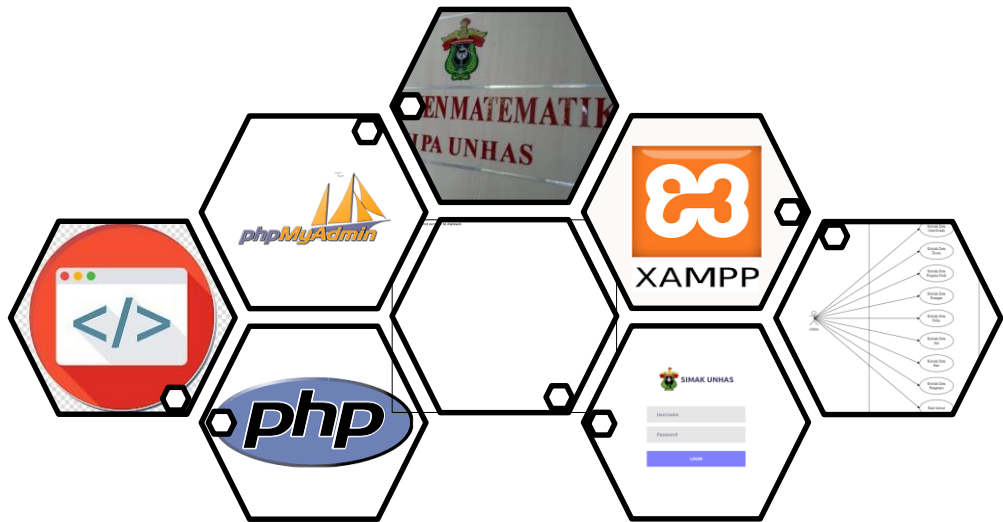


**IMPLEMENTASI ALGORITMA GENETIKA PADA RANCANG BANGUN
SISTEM PENJADWALAN MATA KULIAH DEPARTEMEN BER BASIS
WEBSITE (STUDI KASUS : DEPARTEMEN MATEMATIKA)**



OLEH :

MUHAMMAD ARI PUTRA RAMADHAN H071171313



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR
2024**

**IMPLEMENTASI ALGORITMA GENETIKA PADA RANCANG BANGUN
SISTEM PENJADWALAN MATA KULIAH DEPARTEMEN BER BASIS
WEBSITE (STUDI KASUS : DEPARTEMEN MATEMATIKA)**

**MUHAMMAD ARI PUTRA RAMADHAN
H071171313**



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR
2024**

**IMPLEMENTASI ALGORITMA GENETIKA PADA RANCANG BANGUN
SISTEM PENJADWALAN MATA KULIAH DEPARTEMEN BER BASIS
WEBSITE (STUDI KASUS : DEPARTEMEN MATEMATIKA)**

**MUHAMMAD ARI PUTRA RAMADHAN
H071171313**

Skripsi

Diajukan sebagai salah satu syarat untuk mencapai gelar sarjana

Program Studi Sistem Informasi

pada

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS HASANUDDIN
MAKASSAR
2024**

**IMPLEMENTASI ALGORITMA GENETIKA PADA RANCANG
BANGUN SISTEM PENJADWALAN MATA KULIAH
DEPARTEMEN BERBASIS *WEBSITE*
(STUDI KASUS : DEPARTEMEN MATEMATIKA)**

MUHAMMAD ARI PUTRA RAMADHAN
H071171313

Skripsi,

Telah dipertahankan di depan Panitia Ujian Sarjana Sistem Informasi pada 24 Juli
2024 dan dinyatakan telah memenuhi syarat kelulusan

Pada

Program Studi Sistem Informasi
Departemen Matematika
Fakultas Matematika Dan Ilmu Pengetahuan Alam
Universitas Hasanuddin
Makassar



Mengesahkan:
Pembimbing tugas akhir,

Dr. Hendra, S.Si., M.Kom.
NIP. 197601022002121001

Pembimbing Pendamping,

A. Muh Amil Siddik, S.Si., M.Si
NIP. 199110032019031015

Mengetahui:
Ketua Program Studi,

Prof. Drs. Jeffry Kusuma, Ph.D.
Nip. 196411121987031002

PERNYATAAN KEASLIAN SKRIPSI DAN MELIMPahkan HAK CIPTA

Dengan ini saya menyatakan bahwa, skripsi berjudul “Implementasi Algoritma Genetika Pada Rancang Bangun Sistem Penjadwalan Mata Kuliah Departemen Berbasis *Website* (Studi kasus : Departemen Matematika)” adalah benar karya saya dengan arahan pembimbing dari (Dr. Hendra, S.Si., M.Kom. sebagai Pembimbing Utama dan A. Muh. Amil Siddik., M.Si. sebagai Pembimbing Pendamping). Karya ilmiah ini belum diajukan dan tidak sedang diajukan dalam bentuk apapun kepada perguruan tinggi mana pun. Sumber informasi yang berasal atau dikutip dari karya yang diterbitkan maupun tidak diterbitkan dari penulis lain telah disebutkan dalam teks dan dicantumkan dalam Daftar Pustaka skripsi ini. Apabila dikemudian hari terbukti atau dapat dibuktikan bahwa Sebagian atau keseluruhan skripsi ini adalah karya orang lain, maka saya bersedia menerima sanksi atas perbuatan tersebut berdasarkan aturan yang berlaku.

Dengan ini saya melimpahkan hak cipta (hak ekonomis) dari karya tulis saya berupa skripsi ini kepada Universitas Hasanuddin.

Makassar, 24 Juli 2024



Muhammad Ari Putra Ramadhan

H071171313

UCAPAN TERIMA KASIH

Segala puji dan syukur kehadiran Allah SWT atas rahmat dan hidayah-Nya yang senantiasa dilimpahkan kepada penulis, sehingga bisa menyelesaikan skripsi dengan judul "Implementasi Algoritma Genetika Pada Rancang Bangun Sistem Penjadwalan Mata Kuliah Berbasis *Website* (Studi Kasus: Departemen Matematika)" sebagai syarat memperoleh gelar Sarjana Komputer pada Program Studi Sistem Informasi Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

Dalam penyusunan skripsi ini tentunya banyaknya halangan dan rintangan yang penulis hadapi hingga membuat proses penyelesaian skripsi ini menjadi terhambat namun dari semua proses yang dilalui akhirnya penulis dapat menyelesaikan skripsi ini. Oleh karena itu, penulis ingin menyampaikan ucapan terima kasih kepada orang tua penulis, Ibunda **Hajja Hajar** dan Ayahanda **Suwardi**, yang tak hentinya berdoa dan memberikan motivasi sehingga skripsi ini dapat diselesaikan di waktu yang tepat bukan di waktu yang cepat karena skripsi bukan tentang sebuah perlombaan.

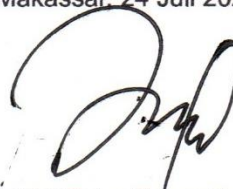
Penghargaan serta ucapan terima kasih dengan penuh rasa hormat dan ketulusan juga penulis ucapkan kepada:

1. Rektor Universitas Hasanuddin Makassar **Prof. Dr. Ir. Jamaluddin Jompa, M.Sc.** beserta seluruh Wakil Rektor dalam lingkup Universitas Hasanuddin.
2. Bapak **Dr. Eng. Amiruddin, M.Si.** selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin dan para Wakil Dekan serta seluruh staf yang telah berkontribusi memberikan bantuan selama penulis menempuh pendidikan di FMIPA Universitas Hasanuddin.
3. Bapak **Dr. Firman, S.Si., M.Si.** selaku Ketua Departemen Matematika FMIPA Unhas, penulis juga berterima kasih kepada dosen-dosen, serta staf Departemen atas dedikasinya dalam membantu dan memberikan ilmu yang sangat bermanfaat.
4. Bapak **Prof. Drs. Jeffry Kusuma, Ph.D.** selaku Ketua Program Studi Sistem Informasi Universitas Hasanuddin, penulis juga berterima kasih pada seluruh dosen-dosen program studi yang selalu memberikan ilmu yang bermanfaat dan motivasi untuk mahasiswa Sistem Informasi.
5. Bapak **Dr. Hendra, S.Si., M.Kom.** selaku dosen pembimbing utama yang selalu meluangkan waktu tenaga dan pikiran untuk memberikan arahan dalam penyusunan skripsi ini.
6. Bapak **A. Muh Amil Siddik, S.Si., M.Si** dosen pembimbing pertama yang selalu meluangkan waktu tenaga dan pikiran untuk memberikan arahan dalam penyusunan skripsi ini.
7. Kedua dosen penguji, bapak **Dr. Eng. Armin Lawi, S.Si., M.Eng**, dan ibu **Prof. Dr. Kasbawati, S.Si., M.Si.** yang telah memberikan masukan, kritik, dan saran dalam penelitian tugas akhir ini sehingga oleh karenanya skripsi ini dapat tersusun.
8. Kepada seluruh teman-teman seperjuangan dari Program Studi Sistem Informasi 2017.

9. Kepada saudara Arya Indrawan, Fadhil Hidayat Amin, Restu Adi Akbar, Muhammad Fitrah, Erwin Syahrul Hidayat, Aris Akhyar Abdillah dan Ihzanul Wajidi sebagai tempat bertukar pikiran, bertukar cerita dan selalu bersedia membantu dalam keseluruhan proses penyelesaian skripsi ini.
10. Kepada kakak-kakak Program Studi Sistem Informasi 2014, 2015, 2016.
11. Kepada kawan sekaligus sahabat seperjuangan Trinada Pamungkas, Jonadap Lambertus Matatula, Edo Bayu Pamungkas, Pardi Susila, Renaldi Akbar Rifai, Sri Rahayu Ningrum, Nurfaidah, dan Risdayanti yang selalu memberikan semangat, mengajak jalan-jalan untuk menyegarkan kembali isi kepala, nongkrong dan bersenda gurau dikala sedang pusing dalam mengerjakan skripsi.
12. Seluruh pihak yang telah berkontribusi untuk penulis baik yang sifatnya berupa materi maupun non-materi yang tidak dapat penulis sebutkan satu persatu, penulis mengucapkan banyak terima kasih atas bantuan dan dukungannya.

Akhir kata, saya berharap semoga Tuhan Yang Maha Esa dapat membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini kedepannya dapat membawa manfaat bagi pengembangan ilmu.

Makassar, 24 Juli 2024



Muhammad Ari Putra Ramadhan

H071171313

ABSTRAK

MUHAMMAD ARI PUTRA RAMADHAN. **Implementasi Algoritma Genetika Pada Rancang Bangun Sistem Penjadwalan Mata Kuliah Departemen Ber Basis Website (Studi Kasus : Departemen Matematika)** (dibimbing oleh Dr. Hendra, S.Si., M.Kom. dan A. Muh. Amil Siddik, S.Si., M.Si.).

Departemen Matematika Universitas Hasanuddin dalam pembuatan jadwal mata kuliah masih melakukan secara manual dengan mengurutkan dosen mana yang tidak bisa mengajar pada harinya. Namun cara tersebut membutuhkan waktu yang relatif lama dan harus dikerjakan secara cermat dan teliti agar tidak terjadi bentrok pada pembuatan jadwal mata kuliah, jika solusinya ditemukan, solusi tersebut mungkin bukanlah solusi yang terbaik. Penelitian ini bertujuan merancang dan membamgun aplikasi penjadwalan berbasis web menggunakan algoritma genetika. Adapun dalam pengimplementasiannya menggunakan bahasa pemrograman *Hypertext Preprocessor (PHP)* dan database yang digunakan adalah *MySQL*. Pada hasil penelitian menunjukkan bahwa algoritma genetika dapat diterapkan dalam penjadwalan mata kuliah dan membantu dalam proses penjadwalan yang lebih efisien.

Kata Kunci: algoritma genetika, aplikasi penjadwalan mata kuliah, *Hypertext Preprocessor (PHP)*, *MySQL*.

ABSTRACT

MUHAMMAD ARI PUTRA RAMADHAN. **Implementation of Genetic Algorithm in the Design and Development of a Web-Based Course Scheduling System** (Case Study: Department of Mathematics) (supervised by Dr. Hendra, S.Si., M.Kom. and A. Muh. Amil Siddik, S.Si., M.Si.).

The Department of Mathematics at Hasanuddin University still manually creates course schedules by arranging which lecturers are unavailable on certain days. However, this method takes a relatively long time and must be done carefully and meticulously to avoid conflicts in the course schedule. Even if a solution is found, it may not be the best solution. This research aims to design and develop a web-based scheduling application using a genetic algorithm. The implementation uses the Hypertext Preprocessor (PHP) programming language and the MySQL database. The research results show that the genetic algorithm can be applied to course scheduling and helps make the scheduling process more efficient.

Keywords: genetic algorithm, course scheduling application, Hypertext Preprocessor (PHP), MySQL.

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
PERNYATAAN PENGAJUAN.....	ii
HALAMAN PENGESAHAN.....	iii
PERNYATAAN KEASLIAN SKRIPS.....	iv
UCAPAN TERIMA KASIH.....	vi
ABSTRAK	viii
ABSTRACT	ix
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Teori	3
1.6 Penelitian Terkait	12
BAB II METODE PENELITIAN	16
2.1 Analisis Penelitian	16
2.2 Analisis Kebutuhan Non Fungsional	16
2.3 Analisis Kebutuhan Proses	16
2.4 Analisis Algoritm Genetika	17
2.5 Perancangan Unified Modeling Language (UML).....	17
2.6 Perancangan Tabel.....	30
2.7. Perancangan Relasi Antar Tabel	34
BAB III HASIL DAN PEMBAHASAN.....	35
3.1 Hasil	35
3.2 Pembahasan	42
BAB IV KESIMPULAN DAN SARAN	51

4.1 Kesimpulan	51
4.2 Saran.....	51
DAFTAR PUSTAKA.....	52

DAFTAR TABEL

Nomor Urut	Halaman
1. Simbol-simbol <i>Use Case Diagram</i> (Munawar, 2005)	10
2. Simbol <i>Activity Diagram</i> (Munawar, 2005)	10
3. Penelitian Terkait	14
4. <i>Use Case Model Diagram</i>	18
5. <i>Narrative</i> Kelola Data Mata Kuliah	20
6. <i>Narrative</i> Kelola Data Dosen	20
7. <i>Narrative</i> Kelola Data Program Studi	21
8. <i>Narrative Kelola Data Ruangan</i>	22
9. <i>Narrative</i> Kelola Data Kelas	23
10. <i>Narrative</i> Kelola Data Jam.....	23
11. <i>Narrative</i> Kelola Data Hari.....	24
12. <i>Narrative</i> Kelola Data Pengampu.....	24
13. <i>Narrative</i> Kelola Data Penjadwalan.....	25
14. Menemukan <i>Potential Object</i>	29
15. Menyeleksi Object yang diusulkan	30
16. Tabel Dosen	30
17. Tabel Hari	31
18. Tabel Jadwal Kuliah	31
19. Tabel Jam	32
20. Tabel Mata Kuliah.....	32
21. Tabel Ruangan	32
22. Tabel Program Studi.....	33
23. Tabel Kelas.....	33
24. Tabel Pengampu	33
25. Tabel Administrator	34
26. Data Dosen.....	43
27. Data Mata Kuliah	43
28. Data Ruang.....	43
29. Data Jam Kuliah	44

30. Data Hari Perkuliahan	44
31. Contoh Model Kromosom (S. Ni Luh Gede Pivin).....	44
32. Aturan dan nilai penalti (S. Ni Luh Gede Pivin)	46
33. Proses Crossover (S. Ni Luh Gede Pivin)	48
34. Proses Mutasi (S. Ni Luh Gede Pivin)	49

DAFTAR GAMBAR

Nomor Urut	Halaman
1. Flowchart Operasi Algoritma Genetika Sumber: (Suhartono, 2015).....	7
2. Use Case Diagram	19
3. <i>Activity Diagram</i> Proses <i>Login</i>	26
4. <i>Activity Diagram</i> Master Data	27
5. Activity Diagram Proses Data	28
6. Relasi Antar Tabel	34
7. Tampilan Halaman <i>Login</i>	35
8. Halaman Dashboard.....	36
9. Halaman Data Matakuliah	37
10. Halaman Data Dosen	37
11. Halaman Program Studi	38
12. Halaman Ruang.....	38
13. Data Kelas	39
14. Data Jam	39
15. Data Hari.....	40
16. Data Pengampu.....	40
17. Halaman Buat Jadwal.....	41
18. Hasil Penjadwalan	42

BAB I PENDAHULUAN

1.1 Latar Belakang

Departemen Matematika merupakan departemen di salah satu Universitas Hasanuddin yang berdiri sejak 17 Agustus 1963. Sejauh ini pihak akademik dalam pembuatan jadwal masih melakukan secara manual dengan mengurutkan dosen mana yang tidak bisa mengajar pada harinya. Namun cara tersebut membutuhkan waktu yang relatif lama dan harus dikerjakan secara cermat dan teliti agar tidak terjadi bentrok pada pembuatan jadwal mata kuliah, jika solusinya ditemukan, solusi tersebut mungkin bukanlah solusi yang terbaik. Ketidakpastian ini mendorong saya untuk melakukan kajian ilmiah dari masalah penjadwalan perkuliahan secara otomatis. Berdasarkan permasalahan tersebut, perlu dikembangkan suatu sistem yang dapat membuat penjadwalan perkuliahan secara otomatis, maka dari itu pada penelitian ini, saya menggunakan sistem metode dari algoritma genetika.

Pada penelitian sebelumnya yang dilakukan Wiga Ayu Puspaningrum, Arif Djunaidy, dan Retno Aulia Vinarty pada tahun 2013 dengan judul penelitian Penjadwalan Mata Kuliah Menggunakan Algoritma Genetika di Jurusan Sistem Informasi ITS. Penelitian ini memiliki tujuan untuk memecahkan masalah dalam pembuatan jadwal mata kuliah di jurusan Sistem Informasi ITS yang masih dilakukan dengan cara manual, sehingga menyebabkan adanya jadwal mata kuliah yang bentrok. Pada penelitian tersebut menggunakan algoritma genetika karena algoritma genetika merupakan salah satu jalan untuk memecahkan masalah yang cukup besar dengan solusi yang cukup baik meskipun masalah tersebut membutuhkan waktu eksekusi yang lama bila dilakukan secara manual. Berdasarkan penelitian tersebut, didapatkan hasil sistem yang menerapkan algoritma genetika dapat memenuhi aturan yang dibuat seperti tidak boleh ada jadwal mata kuliah yang bentrok (Puspaningrum, Djunaidy, & Vinarti, 2013).

Penelitian Fitria Larantika pada tahun 2015 dengan judul penelitian Sistem Penjadwalan Otomatis Menggunakan Algoritma Genetika di Fakultas Sains dan Teknologi. Penelitian ini memiliki tujuan untuk memecahkan masalah dalam pembuatan jadwal mata kuliah di Fakultas Sains dan Teknologi yang masih menggunakan cara manual. Penelitian tersebut menggunakan algoritma genetika untuk memecahkan masalah penjadwalan yang telah disebutkan. Algoritma genetika merupakan salah satu metode optimasi yang kuat dan bisa digunakan pada berbagai macam studi kasus, baik kasus yang sederhana hingga kasus yang rumit karena menggunakan prinsip teori evolusi. Berdasarkan penelitian tersebut, didapatkan hasil algoritma genetika dapat diterapkan pada sistem penjadwalan perkuliahan, sistem yang dibuat dapat menghasilkan jadwal dengan menjamin seorang dosen tidak mengajar dalam sehari penuh, dan sistem dapat menjamin mata kuliah paket semester tidak dijadwalkan berurutan dalam satu hari (Laranika, 2015).

Penelitian Entot Suhartono pada tahun 2015 dengan judul penelitian Optimasi Penjadwalan Mata Kuliah dengan Algoritma Genetika (Studi Kasus di AMIK Semarang). Penelitian ini memiliki tujuan untuk memecahkan masalah dalam

pembuatan jadwal penggunaan ruang laboratorium komputer yang masih menggunakan cara manual, sehingga menyebabkan adanya bentrok jadwal penggunaan laboratorium. Penelitian tersebut menggunakan algoritma genetika karena algoritma genetika cukup baik untuk digunakan dalam penjadwalan mata kuliah sebuah perguruan tinggi. Algoritma genetika merupakan salah satu jalan untuk memecahkan masalah waktu eksekusi yang lama bila dilakukan secara manual. Berdasarkan penelitian tersebut, didapatkan hasil bahwa dengan algoritma genetika penyusunan jadwal mata kuliah dapat dioptimalkan (Suhartono, 2015).

Penelitian selanjutnya dilakukan oleh Oktarina dan Hajjah pada tahun 2019 yang berjudul "Perancangan sistem penjadwalan proposal dan sidang skripsi dengan metode algoritma genetika". Hal inilah yang menjadi latar belakang untuk membuat sistem penjadwalan. Algoritma Genetika merupakan salah satu algoritma optimasi yang cukup handal dan sering dipakai dalam permasalahan penjadwalan (Oktarina & Hajjah, 2019).

Berdasarkan penelitian sebelumnya dan permasalahan yang terjadi maka peneliti mengusulkan Algoritma Genetika (GA) sebagai solusi penjadwalan. Oleh karena itu penelitian ini mengambil judul "Implementasi Algoritma Genetika Pada Rancang Bangun Sistem Penjadwalan Mata Kuliah Berbasis Web", di mana data mata kuliah, data dosen pengajar, dan hari perkuliahan akan di masukan pada sistem tersebut sehingga akan muncul jadwal terbaik yang dihasilkan.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan diatas, permasalahan yang akan dilakukan pada penelitian ini adalah :

1. Bagaimana merancang dan membangun aplikasi penjadwalan mata kuliah yang berbasis pada web di Departemen Matematika, Universitas Hasanuddin?
2. Bagaimana mengimplementasikan algoritma genetika untuk membuat sistem penjadwalan mata kuliah di Departemen Matematika, Universitas Hasanuddin ?

1.3 Batasan Masalah

Agar pengerjaan tugas akhir ini menjadi lebih terarah dan mendapatkan hasil yang lebih spesifik, maka sistem yang akan dirancang dibatasi pada lingkup pembahasan sebagai berikut:

1. Parameter masukan untuk implementasi sistem penjadwalan perkuliahan yaitu mata kuliah , dosen, kelas, ruangan, dan waktu(hari serta jam).
2. Data yang digunakan dalam penelitian ini merupakan data penjadwalan perkuliahan untuk Departemen Matematika.
3. Aplikasi yang akan dikembangkan dengan menggunakan web based dan menggunakan bahasa pemrograman PHP dan XAMPP serta mengimplementasikan Algoritma Genetika didalamnya.

1.4 Tujuan Penelitian

Berdasarkan latar belakang masalah yang telah diuraikan diatas tujuan penelitian yang akan dilakukan dalam penelitian ini adalah sebagai berikut:

1. Untuk merancang dan membangun sistem aplikasi berbasis web yang dapat digunakan untuk memudahkan dalam pembuatan sistem penjadwalan mata kuliah di Departemen Matematika, Universitas Hasanuddin.
2. Untuk mengimplementasikan algoritma genetika yang akan digunakan untuk membuat sistem penjadwalan mata kuliah di Departemen Matematika, Universitas Hasanuddin.

1.5 Teori

1.5.1 Departemen Matematika

Departemen Matematika didirikan pada tanggal 17 Agustus 1963 bersamaan dengan berdirinya Fakultas Ilmu Pengetahuan Alam, Universitas Hasanuddin. Selama beberapa tahun pertama, belum ada siswa yang terdaftar di program studi ini sehingga pada tahun 1974, dimana beberapa siswa diterima di tingkat diploma. Setelah lulus, para siswa tersebut harus melanjutkan studi mereka di Institut Teknologi Bandung untuk menyelesaikan gelar Sarjana mereka. Dimulai sejak tahun 1981, Program Studi ini menyelenggarakan program Sarjana dan sejak tahun 2005, Program Sarjana Matematika di akreditasi oleh Badan Akreditasi Nasional untuk Pendidikan Tinggi di Indonesia (BAN-PT). Akreditasi terakhir diberikan di peringkat BAN-PT "UNGGUL" dengan nomor akreditasi 8197/SK/BAN-PT/AKISK/S/XII/2020. Akreditasi ini berlaku dari tanggal 15-12-2020 hingga 04-09-2025 (Akreditasi Program Studi, 2022). Pada saat awal diselenggarakannya, dosen pada program studi Matematika adalah semua dosen pada Program Studi Matematika namun sejak tahun 1997 dosen pada Program Studi Matematika mulai dikelompokkan ke dalam Program Studi Matematika dan Program Studi Statistika berdasarkan spesifikasi dan bidang keahlian dari masing-masing dosen tersebut. Kemudian pada tahun 2014 dikelompokkan lagi menjadi tiga Program Studi yaitu Program Studi Matematika, Program Studi Statistika, dan Program Studi Ilmu Komputer. Namun pada tanggal 24 juli 2019, terbentuk lagi Program Studi Sarjana baru di Departemen Matematika yaitu Program Studi Ilmu Aktuaria. Dimana pada tahun 2019, Departemen Matematika terdiri dari tiga Program Studi yaitu Program Studi Matematika, Program Studi Ilmu Komputer, dan Program Studi Ilmu Aktuaria, dikarenakan pada tahun tersebut Program Studi Statistika memisahkan dari Departemen Matematika dan membentuk Departemennya sendiri yaitu Departemen Statistika. Dan pada tahun 2021 Program Studi Ilmu Komputer berganti nama menjadi Program Studi Sistem Informasi. (FMIPA MATEMATIKA UNHAS, 2022)

1.5.2 Rancang Bangun

Rancang bangun adalah tahap setelah analisis dari siklus pengembangan sistem yang merupakan pendefinisian dari kebutuhan-kebutuhan fungsional, serta menggambarkan bagaimana suatu sistem dibentuk yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi, termasuk menyangkut mengkonfirmasi dari komponen-komponen perangkat keras dan perangkat lunak dari semua sistem (Jogiyanto, 2005).

Berdasarkan pendapat di atas, dapat disimpulkan bahwa rancang bangun adalah tahap setelah analisis dari siklus pengembangan sistem yang merupakan pendefinisian dari kebutuhan-kebutuhan fungsional, serta menggambarkan bagaimana suatu sistem dibentuk yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi, termasuk menyangkut mengkonfirmasi dari komponen-komponen perangkat keras dan perangkat lunak dari semua sistem. Dalam hal ini, penulis merujuk pengertian rancang bangun menurut (Jogiyanto, 2005)

1.5.3 Website

Website merupakan sebuah media informasi yang ada di internet. Website adalah kumpulan dari halaman-halaman situs, yang biasanya terangkum dalam sebuah domain atau subdomain, yang tempatnya berada di dalam World Wide Web (WWW) di Internet. Sebuah halaman web adalah dokumen yang ditulis dalam format HTML (*Hyper Text Markup Language*), yang hampir selalu bisa diakses melalui HTTP, yaitu protokol yang menyampaikan informasi dari server website untuk ditampilkan kepada para pemakai melalui web browser (Haryana, 2008).

1.5.4 Penjadwalan

Penjadwalan adalah pengurutan pembuatan atau pengerjaan produk secara menyeluruh yang dikerjakan pada beberapa buah mesin. Dengan demikian masalah *sequencing* senantiasa 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 (Ginting, 2009)

Berdasarkan uraian, penulis dapat menyimpulkan bahwa penjadwalan mempunyai fungsi sebagai suatu petunjuk untuk pengaturan waktu serta mengalokasikan sumber-sumber yang ada guna mencapai tujuan yang telah ditetapkan sehingga penjadwalan dapat diselesaikan tepat waktu sesuai rencana yang telah ditetapkan.

1.5.5 Definisi Algoritma Genetika

Algoritma genetika pertama kali dikembangkan oleh John Holland dari Universitas Michigan pada tahun 1975. John Holland menyatakan bahwa setiap masalah yang berbentuk adaptasi (alami maupun buatan) dapat diformulasikan ke dalam terminologi genetika. Algoritma genetika adalah simulasi dari proses evolusi darwin dan operasi genetika atas kromosom (Kusumadewi, 2003).

Algoritma genetika mengikuti prosedur atau tahap-tahap yang menyerupai proses evolusi, yaitu adanya proses seleksi, crossover dan mutasi. Pada setiap generasi, himpunan baru dari deretan individu dibuat berdasarkan kecocokan pada generasi sebelumnya. Keberagaman pada evolusi biologis adalah variasi dari kromosom dalam individu organisme. Variasi kromosom ini akan mempengaruhi laju reproduksi dan tingkat kemampuan organisme untuk tetap hidup (Puspaningrum, Djunaidy, & Vinarti, 2013).

Teknik pencarian yang dilakukan algoritma genetika dari himpunan solusi secara acak disebut dengan populasi. Sedangkan setiap individu dalam populasi disebut kromosom. Kromosom berevolusi dalam suatu proses iterasi yang berkelanjutan yang disebut generasi. Pada setiap generasi, kromosom akan dievaluasi berdasarkan fungsi fitness. Setelah beberapa generasi, maka algoritma genetika akan konvergen pada kromosom terbaik yang diharapkan merupakan solusi optimal.

Pendekatan yang diambil oleh algoritma genetika adalah dengan menggabungkan secara acak berbagai pilihan solusi terbaik di dalam suatu populasi untuk mendapatkan individu baru (offspring) yaitu pada suatu kondisi yang memaksimalkan kecocokan yang disebut fitness. Algoritma genetika adalah algoritma pencarian heuristik yang didasarkan atas mekanisme evolusi biologis (Kusumadewi, 2003).

Algoritma genetika sangat tepat digunakan untuk menyelesaikan permasalahan optimasi yang kompleks. Di dalam algoritma genetika, solusi permasalahan direpresentasikan ke dalam bentuk kromosom. Tiga aspek yang penting untuk penggunaan algoritma genetika yaitu:

1. Fungsi *fitness*.
2. Implementasi representasi genetik berupa kromosom.
3. Implementasi operasi genetik berupa operator *crossover* dan mutasi.

1.5.6 Parameter Algoritma Genetika

Dalam algoritma genetika terdapat beberapa parameter yang digunakan dalam pemrosesannya, diantaranya yaitu:

1. Fungsi *fitness* untuk menentukan tingkat kesesuaian individu tersebut.
2. Populasi jumlah individu pada setiap generasi.
3. Probabilitas terjadinya *crosso*
4. Probabilitas terjadinya mutasi pada setiap generasi.
5. Jumlah generasi yang akan dibentuk.

1.5.7 Karakteristik Algoritma Genetika

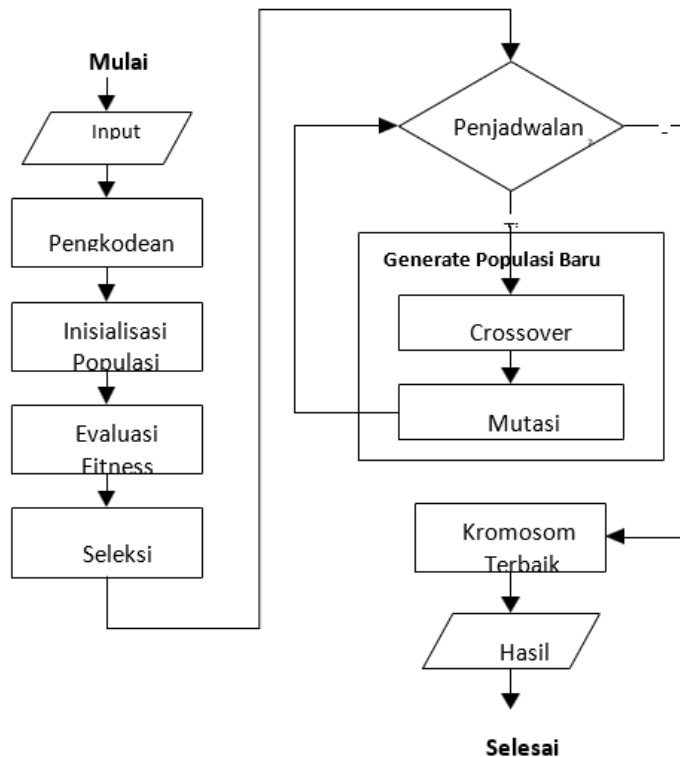
Goldberg (Kusumadewi, 2003) mengemukakan bahwa algoritma genetika mempunyai karakteristik - karakteristik yang perlu diketahui sehingga dapat dibedakan dari prosedur pencarian atau optimasi yang lain, yaitu :

1. Algoritma genetika dengan pengkodean dari himpunan solusi permasalahan berdasarkan parameter yang telah ditetapkan bukan parameter itu sendiri.
2. Algoritma genetika pencarian pada sebuah solusi dari sejumlah individu - individu yang merupakan solusi permasalahan bukan hanya dari sebuah individu.
3. Algoritma genetika informasi fungsi objektif (*fitness*), sebagai cara untuk mengevaluasi individu yang mempunyai solusi terbaik, bukan turunan dari suatu fungsi.
4. Algoritma genetika menggunakan aturan-aturan transisi peluang, bukan aturan - aturan deterministik.

1.5.8 Struktur Umum Algoritma Genetika

Pada ilustrasi Gambar 2.1, struktur umum dari suatu algoritma genetika dapat didefinisikan dengan langkah-langkah sebagai berikut (Hanita, 2011):

1. Menghasilkan populasi awal (generate)
Populasi awal di generate secara random, dimana populasi tersebut terdiri dari beberapa kromosom yang telah didefinisikan sehingga dapat dijadikan solusi awal. Populasi tersebut terdiri dari beberapa kromosom yang mewakili solusi yang diinginkan.
2. Membentuk generasi baru
Membuat generasi baru menggunakan tiga operator yaitu operator seleksi, operator crossover dan operator mutase. Tahapan tersebut dilakukan secara berulang-ulang sehingga diperoleh sejumlah kromosom yang cukup untuk menghasilkan generasi baru, dimana generasi baru tersebut merupakan representasi dari solusi baru.
3. Evaluasi solusi
Pada tiap generasi, kromosom akan diukur dengan fungsi fitness. Nilai fitness suatu kromosom menggambarkan kualitas kromosom dalam populasi tersebut. Proses ini akan mengevaluasi setiap populasi dengan menghitung nilai fitness setiap kromosom dan mengevaluasinya sampai terpenuhi kriteria berhenti. Apabila kriteria berhenti belum terpenuhi maka akan dibentuk lagi generasi baru dengan mengulangi langkah kedua. Beberapa kriteria berhenti yang sering digunakan antara lain:
 - a. Berhenti pada generasi tertentu.
 - b. Berhenti setelah beberapa generasi berturut-turut didapatkan nilai fitness tertinggi.
 - c. Berhenti pada n generasi dimana nilai fitness dari populasi tidak mengalami perubahan.



Gambar 1 Flowchart Operasi Algoritma Genetika Sumber: (Suhartono, 2015)

1.5.9 Komponen-komponen Utama Algoritma Genetika

Ada 6 komponen utama yang terdapat di dalam algoritma genetika, (Hanita, 2011) yaitu:

1. Teknik Pengkodean

Suatu teknik bagaimana mengkodekan gen dari kromosom. Teknik ini merupakan teknik untuk menyatakan populasi awal sebagai calon solusi suatu masalah ke dalam suatu kromosom sebagai suatu kunci pokok masalah. Teknik pengkodean meliputi pengkodean gen dan kromosom. Gen adalah bagian dari kromosom yang dapat direpresentasikan dalam bentuk string bit, tree, array bilangan real, daftar aturan, elemen permutasi, elemen program, atau representasi lain yang dapat diimplementasikan untuk operator genetika.

2. Prosedur Inisialisasi

Suatu proses yang menghasilkan sejumlah individu secara acak (random). Banyaknya populasi tergantung pada masalah yang akan diselesaikan dan jenis operator genetika yang akan diterapkan. Setelah jumlah populasi ditentukan, selanjutnya dilakukan inisialisasi terhadap kromosom yang ada di dalam populasi tersebut. Inisialisasi kromosom dilakukan secara acak, dengan tetap memperhatikan domain solusi dan kendala permasalahan yang ada.

3. Fungsi Evaluasi

Individu dievaluasi berdasarkan fungsi tertentu sebagai ukuran kinerjanya. Individu dengan nilai fitness tinggi pada kromosomnya yang akan dipertahankan, sedangkan individu yang pada kromosomnya bernilai fitness rendah akan diganti. Fungsi fitness tergantung pada permasalahan tertentu dari representasi yang digunakan.

4. Seleksi

Proses seleksi bertujuan untuk memilih individu-individu yang akan dipilih untuk proses persilangan dan mutasi, sehingga akan diperoleh calon induk yang baik. Induk yang baik akan menghasilkan keturunan yang baik. Langkah pertama dalam seleksi yaitu pencarian nilai fitness. Masing-masing individu dalam suatu wadah seleksi akan menerima probabilitas reproduksi yang tergantung pada nilai objektif dirinya sendiri terhadap nilai objektif dari semua individu dalam wadah seleksi tersebut. Nilai fitness inilah yang nantinya akan digunakan pada tahap seleksi berikutnya.

5. Operator Genetika

Algoritma genetika merupakan proses pencarian yang heuristik dan acak sehingga penekanan pemilihan operator yang digunakan sangat menentukan keberhasilan algoritma genetik dalam menemukan solusi optimum suatu masalah yang diberikan. Hal yang harus diperhatikan adalah menghindari terjadinya konvergensi premature, yaitu mencapai solusi optimal yang belum waktunya, dalam arti bahwa solusi yang diperoleh adalah hasil optimum lokal. Ada dua operator genetika yaitu:

a. *Crossover* (Persilangan)

Crossover merupakan proses di dalam algoritma genetika yang bekerja untuk menggabungkan dua kromosom parent menjadi kromosom baru (offspring) pada suatu waktu. Sebuah kromosom yang mengarah pada solusi baik dapat diperoleh melalui proses crossover pada dua buah kromosom. Cara sederhana pada proses crossover yaitu dengan memilih satu titik yang dipisahkan secara acak dan kemudian membentuk offspring dengan mengkombinasikan segmen dari satu induk ke sebelah kiri dari titik yang dipisahkan dengan segmen dari induk yang lain ke sebelah kanan dari titik yang dipisahkan.

b. *Mutation* (Mutasi)

Proses ini merupakan untuk mengubah salah satu atau lebih beberapa gen dari suatu kromosom. Proses ini berfungsi untuk menggantikan gen yang hilang dari populasi akibat proses seleksi yang memungkinkan munculnya kembali gen yang tidak muncul pada inisialisasi populasi. Beberapa cara operasi mutasi yang diterapkan dalam algoritma genetika menurut jenis pengkodeannya.

6. Parameter Kontrol

Parameter kontrol genetika diperlukan untuk mengendalikan operator-operator seleksi. Pemilihan parameter genetika menentukan penampilan kinerja algoritma genetika dalam memecahkan masalah (Desiani dan Arhami, 2006).

Ada dua parameter dasar dari algoritma genetika, yaitu probabilitas crossover (P_c) dan probabilitas mutasi (P_m).

a. Probability Crossover (P_c)

Probabilitas crossover akan mengendalikan operator crossover dalam setiap generasi dalam populasi yang mengalami crossover. Semakin besar nilai probabilitas crossover, akan semakin cepat struktur individu baru terbentuk ke dalam populasi. Apabila nilai probabilitas crossover terlalu besar, maka individu yang merupakan kandidat solusi terbaik mungkin akan dapat hilang lebih cepat pada generasi selanjutnya. Nilai probabilitas crossover yang disarankan adalah berkisar antara 80 % - 95 %.

b. Probability Mutation (P_m)

Probabilitas mutasi akan mengendalikan operator mutasi pada setiap generasi dengan peluang mutasi yang digunakan lebih kecil daripada peluang crossover. Pada seleksi alam murni, mutasi jarang sekali muncul, sehingga operator mutasi pada algoritma genetik tidak selalu terjadi. Nilai probabilitas mutasi yang disarankan kecil antara 0.5% - 1%.

1.5.10 Unified Modeling Language (UML)

UML yang biasa disebut (*Unified Modeling Language*) adalah sebuah bahasa pemodelan untuk sistem atau *software* yang berkonsep berorientasi objek. UML seharusnya digunakan untuk perancangan model sebuah sistem yang lengkap sedemikian rupa sehingga sangat mudah untuk dipelajari dan di pahami (Nugraha & Setiawan, 2016) . UML (*Unified Modelling Language*) adalah alat bantu yang menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain (Munawar, 2005).

1. Tujuan UML

Adapun tujuan dari *UML* adalah sebagai berikut :

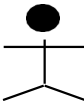


- a. Memberikan model yang siap pakai, bahasa pemodelan visual yang *ekspresif* untuk mengembangkan model dan dimengerti secara umum.
- b. Memberikan bahasa pemodelan yang bebas dari berbagai bahasa pemrograman dan proses rekayasa.
- c. Menyatukan praktek-praktek yang terdapat dalam pemodelan.

2. Diagram-diagram dalam UML




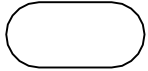

Ada beberapa diagram dalam UML (*Unified Modelling Language*) antara lain:


- a. *Use Case Diagram*, menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-*create* sebuah daftar belanja dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use case diagram* juga mempunyai simbol-simbol yang dapat dilihat di Tabel 1.

Tabel 1 Simbol-simbol *Use Case Diagram* (Munawar, 2005)

No.	Simbol	Nama	Keterangan
1.		<i>Actor</i>	Merupakan kesatuan <i>eksternal</i> yang berinteraksi dengan sistem.
2.		<i>Use Case</i>	Rangkaian / uraian kelompok yang saling terkait dan membentuk sistem.
3.		<i>Relationship</i>	Hubungan antara Pelaku/Aktor dengan <i>Use case</i>

Tabel 2 Simbol *Activity Diagram* (Munawar, 2005)

No.	Simbol	Nama	Keterangan
1.		<i>Initial State</i>	Titik awal dimulai <i>activity</i> .
2.		<i>Final State</i>	<i>Finish Akhir Activity</i>
3.		<i>State</i>	<i>Initial Activity</i>
4.		<i>Action State</i>	<i>Activity</i>
5.		<i>Decision</i>	Pilihan untuk Mengambil keputusan

6.		Fork	Menunjukkan kegiatan yang dilakukan secara paralel
----	---	------	--

- b. *Activity Diagram*, menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir. *Activity diagram* juga mempunyai beberapa simbol yang dapat dilihat pada tabel 2.
- c. *Collaboration Diagram*, menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, dimana *message* dari *level* tertinggi memiliki nomor 1. *Message* dari *level* yang sama memiliki *prefiks* yang sama.
- d. *Statechart Diagram*, menggambarkan transisi dan perubahan keadaan (dari satu *state* ke *state* lainnya) suatu objek pada sistem sebagai akibat dari *stimulus* yang diterima. Pada umumnya *statechart diagram* menggambarkan *class* tertentu (satu *class* dapat memiliki lebih dari satu *statechart diagram*).
- e. *Sequence Diagram*, menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna dan *display*) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* terdiri atas dimensi *vertikal* (waktu) dan dimensi *horizontal* (objek-objek yang terkait).
- f. *Class Diagram*, *Class* adalah sebuah spesifikasi yang jika di instansi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class Diagram* menggambarkan keadaan (atribut/*property*) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).
- g. *Component Diagram*, menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya.
- h. *Deployment Diagram*, menggambarkan detail bagaimana komponen di-*deploy* dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, *server* atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi *server*, dan hal-hal lain yang bersifat fisik.

1.5.11 Hypertext Preprocessor (PHP)

PHP merupakan kepanjangan dari *Hypertext Preprocessor* adalah suatu bahasa pemrograman berbasis kode-kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke web browser menjadi kode HTML (Oktavian, 2010). PHP juga digunakan sebagai bahasa *script server-side* dalam mengembangkan web yang disisipkan pada dokumen HTML (Peranginangin, 2006). Penggunaan PHP memungkinkan web dapat dibuat dinamis sehingga maintenance situs web tersebut menjadi lebih mudah dan efisien. Sintaks program/*script* HP ditulis dalam apitan tanda Khusus PHP.

1.5.12 MySQL

Menurut (Nugroho, 2004), *MySQL* adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user, serta menggunakan perintah standar SQL. *MySQL* merupakan *Free Software* dibawah lisensi GNU/GPL (*General Public License*). *MySql* menggunakan suatu format standar SQL (*Structure query language*) yakni bahasa yang berisi perintah-perintah untuk memanipulasi *data base*, mulai dari melakukan perintah *select* untuk menampilkan isi *database*, menginsert atau menambahkan isi ke dalam data base, mendelete atau menghapus isi *database* dan mengedit *data base*. Dalam SQL terdapat beberapa elemen komponen yang penting, pertama DDL (*Data Definition Language*) adalah *statement* yang berhubungan dengan pembuatan objek dan pengolahan strukturnya.

1.5.13 XAMPP

(Mawaddah & Fauzi, 2018) menyatakan bahwa XAMPP adalah software yang didalamnya terdapat server MySQL dan didukung oleh PHP sebagai bahasa pemrograman untuk membuat website dinamis serta terdapat web server apache yang dapat dijalankan di beberapa platform seperti OS X, Windows, Linux, Mac, dan Solaris. (Iqbal, 2019) menyatakan XAMPP merupakan software server apache dimana dalam XAMPP yang telah tersedia database server seperti MySQL dan PHP programming. XAMPP memiliki keunggulan yaitu cukup mudah dioperasikan, tidak memerlukan biaya serta mendukung instalasi pada Windows dan linux.

Dari pengertian diatas disimpulkan bahwa XAMPP merupakan software server apache di mana memiliki banyak keuntungan seperti mudah untuk digunakan, tidak memerlukan biaya serta mendukung pada instalasi Windows dan Linux. Hal ini juga didukung karena dengan instalasi yang dilakukan satu kali tersedia MySQL, apache web server, Database server PHP support

1.6 Penelitian Terkait

Saat ini SIG telah banyak diaplikasikan, seperti sebuah penelitian yang berjudul Sistem Informasi Geografis Pemetaan Penyebaran Penyakit Berbasis Web, penulis Putu Kurniawan Adi Krisna I Nyoman Piarsa dengan kesimpulan Sistem Informasi Geografis Pemetaan Penyebaran Penyakit Berbasis Web dapat diakses cepat, mudah dan memerlukan jaringan internet serta memberikan informasi mengenai pemetaan penyebaran penyakit yang mencakup wilayah Indonesia.

Dalam penelitian lainnya berjudul Perancangan Sistem Informasi Geografis Pemetaan Sebaran Kasus Covid-19 di Kabupaten Garut, penulis Asri Mulyani dengan kesimpulan sistem informasi geografis pemetaan sebaran kasus Covid-19 ini dirancang menggunakan metodologi Rational Unified Process (RUP), serta implementasi perancangan program sistemnya menggunakan PHP dan library leaflete, Dalam sistem informasi pemetaan sebaran kasus Covid-19 ini terdapat fitur peta sebaran Covid - 19, menampilkan informasi tentang Covid-19 dan kontak layanan Covid-19, serta menampilkan peta lokasi rumah sakit rujukan Covid-19.

Dengan adanya web ini diharapkan mampu mempermudah masyarakat untuk pencarian informasi Covid-19 di kabupaten Garut.

Sebuah penelitian berjudul Aplikasi Pemantauan Dan Sebaran Covid-19 Berbasis Mobile Dan GIS penulis, Hendryco, Wilda Susanti dan Irwan dengan kesimpulan produk yang dihasilkan dari penelitian ini adalah aplikasi pemantauan dan sebaran Covid-19 berbasis mobile yang akan diterapkan kepada pendatang yang akan memasuki zona merah dan zona hitam yang akan perlu dipantau oleh pihak berwajib. Dalam penelitian ini, masih terdapat beberapa kekurangan dikarenakan terdapat hambatan seperti keterbatasan waktu dan ilmu pengetahuan yang dimiliki. Diharapkan untuk penelitian selanjutnya sistem informasi geografis pemetaan sebaran kasus Covid-19 dapat mengelola data yang lebih lengkap dan luas, seperti data mengenai identitas pasien Covid-19 dan status pasiennya, dan mengembangkan fitur lainnya pada sistem agar lebih baik.

Tabel 3. Penelitian Terkait

No	Nama Peneliti	Tahun	Judul	Hasil Penelitian	Perbandingan Penelitian
1	Putu Kurniawan Adi Krisna I Nyoman Piarsa	2014	Sistem Informasi Geografis Pemetaan Penyebaran Penyakit Berbasis Web	Sistem Informasi Geografis Pemetaan Penyebaran Penyakit Berbasis Web dapat diakses cepat, mudah dan memerlukan jaringan internet serta memberikan informasi mengenai pemetaan penyebaran penyakit yang mencakup wilayah Indonesia.	Pada penelitian sebelumnya fitur terbatas pada informasi intansi kesehatan. Sedangkan pada penelitian ini terdapat fitur tambahan mulai dari maker lokasi, detail dan tampilan goggle maps intansi kesehatan
2	Asri Mulyani	2021	Perancangan Sistem Informasi Geografis Pemetaan Sebaran Kasus Covid-19 di Kabupaten Garut	sistem informasi geografis pemetaan sebaran kasus Covid-19 ini dirancang menggunakan metodologi Rational Unified Process (RUP), serta implementasi perancangan program sistemnya menggunakan PHP dan library leaflate, Dalam sistem informasi pemetaan sebaran kasus Covid-19 ini terdapat fitur peta sebaran Covid - 19,	Pada penelitian sebelumnya berbasis <i>website</i> dan dibangun dengan PHP. Sedangkan pada penelitian ini berbasis <i>mobile</i> dan terdapat berbagai fitur tambahan seperti informasi akun, informasi intansi kesehatan dan infografis.

				menampilkan informasi tentang Covid-19 dan kontak layanan Covid-19, serta menampilkan peta lokasi rumah sakit rujukan Covid-19. Dengan adanya web ini diharapkan mampu mempermudah masyarakat untuk pencarian informasi Covid-19 di kabupaten Garut.	
3	Hendryco, Wilda Susanti dan Irwan	2020	Aplikasi Pemantauan Dan Sebaran Covid-19 Berbasis Mobile Dan GIS	Aplikasi pemantauan dan sebaran Covid-19 berbasis mobile yang akan diterapkan kepada pendatang yang akan memasuki zona merah dan zona hitam yang akan perlu dipantau oleh pihak berwajib.	Pada penelitian sebelumnya sistem informasi yang dibangun terbatas pada fitur menu yang di sajikan dan fitur tracking yang belum selesai dibuat Sedangkan pada penelitian ini memiliki fitur menu infografis, intansi kesehatan, informasi statistik covid-19 dan informasi akun. Kemudian terdapat menu tracking yang sudah jadi.

BAB II METODE PENELITIAN

2.1 Analisis Penelitian

Jenis penelitian yang digunakan dalam pengembangan sistem penjadwalan mata kuliah yaitu berdasarkan fungsinya merupakan penelitian terapan dan berdasarkan tujuannya merupakan penelitian kasus karena disusun sesuai dengan permasalahan dalam penjadwalan yang berada di Departemen Matematika Universitas Hasanuddin. Sebelum aplikasi digunakan, maka aplikasi harus dipastikan dapat berjalan dengan semestinya dan menghasilkan *output* yang diharapkan. Perlu dilakukan pengujian produk untuk menemukan kesalahan yang mungkin dapat terjadi dan tahap pengujian ini dilakukan agar aplikasi dapat berjalan dengan baik.

2.2 Analisis Kebutuhan Non Fungsional

2.2.1 Analisis Perangkat

Dalam proses perancangan sistem penjadwalan mata, kebutuhan perangkat dibagi menjadi dua macam, yaitu kebutuhan perangkat lunak (*Software*) dan kebutuhan perangkat keras (*Hardware*).

1. Kebutuhan perangkat lunak
 - a. Sistem Operasi Windows 10 32/64 bit
 - b. Xampp
 - c. Web Browser
2. Kebutuhan perangkat keras yang dibutuhkan berupa komputer dengan spesifikasi:
 - a. *Processor* : Intel® Core™ i5-8300H CPU @ 2.30GHz(8 CPUs), 2.3GHz
 - b. RAM 8 GB

2.2.2 Analisis Kebutuhan Data

Pada penelitian ini penulis menggunakan data pada semester genap tahun akademik 2022-2023. Data tersebut berupa data dosen, data mata kuliah, data ruang baik ruang kelas maupun ruang lab, serta data jam dan hari perkuliahan.

2.3 Analisis Kebutuhan Proses

Kebutuhan proses dalam sistem penjadwalan mata kuliah antara lain:

- a. Proses input data dosen, mata kuliah, jam, hari, ruang, pengampu, dan waktu tidak bersedia dosen.
- b. Proses kelola data seperti tampil, *edit*, cari, dan *delete* pada data yang telah dimasukkan.
- c. Proses pembuatan jadwal mata kuliah berdasarkan semester dan tahun akademik.

2.4 Analisis Algoritm Genetika

2.4.1 Analisis Kebutuhan Sistem

Pembuatan representasi kromosom untuk jadwal perkuliahan, diperlukan slot waktu jam perkuliahan. Diasumsikan bahwa kegiatan perkuliahan diselenggarakan pada pukul 07.30 sampai dengan pukul 18.10 dengan durasi 1 sks adalah 50 menit sehingga terdapat 12 slot waktu yang tersedia setiap harinya. Selanjutnya, kromosom akan menampung nilai-nilai dan parameter *fitness* yang kemudian akan digunakan dalam operasi genetik. Penentuan populasi awal adalah proses membangkitkan sejumlah kromosom secara acak. Kromosom menyatakan salah satu alternatif solusi.

2.4.2 Penentuan Fungsi *Fitness*

Suatu individu dievaluasi berdasarkan suatu fungsi tertentu sebagai ukuran performasinya. Di dalam teori algoritma genetika, individu yang bernilai *fitness* tinggi akan bertahan hidup, sedangkan individu yang bernilai *fitness* rendah akan mati. Fungsi yang digunakan untuk mengukur nilai kecocokan atau derajat optimalitas suatu kromosom disebut dengan *fitness function*. Nilai yang dihasilkan oleh fungsi *fitness* merepresentasikan seberapa banyak persyaratan yang dilanggar, sehingga dalam kasus penjadwalan mata kuliah semakin kecil jumlah pelanggaran yang dihasilkan maka solusi yang dihasilkan akan semakin baik. Untuk setiap pelanggaran akan diberikan nilai 1. Agar tidak terjadi nilai *fitness* yang tak terhingga maka jumlah total semua pelanggaran akan ditambahkan 1.

2.4.3 Crossover Kromosom

Crossover merupakan proses pertukaran nilai gen pada posisi gen yang sama dari kromosom asal. Pada proses *crossover* juga harus dilakukan pengecekan apakah kromosom yang dihasilkan sudah sesuai dengan aturan yang berlaku.

2.4.4 Mutasi Kromosom

Proses mutasi ini adalah suatu proses eksploitasi terhadap terhadap kemungkinan modifikasi pada jadwal yang telah ada. Perubahan posisi beberapa mata kuliah dapat membuat solusi duplikasi ini menjadi memiliki nilai *fitness* yang lebih rendah maupun lebih tinggi. Mutasi dapat dilakukan dengan dua cara, yaitu cara acak dan cara penukaran. Mutasi cara pertama adalah dengan menentukan dua gen yang akan dimutasi. Setelah nilai kedua gen tersebut di acak ulang untuk mendapatkan nilai yang baru. Pada cara kedua adalah dengan menukar langsung nilai dari suatu gen pada posisi gen yang sama.

2.5 Perancangan Unified Modeling Language (UML)

Didalam sistem penjadwalan mata kuliah perancangan UML digambarkan dalam bentuk *Use Case Diagram* dan *Use Case Specification* sehingga dapat diketahui fungsi-fungsi yang terdapat dalam sistem penjadwalan mata kuliah yang akan dibuat.

2.5.1 Identifikasi Pelaku (*actor*)

Actor dalam sistem penjadwalan mata kuliah ini adalah Administrasi Akademik selaku petugas yang memiliki kewenangan untuk membuat jadwal perkuliahan di Departemen Matematika Universitas Hasanuddin.

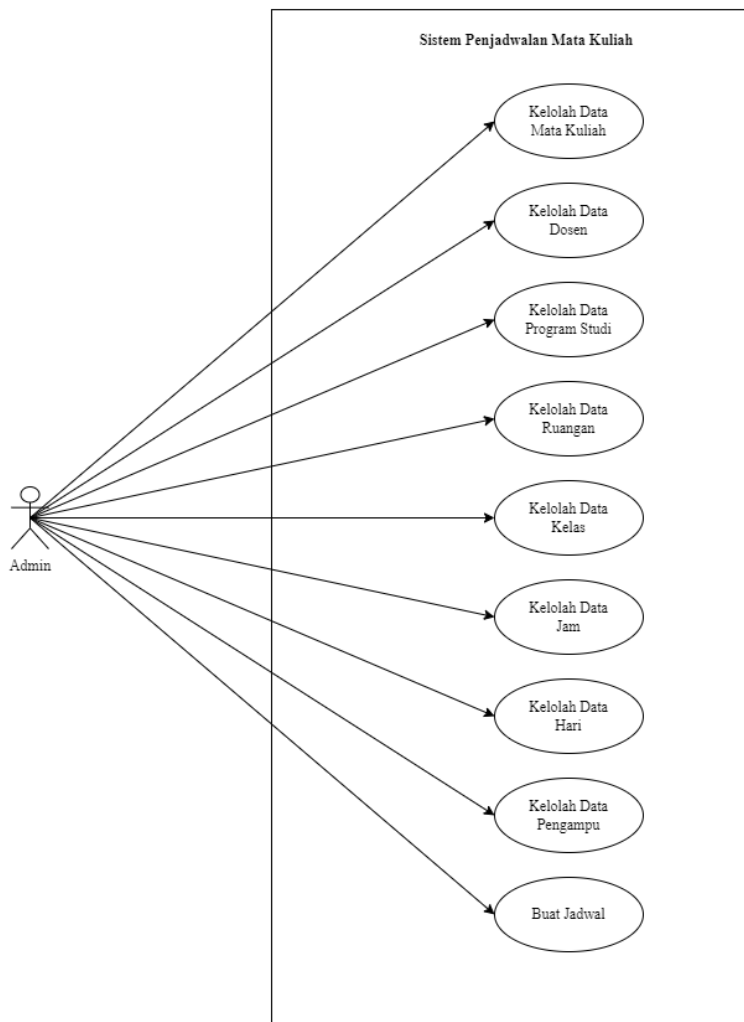
2.5.2 Pembuatan Use Case Model Diagram

Setelah *actor* teridentifikasi, *use case* model diagram digunakan untuk menggambarkan secara grafis lingkup dan Batasan dalam sistem. Tabel berikut ini merupakan tabel *use case* :

Tabel 4 Use Case Model Diagram

No	Nama Usecase	Deskripsi	Pelaku Yang Berpartisipasi
1	Kelola data mata kuliah	<i>Use Case</i> menggambarkan kegiatan <i>input</i> , <i>edit</i> , <i>delete</i> , dan <i>update</i> data mata kuliah	Administrasi Akademik
2	Kelola data dosen	<i>Use Case</i> menggambarkan kegiatan <i>input</i> , <i>edit</i> , <i>delete</i> , dan <i>update</i> data dosen	Administrasi Akademik
3	Kelola data Program studi	<i>Use Case</i> menggambarkan kegiatan <i>input</i> , <i>edit</i> , <i>delete</i> , dan <i>update</i> data program studi	Administrasi Akademik
4	Kelola data ruangan	<i>Use Case</i> menggambarkan kegiatan <i>input</i> , <i>edit</i> , <i>delete</i> , dan <i>update</i> data ruangan	Administrasi Akademik
5	Kelola data kelas	<i>Use Case</i> menggambarkan kegiatan <i>input</i> , <i>edit</i> , <i>delete</i> , dan <i>update</i> data kelas	Administrasi Akademik
6	Kelola data jam	<i>Use Case</i> menggambarkan kegiatan <i>input</i> , <i>edit</i> , <i>delete</i> , dan <i>update</i> data jam	Administrasi Akademik
7	Kelola data hari	<i>Use Case</i> menggambarkan kegiatan <i>input</i> , <i>edit</i> , <i>delete</i> , dan <i>update</i> data hari	Administrasi Akademik
8	Kelola data pengampu	<i>Use Case</i> menggambarkan kegiatan <i>input</i> , <i>edit</i> , <i>delete</i> , dan <i>update</i> data pengampu	Administrasi Akademik
8	Kelola data buat jadwal	<i>Use Case</i> menggambarkan kegiatan pembuatan jadwal berdasarkan semester dan tahun akademik	Administrasi Akademik

Pada *use case* dibawah ini adalah gambaran dari *use case model diagram* sistem penjadwalan mata kuliah.



Gambar 2. Use Case Diagram

2.5.3 Pembuatan Use Case Model Diagram

Narasi *use case* merupakan pemaparan naratif penjelasan tentang kegiatan yang dilakukan oleh *actor* dan respon yang diberikan oleh sistem sesuai dengan yang terjadi pada perangkat lunak Sistem Penjadwalan Mata Kuliah.

1. Kelola Data Mata Kuliah

Tabel 5 merupakan tabel kelola data Mata Kuliah memiliki fungsi yang digunakan untuk mengelola data dosen seperti tambah data, ubah data, lihat data, cari data, dan hapus data mata kuliah.

Tabel 5. *Narrative* Kelola Data Mata Kuliah

Nama <i>usecase</i>	Kelola Data Mata Kuliah	
Aktor	Kaur Administrasi Akademik	
Deskripsi	Menampilkan menu dari modul mata kuliah berupa cari, tampil, input, edit, dan delete data mata kuliah	
Prekondisi	Aktor memilih menu mata kuliah dari navbar	
Nama <i>usecase</i>	Kelola Data Mata Kuliah	
Normal flow	Aktor respon	Sistem respon
	Fungsi cari : user memasukkan nama mata kuliah yang ingin dicari dan klik tombol cari	Menampilkan data mata kuliah yang dicari sesuai dengan nama yang dimasukkan
	Fungsi input : klik tombol tambah data	Menampilkan menu input mata kuliah
	Fungsi edit data : klik ikon edit pada nama mata kuliah	Menampilkan menu edit data mata kuliah
	Fungsi hapus data : klik ikon hapus pada nama mata kuliah	Menampilkan <i>popup</i> pilihan <i>yes/no</i>

2. Kelola Data Dosen

Tabel 6 merupakan tabel kelola data dosen memiliki fungsi yang digunakan untuk mengelola data dosen, seperti tambah data, ubah data, lihat data, cari data, dan hapus data dosen.

Tabel 6 *Narrative* Kelola Data Dosen

Nama <i>usecase</i>	Kelola Data Dosen
Aktor	Kaur Administrasi Akademik
Deskripsi	Menampilkan menu dari modul dosen berupa cari, tampil, input, edit, dan delete data dosen.

Prekondisi	Aktor memilih menu dosen dari navbar	
Normal flow	Aktor respon	Sistem respon
	Fungsi cari : user memasukkan nama dosen yang ingin dicari dan klik tombol cari	Menampilkan data dosen yang dicari sesuai dengan nama yang dimasukkan
	Fungsi input : klik tombol tambah data	Menampilkan menu input dosen
	Fungsi edit data : klik ikon edit pada nama dosen	Menampilkan menu edit data dosen
	Fungsi hapus data : klik ikon hapus pada nama dosen	Menampilkan <i>pop-up</i> pilihan <i>yes/no</i>

3. Kelolah Data Program Studi

Tabel 7 merupakan tabel kelola data program studi memiliki fungsi yang digunakan untuk mengelola data program studi seperti tambah data, ubah data, lihat data, cari data, dan hapus data program studi.

Tabel 7 *Narrative* Kelola Data Program Studi

Nama <i>usecase</i>	Kelola Data Program Studi	
Aktor	Kaur Administrasi Akademik	
Deskripsi	Menampilkan menu dari modul program studi berupa cari, tampil, input, edit, dan delete data program studi.	
Prekondisi	Aktor memilih menu program studi dari navbar	
Normal flow	Aktor respon	Sistem respon

	Fungsi cari : user memasukkan nama program studi yang ingin dicari dan klik tombol cari	Menampilkan data program studi yang dicari sesuai dengan nama yang dimasukkan
	Fungsi input : klik tombol tambah data	Menampilkan menu input program studi
	Fungsi edit data : klik ikon edit pada nama program studi	Menampilkan menu edit data program studi
	Fungsi hapus data : klik ikon hapus pada nama program studi	Menampilkan <i>pop-up</i> pilihan <i>yes/no</i>

4. Kelola Data Ruangan

Tabel 8 merupakan tabel kelola data ruangan memiliki fungsi yang digunakan untuk mengelola data ruangan seperti tambah data, ubah data, lihat data, cari data, dan hapus data ruangan.

Tabel 8 Narrative *Kelola Data Ruangan*

Nama <i>usecase</i>	Kelola Data Ruangan	
Aktor	Kaur Administrasi Akademik	
Deskripsi	Menampilkan menu dari modul ruangan berupa cari, tampil, input, edit, dan delete data ruangan.	
Prekondisi	Aktor memilih menu pengampu dari navbar	
Normal flow	Aktor respon	Sistem respon
	Fungsi cari : user memasukkan nama ruangan atau program studi yang ingin dicari dan klik tombol cari	Menampilkan data ruangan yang dicari sesuai dengan nama yang dimasukkan
	Fungsi input : klik tombol tambah data	Menampilkan menu input ruangan
	Fungsi edit data : klik ikon edit pada nama ruangan	Menampilkan menu edit data ruangan
	Fungsi hapus data : klik ikon hapus pada nama ruangan	Menampilkan <i>pop-up</i> pilihan <i>yes/no</i>

5. Kelola Data Kelas

Tabel 9 merupakan tabel kelola data kelas memiliki fungsi yang digunakan untuk mengelola data kelas seperti tambah data, ubah data, lihat data, dan hapus data kelas.

Tabel 9 *Narrative* Kelola Data Kelas

Nama <i>usecase</i>	Kelola Data Kelas	
Aktor	Kaur Administrasi Akademik	
Deskripsi	Menampilkan menu dari modul ruang berupa tampil, input, edit, dan delete data kelas	
Pre-kondisi	Aktor memilih menu ruang dari navbar	
Normal flow	Aktor respon	Sistem respon
	Fungsi input : klik tombol tambah data	Menampilkan menu input data kelas
	Fungsi edit data : klik ikon edit pada nama kelas	Menampilkan menu edit data kelas
	Fungsi hapus data : klik ikon hapus pada nama kelas	Menampilkan <i>pop-up</i> pilihan <i>yes/no</i>

6. Kelola Data Jam

Tabel 10 merupakan tabel kelola data jam memiliki fungsi yang digunakan untuk mengelola data dosen seperti tambah data, ubah data, lihat data, cari data, dan hapus data jam.

Tabel 10 *Narrative* Kelola Data Jam

Nama <i>usecase</i>	Kelola Data Jam	
Aktor	Kaur Administrasi Akademik	
Deskripsi	Menampilkan menu dari modul jam berupa tampil, input, edit, dan delete data ruang	
Pre-kondisi	Aktor memilih menu jam dari navbar	
Normal flow	Aktor respon	Sistem respon
	Fungsi input : klik tombol tambah data	Menampilkan menu input data jam
	Fungsi edit data : klik ikon edit pada nama jam	Menampilkan menu edit data jam

	Fungsi hapus data : klik ikon hapus pada nama jam	Menampilkan <i>pop-up</i> pilihan <i>yes/no</i>
--	---	---

7. Kelola Data Hari

Tabel 11 merupakan tabel kelola data hari memiliki fungsi yang digunakan untuk mengelola data hari seperti tambah data, edit data, lihat data, dan hapus data hari.

Tabel 11 *Narrative* Kelola Data Hari

Nama <i>usecase</i>	Kelola Data Hari	
Aktor	Kaur Administrasi Akademik	
Deskripsi	Menampilkan menu dari modul ruang berupa tampil, input, edit, dan delete data hari	
Pre-kondisi	Aktor memilih menu hari dari navbar	
Normal flow	Aktor respon	Sistem respon
	Fungsi input : klik tombol tambah data	Menampilkan menu input data hari
	Fungsi edit data : klik ikon edit pada nama hari	Menampilkan menu edit data hari
	Fungsi hapus data : klik ikon hapus pada nama hari	Menampilkan <i>pop-up</i> pilihan <i>yes/no</i>

8. Kelola Pengampu

Tabel 12 merupakan tabel kelola data pengampu memiliki fungsi yang digunakan untuk mengelola data pengampu seperti tambah data, edit data, lihat data, dan hapus data pengampu.

Tabel 12 *Narrative* Kelola Data Pengampu

Nama <i>usecase</i>	Kelola Data Pengampu	
Aktor	Kaur Administrasi Akademik	
Deskripsi	Menampilkan menu dari modul ruang berupa tampil, input, edit, dan delete data pengampu	
Pre-kondisi	Aktor memilih menu hari dari navbar	
Normal flow	Aktor respon	Sistem respon

	Fungsi input : klik tombol tambah data	Menampilkan menu input data pengampu
	Fungsi edit data : klik ikon edit pada nama pengampu	Menampilkan menu edit data pengampu
	Fungsi hapus data : klik ikon hapus pada nama pengampu	Menampilkan <i>pop-up</i> pilihan <i>yes/no</i>

9. Kelola Data Buat Jadwal

Tabel 13 merupakan tabel kelola data buat jadwal memiliki fungsi yang digunakan untuk membuat penjadwalan secara otomatis berdasarkan semester dan tahun akademik.

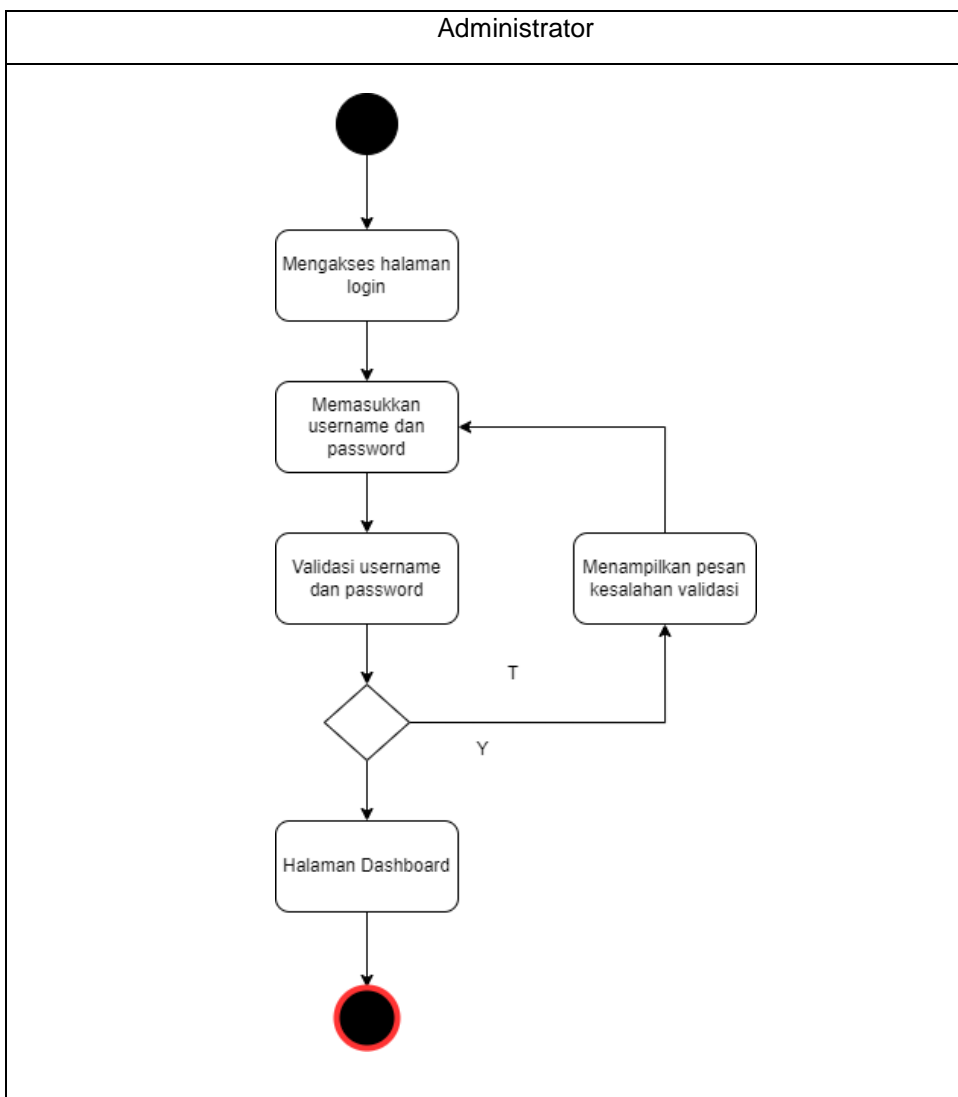
Tabel 13 *Narrative* Kelola Data Penjadwalan

Nama <i>usecase</i>	Kelola Data Buat Jadwal	
Aktor	Kaur Administrasi Akademik	
Deskripsi	Modul ini digunakan untuk melakukan penjadwalan setelah data pengampu, ruang, hari, dan jam dibuat	
Pre-kondisi	Aktor memilih menu penjadwalan dari navbar	
Normal flow	Aktor respon	Sistem respon
	Fungsi penjadwalan : memilih semester yang diinginkan dan tahun akademik kemudian masukkan operator algoritma genetika dan tekan tombol proses	Menampilkan jadwal mata kuliah

2.5.4 Activity Diagram

Berikut ini digambarkan *activity* diagram yang memperlihatkan alur jalannya setiap menu yang ada pada Sistem Penjadwalan Mata Kuliah.

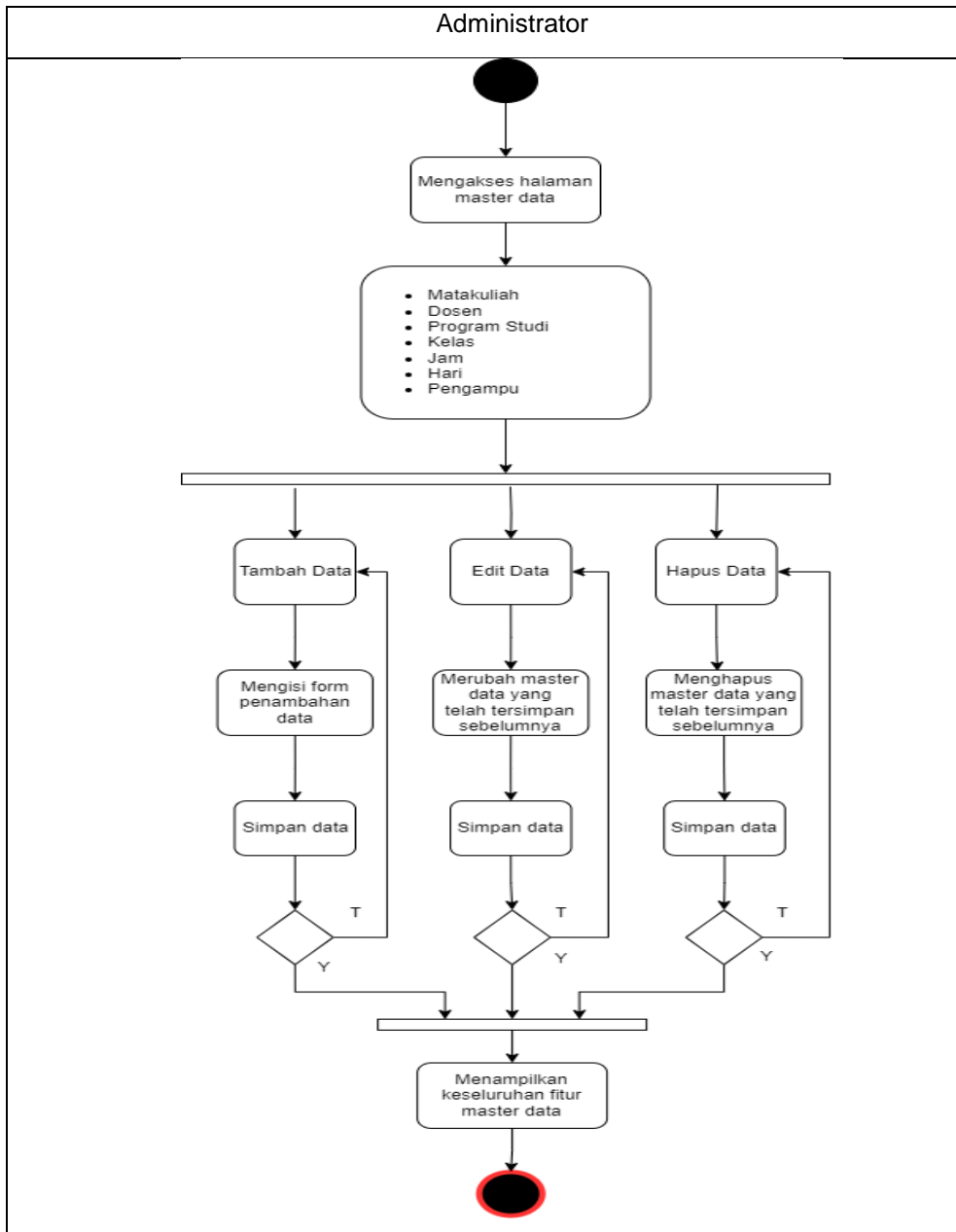
1. Activity Diagram Proses Login



Gambar 3. Activity Diagram Proses Login

Pada Gambar 3. merupakan *activity diagram* dari proses login, Dimana *user* harus terlebih dahulu mengakses halaman *login* dari website ini untuk masuk kedalam menu *dashboard*. Setelah *user* mengakses halaman login selanjutnya *user* diminta untuk memasukkan *username* dan *password* yang telah tersimpan di dalam *database*, kemudian sistem akan melakukan proses validasi, jika sistem berhasil dan data yang dimasukkan sesuai dengan data yang tersedia di *database* maka *user* akan diarahkan kehalaman *dashboard*, dan apabila sistem menemukan adanya kesalahan dalam proses penginputan data maka akan muncul pesan kesalahan "username / password salah" dan akan di arahkan kembali ke halaman login.

2. Activity Diagram Master Data

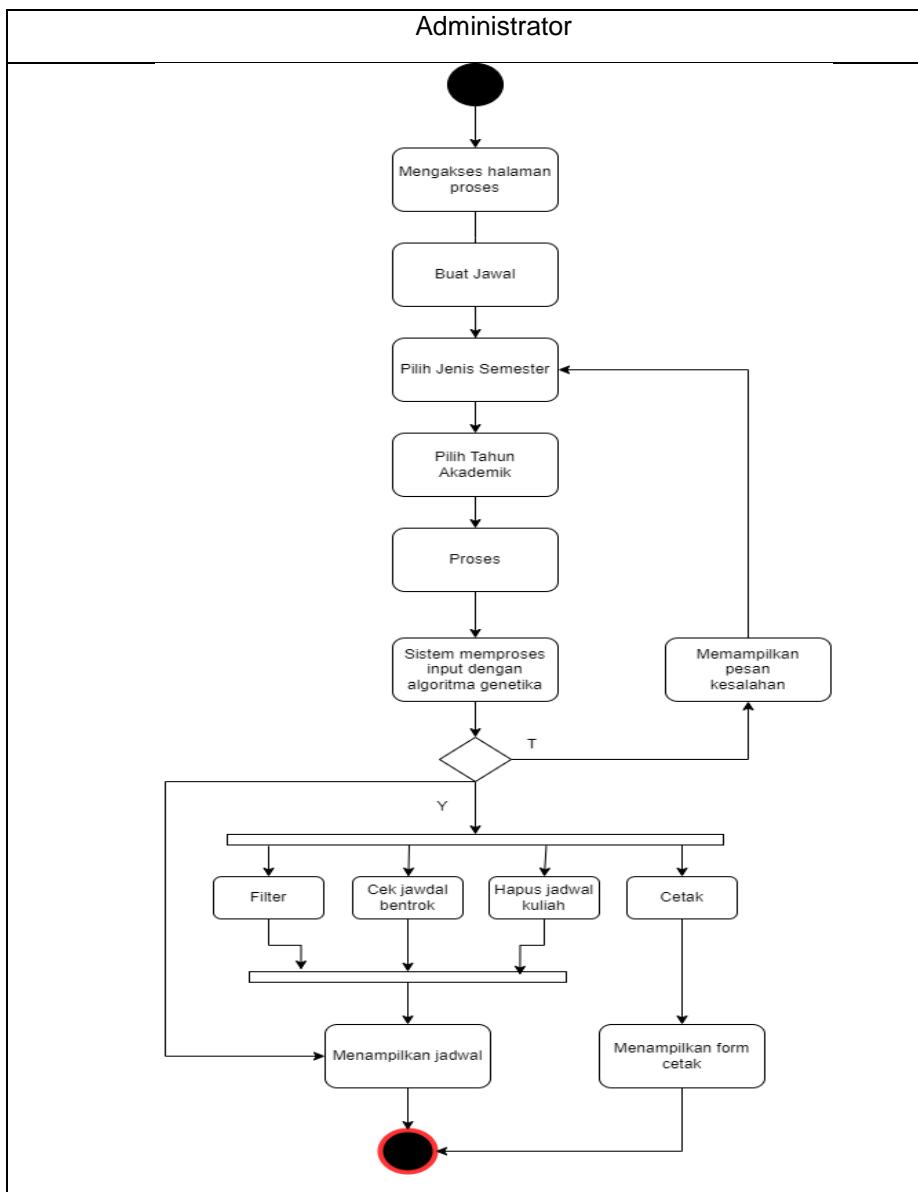


Gambar 4 Activity Diagram Master Data

Pada Gambar 4 merupakan gambar *activity* diagram master data, sistem ini dirancang untuk memudahkan pengelolaan data-data penting seperti mata kuliah, dosen, program studi, dan lainnya. Prosesnya meliputi tiga fungsi utama:

penambahan data baru, pengeditan data yang sudah ada, dan penghapusan data yang tidak diperlukan. Pengguna dapat mengakses halaman utama untuk memilih tindakan yang ingin dilakukan. Setelah itu, sistem akan memandu pengguna melalui proses penambahan, pengeditan, atau penghapusan data dengan menyediakan formulir yang mudah diisi. Setelah perubahan dilakukan, sistem akan menampilkan kembali seluruh data yang telah diperbarui, memastikan data selalu akurat dan *up-to-date*.

3. Activity Diagram Proses



Gambar 5 Activity Diagram Proses Data

Gambar 5 merupakan proses pembuatan dan pengecekan jadwal kuliah menggunakan algoritma genetika. Proses dimulai dengan pengguna mengakses halaman untuk memproses jadwal. Setelah itu, pengguna memilih opsi untuk membuat jadwal baru dan memilih jenis semester yang diinginkan serta tahun akademik yang sesuai. Selanjutnya, sistem mulai memproses data yang dimasukkan. Sistem menggunakan algoritma genetika untuk memproses data input tersebut. Setelah memproses input, sistem memeriksa apakah proses tersebut berhasil. Jika proses berhasil, sistem akan melakukan penyaringan data jadwal, memeriksa apakah ada bentrok jadwal, dan memberikan opsi untuk menghapus jadwal kuliah yang tidak sesuai atau bentrok. Selain itu, sistem memberikan opsi untuk mencetak jadwal dan menampilkan jadwal yang telah diproses serta menampilkan form untuk mencetak jadwal tersebut. Namun, jika proses gagal, sistem akan menampilkan pesan kesalahan yang terjadi selama proses. Setelah semua langkah di atas dilakukan, alur proses berakhir. Diagram ini mengilustrasikan langkah-langkah yang diambil dalam sistem untuk memastikan jadwal kuliah diatur dengan baik menggunakan algoritma genetika, serta memberikan opsi bagi pengguna untuk menangani dan mencetak jadwal yang telah dibuat.

2.5.4 Class Diagram

Visualisasi dari struktur *object* sistem yang diajukan, digambarkan dalam *class* diagram. Adapun langkah-langkah untuk menentukan *object* yang terlibat adalah sebagai berikut:

1. Menemukan *Potencial Object*

Dari narasi *use case analysis*, didapatkan sejumlah *potential object* berikut:

Tabel 14 Menemukan *Potencial Object*

No	Nama <i>potential object</i>
1	Mata Kuliah
2	Dosen
3	Program Studi
4	Ruangan
5	Kelas
6	Jam
7	Hari
8	Pengampu
9	Buat Jadwal
10	Pengguna
11	Sistem

12	Menu
----	------

2. Menyeleksi *Object* yang Diusulkan

Dari daftar *potential object* di atas, maka dilakukan analisa untuk memilih *object* yang digunakan. Tabel 15 merupakan tabel penyeleksian *object* yang diusulkan dan diseleksi berdasarkan fungsinya.

Tabel 15 Menyeleksi Object yang diusulkan

No	Potential Object	#	Alasan
1	Mata Kuliah	Yes	Tabel mata kuliah
2	Dosen	Yes	Tabel dosen
3	Program Studi	Yes	Tabel program studi
4	Ruangan	Yes	Tabel ruangan
5	Kelas	Yes	Tabel Kelas
6	Jam	Yes	Tabel jam
7	Hari	Yes	Tabel hari
8	Pengampu	Yes	Tabel pengampu
9	Buat Jadwal	Yes	Tabel buat jadwal
10	Pengguna	No	Jenis pengguna
11	Sistem	No	Tidak sesuai
12	Menu	No	Tidak sesuai

2.6 Perancangan Tabel

Berikut adalah struktur tabel yang dihasilkan:

1. Tabel Dosen

Tabel 16 merupakan tabel dosen digunakan untuk menyimpan data dosen yang telah dimasukkan.

Tabel 16. Tabel Dosen

No	Nama Field	Type Data	Null	Keterangan
1	Id	Int	None	Primary, kode dosen
2	nip	Varchar	Null	Nip dosen
3	nama	Varchar	Null	Nama dosen

4	title	Varchar	Null	Title dosen
5	alamat	Varchar	Null	Alamat dosen
6	telp	Varchar	Null	No telephone dosen

2. Tabel Hari

Tabel 17 merupakan tabel hari digunakan untuk menyimpan data hari yang akan digunakan untuk menentukan hari mengajar.

Tabel 17 Tabel Hari

No	Nama Field	Type Data	Null	Keterangan
1	id	Int	None	Primary, kode hari
2	nama	varchar	Null	Nama hari
3	kelas	varchar	Null	Nama kelas

3. Tabel Jadwal Kuliah

Tabel 18 merupakan tabel jadwal kuliah yang digunakan untuk menyimpan hasil penjadwalan yang dilakukan pada proses penjadwalan.

Tabel 18 Tabel Jadwal Kuliah

No	Nama Field	Type Data	Null	Keterangan
1	id	Int	None	Primary, kode jadwal kuliah
2	id_pengampu	Int	Null	Kode pengampu matakuliah
3	id_jam	Int	Null	kode jam yang digunakan
4	id_hari	Int	Null	Kode hari yang digunakan
5	id_ruang	int	null	Kode ruang yang digunakan
6	kelas	varchar	null	Kelas yang digunakan

4. Tabel Jam

Tabel 19 merupakan tabel jam yang digunakan untuk menyimpan data range jam yang digunakan untuk mengajar.

Tabel 19 Tabel Jam

No	Nama Field	Type Data	Null	Keterangan
1	id	Int	None	Primary, kode jam
2	range_jam	Varchar	Null	Range jam yang digunakan
3	waktu_sholat	text	Null	Pembagian waktu sholat

5. Tabel Mata Kuliah

Tabel 20 merupakan tabel mata kuliah yang digunakan untuk menyimpan data mata kuliah yang akan digunakan dalam proses penjadwalan.

Tabel 20 Tabel Mata Kuliah

No	Nama Field	Type Data	Null	Keterangan
1	id	Int	None	Primary, kode matakuliah
2	kode_mk	Varchar	Null	Kode matakuliah
3	nama	Varchar	Null	Nama matakuliah
4	sks	Int	Null	Jumlah sks matakuliah
5	semester	Int	Null	Semester dilaksanakannya matakuliah
6	jenis	Varchar	Null	Jenis matakuliah
7	id_prodi	Int	Null	Kode program Studi

6. Tabel Ruang

Tabel 21 merupakan tabel ruang digunakan untuk menyimpan data ruang yang digunakan dalam proses penjadwalan matakuliah.

Tabel 21 Tabel Ruang

No	Nama Field	Type Data	Null	Keterangan
1	id	Int	None	Primary, kode ruang
2	nama	Varchar	Null	Nama ruang
3	jenis	Varchar	Null	Jenis Ruangan
4	id_prodi	int	Null	Kode program studi

7. Tabel Program Studi

Tabel 22 merupakan tabel program studi yang digunakan untuk menyimpan data program studi apa saja yang tersedia.

Tabel 22 Tabel Program Studi

No	Nama Field	Type Data	Null	Keterangan
1	id	Int	None	Primary, kode program studi
2	nama	Varchar	Null	Nama program Studi

8. Tabel Kelas

Tabel 23 merupakan tabel kelas yang digunakan dalam proses pembelajaran.

Tabel 23 Tabel Kelas

No	Nama Field	Type Data	Null	Keterangan
1	id	Int	None	Primary, kode kelas
2	nama	Longtext	Null	Nama kelas
3	jenis	Varchar	Null	Jenis kelas
4	tahun_angkatan	Varchar	Null	Angkatan mahasiswa
5	id_prodi	Int	Null	Kode program studi

9. Tabel Pengampu

Tabel 24 merupakan tabel pengampu yang digunakan untuk menyimpan data perkuliahan yang kemudian data tersebut di proses dalam pembuatan jadwal kuliah.

Tabel 24 Tabel Pengampu

No	Nama Field	Type Data	Null	Keterangan
1	id	Int	None	Primary, kode Pengampu
2	id_mk	Int	Null	Kode mata kuliah
3	id_dosen	Int	Null	Kode dosen
4	kelas	Longtext	Null	Kelas yang digunakan
5	tahun_akademik	Varchar	Null	Tahun akademik yang digunakan

10. Tabel Administrator

Tabel 25 merupakan tabel administrator yang digunakan dalam proses masuk ke dalam aplikasi yang dibuat.

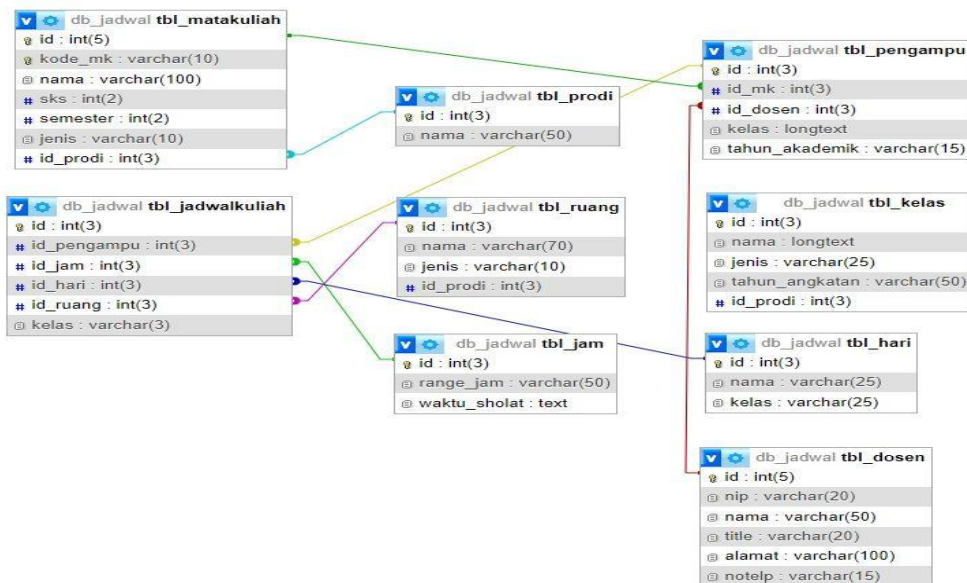
Tabel 25 Tabel Administrator

No	Nama Field	Type Data	Null	Keterangan
1	username	Varchar	None	Username yang digunakan
2	password	Varchar	Null	Password yang digunakan
3	nama	Varchar	Null	Nama
4	email	Varchar	Null	Email yang digunakan

2.7. Perancangan Relasi Antar Tabel

Pada model data relasional hubungan antar tabel direlasikan dengan kunci utama (*primary key*) dari masing – masing tabel. Perancangan relasi antar tabel pada sistem yang akan dibangun ini dapat dilihat pada gambar berikut:

Gambar 12 merupakan relasi antar tabel yang digunakan pada sistem penjadwalan mata kuliah.

**Gambar 6** Relasi Antar Tabel