANCANGAN SISTEM

3.1 Metodologi Penelitian

Beberapa metode penelitian yang digunakan dalam penelitian ini adalah.

1. Studi Fisibilitas

Pada tahap ini, dilakukan pembelajaran berbagai bentuk informasi pada penelitian sebelumnya seperti apakah algoritma Genetika dapat menyelesaikan *Job-Shop Scheduling Problem* dan *Knapsack Problem*. Berdasarkan hasil pengumpulan data melalui wawancara dengan Sekertaris Prodi Informatika UMN, maka penelitian ini dianggap layak untuk diteliti lebih lanjut.

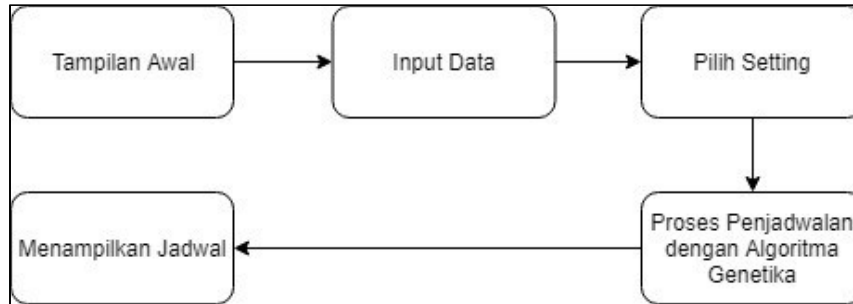
2. Studi Literatur

Kegiatan yang dilakukan pada tahap ini adalah, mengumpulkan informasi yang berhubungan dengan algoritma Genetika seperti skema pengkodean, nilai *fitness*, seleksi orang tua, *crossover*, dan *mutation*, lalu tentang *Job-Shop Scheduling Problem* dan *Knapsack*, serta informasi yang berhubungan dengan EUCS seperti skala *Likert*.

3. Analisa dan Perancangan

Dalam tahap ini, dilakukan analisis terhadap metode, data, dan proses yang akan digunakan pada penelitian ini yang berhubungan dengan penjadwalan sidang magang.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.1 Gambaran Umum Aplikasi

Input yang dimasukkan adalah nama mahasiswa, judul laporan, dan dosen pembimbing. Kemudian penjadwalan akan dilakukan menggunakan Algoritma Genetika. Dalam tahap perancangan akan dibuat *flowchart* untuk menggambarkan keseluruhan cara kerja *web service*. Hasil analisis menunjukkan bahwa penjadwalan sidang magang telah memenuhi seluruh kriteria untuk disebut sebagai *Job Shop Scheduling Problem* dan termasuk dalam kategori *Single Knapsack Problem* dengan *multiple constraints*

4. Implementasi

Pada tahap ini, penjadwalan sidang magang akan diimplementasikan dalam bentuk web service, kemudian akan dilakukan penulisan beberapa baris kode dalam bahasa Python berupa inisialisasi populasi, seleksi orang tua, *crossover*, mutasi, dan evaluasi fungsi *fitness*.

5. Uji Coba dan Evaluasi

Pada tahap ini, akan dilakukan uji coba dan evaluasi untuk mendapatkan hasil keluaran yang sesuai dengan kebutuhan pengguna yang didapatkan dari hasil wawancara dengan Sekertaris Prodi Informatika UMN. Uji Coba dan Evaluasi akan dilakukan untuk mengetahui apakah hasil implementasi algoritma Genetika untuk penjadwalan sidang magang telah memenuhi tujuan dan memberikan keluaran yang

diharapkan oleh pengguna. Uji coba akan dilakukan dengan metode *White Box*, dan evaluasi akan dilakukan untuk mendapatkan nilai tingkat kepuasan menggunakan EUCS dengan skala *Likert* 6 tingkat.

6. Konsultasi dan Penulisan

Tahapan ini dilakukan untuk mendokumentasikan seluruh proses penelitian yang akan dibuat menjadi sebuah laporan. Laporan ini akan dijadikan sebagai bukti bahwa penelitian telah selesai.

3.2 Perancangan Sistem

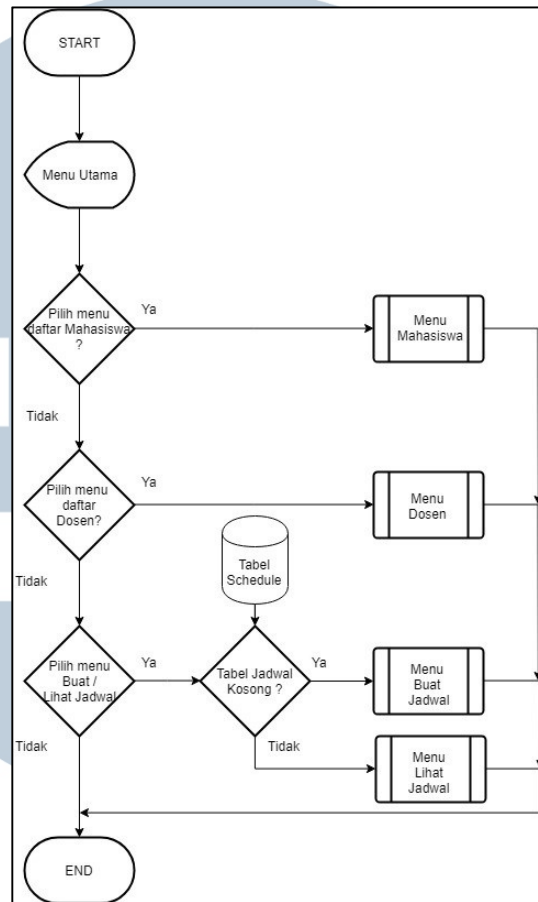
Perancangan menu dalam penelitian ini dilakukan dengan membuat *flowchart* dan diagram perencanaan *web service*.

3.2.1 Flowchart

Flowchart digunakan untuk menggambarkan alur dari menu yang telah dibuat yang terdiri dari *flowchart* menu mahasiswa, *flowchart* menu dosen, dan *flowchart* menu jadwal.



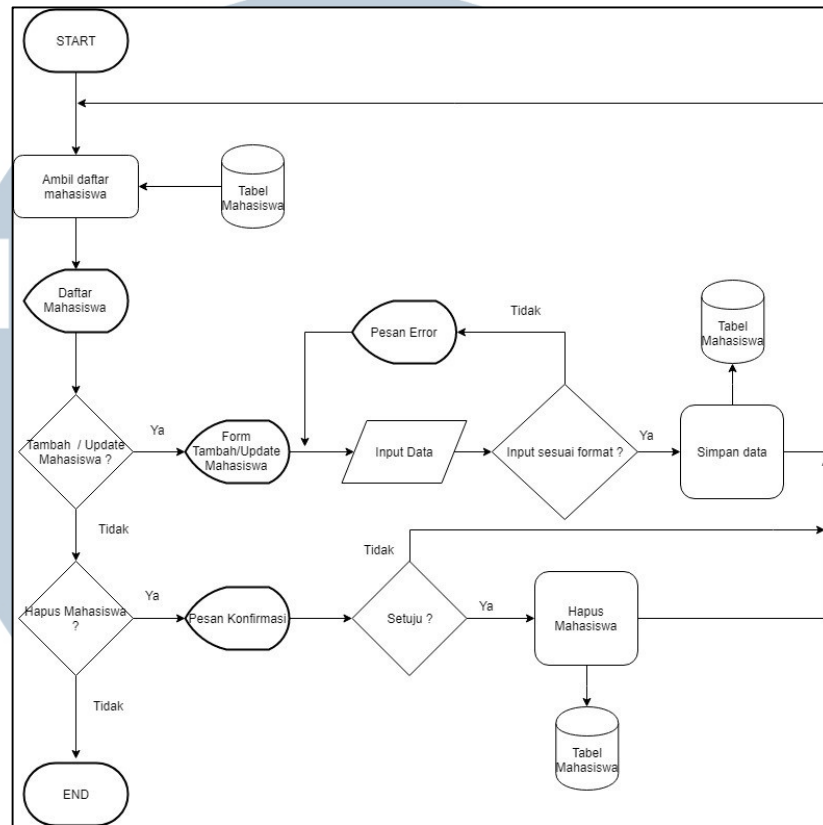
A. Flowchart Utama



Gambar 3.2 Flowchart Utama

Gambar 3.2 merupakan *flowchart* utama dari *web service* yang terdiri dari 3 menu yang terdiri dari menu Mahasiswa, menu Dosen, dan menu Jadwal. Pengguna dapat memilih menu yang tersedia, jika menu Mahasiswa dipilih, pengguna akan masuk ke menu Mahasiswa, jika menu Dosen dipilih, pengguna akan masuk ke menu Dosen, dan jika menu Jadwal dipilih maka pengguna akan masuk ke menu Jadwal.

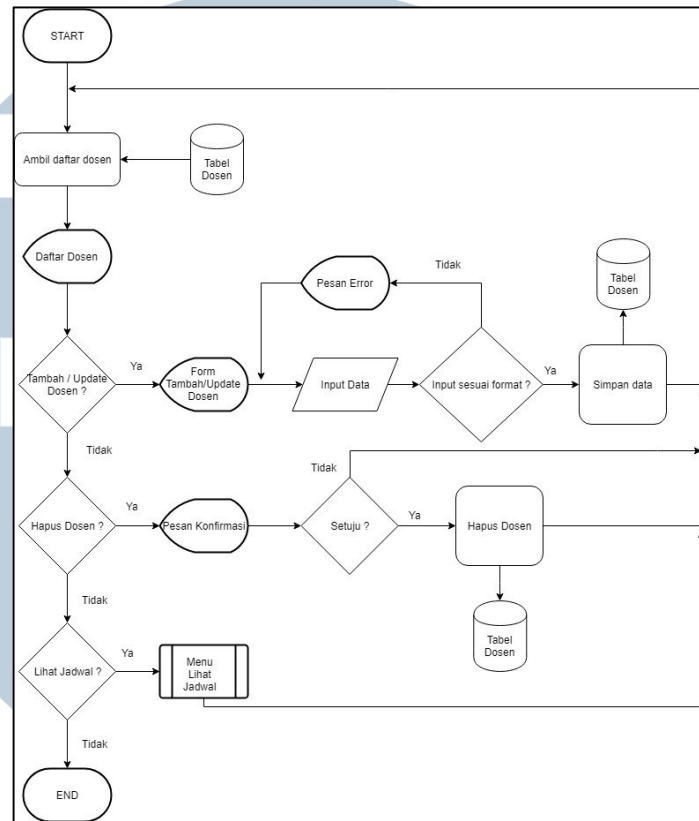
B. Flowchart Menu Mahasiswa



Gambar 3.3 Flowchart Menu Mahasiswa

Gambar 3.3 merupakan *flowchart* dari menu Mahasiswa. Di dalam menu Mahasiswa, pengguna dapat melihat seluruh mahasiswa yang telah mengambil mata kuliah magang atau *internship* yang akan maju ke tahap sidang. Pengguna juga dapat memodifikasi daftar mahasiswa dengan menambahkan mahasiswa yang akan maju sidang dengan memasukkan data berupa nomor induk mahasiswa atau NIM, nama mahasiswa, judul laporan, dan dosen pembimbing. Pengguna juga dapat menghapus mahasiswa dari daftar mahasiswa yang akan maju sidang.

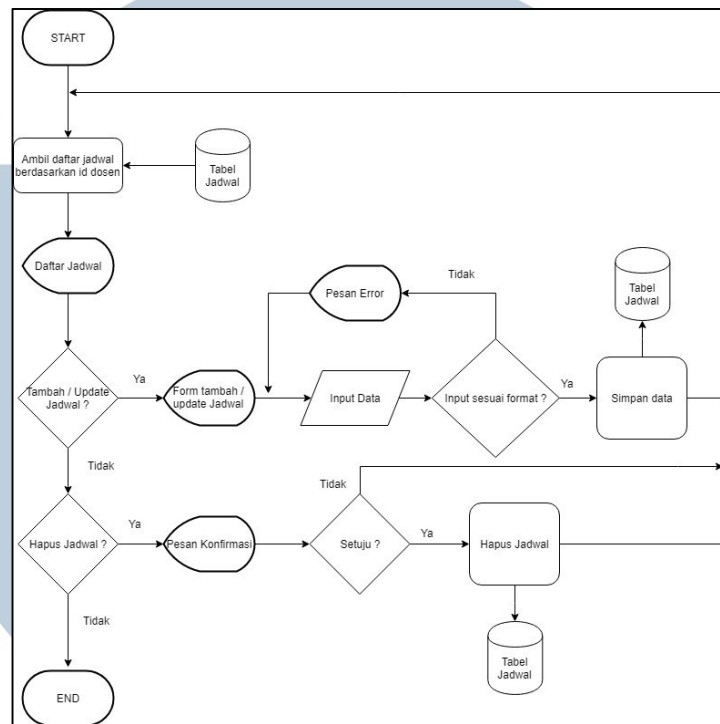
C. Flowchart Menu Dosen



Gambar 3.4 Flowchart Menu Dosen

Gambar 3.4 merupakan *flowchart* dari menu Dosen. Di dalam menu Dosen, pengguna dapat melihat seluruh dosen yang terdaftar sebagai dosen di Universitas Multimedia Nusantara Prodi Informatika. Pengguna juga dapat memodifikasi daftar dosen dengan memasukkan data berupa nama dosen dan jabatan. Pengguna juga dapat menghapus dosen dari daftar dosen yang tersedia. Selain itu, pengguna dapat melihat jadwal dari dosen yang bersangkutan dengan memilih menu jadwal pada daftar dosen yang tersedia.

D. Flowchart Menu Lihat Jadwal

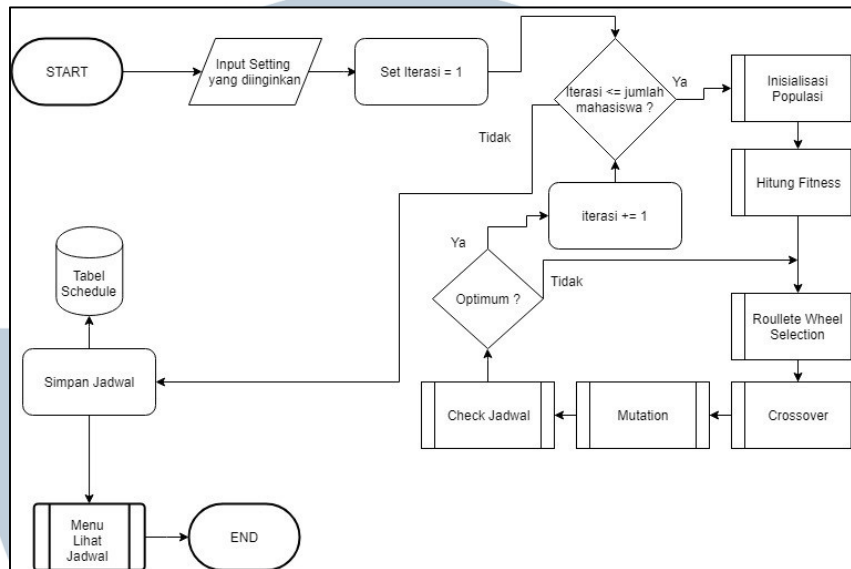


Gambar 3.5 Flowchart Menu Lihat Jadwal

Gambar 3.5 adalah *flowchart* untuk menu lihat jadwal. Pengguna dapat melihat daftar jadwal dosen yang telah dipilih sebelumnya pada menu dosen. Jadwal dari dosen yang bersangkutan dapat ditambahkan dengan memasukkan data berupa hari mengajar dan jam mengajar. Pengguna juga dapat menghapus jadwal dari daftar jadwal yang telah ada.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

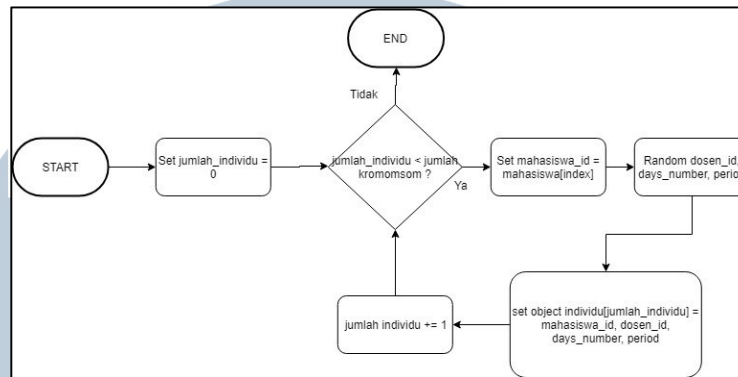
E. Flowchart Buat Jadwal



Gambar 3.6 Flowchart Buat Jadwal

Gambar 3.6 adalah *flowchart* buat jadwal, dimana pengguna akan diminta untuk memasukkan pengaturan yang dikehendaki untuk jadwal yang akan dibuat. Jadwal akan dibuat dengan mengimplementasikan Algoritma Genetika. Proses dari Algoritma Genetika akan diawali dengan inisialisasi populasi awal dengan segala data yang diperlukan. Kemudian setiap individu akan memiliki nilai *Fitness* yang akan menentukan perhitungan *Probability Distribution Function* (PDF). Seleksi akan dilakukan untuk mencari individu yang lebih baik untuk dilakukan *crossover*. *Crossover* akan dilakukan setelah individu yang unggul terpilih, kemudian beberapa individu akan mengalami proses mutasi. Setelah proses mutasi selesai, nilai *fitness* dari tiap individu akan dihitung kembali dan dicek apakah solusi sudah optimal, jika solusi dianggap belum optimal, maka proses akan diulang dari tahap seleksi. Hal ini akan terus berulang sampai seluruh mahasiswa terjadwal. Solusi yang sudah dianggap optimal akan disimpan dalam tabel *Schedule*, dan akan ditampilkan pada halaman Jadwal Sidang.

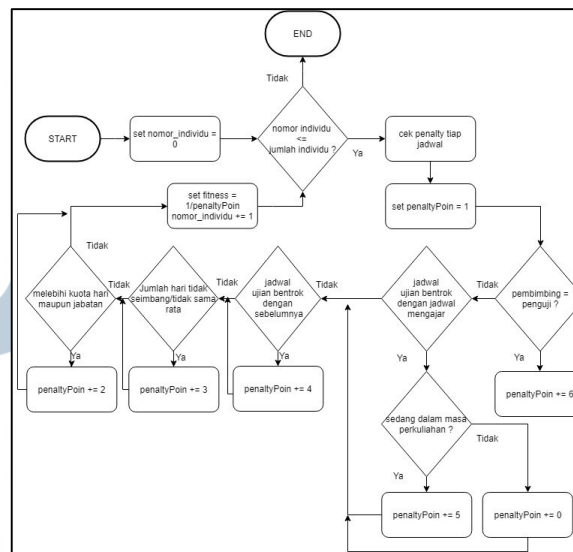
F. Flowchart Inisialisasi Populasi



Gambar 3.7 Flowchart Inisialisasi Populasi

Gambar 3.7 adalah *flowchart* inisialisasi populasi. Tahap ini adalah awal dari proses Algoritma Genetika. Individu didalam populasi akan dibuat sebanyak hasil dari *input* dari pengguna pada menu Buat Jadwal, setiap individu bersifat acak terdiri dari ID dari mahasiswa, ID dari dosen, hari, dan jam ujian sidang.

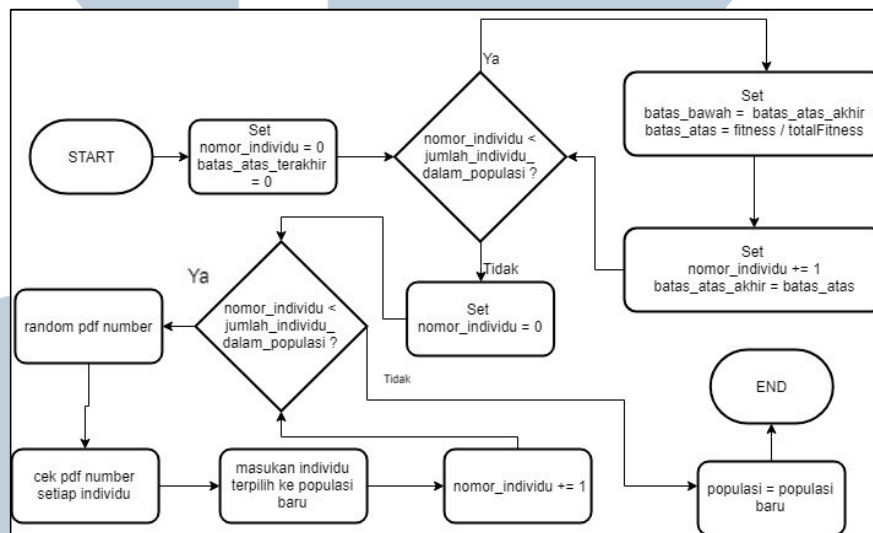
G. Flowchart Hitung Fitness



Gambar 3.8 Flowchart Hitung Fitness

Gambar 3.8 adalah *flowchart* hitung *fitness*. Pada tahap ini, nilai *fitness* setiap individu yang berada dalam populasi akan dihitung. Batasan-batasan dalam perhitungan *fitness* adalah apakah dosen pembimbing sama dengan dosen penguji, apakah jadwal milik dosen penguji dan dosen pembimbing bentrok dengan jadwal mengajar dan sidang sebelumnya, dan apakah sudah melebihi kuota yang ditetapkan baik per hari maupun per dosen. Setiap pelanggaran Batasan yang telah ditentukan akan diberikan poin penalti, nilai *fitness* akan dihitung setelah perhitungan poin penalti setiap individu telah selesai. Pada penelitian ini fungsi optimal yang dicari adalah nilai *minimum*.

H. Flowchart Roulette Wheel Selection

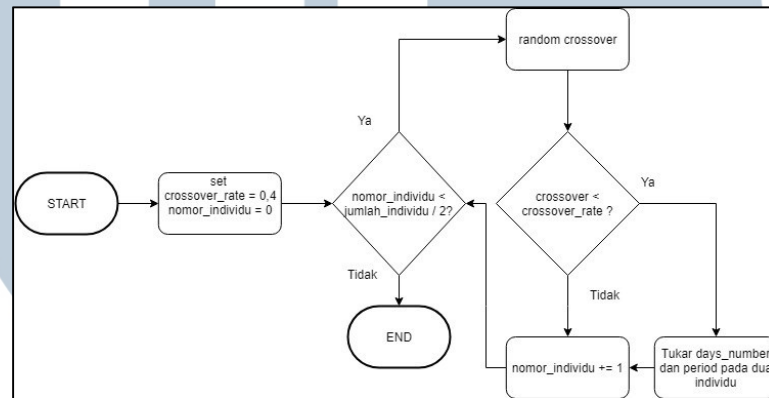


Gambar 3.9 Flowchart Roulette Wheel Selection

Gambar 3.9 adalah *Flowchart Roulette Wheel Selection*. Tahap ini akan melahirkan populasi baru dengan cara melakukan seleksi terhadap individu-individu yang ada. Pemilihan individu baru dilakukan dengan perhitungan *Probability Distribution Function* (PDF). Batas bawah dari PDF adalah batas atas terakhir. Kemudian untuk batas atas didapatkan dari perhitungan *fitness* individu

dibagi dengan total *fitness* seluruh individu. Lalu akan dibangkitkan bilangan acak sebanyak jumlah individu yang akan dicocokkan dengan PDF yang ada. Individu dengan hasil nilai acak akan menjadi populasi baru yang akan menggantikan populasi yang lama.

I. Flowchart Crossover

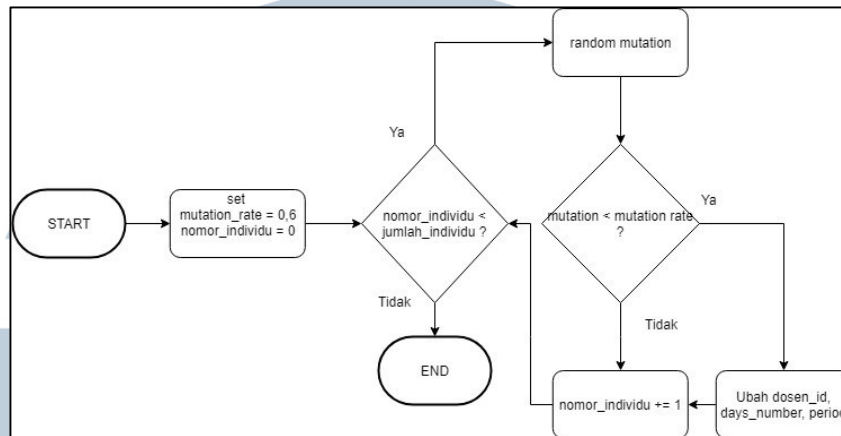


Gambar 3.10 Flowchart Crossover

Gambar 3.10 adalah *flowchart Crossover*. Pada bagian ini setengah dari individu akan mendapatkan sebuah bilangan acak dari 0 - 99, jika angka yang didapatkan sebuah individu lebih kecil dari *crossover rate* yang telah ditentukan yaitu sebesar 40, maka individu tersebut akan melalui tahap *crossover*.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

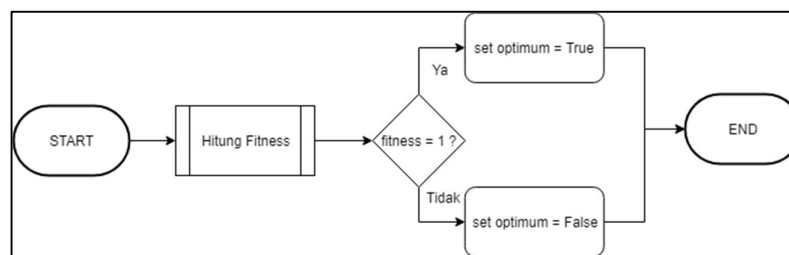
J. Flowchart Mutation



Gambar 3.11 Flowchart Mutation

Gambar 3.11 adalah *flowchart Mutation*. Pada bagian ini setiap individu akan mendapatkan sebuah bilangan acak dari 1 – 100, jika angka yang didapatkan sebuah individu lebih kecil dari *mutation rate* yang telah ditentukan yaitu sebesar 6, maka individu tersebut akan melalui tahap *mutation*.

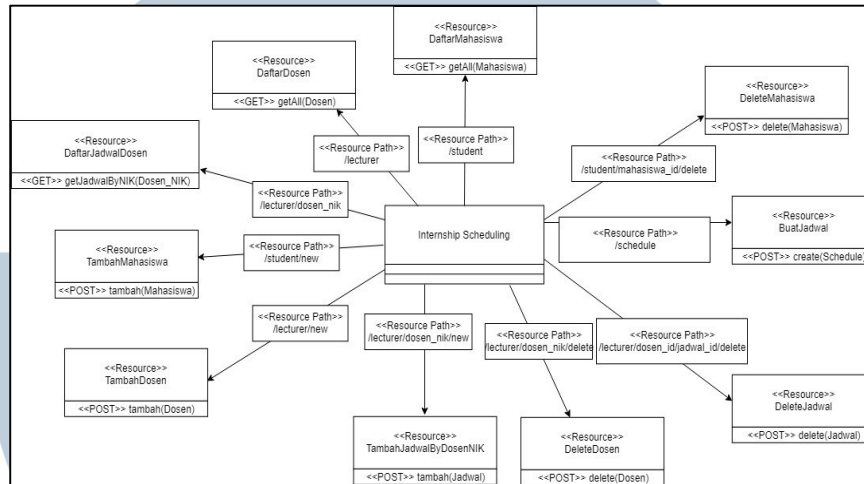
K. Flowchart CheckJadwal



Gambar 3.12 Flowchart CheckJadwal

Gambar 3.12 adalah *flowchart CheckJadwal*. Pada bagian ini setiap individu yang telah melalui tahap seleksi, *crossover*, dan *mutation* akan kembali dicek nilai *fitness*nya. Jika terdapat 1 individu yang memiliki nilai fitness 1 maka individu tersebut dinyatakan telah memenuhi kriteria.

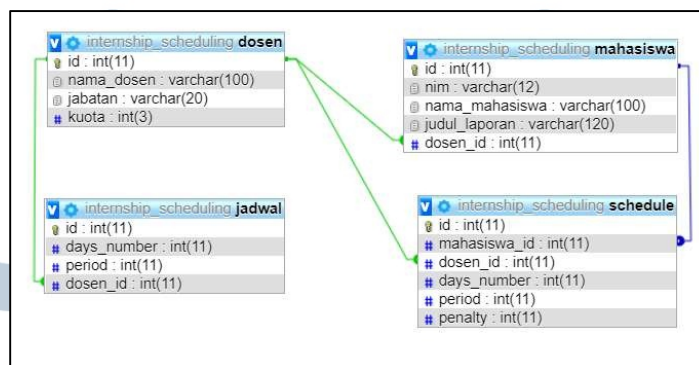
3.2.2 Web Service Model



Gambar 3.13 Web Service Model

Gambar 3.13 merupakan gambar diagram *web service* dari aplikasi penjadwalan sidang magang. Diagram ini memiliki *path* untuk mengakses *function* yang akan dijalankan beserta dengan *method* yang akan digunakan.

3.2.3 Database Schema



Gambar 3.14 Database Schema

Gambar 3.14 merupakan keseluruhan dari *database* yang digunakan dalam *web service* penjadwalan sidang magang. Terdiri dari 4 tabel yaitu tabel mahasiswa, tabel dosen, tabel jadwal dan tabel *schedule*.

3.2.4 Struktur Tabel

Berikut ini adalah struktur dari keseluruhan tabel yang digunakan pada *web service* penjadwalan sidang magang.

1. Tabel Mahasiswa

Tabel mahasiswa menyimpan seluruh mahasiswa yang akan melakukan ujang sidang magang. Setiap mahasiswa memiliki atribut id, nim, nama mahasiswa, dosen id sebagai dosen pembimbing, dan judul laporan.

Tabel 3.1 Struktur Tabel Mahasiswa

Nama Kolom	Tipe Data	Identitas	Keterangan
Id	Int	PK	Auto Increment
nim	Int(11)		
nama_mahasiswa	Varchar(100)		
judul_laporan	Varchar(120)		
dosen_id	Int	FK	Sebagai Dosen Pembimbing

2. Tabel Dosen

Tabel dosen menyimpan seluruh dosen yang terdaftar sebagai dosen fakultas Informatika Universitas Multimedia Nusantara, setiap dosen memiliki id, nama_dosen, jabatan, dan kuota yang akan digunakan untuk menentukan beban sidang.

Tabel 3.2 Tabel Dosen

Nama Kolom	Tipe Data	Identitas	Keterangan
id	Int	PK	Auto Increment
Nama_dosen	Varchar(100)		
Jabatan	Varchar(20)		
Kuota	Int(3)		

3. Tabel Jadwal

Tabel *jadwal* menyimpan seluruh jadwal mengajar dari dosen yang bersangkutan. Tabel ini menyimpan *id*, *days_number*, *period*, dan *dosen_id* sebagai pemilik dari jadwal tersebut

Tabel 3.3 Tabel Jadwal

Nama Kolom	Tipe Data	Identitas	Keterangan
Id	Int	PK	Auto Increment
Days_number	Int(11)		
period	Int(11)		
Dosen_id	Int	FK	Pemilik Jadwal

4. Tabel Schedule

Tabel *schedule* digunakan untuk menyimpan jadwal sidang magang yang akan dibuat. Tabel ini menyimpan *id*, *mahasiswa_id*, *dosen_id*, *days_number*, dan *period*.

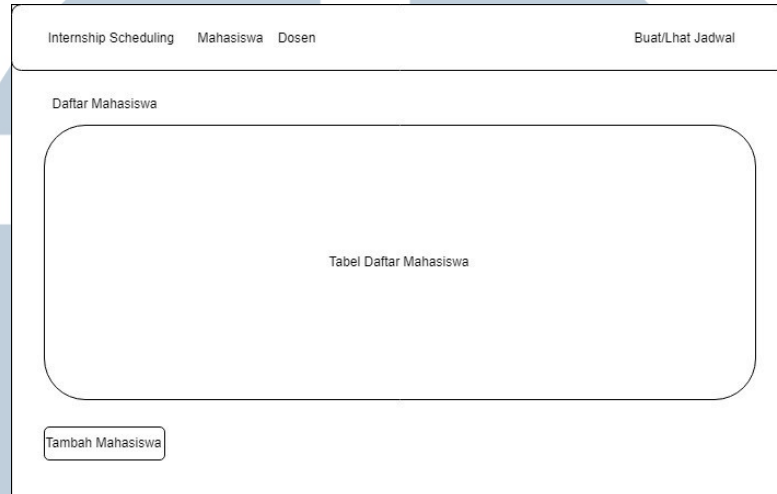
Tabel 3.4 Tabel Schedule

Nama Kolom	Tipe Data	Identitas	Keterangan
Id	Int	PK	Auto Increment
Mahasiswa_id	Int	FK	
Dosen_id	Int	FK	Sebagai penguji
Days_number	Int		
Period	Int		

3.2.5 Rancangan Antarmuka

Dalam mengimplementasi web service penjadwalan sidang magang ini, dilakukan perancangan antarmuka terlebih dahulu. Gambar 3.15 merupakan rancangan tampilan awal ketika pengguna mengakses *web service* penjadwalan sidang magang. Pengguna dapat melihat seluruh mahasiswa yang akan melakukan

ujian sidang magang, menambahkan mahasiswa ke daftar, dan menghapus mahasiswa dari daftar



The image shows a web application interface for 'Daftar Mahasiswa' (Student List). At the top, there is a navigation bar with links: 'Internship Scheduling', 'Mahasiswa', 'Dosen', and 'Buat/Lihat Jadwal'. Below the navigation bar, the main content area is titled 'Daftar Mahasiswa'. Inside this area, there is a large rectangular box labeled 'Tabel Daftar Mahasiswa', which is currently empty. At the bottom left of the main content area, there is a button labeled 'Tambah Mahasiswa'.

Gambar 3.15 Rancangan Menu Mahasiswa

Gambar 3.16 merupakan rancangan tampilan untuk Tambah Mahasiswa. Pengguna dapat mengisi form tambah mahasiswa yang berisi NIM, Nama Mahasiswa, Judul Laporan, dan Dosen Pembimbing untuk menambahkan mahasiswa yang akan mengikuti ujian sidang magang.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Internship Scheduling Mahasiswa Dosen Buat/Lihat Jadwal

Tambah Mahasiswa

NIM

Nama Mahasiswa

Judul Laporan

Dosen Pembimbing

Gambar 3.16 Rancangan Menu Tambah Mahasiswa

Gambar 3.17 merupakan rancangan tampilan menu dosen ketika pengguna mengakses dengan menekan tombol Dosen pada *menu bar*. Pengguna dapat melihat seluruh daftar dosen yang terdaftar sebagai dosen Program Studi Informatika Universitas Multimedia Nusantara, menambah dosen ke dalam daftar, dan menghapus dosen yang telah ada pada daftar.

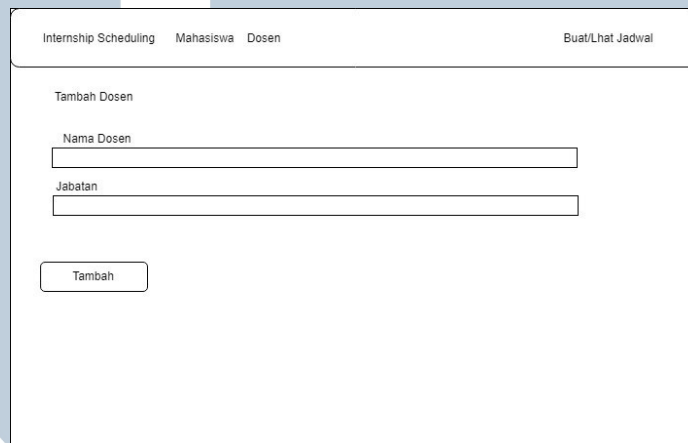
Internship Scheduling Mahasiswa Dosen Buat/Lihat Jadwal

Daftar Dosen

Tabel Daftar Dosen

Gambar 3.17 Rancangan Menu Dosen

Gambar 3.18 merupakan rancangan tampilan untuk Tambah Dosen. Pengguna dapat mengisi form tambah dosen yang berisi Nama Dosen, dan Jabatan yang terdiri dari Pejabat dan Non Pejabat.



The screenshot shows a web application window titled 'Internship Scheduling'. It has a navigation bar with 'Mahasiswa' and 'Dosen' tabs, and a 'Buat/Lihat Jadwal' button. The main content area is titled 'Tambah Dosen' and contains two text input fields: 'Nama Dosen' and 'Jabatan'. Below the fields is a 'Tambah' button.

Gambar 3.18 Halaman Tambah Dosen

Gambar 3.19 merupakan rancangan tampilan untuk menu buat jadwal. Pengguna dapat membuat jadwal sidang magang dengan memasukan beberapa *setting* untuk jadwal yang akan dibuat seperti, banyaknya beban sidang untuk dosen pejabat dan non pejabat, batch, beban sidang perhari, dan jumlah individu dalam populasi atau kromosom.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

Internship Scheduling Mahasiswa Dosen Buat/Lhat Jadwal

Buat Jadwal

Banyak beban sidang untuk dosen pejabat

Banyak beban sidang untuk dosen non-pejabat

Batch

Banyak beban sidang untuk dosen dalam 1 hari

Jumlah Individu dalam Populasi

Gambar 3.19 Rancangan Halaman Buat Jadwal

Gambar 3.20 merupakan rancangan tampilan untuk melihat hasil jadwal sidang magang yang telah dibuat berdasarkan pengaturan yang dilakukan sebelumnya. Pengguna dapat melihat seluruh jadwal sidang magang mahasiswa beserta dengan dosen pengujinya. Pengguna juga dapat menghapus seluruh jadwal untuk membuat jadwal yang baru.

Internship Scheduling Mahasiswa Dosen Buat/Lhat Jadwal

Jadwal Sidang Magang

Tabel Daftar Jadwal Dosen

Gambar 3.20 Rancangan Menu Jadwal Sidang Magang