

**PERANCANGAN SISTEM PENJADWALAN KEGIATAN
BELAJAR MAHASISWA BERBASIS AI MENGGUNAKAN
ALGORITMA GENETIKA DI FAKULTAS ILMU
KOMPUTER DAN TEKNOLOGI INFORMASI**

SKRIPSI

Disusun Oleh :

Bahdi Rizky Al wahya

2109020135



UMSU
Unggul | Cerdas | Terpercaya

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
MEDAN**

2025

**PERANCANGAN SISTEM PENJADWALAN KEGIATAN BELAJAR
MAHASISWA BERBASIS AI MENGGUNAKAN ALGORITMA
GENETIKA DI FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI**

SKRIPSI

**Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer
(S.Kom) dalam Program Studi Teknologi Informasi pada Fakultas Ilmu Komputer
dan Teknologi Informasi, Universitas Muhammadiyah Sumatera Utara**

BAHDI RIZKY AL WAHYA

NPM. 2109020135

**PROGRAM STUDI TEKNOLOGI INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
MEDAN**

2025

LEMBAR PENGESAHAN

Judul Skripsi : **PERANCANGAN SISTEM PENJADWALAN
KEGIATAN BELAJAR MAHASISWA BERBASIS
AI MENGGUNAKAN ALGORITMA GENETIKA
DI FAKULTAS ILMU KOMPUTER DAN
TEKNOLOGI INFORMASI**

Nama Mahasiswa : **BAHDI RIZKY AL WAHYA**

NPM : **2109020135**

Program Studi : **TEKNOLOGI INFORMASI**

Menyetujui
Komisi Pembimbing



(Indah Purnama Sari, S.T.,M.Kom)
NIDN. 0116049001

Ketua Program Studi



(Fatma Sari Hutagalung, M.Kom)
NIDN. 0117019301

Dekan



(Dr. Al-Khowarizmi, S.Kom., M.Kom.)
NIDN. 0127099201

PERNYATAAN ORISINALITAS

PERANCANGAN SISTEM PENJADWALAN KEGIATAN BELAJAR MAHASISWA BERBASIS AI MENGGUNAKAN ALGORITMA GENETIKA DI FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, Agustus 2025

Yang membuat pernyataan



Bahdi Rizky Al wahya

NPM. 2109020135

**PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama	: Bahdi Rizky Al wahya
NPM	: 2109020135
Program Studi	: Teknologi Informasi
Karya Ilmiah	: Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bedas Royalti Non-Eksekutif (*Non-Exclusive Royalty free Right*) atas penelitian skripsi saya yang berjudul:

**“PERANCANGAN SISTEM PENJADWALAN KEGIATAN BELAJAR
MAHASISWA BERBASIS AI MENGGUNAKAN ALGORITMA
GENETIKA DI FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI”**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, Agustus 2025

Yang membuat pernyataan



Bahdi Rizky Al wahya

NPM. 2109020135

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap : Bahdi Rizky Al wahya
Tempat dan Tanggal Lahir : Gunting Saga, 29 Juni 2003
Alamat Rumah : Lk. VI Panjang Bidang II
Telepon/Faks/HP : 082164552148
E-mail : bahdirizkyalwahya@gmail.com
Instansi Tempat Kerja : -
Alamat Kantor : -

DATA PENDIDIKAN

SD	: SD SWASTA ALWASHLIYAH 83	TAMAT : 2015
SMP	: MTS SWASTA ALWASHLIYAH 93	TAMAT : 2018
SMA	: MAN 2 LABUHANBATU UTARA	TAMAT : 2021

KATA PENGANTAR



Segala puji dan syukur penulis panjatkan kehadirat Allah SWT yang telah memberikan rahmat, nikmat dan anugrah-nya, bahwa penulis dapat menyelesaikan skripsi yang berjudul **“PERANCANGAN SISTEM PENJADWALAN KEGIATAN BELAJAR MAHASISWA BERBASIS AI MENGGUNAKAN ALGORITMA GENETIKA DI FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI”**.

Penulis tentunya berterima kasih kepada berbagai pihak dalam dukungan serta doa dalam penyelesaian skripsi. Penulis juga mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Agussani, M.AP., Rektor Universitas Muhammadiyah Sumatera Utara (UMSU)
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. Dekan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
3. Ibu Fatma Sari Hutagalung Ketua Program Studi Teknologi Informasi
4. Bapak Mhd.Basri, S.Si, M.Kom Sekretaris Program Studi Teknologi Informasi
5. Ibu Indah Purnama Sari, S.T., M.Kom, selaku dosen pembimbing yang telah dengan sabar dan tulus mengarahkan penyusunan skripsi ini. Terima kasih yang tulus penulis sampaikan. Ibu tidak hanya membimbing secara teknis, melainkan juga membuka wawasan penulis, menantang untuk berpikir kritis, dan dengan sabar mengarahkan setiap langkah penelitian ini dari ide mentah hingga menjadi karya yang utuh. Setiap diskusi, koreksi, dan masukan yang Ibu berikan merupakan pelajaran yang sangat berharga bagi pendewasaan akademis penulis.
6. Kepada Ayah dan Mamak serta Adek yang paling aku sayang dan cintai, terima kasih telah menjadi Support Sistem terbaik dan menjadi alasan terkuatku untuk terus berjuang. Ayah, terima kasih atas segala kerja keras yang telah kamu lakukan selama ini yang mungkin tak akan mampu bisa kubalas. Mamak, terima kasih atas doa di setiap sujud dan pelukan yang tak

pernah lelah menerima kepulanganku, dalam keadaan apa pun. Terima kasih.... Terima kasih.... Terima kasih.... Jika hari ini aku bisa sampai sejauh ini, itu karena kalian berdua tidak pernah lelah berdiri di belakangku.

7. Untuk Nenek dan Atok yang paling aku sayangi, Terima kasih karena selalu menjadi penyemangat di setiap pagi mendorongku untuk tidak menyerah dan terus melangkah, bahkan saat semangatku mulai menipis. Kehadiran kalian di rumah selama ini bukan hanya membuatku merasa ditemani, tapi juga mengisi kekosongan yang tak bisa dijelaskan dengan kata-kata. Terima kasih karena telah menjadi pelipur sepi, tempatku berbagi diam-diam, dan sumber kehangatan yang sederhana namun berarti. Dalam proses yang panjang dan melelahkan ini, kalian menjadi bagian penting yang membuatku tetap kuat dan tidak merasa sendiri. Dan karena kalian berdualah aku ingin menyelesaikan penulisan skripsi ini.
8. Untuk diriku sendiri. Terima kasih telah menjadi kuat dan menolak untuk menyerah walupun banyak hal terjadi selama pengerjaan skripsi. Terima kasih telah bertahan melalui malam-malam panjang tanpa tidur, keraguan diri, dan rasa lelah yang tak berujung. Kamu telah berjuang dengan hebat, dan kamu berhasil melaluinya
9. Untuk seorang perempuan yang pernah berjalan berdampingan dengan saya, terima kasih bukan hanya atas waktu yang pernah kita lewati bersama, tapi juga untuk patah hati yang kamu tinggalkan, yang diam-diam menjadi bahan bakar bagi saya untuk terus melangkah. Di tengah kesulitan menyusun skripsi ini, rasa kehilanganmu berubah menjadi kekuatan, menjadi alasan bagi saya untuk membuktikan, bahwa saya bisa berdiri meski pernah dijatuhkan oleh harapan. Terima kasih karena pernah menjadi kisah yang menyakitkan, tapi juga penuh pelajaran tentang tumbuh, tentang bagaimana menjadi manusia yang lebih kuat, lebih dewasa, dan lebih layak dicintai dengan cara yang lebih baik.
10. Untukmu, seorang teman dengan NPM 2109020102, terima kasih telah membersamaku sejak awal. Kau bukan hanya membantu menyusun skripsi ini, tapi juga membantu menyusun kembali semangatku yang sempat hilang sekaligus merangkai ulang hatiku yang sempat runtuh oleh rasa lelah dan

viii keinginan untuk menyerah. Kau adalah teman berproses yang membuat perjalanan ini indah dan berarti. Semoga kita tak berhenti di sini, tapi terus menjadi teman berproses saling tumbuh, saling melengkapi, hingga masa depan yang masih sama-sama kita rahasiakan.

11. Untuk seseorang yang tak bisa kusebutkan namanya hari ini, karena semesta masih menyimpannya untuk masa depan, mungkin kita akan bertem atau mungkin kamu sedang membaca ini entah di mana, tanpa tahu bahwa ini tentangmu. Terima kasih telah menjadi bayangan yang kurindukan, doa yang tak pernah benar-benar selesai kuucapkan setiap malam. Kamu adalah alasan mengapa aku terus berusaha, mengapa aku ingin menjadi versi terbaik dari diriku. Jika takdir mempertemukan kita nanti, aku ingin kamu tahu, bahwa sejak pertama bertemu, aku sudah mulai mencintaimu dalam bentuk paling sederhana dengan menjadi lebih baik setiap harinya.
12. Kepada Sepupu-sepupuku dan Teman-temanku yang menemani di tengah patah hatiku. Di tengah hati yang retak dan semangat yang nyaris padam, kalian datang bukan untuk memperbaiki segalanya, tapi cukup untuk duduk bersamaku dalam diam, menemani, dan mengingatkanku bahwa aku tidak sendiri. Terima kasih karena kalian tidak hanya membantu dalam hal akademis, tapi juga menyelamatkan sebagian hatiku yang nyaris hancur. Kebaikan kecil kalian adalah hal besar yang takkan pernah kulupakan.
13. Kepada seluruh teman-teman di kelas C1 TI, penulis mengucapkan terima kasih yang tulus atas solidaritas dan atmosfer positif yang kita ciptakan bersama. Semoga ikatan pertemanan ini tetap terjaga dan kita semua dapat meraih kesuksesan di jalan yang kita tuju.
14. Semua pihak yang terlibat langsung ataupun tidak langsung yang tidak dapat penulis ucapkan satu-persatu yang telah membantu penyelesaian skripsi ini.

PERANCANGAN SISTEM PENJADWALAN KEGIATAN BELAJAR
MAHASISWA BERBASIS AI MENGGUNAKAN ALGORITMA GENETIKA
DI FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

ABSTRAK

Penelitian ini mengembangkan sistem penjadwalan perkuliahan otomatis berbasis Artificial Intelligence (AI) dengan menggunakan algoritma genetika untuk menyelesaikan masalah kompleks dalam penyusunan jadwal di Fakultas Ilmu Komputer dan Teknologi Informasi. Sistem dibangun menggunakan Python, Flask, dan library DEAP, serta diintegrasikan ke dalam platform web agar dapat diakses oleh admin, dosen, dan mahasiswa. Hasil pengujian menunjukkan sistem mampu menghasilkan hingga 787 jadwal dengan tingkat keberhasilan tinggi dan nilai fitness maksimal (1.00) tanpa konflik. Fitur tambahan seperti chatbot interaktif, status ruangan real-time, dan export data ke Excel meningkatkan kemudahan penggunaan dan efisiensi dalam pengelolaan akademik. Sistem ini terbukti efektif, efisien, dan layak untuk diimplementasikan dalam manajemen jadwal akademik secara otomatis.

Kata kunci: Penjadwalan otomatis, algoritma genetika, kecerdasan buatan, sistem informasi akademik, chatbot.

DESIGN OF A STUDENT LEARNING ACTIVITY SCHEDULING SYSTEM
BASED ON AI USING GENETIC ALGORITHM IN THE FACULTY OF
COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

ABSTRACT

This research develops an automated course scheduling system based on Artificial Intelligence (AI) using a genetic algorithm to address the complex scheduling problems at the Faculty of Computer Science and Information Technology. The system was built using Python, Flask, and the DEAP library, and integrated into a web-based platform accessible to administrators, lecturers, and students. Test results show that the system can generate up to 787 schedules with high success rates and a maximum fitness value (1.00) without conflicts. Additional features such as an interactive chatbot, real-time room status, and data export to Excel enhance usability and efficiency in academic schedule management. The system has proven to be effective, efficient, and feasible for implementation in automated academic scheduling.

Keywords: Automated scheduling, genetic algorithm, artificial intelligence, academic information system, chatbot.

DAFTAR ISI

LEMBAR PENGESAHAN	ii
PERNYATAAN ORISINALITAS.....	iii
PERNYATAAN PERSETUJUAN PUBLIKASI.....	iv
RIWAYAT HIDUP	v
KATA PENGANTAR.....	vi
ABSTRAK	ix
ABSTRACT	x
DAFTAR ISI.....	xi
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR.....	xv
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	4
1.3 Batasan Masalah.....	5
1.4 Tujuan Penelitian	6
1.5 Manfaat Penelitian	7
BAB II LANDASAN TEORI	8
2.1 Rancang Bangun	8
2.1.1 Rancang Bangun Sistem	8
2.1.2 Rancang Bangun Website	10
2.2 Kecerdasan Buatan (Artificial Intelligence).....	12
2.2.1 Definisi dan Sejarah AI.....	12
2.2.2 Pendekatan Dan Klasifikasi AI.....	13
2.2.3 AI Dalam Sistem Informasi dan Pendidikan	15
2.2.4 Relevansi AI Terhadap Penjadwalan Akademik	16
2.3 Sistem Penjadwalan Berbasis AI	17
2.3.1 Konsep Penjadwalan.....	17
2.3.2 Masalah Penjadwalan Kuliah.....	18
2.3.3 AI Dalam Penyelesaian Penjadwalan	19
2.4 Algoritma Genetika.....	20
2.4.1 Definisi dan Sejarah Algoritma Genetika	20

2.4.2 Komponen Utama Algoritma Genetika	23
2.4.3 Tahapan Algoritma Genetika.....	26
2.5 Chatbot.....	30
2.5.1 Definisi dan Konsep Dasar Chatbot	30
2.5.2 Arsitektur dan Komponen Chatbot	31
2.5.3 Teknologi dan Bahasa Pemrograman Chatbot.....	33
2.6 Penelitian Terdahulu	34
BAB III METODOLOGI PENELITIAN	36
3.1 Jenis dan Pendekatan Penelitian.....	36
3.1.1 Pendekatan Research And Development (R&D).....	36
3.1.2 Model Pengembangan Sistem.....	37
3.2 Pengumpulan Data	39
3.2.1 Data Mata kuliah	39
3.2.2 Data Dosen.....	43
3.2.3 Data Jam Perkuliahan Dan Data Ruangan Kuliah	51
3.3 Perancangan Sistem	52
3.3.1 Alur Kerja Sistem	53
3.3.2 Penjelasan Alur Kerja Sistem	54
3.4 Waktu dan Tempat Penelitian	56
3.4.1 Waktu Penelitian.....	56
3.4.2 Tempat Penelitian	56
BAB IV HASIL DAN PEMBAHASAN.....	58
4.1 Hasil Implementasi Sistem.....	58
4.1.1 Fitur Utama	58
4.1.2 Fitur Biro Akademik.....	60
4.1.3 Fitur Dosen	74
4.2 Pengujian Sistem.....	77
4.2.1 Pengujian Chatbot.....	77
4.2.2 Pengujian Algoritma Genetika.....	79
4.2.3 Pengujian Fitur Tambahan.....	82
4.3 Analisis Sistem.....	86
4.3.1 Pengembangan Sistem Penjadwalan Otomatis	86

4.3.2 Penerapan Algoritma Genetika dalam Mengurangi Konflik	87
4.3.3 Integrasi Sistem dengan Platform Website	88
4.3.4 Efektivitas Sistem dibanding Metode Manual	89
4.4 Evaluasi Sistem	90
BAB V KESIMPULAN DAN SARAN	92
5.1 Kesimpulan	92
5.2 Saran.....	93
DAFTAR PUSTAKA.....	96
LAMPIRAN	100

DAFTAR TABEL

Table 2.1 Penelitian Terdahulu	34
Tabel 3.1 Data Mata Kuliah Teknologi Informasi Tahun 2022.....	40
Tabel 3.2 Data Mata Kuliah Teknologi Informasi Tahun 2023.....	40
Tabel 3.3 Data Mata Kuliah Teknologi Informasi Tahun 2024.....	41
Tabel 3.4 Data Mata Kuliah Sistem Informasi Tahun 2021	41
Tabel 3.5 Data Mata Kuliah Sistem Informasi Tahun 2022	41
Tabel 3.6 Data Mata Kuliah Sistem Informasi Tahun 2023	42
Tabel 3.7 Data Mata Kuliah Sistem Informasi Tahun 2024	42
Tabel 3.8 Data Mata Kuliah Sains Data Tahun 2023.....	42
Tabel 3.9 Data Mata Kuliah Sains Data Tahun 2024.....	43
Tabel 3.10 Data Dosen.....	44
Table 3.11 Data Jam Perkuliahan.....	51
Tabel 3.12 Data Ruangan Kuliah	52
Tabel 3.13 Waktu Penelitian.....	56
Tabel 4.1 Pengujian Chatbot.....	78
Tabel 4.2 Hasil Generate Algoritma Genetika.....	80

DAFTAR GAMBAR

Gambar 2.1 Tahapan Algoritma Genetika	29
Gambar 3.1 Model Waterfall	38
Gambar 3.2 Alur Kerja Sistem.....	54
Gambar 4.1 Halaman Login.....	58
Gambar 4.2 Chat Bot Akademik.....	59
Gambar 4.3 Monitoring Status Ruangan Real Time	60
Gambar 4.4 Dashboard Biro Akademik.....	61
Gambar 4.5 Generate Jadwal (Algoritma Genetika).....	61
Gambar 4.6 Pengaturan Semester	62
Gambar 4.7 Jadwal Kuliah.....	63
Gambar 4.8 Form Tambah Jadwal Kuliah	63
Gambar 4.9 Form Edit Jadwal Kuliah.....	64
Gambar 4.10 Jadwal Dosen dan Mata Kuliah.....	64
Gambar 4.11 Riwayat Pindah Jadwal	65
Gambar 4.12 Daftar Mata Kuliah.....	66
Gambar 4.13 Form Tambah Mata Kuliah	66
Gambar 4.14 Edit Mata Kuliah	67
Gambar 4.15 Daftar Dosen	67
Gambar 4.16 Form Tambah Dosen.....	68
Gambar 4.17 Form Edit Dosen	68
Gambar 4.18 Daftar Ruang Kuliah	69
Gambar 4.19 Form Tambah Ruang Kuliah.....	69
Gambar 4.20 Form Edit Ruang Kuliah	70
Gambar 4.21 Manajemen Kelas.....	70
Gambar 4.22 Form Tambah Kelas	71
Gambar 4.23 Form Buat Kelas Massal	72
Gambar 4.24 Dosen dan Mata Kuliah.....	72
Gambar 4.25 Export Data	73
Gambar 2.26 Log Perintah AI.....	74
Gambar 4.27 Halaman Dashboard Dosen.....	75

Gambar 4.28 Jadwal Saya	75
Gambar 4.29 Kehadiran Dosen	76
Gambar 4.30 Mata Kuliah Yang Diampu	76
Gambar 4.31 Contoh Penggunaan ChatBot	77
Gambar 4.32 Filter Generate Algoritma Genetika	79
Gambar 4.33 Pengujian Status Ruangan Real-time	82
Gambar 4.34 Filter Export Data	84
Gambar 4.35 Hasil Export Data	85

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Penjadwalan perkuliahan merupakan salah satu aspek penting dalam manajemen akademik di perguruan tinggi. Proses ini melibatkan alokasi waktu, ruang, dosen, dan mata kuliah secara efisien dan juga efektif. Namun, seiring dengan meningkatnya jumlah mahasiswa dan kompleksitas kurikulum, institusi pendidikan tinggi di Indonesia masih banyak menghadapi tantangan dalam menyusun jadwal perkuliahan yang optimal. Proses penjadwalan yang dilakukan secara manual sering kali menimbulkan konflik berupa jadwal perkuliahan yang bertabrakan, penggunaan ruang yang tidak efisien, dan ketidakpuasan dari berbagai pihak terkait.

Di Universitas Muhammadiyah Sumatera Utara (UMSU), peningkatan jumlah mahasiswa dan mata kuliah setiap tahunnya bertambah, sehingga penjadwalan kegiatan belajar kurang kompleksitas. Proses penjadwalan manual yang masih diterapkan sering kali mengakibatkan bentroknya jadwal antara dosen dan mahasiswa, serta penggunaan ruang kelas yang tidak optimal. Hal ini berdampak negatif terhadap efektivitas proses belajar mengajar dan efisiensi operasional universitas.

Kondisi ini juga semakin terasa di Fakultas Ilmu Komputer dan Teknologi Informasi yang di mana membawahi tiga program studi, yaitu Teknologi Informasi, Sistem Informasi, dan Sains Data. Dengan jumlah kelas

yang terbatas dan dosen yang sering mengampu mata kuliah lebih dari satu program studi, penjadwalan yang tidak terstruktur sering kali menyebabkan tumpang tindih antara jadwal kuliah dan ketersediaan ruang. Misalnya, satu ruang kelas dapat di jadwalkan untuk dua kelas berbeda pada waktu yang sama, atau seorang dosen di jadwalkan untuk dua kelas yang berbeda pada waktu yang bersamaan. Tantangan ini membuat proses penjadwalan menjadi semakin kompleks dan rentan terhadap konflik. Oleh karena itu, diperlukan sebuah sistem penjadwalan otomatis yang tidak hanya mempertimbangkan banyaknya variabel, tetapi juga mampu menghasilkan jadwal yang bebas konflik dan memkasimalkan pemanfaatan ruang kelas dan mengakomodasi ketersediaan dosen secara akurat.

Menurut penelitian oleh Amrulloh dan Sela (2021) menjelaskan bahwa penjadwalan manual sering kali menghasilkan jadwal yang tidak optimal dan rentan terhadap konflik, terutama dalam hal alokasi ruang dan waktu perkuliahan. Mereka mengusulkan penggunaan *algoritma genetika* dan pencarian tabu untuk mengoptimalkan proses penjadwalan, yang terbukti mampu menghasilkan jadwal tanpa konflik dalam waktu yang lebih singkat dibandingkan metode manual .

Untuk mengatasi permasalahan tersebut, penerapan teknologi kecerdasan buatan (*Artificial Intelligence*) dalam sistem penjadwalan perkuliahan menjadi salah satu solusi yang menjanjikan. *Algoritma genetika*, sebagai salah satu metode dalam *AI*, telah banyak digunakan dalam berbagai penelitian untuk menyelesaikan masalah penjadwalan yang kompleks. Selain itu, penerapan *AI* juga dapat diperluas dengan

mengintegrasikan fitur *chatbot* ke dalam sistem. *Chatbot* bertindak sebagai antarmuka interaktif yang memungkinkan pengguna baik dosen maupun staf akademik untuk mengakses informasi jadwal secara langsung dan *real-time* hanya melalui percakapan sederhana. Dengan cara ini, sistem penjadwalan tidak hanya menjadi *backend* otomatisasi, tetapi juga dapat diakses dengan mudah melalui platform yang *user-friendly*.

Algoritma genetika bekerja dengan meniru proses seleksi alam untuk mencari solusi optimal dalam ruang pencarian yang kompleks. Dengan mengadaptasi mekanisme *evolusi* seperti *seleksi*, *crossover*, dan *mutasi*, algoritma ini dapat digunakan untuk mengembangkan sistem penjadwalan otomatis yang memperhatikan berbagai kendala dan preferensi, seperti waktu, ruang, dan dosen. Di dalam konteks pengembangan sistem di FIKTI, *algoritma genetika* menjadi inti dari proses optimasi jadwal yang terintegrasi dalam platform berbasis *website*.

Sebagai salah satu *algoritma evolusioner*, *algoritma genetika* dimulai dengan membangkitkan sejumlah populasi awal secara acak yang direpresentasikan dalam bentuk *kromosom*. Setiap *kromosom* mewakili kombinasi jadwal, seperti waktu, ruang, dan pengampu mata kuliah. Selanjutnya, setiap *kromosom* dievolusi menggunakan *fungsi fitness*, yaitu ukuran seberapa baik jadwal tersebut bebas dari konflik, efisien dalam penggunaan ruang, dan sesuai dengan alokasi dosen. *Kromosom* terbaik akan dipilih melalui proses seleksi untuk direkombinasi (*crossover*) dan dimodifikasi secara acak (*mutasi*) guna menghasilkan generasi solusi baru

yang lebih baik. Proses ini terus berulang hingga ditemukan solusi yang optimal atau memenuhi ambang kualitas tertentu. Pendekatan ini sangat sesuai untuk permasalahan penjadwalan karena mampu menangani banyak variabel secara *simultan* dan menghasilkan solusi yang mendekati optimal dalam waktu yang relatif singkat. Solusi terbaik inilah yang kemudian digunakan sebagai jadwal akhir dalam sistem penjadwalan otomatis.

Dengan mempertimbangkan berbagai penelitian tersebut, pengembangan *website* penjadwalan otomatis berbasis *AI* menggunakan *algoritma genetika* di FIKTI diharapkan dapat mengatasi permasalahan penjadwalan yang ada. Sistem ini tidak hanya akan mengurangi konflik jadwal dan memaksimalkan penggunaan ruang kelas, tetapi juga memberikan kemudahan akses bagi mahasiswa, dosen, dan staf akademik dalam mengelola informasi jadwal secara *real-time* dan terintegrasi.

1.2. Rumusan Masalah

1. Bagaimana cara mengembangkan sistem penjadwalan perkuliahan otomatis berbasis *Artificial Intelligence (AI)* menggunakan *Algoritma Genetik* di Fakultas Ilmu Komputer dan Teknologi Informasi?
2. Bagaimana *Algoritma Genetik* dapat diterapkan untuk mengurangi konflik jadwal antara dosen, mahasiswa, dan penggunaan ruang kelas?
3. Bagaimana sistem penjadwalan otomatis ini dapat diintegrasikan ke dalam platform *website* agar memudahkan akses bagi mahasiswa, dosen, dan staf akademik?
4. Seberapa efektif dan optimal kinerja sistem penjadwalan otomatis dibandingkan dengan metode manual, khususnya dalam hal kecepatan

pemrosesan, pengurangan konflik jadwal, dan efisiensi penggunaan ruang?

1.3. Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Sistem yang dikembangkan hanya berfokus pada penjadwalan perkuliahan reguler di Fakultas Ilmu Komputer dan Teknologi Informasi dan tidak mencakup jadwal kegiatan akademik lain seperti seminar, workshop, maupun ujian susulan.
2. Sistem ini tidak mempertimbangkan variabel eksternal seperti jadwal kegiatan mahasiswa di luar akademik atau perubahan mendadak akibat kebijakan universitas. Sistem hanya menggunakan data mata kuliah, dosen, mahasiswa, dan ruang kelas yang telah terdaftar.
3. *Algoritma Genetik* yang digunakan hanya difokuskan pada optimasi penjadwalan untuk mengurangi bentrok jadwal dosen, mahasiswa, dan ruang kelas. Preferensi individu tidak dipertimbangkan kecuali telah ditentukan dalam aturan akademik.
4. Evaluasi sistem dilakukan berdasarkan tingkat kepuasan pengguna, jumlah konflik jadwal yang berhasil di kurangi, dan kecepatan pemrosesan jadwal di bandingkan metode manual sebelumnya.
5. Sistem penjadwalan yang dikembangkan menggunakan *algoritma genetika* membutuhkan waktu komputasi yang relatif lama saat memproses data dalam jumlah besar, hal ini disebabkan oleh proses perhitungan *fitness* yang kompleks dan lingkungan pengujian yang masih berjalan di *localhost*, sehingga performa belum optimal seperti ketika sistem di-*hosting* pada server produksi.
6. Fitur penggantian jadwal melalui *chatbot* masih memiliki keterbatasan dari sisi kemudahan penggunaan, karena format input yang harus dimasukkan pengguna cukup rumit. Pengguna perlu memasukkan detail seperti kode mata kuliah (*kode_mk*), kelas, NIDN

dosen, hari, jam, dan durasi, sehingga memerlukan ketelitian tinggi dan dapat menyulitkan pengguna awam.

7. Pengujian sistem dilakukan dalam lingkungan simulasi berbasis data akademik di Universitas Muhammadiyah Sumatra Utara.
8. Pengujian sistem hanya dilakukan di fakultas Ilmu Komputer dan Teknologi, sebelum digunakan lebih lanjut untuk penggunaan di Universitas Muhammadiyah Sumatera Utara

1.4. Tujuan Penelitian

Adapun tujuan dari penelitian ini adalah sebagai berikut:

1. Mengembangkan *website* berbasis *chatbot* Penjadwalan Kegiatan Belajar Berbasis AI Menggunakan *Algoritma Genetika* yang dapat menyusun jadwal perkuliahan secara otomatis, optimal, dan efisien di Fakultas Ilmu Komputer dan Teknologi Informasi.
2. Menerapkan *Algoritma Genetik* dalam optimasi penjadwalan untuk mengurangi bentrokan antara jadwal dosen, mahasiswa, dan ruang kelas, serta meningkatkan efisiensi dalam alokasi waktu perkuliahan.
3. Mengintegrasikan sistem penjadwalan berbasis AI ke dalam platform *chatbot* agar mempermudah akses pengguna, baik mahasiswa, dosen, maupun staf akademik.
4. Menguji efektifitas dan performa sistem penjadwalan otomatis, dengan membandingkan hasilnya terhadap metode manual berdasarkan kecepatan pemrosesan, tingkat pengurangan konflik jadwal, dan optimalisasi penggunaan ruang kelas.

5. Menganalisis tingkan kepuasan pengguna, serta menjadikan sistem ini sebagai model inovasi digital berbasis *AI* dalam manajemen akademik.

1.5. Manfaat Penelitian

1. Penelitian ini diharapkan dapat meningkatkan efisiensi pengelolaan jadwal akademik di FIKTI dengan mengurangi konflik jadwal serta mengoptimalkan pemanfaatan ruang kelas secara sistematis dan otomatis.
2. Penerapan *Algoritma Genetik* dalam sistem penjadwalan akademik dapat menjadi acuan bagi pengembang sistem lain dalam mengintegrasikan teknologi kecerdasan buatan (*AI*) untuk manajemen akademik yang lebih adaptif dan efisien.
3. Sistem yang dikembangkan berpotensi menjadi model inovasi digital berbasis *AI* di bidang pendidikan, khususnya dalam otomatisasi proses administrasi akademik di perguruan tinggi.
4. Penelitian ini memberikan pengalaman praktis bagi peneliti dalam merancang dan mengimplementasikan sistem berbasis *AI*, termasuk dalam penerapan *algoritma optimasi* dan proses pengembangan perangkat lunak.
5. Hasil dari penelitian ini diharapkan dapat memberikan kontribusi ilmiah bagi pengembangan literatur di bidang sistem informasi akademik, terutama yang berkaitan dengan pemanfaatan teknologi *AI* dalam meningkatkan efektivitas dan efisiensi operasional institusi pendidikan tinggi.

BAB II

LANDASAN TEORI

2.1 Rancang Bangun

Rancang bangun adalah proses sistematis dalam pengembangan sistem informasi yang mencakup tahap perencanaan, perancangan, implementasi, hingga pengujian dan evaluasi sistem. Proses ini bertujuan untuk membangun sistem yang dapat memenuhi kebutuhan pengguna, bekerja secara efisien, serta mudah dipelihara dan dikembangkan.

Menurut Sutedjo (2016), rancang bangun dalam konteks teknologi informasi merupakan metode untuk menghasilkan sistem yang mampu menyelesaikan masalah secara tepat, melalui langkah-langkah yang terstruktur mulai dari analisis kebutuhan hingga implementasi dan pengujian sistem. Rancang bangun tidak hanya terbatas pada aspek teknis pemrograman, melainkan juga mencakup pemahaman mendalam terhadap kebutuhan pengguna, alur proses, dan struktur data yang akan digunakan dalam sistem.

Dalam penelitian ini, rancang bangun digunakan sebagai pendekatan untuk mengembangkan sistem penjadwalan kegiatan belajar mahasiswa berbasis *AI*. Prosesnya dimulai dengan analisis kebutuhan jadwal akademik, kemudian dilanjutkan dengan perancangan sistem berbasis *website* menggunakan bahasa pemrograman *Python* dan *framework Flask*.

2.1.1 Rancang Bangun Sistem

Rancang bangun sistem adalah kegiatan merancang dan mengimplementasikan sistem informasi agar sesuai dengan

kebutuhan pengguna dan tujuan organisasi. Tujuan utama dari proses ini adalah memastikan bahwa sistem yang dikembangkan dapat menjalankan fungsinya secara optimal, efisien, dan dapat ditingkatkan di masa depan.

Menurut Jogyanto (2005), rancang bangun sistem terdiri dari kegiatan seperti analisis kebutuhan, perancangan proses, perancangan data, desain antarmuka, implementasi, serta pengujian dan pemeliharaan sistem.

Tahapan umum dalam rancang bangun sistem meliputi:

1. Analisis Kebutuhan

Identifikasi kebutuhan pengguna, baik fungsional (apa yang dilakukan sistem) maupun non-fungsional (kinerja, keamanan, skalabilitas).

2. Perancangan Sistem

- Desain Proses: Menggambarkan alur kerja sistem menggunakan diagram alir atau UML.
- Desain Data: Perancangan struktur *database* dan relasi antar entitas.
- Desain Antarmuka: Perancangan tampilan yang akan digunakan pengguna (*UI/UX*).

3. Implementasi

Penerjemahan rancangan ke dalam kode program menggunakan bahasa pemrograman dan alat bantu tertentu.

4. Pengujian

Pengujian ini meliputi pengujian modul, integrasi, dan sistem secara keseluruhan untuk memastikan fungsionalitas sistem berjalan sesuai spesifikasi.

5. Pemeliharaan

Proses perbaikan atau peningkatan sistem setelah digunakan, untuk menjamin sistem tetap relevan dan berfungsi dengan baik.

Dalam konteks sistem penjadwalan akademik, rancang bangun sistem mencakup pengumpulan data tentang jadwal kuliah, dosen, ruang kelas, serta penerapan logika algoritma untuk menghindari konflik waktu dan ruang.

2.1.2 Rancang Bangun Website

Rancang bangun *website* adalah proses perencanaan, perancangan, pengembangan, dan pengujian aplikasi berbasis *website* yang dapat diakses melalui jaringan *internet*. *Website* merupakan platform ideal dalam pengembangan sistem informasi karena kemudahannya dalam diakses dari berbagai perangkat dan lokasi.

Menurut Fahrezi et al. (2021), pengembangan sistem informasi akademik berbasis *website* dapat dilakukan dengan menerapkan model *Waterfall*, yang mencakup tahapan analisis kebutuhan, desain, implementasi, pengujian, dan pemeliharaan. Sistem ini memungkinkan guru dan siswa untuk mengakses

informasi akademik secara *real-time* melalui *browser*, sehingga meningkatkan efisiensi dan efektivitas dalam pengelolaan data akademik .

Komponen penting dalam rancang bangun *website* meliputi:

- Struktur Navigasi: Menentukan bagaimana pengguna berpindah antar halaman atau fitur.
- Desain Antarmuka (*UI/UX*): Menentukan tata letak, warna, ikon, dan interaksi visual.
- Manajemen Data dan Backend: Menentukan logika pemrosesan data, penyimpanan ke *database*, dan manajemen akses pengguna.
- Integrasi dengan *AI*: *Website* dalam penelitian ini mengintegrasikan sistem *AI* berbasis *algoritma genetika* untuk menghasilkan penjadwalan otomatis.

Pengembangan *website* dilakukan dengan *Python* sebagai bahasa pemrograman utama dan *Flask* sebagai *framework backend*. *Flask* dipilih karena fleksibel, ringan, dan mendukung integrasi dengan pustaka-pustaka *AI Python* seperti *DEAP* (untuk *algoritma genetika*) dan *NumPy* (untuk *komputasi numerik*).

Website ini dirancang agar:

- Mudah digunakan oleh dosen dan juga biro.
- Responsif, dapat dibuka di berbagai perangkat (desktop, tablet, ponsel).

- Terintegrasi dengan modul *AI* untuk proses penjadwalan otomatis yang cepat dan efisien.

2.2 Kecerdasan Buatan (*Artificial Intelligence*)

2.2.1 Definisi dan Sejarah AI

Kecerdasan Buatan atau *Artificial Intelligence* (*AI*) merupakan cabang ilmu komputer yang bertujuan menciptakan sistem atau mesin yang mampu melakukan tugas-tugas yang membutuhkan kecerdasan manusia, seperti pengambilan keputusan, pembelajaran dari pengalaman, pemrosesan bahasa alami, serta pengenalan pola dan gambar. *AI* dirancang agar sistem komputer dapat berpikir, bertindak, dan belajar secara mandiri melalui pemrosesan data dan algoritma canggih.

Menurut Russell dan Norvig (2010), *AI* adalah studi tentang agen cerdas yang dapat mempersepsi lingkungan dan mengambil tindakan yang memaksimalkan peluang keberhasilan dalam mencapai tujuan tertentu. Definisi ini mencerminkan peran *AI* dalam menyimulasikan kecerdasan manusia secara sistematis dan terstruktur.

Sejarah *AI* dimulai sejak tahun 1956 ketika John McCarthy, Marvin Minsky, Nathaniel Rochester, dan Claude Shannon memperkenalkan istilah "*Artificial Intelligence*" pada konferensi Dartmouth. Sejak saat itu, *AI* berkembang pesat, melewati beberapa fase penting seperti era sistem pakar (*expert system*), kebangkitan *machine learning*, dan sekarang memasuki

era *deep learning* dan integrasi *AI* dalam *Internet of Things (IoT)*, *big data*, serta *cloud computing*.

AI juga mengalami pasang surut (*AI winters*) karena keterbatasan teknologi dan harapan yang terlalu tinggi. Namun, dengan kemajuan perangkat keras, metode komputasi modern, dan ketersediaan data besar, *AI* kini telah menjadi tulang punggung dari berbagai inovasi teknologi.

2.2.2 Pendekatan dan Klasifikasi *AI*

Pendekatan dalam pengembangan *AI* diklasifikasikan berdasarkan metode dan aplikasi yang digunakan. Beberapa pendekatan utama dalam *AI* meliputi:

1. *Symbolic AI (Good Old-Fashioned AI / GOF AI)*:

Menggunakan representasi pengetahuan dan logika aturan (*rule-based systems*) untuk menyelesaikan masalah. Cocok untuk permasalahan yang dapat dipetakan dengan aturan yang jelas.

2. *Machine Learning (ML)*: Sistem belajar dari data melalui pelatihan model matematika. Terdiri dari:

- *Supervised Learning*: Model belajar dari data berlabel (contoh: regresi, klasifikasi).
- *Unsupervised Learning*: Model mencari pola dari data tanpa label (contoh: *clustering*).
- *Reinforcement Learning*: Sistem belajar melalui umpan balik dan penghargaan dari lingkungan.

3. *Natural Language Processing (NLP)*: Teknik untuk memproses dan memahami bahasa manusia, digunakan pada *chatbot*, *voice assistant*, dan analisis sentimen.
4. *Computer Vision*: Menganalisis dan menafsirkan konten visual seperti gambar dan video, digunakan dalam sistem pengenalan wajah dan kendaraan otonom.
5. *Evolutionary Computation*: Pendekatan yang meniru proses evolusi biologis, seperti *algoritma genetika*, *genetic programming*, dan *particle swarm optimization*, yang digunakan untuk menyelesaikan masalah optimasi kompleks.

Klasifikasi AI berdasarkan kemampuannya terbagi menjadi tiga kategori:

- *Artificial Narrow Intelligence (ANI)*: AI yang hanya mampu melakukan satu tugas spesifik. Contoh: sistem rekomendasi film.
- *Artificial General Intelligence (AGI)*: AI yang mampu meniru kecerdasan manusia secara menyeluruh dalam berbagai konteks. Masih dalam tahap penelitian.
- *Artificial Superintelligence (ASI)*: AI yang secara hipotetis melampaui kecerdasan manusia. Konsep ini masih bersifat teoritis dan menjadi topik etika dan keamanan teknologi.

2.2.3 AI Dalam Sistem Informasi dan Pendidikan

Kecerdasan Buatan (*AI*) telah menjadi elemen integral dalam pengembangan sistem informasi modern, termasuk dalam sektor pendidikan. *AI* digunakan untuk mendukung sistem manajemen pembelajaran, evaluasi otomatis, prediksi hasil belajar, hingga pembelajaran adaptif.

Menurut Oktavia dan Suseno (2024), *AI* memiliki potensi besar dalam mendukung proses belajar mengajar yang lebih individual, interaktif, dan inklusif. Teknologi seperti *chatbot* akademik, sistem penilaian otomatis, serta sistem informasi akademik berbasis *AI* membantu mempercepat layanan dan mengurangi beban administratif tenaga pendidik.

Selain itu, penelitian oleh Sihaloho dan Napitupulu (2023) menunjukkan bahwa *AI* dapat meningkatkan efisiensi dan efektivitas pembelajaran, memberikan umpan balik yang lebih personal, dan membantu pengambilan keputusan oleh pendidik. Namun, implementasi *AI* juga menghadapi berbagai tantangan seperti biaya tinggi, kebutuhan akan infrastruktur teknologi, dan isu etika.

Contoh penerapan *AI* dalam pendidikan meliputi:

1. *Chatbot* Akademik: Memberikan informasi akademik otomatis kepada mahasiswa.
2. *Adaptive Learning Systems*: Sistem yang menyesuaikan materi berdasarkan performa dan kebutuhan siswa.

3. Sistem Prediktif Akademik: Memprediksi performa belajar siswa berdasarkan data historis.
4. AI dalam Sistem Informasi Akademik: Mengelola jadwal, absensi, nilai, dan kegiatan belajar secara otomatis.

Integrasi AI dalam sistem informasi pendidikan memungkinkan penciptaan lingkungan belajar yang inklusif dan fleksibel melalui platform *e-learning* yang cerdas dan terintegrasi.

2.2.4 Relevansi AI terhadap Penjadwalan Akademik

Permasalahan penjadwalan akademik merupakan salah satu tantangan yang sangat kompleks dan sering ditemukan dalam dunia pendidikan, terutama di perguruan tinggi. Masalah ini mencakup banyak variabel, seperti jumlah mata kuliah, ketersediaan dosen, keterbatasan ruang, waktu, dan jumlah mahasiswa.

Menurut Pambudi et al. (2021), penyusunan jadwal kuliah tergolong *NP-Hard Problem*, karena tidak dapat diselesaikan secara eksak dalam waktu yang wajar jika menggunakan pendekatan konvensional. Oleh karena itu, diperlukan metode alternatif seperti AI yang mampu melakukan optimasi dalam ruang solusi besar.

Penggunaan *algoritma genetika* sebagai bagian dari AI telah terbukti efektif dalam menyelesaikan masalah penjadwalan. Algoritma ini bekerja dengan prinsip seleksi alam, menggunakan proses seperti *inisialisasi populasi*, *evolusi fitness*, *seleksi*,

crossover, dan *mutasi* untuk menghasilkan solusi yang optimal atau mendekati optimal.

Manfaat penggunaan *AI* dalam penjadwalan akademik:

1. Menghindari konflik jadwal (kelas, dosen, dan ruang)
2. Mengoptimalkan alokasi sumber daya (ruang dan waktu)
3. Fleksibel terhadap perubahan jadwal mendadak
4. Otomatisasi dan efisiensi waktu penyusunan jadwal

Integrasi sistem *AI* dengan antarmuka *website* memungkinkan pengguna seperti dosen atau admin akademik untuk mengelola penjadwalan secara *real-time*, responsif, dan mudah digunakan.

2.3 Sistem Penjadwalan Berbasis AI

2.3.1 Konsep Penjadwalan

Penjadwalan (*scheduling*) adalah proses pengalokasian sumber daya terbatas terhadap sejumlah tugas dalam suatu periode waktu tertentu dengan tujuan untuk mengoptimalkan kinerja sistem. Penjadwalan sering digunakan dalam berbagai bidang seperti manufaktur, transportasi, pendidikan, dan sistem operasi komputer.

Ruang lingkup penjadwalan melibatkan elemen-elemen seperti:

1. Tugas : Aktivitas atau pekerjaan yang harus dijadwalkan.
2. Sumber daya: Objek terbatas yang dibutuhkan oleh tugas (misalnya, dosen, ruang, waktu).

3. Kendala: Aturan atau batasan yang harus dipenuhi (misalnya, tidak boleh ada jadwal bentrok).

Adapun jenis-jenis penjadwalan antara lain:

1. Penjadwalan statis: Seluruh jadwal dirancang di awal dan tidak berubah selama periode pelaksanaan.
2. Penjadwalan dinamis: Jadwal disesuaikan berdasarkan kondisi yang berubah secara *real-time*.
3. Penjadwalan *real-time*: Digunakan dalam sistem yang memerlukan reaksi cepat terhadap input eksternal, seperti sistem kendali atau perangkat medis.

Permasalahan umum dalam penjadwalan meliputi:

1. Konflik penggunaan sumber daya (misalnya, ruang kelas yang sama digunakan pada waktu yang sama).
2. Prioritas tugas yang berbeda.
3. Keterbatasan waktu atau ketersediaan sumber daya.

2.3.2 Masalah penjadwalan kuliah

Dalam lingkungan pendidikan tinggi, penjadwalan kuliah merupakan proses kompleks yang melibatkan berbagai entitas seperti dosen, mahasiswa, ruang kelas, dan waktu kuliah.

Penjadwalan harus mempertimbangkan banyak kendala seperti:

1. Dosen tidak dapat mengajar dua kelas sekaligus.
2. Mahasiswa tidak boleh memiliki kelas yang tumpang tindih.
3. Kapasitas ruang kelas harus mencukupi jumlah mahasiswa.

4. Preferensi waktu dosen atau pembatasan jam mengajar.

Permasalahan penjadwalan kuliah dikategorikan sebagai *NP-hard problem*, yaitu permasalahan komputasional yang tidak dapat diselesaikan secara efisien menggunakan pendekatan *brute-force* karena jumlah kombinasi solusi yang sangat besar. Oleh karena itu, diperlukan pendekatan cerdas yang mampu mengeksplorasi ruang solusi secara optimal dalam waktu yang terbatas.

Menurut Pambudi et al. (2021), penyusunan jadwal kuliah yang optimal memerlukan pertimbangan berbagai kendala yang saling berkaitan, sehingga metode konvensional seringkali tidak mampu menghasilkan solusi yang efisien dan bebas konflik.

2.3.3 AI dalam Penyelesaian Penjadwalan

Kecerdasan Buatan (AI) menawarkan pendekatan yang lebih efisien dan adaptif untuk menyelesaikan masalah penjadwalan kuliah. Salah satu metode yang umum digunakan adalah algoritma genetika (*genetic algorithm*), yaitu algoritma pencarian berbasis populasi yang terinspirasi dari proses seleksi alam dalam *biologi evolusioner*.

Algoritma genetika bekerja dengan prinsip sebagai berikut:

1. *Inisialisasi*: Membuat populasi awal yang terdiri dari solusi jadwal acak.
2. *Evolusi*: Menghitung *fitness* setiap individu berdasarkan jumlah pelanggaran kendala.

3. Seleksi: Memilih individu terbaik untuk proses reproduksi.
4. *Crossover*: Menggabungkan dua individu untuk menghasilkan solusi baru.
5. *Mutasi*: Mengubah sebagian kecil individu untuk meningkatkan keberagaman solusi.
6. *Iterasi*: Proses diulang hingga diperoleh solusi terbaik atau jumlah generasi maksimum tercapai.

Menurut Amrulloh dan Sela (2021), *algoritma genetika* mampu menghasilkan solusi penjadwalan yang lebih optimal dibanding metode *heuristik* konvensional karena kemampuannya dalam menelusuri ruang solusi yang sangat besar secara efisien dan adaptif terhadap perubahan kendala.

Dengan menerapkan *AI* dalam bentuk *algoritma genetika*, sistem penjadwalan kuliah dapat menghasilkan jadwal otomatis yang :

1. Bebas dari konflik waktu, ruang, dan dosen.
2. Dapat disesuaikan dengan preferensi pengguna.
3. Dioptimalkan secara komputasional dalam waktu singkat.

Hal ini membuat *AI* menjadi pilihan teknologi yang sangat relevan untuk pengembangan sistem informasi akademik modern yang bersifat dinamis dan kompleks.

2.4 Algoritma Genetika

2.4.1 Definisi dan Sejarah Algoritma Genetika

Algoritma genetika (GA) adalah salah satu teknik dalam kecerdasan buatan (AI) yang digunakan untuk menyelesaikan

permasalahan optimasi dan pencarian solusi. GA terinspirasi oleh mekanisme evolusi makhluk hidup di alam, khususnya teori seleksi alam yang dikemukakan oleh Charles Darwin. Dalam GA, solusi terhadap suatu permasalahan di analogikan sebagai individu dalam suatu populasi yang akan berevolusi menuju solusi terbaik dari generasi ke generasi (Haupt & Haupt, 2004).

GA pertama kali diperkenalkan secara formal oleh John Holland pada tahun 1975 melalui bukunya yang berjudul *Adaptation in Natural and Artificial Systems*. Holland dan murid-muridnya mengembangkan AG sebagai metode simulasi evolusi buatan dengan pendekatan matematis dan komputasional (Holland, 1975). Mereka membentuk dasar pengembangan algoritma yang memanfaatkan proses *seleksi*, *crossover* (penyilangan), dan mutasi yang merupakan komponen inti dalam GA.

Dalam perkembangannya, *algoritma genetika* mulai banyak digunakan dalam berbagai bidang, seperti sistem pakar, pemodelan jaringan syaraf tiruan, pemrosesan citra, dan sistem penjadwalan. GA sangat efektif dalam menyelesaikan masalah yang memiliki ruang solusi besar dan tidak dapat dipecahkan dengan metode konvensional. Hal ini karena GA mampu mengeksplorasi berbagai kemungkinan solusi secara paralel dan menghindari jebakan pada solusi lokal (Mitchell, 1998).

GA termasuk dalam kategori *evolutionary computation*, yaitu pendekatan dalam kecerdasan buatan yang mengandalkan

prinsip-prinsip evolusi biologis sebagai model penyelesaian masalah. Dengan pendekatan ini, *algoritma genetika* dapat digunakan untuk menyelesaikan permasalahan kompleks, seperti penjadwalan akademik, yang melibatkan banyak variabel dan kendala, serta memiliki ruang pencarian solusi yang sangat luas dan tidak linier.

Karakteristik utama dari algoritma genetika meliputi:

- Representasi solusi dalam bentuk *kromosom* (biasanya sebagai string biner atau struktur data lainnya)
- Penggunaan fungsi *fitness* untuk mengevaluasi kualitas solusi
- Penggunaan operator *seleksi*, *crossover*, dan *mutasi*
- Proses *iteratif* yang melibatkan pembentukan generasi baru

Dalam konteks sistem penjadwalan kuliah, *algoritma genetika* dimanfaatkan untuk menghasilkan kombinasi jadwal optimal secara otomatis dengan mempertimbangkan berbagai kendala seperti ketersediaan dosen, waktu, ruang kelas, dan distribusi mata kuliah.

Tujuan utama dari *algoritma genetika* adalah menemukan solusi optimal atau mendekati optimal dari suatu permasalahan, terutama permasalahan yang bersifat kompleks dan tidak dapat diselesaikan dengan *algoritma deterministik* dalam waktu komputasi yang wajar.

2.4.2 Komponen Utama Algoritma Genetika

Algoritma genetika terdiri atas beberapa komponen utama yang membentuk proses evolusi buatan dalam pencarian solusi optimal. Komponen-komponen ini memungkinkan algoritma untuk beradaptasi dan menemukan solusi yang lebih baik secara iteratif. Berikut adalah penjelasan rinci dari setiap komponen:

1. Kromosom

Kromosom merupakan representasi dari solusi dalam ruang pencarian. Dalam konteks penjadwalan kuliah, satu kromosom berisi serangkaian gen, di mana setiap gen menyatakan satu kombinasi mata kuliah, dosen, waktu, dan ruang kuliah. Representasi ini bisa berupa *array*, *string*, atau struktur data yang disesuaikan dengan kebutuhan aplikasi.

2. Populasi

Populasi adalah kumpulan *kromosom* yang terdiri dari solusi-solusi potensial. Ukuran populasi memengaruhi eksplorasi solusi: terlalu kecil menyebabkan *konvergensi prematur*, terlalu besar meningkatkan waktu komputasi.

3. Fitness Function

Fitness function digunakan untuk mengukur seberapa baik solusi memenuhi tujuan. Dalam penjadwalan, fungsi ini menilai apakah solusi bebas konflik dan sesuai preferensi.

Rumus dasar *fitness function* :

$$f(x) = \frac{1}{1+c(x)} \dots\dots\dots (2.1)$$

Untuk multi kendala :

$$f(x) = \frac{1}{1+\sum_{i=1}^n w_i \cdot c_i(x)} \dots\dots\dots (2.2).$$

Dimana :

$c(x)$: Total konflik

w_i : Bobot kendala ke-i

w_i : Konflik Kendala pada kendala ke-i

4. Seleksi

Seleksi bertujuan memilih individu terbaik sebagai induk generasi baru. Salah satu metode yang umum adalah

Roulette Wheel Selection:

$$P = \frac{F_i}{\sum_{j=1}^N F_j} \dots\dots\dots (2.3)$$

Dimana :

P_i : Peluang individu ke-iii dipilih untuk crossover

F_i : Nilai fitness individu ke-iii

N : Jumlah total individu dalam populasi

Diaman semakin tinggi nilai *fitness* F_i , semakin besar peluangnya dipilih.

5. Crossover

Crossover bertujuan menghasilkan kombinasi genetik baru dengan menggabungkan bagian dari dua individu induk.

Salah satu metode yang digunakan adalah **One-Point**

Crossover:

$$Anak_1 = [a_1, \dots, a_k, b_k + 1, \dots, b_n]$$

$$Anak_2 = [b_1, \dots, b_k, a_k + 1, \dots, a_n] \dots \dots \dots (2.4)$$

Tingkat *crossover* umumnya:

$$R_c \in [0.6, 0.9] \dots \dots \dots (2.5)$$

Dimana :

$a_1, \dots, a_n$: Gen dari induk pertama

$b_1, \dots, b_n$: Gen dari induk kedua

k : Titik potong (*crossover point*), dipilih secara acak
antara 1 hingga $n-1$

$Anak_1$: Anak hasil gabungan bagian awal induk 1
dan akhir induk 2

$Anak_2$: Anak hasil gabungan bagian awal induk 2
dan akhir induk 1

R_c : Peluang dua individu (induk) melakukan proses
crossover.

0.6 : 60% pasangan induk akan mengalami *crossover*.

0.9 : *crossover* dilakukan secara agresif, 90% pasangan
mengalami pertukaran gen.

6. Mutasi

Mutasi menjaga keberagaman populasi dan mencegah
konvergensi terhadap solusi lokal. Mutasi dilakukan

dengan mengganti nilai suatu gen dengan nilai baru secara acak.

$$Gen'_i = \begin{cases} 1, & \text{jika } Gen_i = 0 \\ 0, & \text{jika } Gen_i = 1 \end{cases} \dots\dots\dots (2.6)$$

Tingkat Mutasi

$$R_m \in [0.01, 0.1] \dots\dots\dots (2.7)$$

Dimana :

Gen_i : Nilai gen ke-i sebelum mutasi

Gen'_i : nilai gen ke-I setelah mutasi

R_m : probabilitas mutasi per gen

0.01 : 1% gen dalam populasi dapat di mutasi setiap waktu

0.1 : 10% gen dalam populasi dapat di mutasi setiap waktu

2.4.3 Tahapan Algoritma Genetika

Tahapan dalam *algoritma genetika* mencerminkan prinsip-prinsip evolusi biologis yang diterapkan dalam pencarian solusi optimal terhadap suatu permasalahan. Setiap tahapan memainkan peran penting dalam mengarahkan populasi menuju solusi terbaik. Berikut adalah tahapan-tahapan utama *algoritma genetika*:

1. Inisialisasi Populasi

Tahapan pertama adalah pembuatan populasi awal yang terdiri atas sejumlah individu (solusi). Setiap individu atau *kromosom* dibentuk secara acak atau berdasarkan *heuristik* tertentu yang sesuai dengan permasalahan.

Misalnya, dalam sistem penjadwalan, satu *kromosom* dapat berisi urutan jadwal mata kuliah, ruang, dosen, dan waktu. Tujuan dari inisialisasi ini adalah memberikan variasi solusi awal yang dapat dievaluasi dalam generasi berikutnya.

2. Evolusi Fitness

Setelah populasi awal terbentuk, setiap individu dievolusi menggunakan fungsi *fitness*. Fungsi ini mengukur kualitas solusi dengan memperhatikan sejumlah kriteria atau kendala. Dalam konteks penjadwalan, penilaian bisa dilakukan dengan menghitung jumlah konflik (misalnya bentrok ruang atau waktu) dan pelanggaran terhadap preferensi dosen. Semakin sedikit konflik, semakin tinggi nilai *fitness* individu tersebut.

3. Seleksi

Seleksi dilakukan untuk memilih individu-individu terbaik yang akan digunakan sebagai induk dalam proses *reproduksi*. Individu dengan nilai *fitness* lebih tinggi memiliki peluang lebih besar untuk terpilih. Salah satu metode yang umum digunakan adalah *Roulette Wheel Selection*, yaitu metode pemilihan berdasarkan proporsi nilai *fitness* masing-masing individu terhadap total nilai *fitness* populasi.

4. Crossover

Crossover adalah proses rekombinasi informasi genetik antara dua induk untuk menghasilkan satu atau dua keturunan. Salah satu metode umum adalah *One-Point Crossover*, di mana titik potong dipilih secara acak dan bagian gen setelah titik tersebut dipertukarkan antar induk.

5. Mutasi

Mutasi dilakukan untuk menjaga keberagaman genetik dalam populasi. Proses ini mengubah nilai satu atau lebih gen dalam individu secara acak, dengan probabilitas rendah. Pada representasi biner, mutasi dilakukan dengan membalik nilai bit dari 0 ke 1 atau sebaliknya.

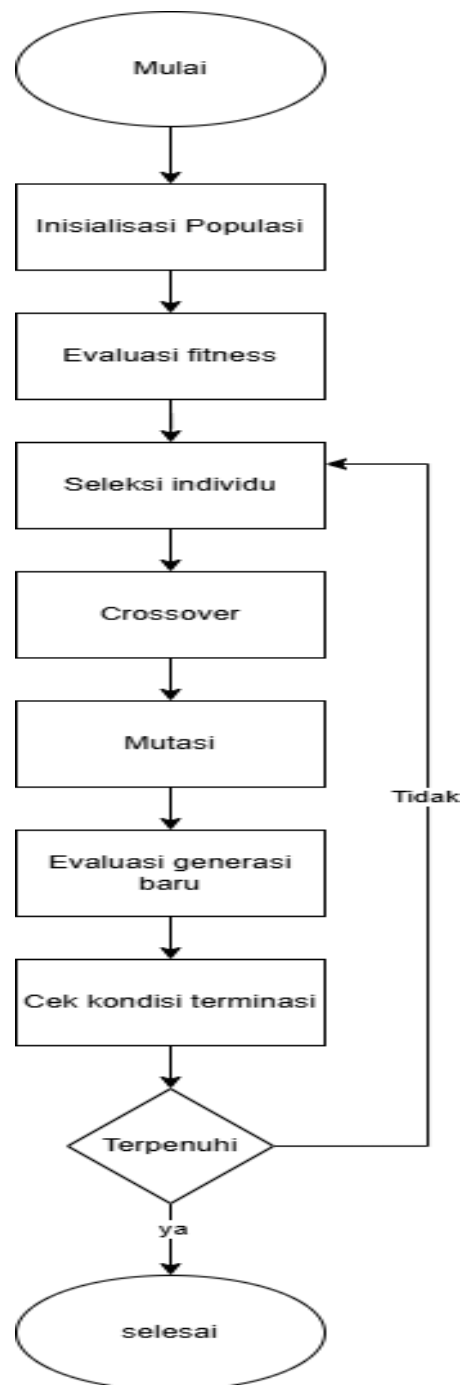
6. Evolusi pupulasi baru

Setelah proses *crossover* dan mutasi menghasilkan individu-individu baru, populasi diperbarui dan kembali dievolusi menggunakan fungsi *fitness*. Evolusi ini bertujuan mengidentifikasi individu-individu terbaik dari generasi baru untuk digunakan pada iterasi berikutnya.

7. Terminasi

Proses *algoritma genetika* diakhiri jika telah memenuhi kondisi terminasi. Kondisi ini dapat berupa jumlah

generasi maksimal yang tercapai, nilai *fitness* yang telah memadai, atau tidak adanya peningkatan kualitas solusi setelah beberapa generasi.



Gambar 2.1 Tahapan Algoritma Genetika

2.5 Chatbot

Sebagai bagian dari sistem berbasis kecerdasan buatan (AI), *chatbot* dalam penelitian ini berfungsi sebagai komponen pendukung interaksi pengguna dengan sistem penjadwalan akademik yang dibangun menggunakan *algoritma genetika*. *Chatbot* memungkinkan pengguna, baik mahasiswa maupun dosen, untuk mengakses informasi jadwal secara cepat dan interaktif tanpa perlu menelusuri seluruh sistem secara manual. Dengan demikian, *chatbot* menjadi jembatan antara pengguna dan sistem penjadwalan, sekaligus meningkatkan efisiensi layanan dalam lingkungan pendidikan tinggi.

2.5.1 Definisi dan Kosep Dasar Chatbot

Chatbot merupakan singkatan dari “*chatting robot*”, yaitu program komputer yang dirancang untuk mensimulasikan percakapan manusia secara otomatis melalui antarmuka teks maupun suara. *Chatbot* bekerja dengan menerima input dari pengguna, memprosesnya menggunakan skrip atau algoritma berbasis kecerdasan buatan, kemudian memberikan respons yang sesuai. Tujuan dari *chatbot* adalah memberikan layanan interaktif secara cepat, efisien, dan tersedia sepanjang waktu tanpa keterlibatan manusia secara langsung (Adamopoulou & Moussiades, 2020).

Pada dasarnya, *chatbot* dibagi menjadi dua kategori:

1. *Rule-based*, menggunakan skenario percakapan yang telah ditentukan sebelumnya

2. *AI-based*, yang memanfaatkan *Natural Language Processing (NLP)* dan pembelajaran mesin untuk memahami konteks dan memberikan jawaban dinamis.

Perkembangan teknologi chatbot yang semakin cepat, terutama dengan kemajuan model bahasa besar (*large language models, LLM*) seperti GPT-3/4, yang memperluas kemampuan chatbot untuk menjawab secara kontekstual dan realistis. Penggunaan *chatbot* sudah meluas dalam layanan pelanggan, sistem informasi, dan pendidikan, termasuk sistem informasi akademik, karena kemampuannya menjawab pertanyaan jadwal, pengumuman, layanan administratif, dan lain-lain secara *real-time* dan personal (Qalimaturrahmah & Santoso, 2024).

Dalam konteks skripsi ini, *chatbot* dikembangkan sebagai antarmuka pengguna terhadap sistem penjadwalan berbasis *algoritma genetika*, sehingga mahasiswa dapat mengakses jadwal mereka dengan cukup mengirimkan pertanyaan melalui teks, tanpa harus membuka dan mencari manual dalam halaman web.

2.5.2 Arsitektur dan Komponen Chatbot

Pendekatan arsitektur *chatbot* dalam konteks sistem informasi akademik mengacu pada struktur modular yang umum: *NLU (Natural Language Understanding)*, *Dialog Manager*, dan *NLG (Natural Language Generation)*, serta modul pendukung seperti integrasi kanal, *database*, dan *logging*.

Natural Language Understanding (NLU) merupakan tahap awal yang berfungsi untuk menganalisis dan memahami input pengguna. Teknologi *NLU* memproses kalimat dengan cara memetakan maksud pengguna (*intent recognition*) serta mengekstraksi informasi penting (*entity extraction*). Contohnya, dalam pertanyaan “Kapan kuliah Pemrograman Web?”, *NLU* bertugas mengenali bahwa maksud pengguna adalah menanyakan jadwal kuliah, serta mengenali entitas “Pemrograman Web” sebagai mata kuliah.

Setelah maksud dan entitas dikenali, proses dilanjutkan ke *Dialog Management*, yang bertugas menentukan langkah percakapan berikutnya berdasarkan konteks. *Dialog manager* menyimpan *state* (keadaan) dari percakapan dan memutuskan tanggapan sistem berdasarkan logika atau skenario tertentu. Sistem ini bisa berbasis aturan (*rule-based*) atau menggunakan model pembelajaran mesin untuk menentukan respons yang lebih fleksibel (Serban et al., 2017).

Komponen terakhir adalah *Natural Language Generation (NLG)*, yang bertugas menyusun dan menghasilkan kalimat respons dalam bentuk bahasa alami. *NLG* menerima output dari *dialog manager* dalam bentuk struktur data atau template, lalu mengubahnya menjadi kalimat yang mudah dipahami pengguna.

Dalam sistem penjadwalan akademik, *chatbot* yang dibangun dapat diintegrasikan dengan basis data jadwal dan

sistem *AI* seperti *algoritma genetika*. Dengan demikian, *chatbot* tidak hanya menjawab pertanyaan statis, tetapi juga dapat memberikan informasi dinamis seperti: “Mata kuliah apa saja yang dosen Ajaran ajar hari ini?” atau “Kapan saya ada kelas kosong?” Ini memungkinkan sistem memberikan dukungan interaktif bagi mahasiswa dan dosen dalam mengakses jadwal perkuliahan secara efisien.

2.5.3 Teknologi dan Bahasa Pemrograman Chatbot

Dalam pengembangan *chatbot*, pemilihan bahasa pemrograman dan platform menjadi aspek penting yang menentukan fleksibilitas, kecanggihan, serta kemudahan integrasi dengan sistem yang lebih besar. Salah satu bahasa pemrograman yang paling banyak digunakan dalam pembangunan *chatbot* adalah Python, karena sintaksisnya yang sederhana dan ketersediaan berbagai pustaka (*library*) pendukung *chatbot* maupun *Natural Language Processing (NLP)*.

Python menyediakan banyak *framework* yang mendukung pembuatan *chatbot*, seperti *Rasa*, *ChatterBot*, dan *AIML (Artificial Intelligence Markup Language)*. *Rasa* adalah *framework open-source* yang sangat populer untuk *chatbot* berbasis *AI* karena menyediakan modul pemahaman bahasa (*NLU*) dan pengelolaan dialog (*Core*). *Framework* ini dapat dikombinasikan dengan pustaka *NLP* seperti *spaCy* atau *TensorFlow* untuk meningkatkan akurasi respons. Penelitian oleh Qalimaturrahmah dan Santoso

(2024) menunjukkan bahwa *chatbot* yang dibangun menggunakan *Rasa* dan di integrasikan dengan Telegram mampu memberikan respons cepat dan mendukung proses layanan akademik yang efisien.

Selain *Rasa*, *AIML* merupakan *framework* scripting berbasis *XML* yang banyak digunakan untuk membangun *chatbot rule-based*. *AIML* bersifat ringan dan cocok untuk pembuatan chatbot yang menangani pertanyaan tetap (statik) dalam domain terbatas, seperti informasi jadwal kuliah atau prosedur akademik. Ajiz et al. (2023) dalam penelitiannya mengembangkan chatbot akademik berbasis *AIML* dan *Python*, serta mengintegrasikannya ke dalam aplikasi web. Hasilnya, *chatbot* tersebut mampu menjawab pertanyaan-pertanyaan akademik secara cepat dan konsisten.

Dengan demikian, chatbot bukan hanya berfungsi sebagai antarmuka interaktif, tetapi juga sebagai bagian dari sistem *AI* yang mendukung layanan penjadwalan akademik berbasis *algoritma genetika*.

2.6 Penelitian Terdahulu

Tabel 2.1. Penelitian Terdahulu

No	Penulis	Tahun	Judul
1	Evi Ferawety	2010	Optimasi Penjadwalan Mata Kuliah Pada Perguruan Tinggi

			Mennguna-kan Algoritma Genetika
2	Muhammad fauzi, Zulfikri Akbarl, Kevin Kurniawansyah	2024	Sistem Penjadwalan Sidang Skripsi Dengan Algoritma Genetika Pada Program Studi Informatik Universitas Muhammadiyah Jambi
3	Daniel Jeriko Panjaitan	2022	Sistem Informasi Penjadwalan Mata Pelajaran Berbasis Web Menggunakan Algoritma Genetika Pada Smk Negeri 3 Medan
4	Asep Toyib Hidayat, Lukman Hakim, Rio, M Afif Ravanza	2023	Implementasi Sistem Penjadwalan Mata Kuliah Menggunakan Metode Algoritma Genetika Berbasis Web
5	Andrie Tri Laksono, Meinarini Catur Utami, Yuni Sugiarti	2016	Sistem Penjadwalan Kuliah Menggunakan Metode Algoritma Genetika (Studi Kasus: Fakultas Kedokteran Dan Kesehatan Universitas Muha-mmadiyah Jakarta)

BAB III

METODOLOGI PENELITIAN

3.1 Jenis dan Pendekatan Penelitian

Penelitian ini termasuk dalam jenis penelitian rekayasa perangkat lunak (*Software Engineering Research*) yang berfokus pada pengembangan sistem informasi akademik dalam bentuk website penjadwalan kegiatan belajar mahasiswa berbasis kecerdasan buatan (*Artificial Intelligence*), khususnya menggunakan *algoritma genetika*. Penelitian ini memadukan aspek teoritis dan teknis dengan tujuan menghasilkan produk perangkat lunak nyata yang dapat digunakan langsung di lingkungan Fakultas Ilmu Komputer dan Teknologi Informasi.

3.1.1 Pendekatan Research and Development (R&D)

Pendekatan yang digunakan adalah *Research and Development (R&D)*, yaitu pendekatan penelitian yang bertujuan untuk menemukan, merancang, mengembangkan, dan menguji validitas suatu produk tertentu (Sugiyono, 2016). Dalam konteks ini, produk yang dikembangkan berupa website penjadwalan otomatis yang mengintegrasikan *algoritma genetika* sebagai mesin optimasi jadwal, serta sistem *chatbot AI* sebagai antarmuka pengguna.

Pendekatan *R&D* dianggap sesuai karena fokus penelitian tidak hanya pada analisis atau evaluasi sistem yang sudah ada, tetapi juga pada proses penciptaan sistem baru, mulai dari tahap konsep hingga implementasi, pengujian, dan evaluasi.

3.1.2 Model Pengembangan Sistem

Model pengembangan sistem yang digunakan adalah model *Waterfall*, yang dikenal sebagai model klasik dalam rekayasa perangkat lunak (Pressman, 2010). *Waterfall* bersifat linier dan sistematis, cocok untuk proyek dengan kebutuhan yang telah ditentukan secara jelas dari awal. Tahapan dalam model ini meliputi:

1. Analisis Kebutuhan

Mengumpulkan kebutuhan sistem dari pengguna (admin akademik dan dosen) serta mendefinisikan batasan masalah dan tujuan pengembangan sistem. Kebutuhan berupa struktur data jadwal, mata kuliah, waktu, dosen, dan ruang kelas

2. Perancangan Sistem

Merancang struktur basis data, arsitektur sistem, antarmuka pengguna (*UI*), dan logika algoritma genetika yang digunakan untuk mengoptimalkan jadwal.

3. Implementasi

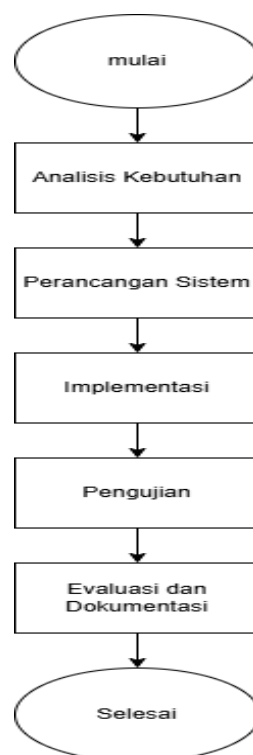
Menerapkan desain menjadi kode program menggunakan *Python* dengan *framework Flask*, serta menggunakan *library DEAP* untuk membangun *algoritma genetika*.

4. Pengujian Sistem

Menguji sistem menggunakan metode *black-box testing* untuk memastikan fungsionalitas sistem berjalan sesuai spesifikasi. Pengujian juga dilakukan untuk mengevaluasi performa algoritma dalam menghasilkan jadwal yang bebas konflik.

5. Evaluasi dan Dokumentasi

Melakukan evaluasi terhadap hasil implementasi sistem, berdasarkan metrik seperti jumlah konflik yang tersisa, waktu proses penjadwalan, serta tingkat kepuasan pengguna dari hasil simulasi internal di lingkungan Fakultas Ilmu Komputer dan Teknologi Informasi.



Gambar 3.1 Model Waterfall

3.2 Pengumpulan Data

Data yang digunakan dalam penelitian ini merupakan data sekunder yang diperoleh dari data data perkuliahan yang diselenggarakan di Fakultas Ilmu Komputer dan Teknologi Informasi untuk jenjang pendidikan S1 Reguler pada tahun ajaran 2024/2025 yang diperoleh dari Biro Administrasi Fakultas Ilmu Komputer dan Teknologi Informasi. Data yang digunakan untuk penelitian penjadwalan kegiatan belajar mahasiswa adalah sebagai berikut :

1. Data mata kuliah

Berisi informasi tentang Jurusan, Semester, nama mata kuliah dan jumlah satuan kredit semester (SKS)

2. Data dosen

Memuat daftar nama dosen beserta mata kuliah yang mereka ajarkan, serta kemungkinan preferensi waktu mengajar (jika tersedia).

3. Data jam Perkuliahan dan Data ruangan kuliah

Meliputi daftar ruang kuliah yang tersedia di lingkungan FIKTI serta kode ruangan dan berisi pembagian jam kuliah dalam satu hari, baik untuk hari efektif (Senin–Sabtu), mencakup jam mulai dan jam selesai untuk setiap sesi kuliah.

3.2.1 Data Mata Kuliah

Data mata kuliah yang digunakan dalam penelitian ini berdasarkan pada kurikulum Fakultas Ilmu Komputer dan Teknologi Informasi yang berlaku untuk tahun ajaran 2024/2025.

Semua mata kuliah akan di kelompokkan terlebih dahulu berdasarkan jurusan, semester, mata kuliah dan jumlah sks mata kuliah yang bersangkutan. Struktur mata kuliah dapat di lihat pada tabel-tabel berikut.

Tabel 3.1. Data Mata Kuliah Teknologi Informasi Tahun 2022

JURUSAN & SEMESTER	MATA KULIAH	SKS
TI/2022	Big Data Analitik **	3
TI/2022	Enterprise Development Software	2
TI/2022	Metodelogi Penelitian	2
TI/2022	Neurofuzzy **	3
TI/2022	Pemograman Integrative	2
TI/2022	Penulisan Proposal	2
TI/2022	Prak. Pemograman Integrative	1
TI/2022	Prak. Sistem Multimedia	1
TI/2022	Prak. Web Semantik	1
TI/2022	Sistem Multimedia	2
TI/2022	Sistem Sensor & Aplikasi **	3
TI/2022	Web Semantik	2

Tabel 3.2. Data Mata Kuliah Teknologi Informasi Tahun 2023

JURUSAN & SEMESTER	MATA KULIAH	SKS
TI/2023	Islam & Iptek	2
TI/2023	Keamanan Informasi	2
TI/2023	Kecerdasan Buatan	3
TI/2023	Manajemen Proyek TI	2
TI/2023	Pengolahan Citra Digital	2
TI/2023	Prak. Pengolahan Citra Digital	1
TI/2023	Prak. Sistem Mikroprosesor & Assembly	1
TI/2023	Prak. Statistika & Probabilitas	1
TI/2023	Prak. Teknologi Antar Jaringan	1
TI/2023	Sistem Mikroprosesor & Assembly	2
TI/2023	Statistika & Probabilitas	2
TI/2023	Teknologi Antar Jaringan	2

Tabel 3.3. Data Mata Kuliah Teknologi Informasi Tahun 2024

JURUSAN & SEMESTER	MATA KULIAH	SKS
TI/2024	Aljabar, Vektor & Matrik	3
TI/2024	Bahasa Inggris	2
TI/2024	Ibadah & Muamalah	2
TI/2024	Kewarganegaraan	2
TI/2024	Kewirausahaan IT	2
TI/2024	Pengantar Organisasi & Arsitektur TI	2
TI/2024	Prak. Sistem Operasi	1
TI/2024	Prak. Struktur Data	1
TI/2024	Sistem Operasi	2
TI/2024	Struktur Data	2
TI/2024	Teknologi Open Source	2

Tabel 3.4. Data Mata Kuliah Sistem Informasi Tahun 2021

JURUSAN & SEMESTER	MATA KULIAH	SKS
SI/2021	Etika Profesi IT	2

Tabel 3.5. Data Mata Kuliah Sistem Informasi Tahun 2022

JURUSAN & SEMESTER	MATA KULIAH	SKS
SI/2022	Enterprise Application Integration **	3
SI/2022	Customer Relationship Management **	3
SI/2022	Desain Pengalaman Pengguna (UI UX Exp)	2
SI/2022	Enterprise Resource Planning **	3
SI/2022	Kontrol & Audit sistem Informasi **	3
SI/2022	Metodelogi Penelitian	3
SI/2022	Pemograman Mobile	2
SI/2022	Pemograman Web Lanjut	2
SI/2022	Prak. Desain Pengalaman Pengguna (UI UX Exp)	1
SI/2022	Prak. Pemograman Mobile	1
SI/2022	Prak. Pemograman Web Lanjut	1

Tabel 3.6. Data Mata Kuliah Sistem Informasi Tahun 2023

JURUSAN & SEMESTER	MATA KULIAH	SKS
SI/2023	Kecerdasan Buatan	3
SI/2023	Tata Kelola Sistem Informasi	3
SI/2023	Kriptografi & Sekuriti Sistem	3
SI/2023	Pemograman Desktop	2
SI/2023	Islam & Iptek	2
SI/2023	Manajemen & Desain Jaringan Komputer	2
SI/2023	Pemograman Web Dasar	2
SI/2023	Prak. Pemograman Desktop	1
SI/2023	Prak. Manajemen & Desain Jaringan Komputer	1
SI/2023	Prak. Pemograman Web Dasar	1

Tabel 3.7. Data Mata Kuliah Sistem Informasi Tahun 2024

JURUSAN & SEMESTER	MATA KULIAH	SKS
SI/2024	Prak. Sistem Operasi	1
SI/2024	Prak. Teknologi Open Source	1
SI/2024	Prak. Struktur Data	1
SI/2024	Ibadah & Muamalah	2
SI/2024	Sistem Operasi	2
SI/2024	Teknologi Open Source	2
SI/2024	Organisasi Manajemen	2
SI/2024	Kewarganegaraan	2
SI/2024	Struktur Data	2
SI/2024	Islam & Iptek	2
SI/2024	Matematika Diskrit	3

Tabel 3.8. Data Mata Kuliah Sains Data Tahun 2023

JURUSAN & SEMESTER	MATA KULIAH	SKS
SD/2023	Kecerdasan & Analisis Bisnis	2
SD/2023	Pemodelan Data Bisnis	2
SD/2023	Teknologi Big Data	2
SD/2023	Visualisasi Data & Informasi	2
SD/2023	Prak. Visualisasi Data & Informasi	1

SD/2023	Web Programming	2
SD/2023	Prak. Web Programming	1
SD/2023	AI Fundamental	3
SD/2023	Islam & Iptek	2
SD/2023	Statistika Regresi	2
SD/2023	Riset Operasi	2

Tabel 3.9. Data Mata Kuliah Sains Data Tahun 2024

JURUSAN & SEMESTER	MATA KULIAH	SKS
SD/2024	Aljabar linier Elementer	2
SD/2024	Interaksi Manusia & Komputer	2
SD/2024	Struktur Data	2
SD/2024	Prak. Struktur Data	1
SD/2024	Ibadah & Muamalah	2
SD/2024	Bahasa Inggris (Communicative English)	2
SD/2024	Teori Peluang	2
SD/2024	Statistika Komputasi	2
SD/2024	Kewarganegaraan	2
SD/2024	Digital Marketing Strategy	3

3.2.2 Data Dosen

Data dosen yang digunakan dalam penelitian ini yaitu berupa nama-nama dosen yang mengajar di Fakultas Ilmu Komputer dan Teknologi Informasi beserta mata kuliah yang diajarkan pada perkuliahan yang diadakan di semester genap. Data dosen dapat di lihat pada Tabel 3.10.

Data dosen ini akan digunakan untuk melihat hubungan antar mata kuliah yang satu dengan mata kuliah yang lain. Mata kuliah yang di ajarkan oleh dosen yang sama tidak boleh di jadwalkan pada saat bersamaan untuk menghindari terjadinya

bentrok. Jika bentrok, maka mata kuliah tersebut akan di jadwalkan lagi pada waktu yang lain.

Tabel 3.10 Data Dosen

NAMA DOSEN	MATA KULIAH
Ahmad Rahmatika, S.Pd., M.Pd	Aljabar, Vektor & Matrik
	Aljabar linier Elementer
Sa'adah Ritonga, S.Th.I., M.A	Islam & Iptek
	Ibadah & Muamalah
Sudarto, M.Kom	Enterprise Resources Planning **
	Kontrol & Audit sistem Informasi **
	Customer Relationship Management **
Tri Andre Anu, S.Kom., M.Kom	Pemograman Web Lanjut
	Prak. Pemograman Web Lanjut
Wilda Rina Hasibuan., S.Kom., M.Kom	Sistem Operasi
	Prak. Sistem Operasi
	Organisasi Manajemen
	Teknologi Open Source
	Prak. Teknologi Open Source
	Enterprise Resource Planning **
Yohanni Syahra., S.Si., M.Kom	Kecerdasan Buatan
	Manajemen Proyek TI
	Neurofuzzy **
	Pengantar Organisasi & Arsitektur TI
	Kecerdasan & Analisis Bisnis
Yoshida Sary., S.Kom.,M.Kom	Customer Relationship Management **
	Tata Kelola Sistem Informasi
	Kewirausahaan IT

Zuli Agustina Gultom, M.Si	Matematika Diskrit
	Statistika & Probabilitas
	Prak. Statistika & Probabilitas
	Pemodelan Data Bisnis
Muhammad Iqbal Pradipta S.Si, M.Kom	Enterprise Application Integration **
	Pemograman Desktop
	Prak. Pemograman Desktop
	Web Semantik
	Prak. Web Semantik
Muhammad Haris., S.Kom., M.Kom	Teknologi Open Source
	Teknologi Antar Jaringan
	Prak. Teknologi Antar Jaringan
	Teknologi Open Source
	Sistem Operasi
	Prak. Sistem Operasi
	Enterprise Application Integration **
Mulkan Azhari, S.Kom., M.Kom	Customer Relationship Management **
	Organisasi & Manajemen
	Sistem Operasi
	Prak. Sistem Operasi
	Manajemen Proyek TI
	Teknologi Big Data
	Interaksi Manusia & Komputer
Nadlrah Naimi, S.Ag., M.A	Islam & Iptek
	Ibadah & Muamalah
Nugra Autary., S.Kom., M.Kom	Desain Pengalaman Pengguna (UI UX Exp)
	Prak. Desain Pengalaman Pengguna (UI UX Exp)
	Pemograman Integrative
	Prak. Pemograman Integrative

Okvi Nugroho., S.Kom., M.Kom	Manajemen & Desain Jaringan Komputer
	Prak. Manajemen & Desain Jaringan Komputer
	Teknologi Open Source
	Prak. Teknologi Open Source
	Pengolahan Citra Digital
	Prak. Pengolahan Citra Digital
Putri Annisa, S.Kom., M.Kom	Sistem Multimedia
	Prak. Sistem Multimedia
Rajarif Syah Akbar S, S.H., M.H	Kewarganegaraan
Rama Prameswara Ritonga, S.T., M.Kom	Sistem Operasi
	Prak. Sistem Operasi
	Pemograman Mobile
	Prak. Pemograman Mobile
	Desain Pengalaman Pengguna (UI UX Exp)
	Prak. Desain Pengalaman Pengguna (UI UX Exp)
Rizaldy Khair., S.Kom., M.Kom	Struktur Data
	Prak. Struktur Data
	Kontrol & Audit Sistem Informasi **
	Organisasi Manajemen
	Big Data Analitik **
	Keamanan Informasi
Halim Maulana., S.T., M.Kom	Metodelogi Penelitian
	Manajemen & Desain Jaringan Komputer
	Prak. Manajemen & Desain Jaringan Komputer
	Struktur Data
	Prak. Struktur Data
	Teknologi Antar Jaringan
	Prak. Teknologi Antar Jaringan

Hevlie Winda Nazry S, S.Pd., M.Si	Matematika Diskrit
	Statistika & Probabilitas
	Prak. Statistika & Probabilitas
Ina Zainah, M.A	Islam & Iptek
Indah Purnama Sari, S.T., M.Kom	Metodelogi Penelitian
	Penulisan Proposal
	Struktur Data
	Prak. Struktur Data
	Kecerdasan Buatan
Isman Efendi Limbong, S.PdI., M.PdI	Ibadah & Muamalah
Lailan Sofinah Harahap, M.Kom	Enterprise Development Software
	Big Data Analitik **
	Visualisasi Data & Informasi
	Prak. Visualisasi Data & Informasi
Mahardika Abdi Prawira., S.Kom., M.Kom	Enterprise Application Integration **
	Keamanan Informasi
	Web Semantik
	Prak. Web Semantik
	Enterprise Development Software
	Web Programming
	Prak. Web Programming
	AI Fundamental
Martiano., S.Kom., M.Kom	Etika Profesi IT
	Kontrol & Audit Sistem Informasi **
	Metodelogi Penelitian
	Tata Kelola Sistem Informasi
	Sistem Sensor & Aplikasi **
	Penulisan Proposal

Mhd Basri., S.Si., M.Kom	Sistem Mikroprosesor & Assembly
	Prak. Sistem Mikroprosesor & Assembly
	Penulisan Proposal
	Sistem Sensor & Aplikasi **
Miftah Fariz, M.A	Ibadah & Muamalah
	Islam & Iptek
Dr. Mandra Saragih, S.Pd., M.Hum	Bahasa Inggris
Dr. Lutfi Basit., M.I.Kom	Kewarganegaraan
Dr. Marah Dolly Nasution., M.Si	Aljabar, Vektor & Matrik
Dr. Rakhmad Wahyudin Sagala, M.Hum	B. Inggris
	Bahasa Inggris (Communicative English)
Dr. Robbie Fanreza., M.Pd	Ibadah & Muamalah
	Islam & Iptek
Dr. Zainal Azis., M.Si	Aljabar, Vektor & Matrik
Eka Rahayu., S.Kom., M.Kom	Pemrograman Web Dasar
	Prak. Pemrograman Web Dasar
	Neurofuzzy **
Elvin Syahrin, S.Kom., M.Kom	Customer Relationship Management **
	Pengantar Organisasi & Arsitektur TI
Erwin Asmadi, S.H., M.H	Kewarganegaraan
Faisal Amri, M.A	Islam & Iptek
Farid Akbar Siregar., S.Kom., M.Kom	Struktur Data
	Prak. Struktur Data
	Pengolahan Citra Digital
	Prak. Pengolahan Citra Digital

Fatma Sari Hutagalung., S.Kom., M.Kom	Metodelogi Penelitian
	Kewirausahaan IT
	Manajemen Proyek TI
	Penulisan Proposal
	Sistem Multimedia
	Prak. Sistem Multimedia
Ferdy Riza., S.T., M.Kom	Kriptografi & Sekuriti Sistem
	Keamanan Informasi
Hafiz Irwandi, S.Kom., M.Kom	Pemograman Web Dasar
	Prak. Pemograman Web Dasar
Andy Satria, S.Kom., M.Kom	Pemograman Desktop
	Prak. Pemograman Desktop
	Pemrograman Mobile
	Prak. Pemograman Mobile
	Sistem Sensor & Aplikasi **
Andi Zulherry., S.Kom., M.Kom	Sistem Operasi
	Prak. Sistem Operasi
	Teknologi Open Source
	Prak. Teknologi Open Source
	Manajemen & Desain Jar Komputer
	Prak. Manajemen & Desain Jar Komputer
	Teknologi Antar Jaringan
	Prak. Teknologi Antar Jaringan
Angga Fahmi., S.Pd., M.Pd	Ibadah & Muamalah
	Islam & Iptek
Arni Susanti, S.Si., M.Sc	Matematika Diskrit
	Statistika & Probabilitas
	Prak. Statistika & Probabilitas
	Statistika Regresi
	Teori Peluang

Asrar Aspia Manurung, S.Pd., M.Pd	Riset Operasi
	Statistika Komputasi
Asrul Siregar, M.PdI	Islam & Iptek
Boni Oktaviana Sembiring, S.Kom., M.Kom	Neurofuzzy **
	Sistem Multimedia
	Prak. Sistem Multimedia
	Teknologi Open Source
	Prak. Teknologi Open Source
Dedi Irawan., S.Kom., M.Kom	Desain Pengalaman Pengguna (UI UX Exp)
	Prak. Desain Pengalaman Pengguna
	Big Data Analitik **
Dr. Al-Khowarizmi., M.Kom	Metodelogi Penelitian
	Kecerdasan Buatan
Dr. Dewi Kusuma Nst, S.Si., M.Hum	Bahasa Inggris
Dr. Firahmi Rizky, M.Kom	Kecerdasan Buatan
	Neurofuzzy **
Dr. Heru Mardiansyah, S.Si., M.Kom	Sistem Sensor & Aplikasi **
Dr. Irfan Dahnial, S.Pd., M.Pd	Kewarganegaraan
Dr. Irvan., M.Si	Aljabar, Vektor & Matrik
Amrullah, S.Kom., M.Kom	Enterprise Application Integration **
	Enterprise Resources Planning **
	Kontrol & Audit Sistem Informasi **
	Enterprise Development Software
	Pemograman Integrative
	Prak. Pemograman Integrative
	Digital Marketing Strategy

3.2.3 Data Jam Perkuliahan dan Data Ruangan Kuliah

Data jam perkuliahan merujuk pada slot waktu kuliah harian yang digunakan di lingkungan Fakultas Ilmu Komputer dan Teknologi Informasi. Jam perkuliahan ini biasanya dimulai dari pagi hingga sore hari, dan dibagi ke dalam beberapa sesi tetap. Informasi ini digunakan untuk membatasi ruang solusi dalam proses penjadwalan, serta memastikan setiap mata kuliah memiliki alokasi waktu yang sesuai. Data ruangan kuliah merupakan nama-nama ruangan yang tersedia untuk menyelenggarakan kegiatan perkuliahan di Fakultas Ilmu Komputer dan Teknologi Informasi.

Data akan berupa jam perkuliahan, ruangan yang digunakan dan lokasi ruangan. Data dapat di lihat pada Tabel 3.11 dan Tabel 3.12

Table 3.11 Data Jam Perkuliahan

HARI	JADWAL
Senin - Sabtu	07.30 - 08.20
	08.20 - 09.10
	09.10 - 10.00
	10.00 - 10.50
	10.50 - 11.40
	11.40 - 12.30
	12.30 - 13.20
	13.30 - 14.20
	14.20 - 15.10
	15.10 - 16.00
	16.00 - 16.50
	16.50 - 17.40
	17.40 - 18.30
	18.30 - 19.20

Tabel 3.12 Data Ruangan Kuliah

Ruang Kuliah	Lokasi Ruangan
601	Lantai 6 Gedung FIKTI
602	Lantai 6 Gedung FIKTI
603	Lantai 6 Gedung FIKTI
604	Lantai 6 Gedung FIKTI
605	Lantai 6 Gedung FIKTI
606	Lantai 6 Gedung FIKTI
607	Lantai 6 Gedung FIKTI
608	Lantai 6 Gedung FIKTI
701	Lantai 7 Gedung FIKTI
702	Lantai 7 Gedung FIKTI
703	Lantai 7 Gedung FIKTI
704	Lantai 7 Gedung FIKTI
705	Lantai 7 Gedung FIKTI
706	Lantai 7 Gedung FIKTI
707	Lantai 7 Gedung FIKTI
708	Lantai 7 Gedung FIKTI
Lab 605	Lantai 6 Gedung FIKTI
Lab 606	Lantai 6 Gedung FIKTI
Lab 607	Lantai 6 Gedung FIKTI
Lab 608	Lantai 6 Gedung FIKTI
F303	Lantai 3 Gedung PAI
F304	Lantai 3 Gedung PAI
F305	Lantai 3 Gedung PAI
F306	Lantai 3 Gedung PAI
F307	Lantai 3 Gedung PAI
F308	Lantai 3 Gedung PAI
F309	Lantai 3 Gedung PAI
F310	Lantai 3 Gedung PAI
F401	Lantai 4 Gedung PAI

3.3 Perancangan Sistem

Perancangan sistem ini mengacu pada pembangunan sistem informasi penjadwalan kegiatan belajar mahasiswa berbasis *website* dan *AI*. Sistem dirancang agar mampu mengelola jadwal kuliah secara otomatis, mendeteksi konflik, dan menyajikan hasil kepada pengguna melalui antarmuka *web* maupun *chatbot*.

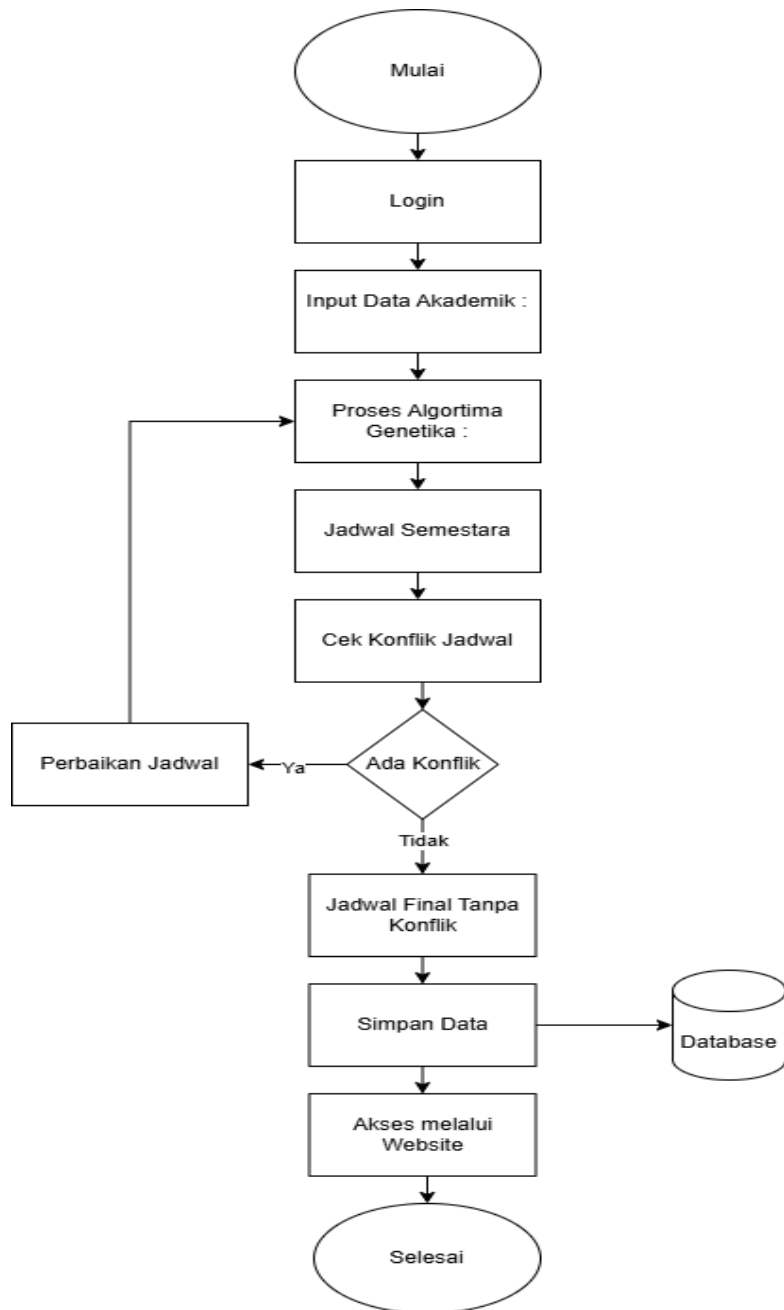
Sistem ini dibagi dalam beberapa komponen utama, yaitu:

1. Modul Input Akademik
2. Modul Optimasi Jadwal (Algoritma Genetika)
3. Modul Validasi
4. Modul Manajemen Database
5. Antarmuka Website
6. Antarmuka Chatbot

3.3.1 Alur Kerja sistem

Sistem nantinya akan memiliki alur kerja sebagai berikut:

1. Admin menginput data seperti dosen, mata kuliah, ruangan, dan slot waktu.
2. Sistem memproses data melalui algoritma genetika, menghasilkan populasi awal solusi jadwal.
3. Proses seleksi, *crossover*, dan mutasi dilakukan untuk menghasilkan jadwal terbaik.
4. Jadwal sementara akan diperiksa apakah memiliki konflik.
5. Jika konflik ditemukan, sistem melakukan perbaikan dengan mengulang proses *genetika*.
6. Setelah jadwal final bebas tanpa konflik dihasilkan, sistem akan menyimpannya ke dalam *database*.
7. Dosen dan biro dapat mengakses jadwal melalui *webchat*



Gambar 3.2 Alur Kerja Sistem

3.3.2 Penjelasan Alur Kerja Sistem

1. Input Akademik

Admin akan memasukkan data seperti :

- Nama mata kuliah & jumlah SKS
- Nama dosen & mata kuliah yang diajarkan

- Slot waktu yang tersedia
- Daftar ruang kuliah

2. Proses Algoritma Genetika

- *Representasi kromosom*: satu jadwal lengkap
- *Fitness function*: menghitung jumlah konflik
- Seleksi: Tournament / *Roulette Wheel*
- *Crossover*: One-point / *PMX*
- Mutasi: random swap posisi jadwal
- Berhenti saat konflik = 0 atau generasi max tercapai

3. Validasi Konflik

- Validasi: tidak ada dua kuliah bersamaan di ruangan sama
- Validasi dosen tidak mengajar dua kelas bersamaan
- Validasi waktu tidak tumpang tindih per mahasiswa

4. Antarmuka Website

Menampilkan jadwal akhir yang sudah optimal dan dapat diakses oleh:

- Dosen (mata kuliah yang diajar)
- Admin (kontrol dan monitoring)

5. Modul Chatbot

Dibangun menggunakan teknologi *NLP* berbasis *Python* (*Rasa/AIML*), yang mampu:

- Menjawab pertanyaan “Ubah Jadwal Kuliah?” atau Lokasi Ruangan Yang Kosong?”
- Mengambil data langsung dari database jadwal
- Tampilan interaktif dan mendukung webchat

3.4 Waktu dan Tempat Penelitian

3.4.1 Waktu Penelitian

Tabel 3.13 Waktu Penelitian

Kegiatan	Bulan							
	Januari	Februari	Maret	April	Mei	Juni	Juli	Agustus
Pengajuan judul								
Observasi								
Pengumpulan Data								
Seminar Proposal								
Pembuatan Sistem								
Sidang								

3.4.2 Tempat Penelitian

Penelitian dilaksanakan di lingkungan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) Universitas Muhammadiyah Sumatera Utara (UMSU). Pemilihan lokasi ini didasarkan pada relevansi

langsung antara objek penelitian dan sistem akademik yang diterapkan di fakultas tersebut, khususnya pada Program Studi Teknologi Informasi jenjang Strata Satu (S1) kelas reguler.

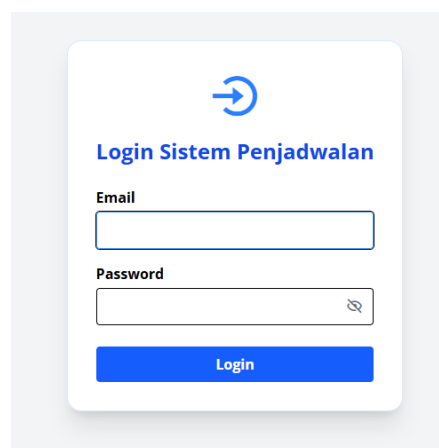
BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Implementasi Sisten

Pada tahap ini, sistem penjadwalan perkuliahan otomatis berbasis algoritma genetika yang telah di rancang dan dibangun diimplementasikan ke dalam bentuk website interaktif. Implementasi ini bertujuan untuk menguji sejauh mana sistem mampu menjalankan fungsi utamanya, yaitu menghasilkan jadwal perkuliahan yang optimal dan bebas konflik, serta memberikan kemudahan akses informasi melalui integrasi chatbot. Hasil implementasi ini mencakup antarmuka pengguna (*User Interface*), fungsi utama sistem, serta interaksi antara pengguna dengan sistem melalui fitur-fitur yang tersedia.

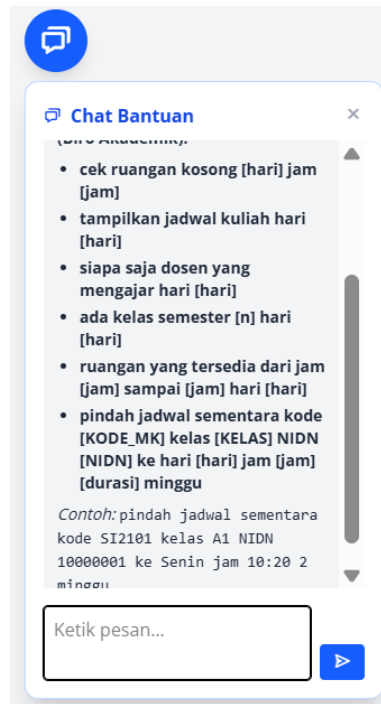
4.1.1 Fitur Utama



Gambar 4.1 Halaman Login

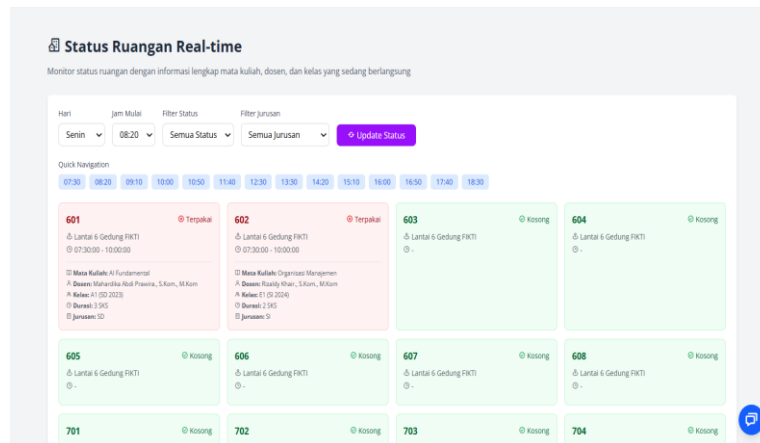
Merupakan pintu masuk utama bagi pengguna sistem penjadwalan, baik dari pihak biro akademik dan juga dosen. Pengguna diminta untuk memasukkan alamat email dan kata sandi yang telah terdaftar untuk dapat mengakses fitur-fitur di dalam sistem. Proses

autentikasi ini dirancang untuk menjaga keamanan data dan memastikan bahwa hanya pengguna yang berwenang yang dapat mengakses informasi penjadwalan.



Gambar 4.2 Chat Bot Akademik

Fitur *chatbot* merupakan salah satu inovasi utama dalam sistem ini. *Chatbot* ini berfungsi sebagai asisten virtual yang siap membantu biro akademik dan juga dosen dalam mengakses informasi secara cepat dan praktis. Melalui perintah teks, pengguna dapat memperoleh data terkait ketersediaan ruang kelas, jadwal dosen, jadwal kuliah harian, hingga informasi kelas berdasarkan semester. Fitur ini mendukung berbagai format perintah yang telah dirancang agar mudah dipahami. *Chatbot* ini juga dirancang untuk aktif sepanjang waktu dan dapat mempercepat proses pengambilan keputusan tanpa harus membuka banyak menu secara manual.

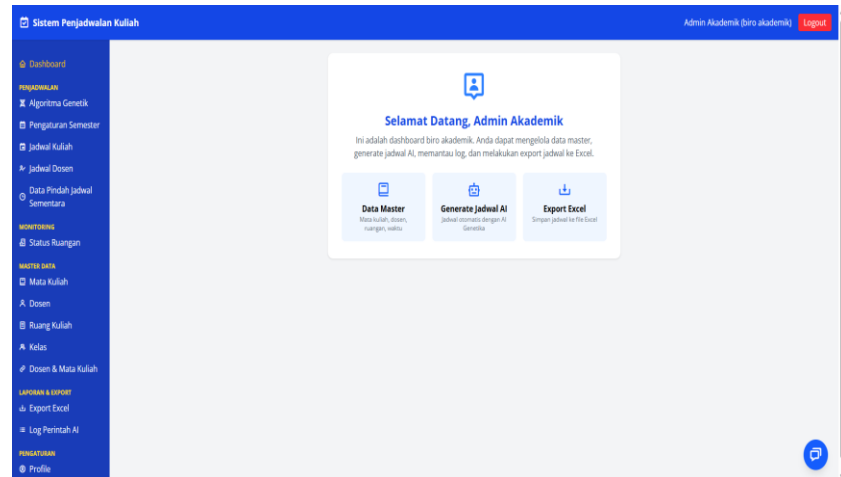


Gambar 4.3 Monitoring Status Ruangan Real Time

Fitur ini memungkinkan biro akademik maupun dosen untuk memantau status ketersediaan ruangan secara langsung berdasarkan waktu dan hari yang dipilih. Informasi yang disajikan mencakup status ruang (kosong atau terpakai), mata kuliah yang sedang berlangsung, nama dosen pengampu, kelas, jurusan, serta durasi dan lokasi kegiatan. Tampilan dibuat dalam format visual yang mudah dipahami, dengan warna berbeda antara ruang kosong dan terpakai, serta dilengkapi navigasi waktu untuk mempermudah pencarian. Fitur ini sangat berguna untuk pengambilan keputusan cepat terkait alokasi ruang secara efisien dan menghindari bentrokan jadwal secara *real-time*.

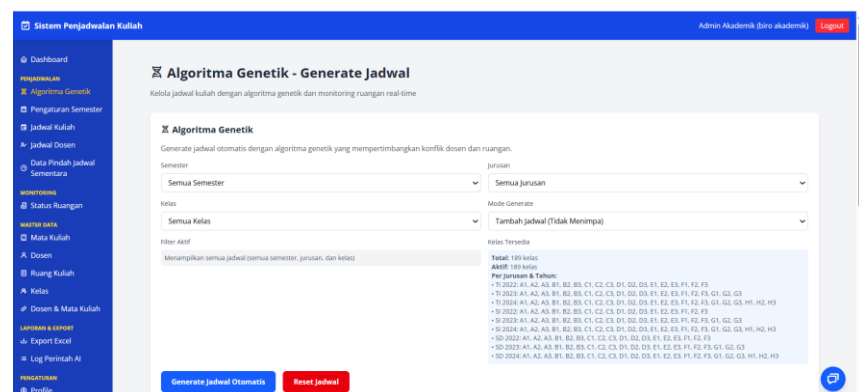
4.1.2 Fitur Biro Akademik

Fitur-fitur yang tersedia untuk biro akademik bertujuan untuk mempermudah proses manajemen jadwal dan administrasi akademik. Berikut adalah penjelasan tiap fitur disertai tampilan antarmukanya:



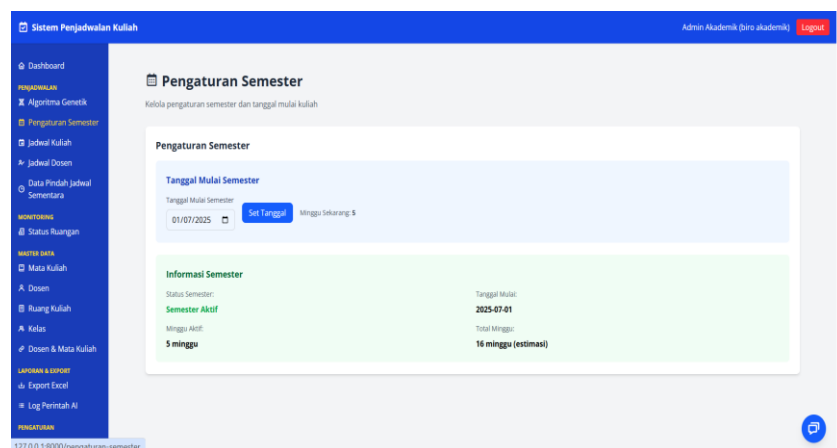
Gambar 4.4 Dashboard Biro Akademik

Tampilan utama (*dashboard*) untuk pengguna dari pihak biro akademik. Melalui halaman ini, admin akademik dapat mengakses seluruh fitur utama sistem seperti pengelolaan data master (mata kuliah, dosen, ruang, waktu), pembuatan jadwal otomatis menggunakan *algoritma genetika*, serta ekspor hasil penjadwalan ke dalam format Excel. Navigasi menu yang terstruktur di sisi kiri juga mempermudah biro akademik dalam mengakses berbagai fungsi sistem mulai dari penjadwalan, monitoring ruang, hingga pengaturan data dosen dan mata kuliah.



Gambar 4.5 Generate Jadwal (Algoritma Genetika)

Halaman ini merupakan inti dari sistem penjadwalan otomatis yang menggunakan *algoritma genetika*. Admin biro akademik dapat melakukan proses generate jadwal kuliah berdasarkan parameter seperti semester, jurusan, dan kelas. Sistem akan memperhitungkan seluruh variabel seperti ketersediaan ruang dan dosen untuk menghasilkan jadwal yang bebas konflik. Terdapat dua tombol utama, yaitu *Generate Jadwal Otomatis* untuk memulai proses optimasi dan *Reset Jadwal* untuk menghapus jadwal yang telah dibuat. Informasi detail mengenai jumlah kelas yang aktif juga ditampilkan untuk membantu admin dalam mengelola cakupan jadwal yang akan di *generate*.



Gambar 4.6 Pengaturan Semester

Fitur Pengaturan Semester pada sistem penjadwalan ini memungkinkan biro akademik untuk mengelola tanggal dimulainya kegiatan perkuliahan pada awal semester. Penetapan tanggal mulai ini menjadi dasar bagi sistem dalam menentukan minggu aktif dan menghitung durasi perkuliahan hingga akhir semester. Setelah tanggal mulai ditetapkan, sistem secara otomatis akan menampilkan informasi

minggu berjalan dan memperkirakan total minggu perkuliahan, seperti estimasi 16 minggu dalam satu semester.

Hari	Jam	Mata Kuliah	Dosen	Ruang	Kelas	Aksi
Senin	07:30:00 - 10:00:00	Kontrol & Audit sistem informasi	Amrullah, S.Kom., M.Kom	706	F1 01 2023	[Edit] [Delete] [Add]
Senin	07:30:00 - 08:20:00	Prak. Teknologi Open Source	Andi Zuherry, S.Kom., M.Kom	F305	F1 01 2023	[Edit] [Delete] [Add]
Senin	07:30:00 - 10:00:00	Organisasi Manajemen	Rizaldi Khair., S.Kom., M.Kom	602	F1 01 2023	[Edit] [Delete] [Add]
Senin	07:30:00 - 10:00:00	AI Fundamental	Mahardika Abdi Pratiwi, S.Kom., M.Kom	601	A1 02 2023	[Edit] [Delete] [Add]
Senin	08:20:00 - 09:10:00	Prak. Teknologi Open Source	Andi Zuherry, S.Kom., M.Kom	F306	F2 01 2023	[Edit] [Delete] [Add]
Senin	09:10:00 - 10:00:00	Prak. Teknologi Open Source	Andi Zuherry, S.Kom., M.Kom	F307	F3 01 2023	[Edit] [Delete] [Add]
Senin	10:00:00 - 12:30:00	Kontrol & Audit sistem informasi	Amrullah, S.Kom., M.Kom	707	F2 01 2023	[Edit] [Delete] [Add]
Senin	10:00:00 - 10:50:00	Prak. Teknologi Open Source	Andi Zuherry, S.Kom., M.Kom	F308	F1 01 2023	[Edit] [Delete] [Add]
Senin	10:00:00 - 11:40:00	Organisasi Manajemen	Rizaldi Khair., S.Kom., M.Kom	603	F2 01 2023	[Edit] [Delete] [Add]
Senin	10:00:00 - 17:30:00	AI Fundamental	Mahardika Abdi Pratiwi, S.Kom., M.Kom	602	A2 02 2023	[Edit] [Delete] [Add]

Gambar 4.7 Jadwal Kuliah

Sistem ini berfungsi untuk menampilkan seluruh daftar jadwal perkuliahan yang telah tersusun, lengkap dengan informasi hari, jam, mata kuliah, dosen pengampu, ruang kelas, serta kelas atau angkatan yang terkait. Tampilan jadwal tersaji dalam bentuk tabel yang mudah dibaca, sehingga biro akademik dapat dengan cepat melakukan pengecekan, penyuntingan, atau penghapusan jadwal jika diperlukan.

Tambah Jadwal Kuliah

Mata Kuliah
- Pilih Mata Kuliah -

Dosen
- Pilih Dosen -

Ruang Kuliah
- Pilih Ruang -

Hari
- Pilih Hari -

Jam Mulai
- Pilih Jam Mulai -

SKS
- Pilih SKS -

Jam Selesai (Otomatis)
Dihitung otomatis berdasarkan jam mulai dan SKS

Kelas
- Pilih Kelas -

Simpan **Batal**

Gambar 4.8 Form Tambah Jadwal Kuliah

Berfungsi sebagai alat input manual untuk menambahkan data jadwal kuliah secara langsung oleh biro akademik. Form ini memungkinkan admin memilih komponen-komponen penting dalam penjadwalan seperti mata kuliah, dosen pengampu, ruang kuliah, hari dan jam pelaksanaan, jumlah SKS, serta kelas yang akan mengikuti perkuliahan tersebut.

Edit Jadwal Kuliah

Mata Kuliah
SI2205 - Kontrol & Audit sistem Informasi

Dosen
Amrullah, S.Kom., M.Kom

Ruang Kuliah
706 - 706 (Lantai 7 Gedung FIKTI)

Hari
Senin

Jam Mulai
- Pilih Jam Mulai -

SKS
3 SKS (150 menit)

Jam Selesai (Otomatis)
10:00:00
Jam selesai dihitung otomatis berdasarkan SKS (1 SKS = 50 menit)

Kelas
F1 - SI 2022 (34 mahasiswa)

Simpan Batal

Gambar 4.9 Form Edit Jadwal Kuliah

Sistem Penjadwalan Kuliah Admin Akademik (biro akademik) Logout

Jadwal Dosen dan Matakuliah
Lihat jadwal mengajar setiap dosen dan mata kuliah yang diajarkan

Filter dan Pengurutan

Hari: Semua Hari | Jurusan: Semua Jurusan | Semester: Semua Semester | Urutan Berdasarkan: Nama Dosen | Urutan: A-Z / Awal-Akhir | Terapkan

Jadwal Mengajar Dosen 50 dosen

Ahmad Rahmatika, S.Pd., M.Pd. (10000001)

Mata Kuliah	Kode	Hari	Jam	Ruang	Kelas	SKS
Aljabar linier Elementer	SD2401	Rabu	07:30:00 - 08:20:00	601	A1 (SD 2024)	2
Aljabar linier Elementer	SD2401	Rabu	17:40:00 - 19:20:00	605	B2 (SD 2024)	2
Aljabar, Vektor & Matrik	TIG401	Kamis	17:40:00 - 20:10:00	601	B2 (TI 2024)	3
AI Fundamental	SD2308	Senin	16:00:00 - 18:30:00	702	D3 (SD 2023)	3
Aljabar linier Elementer	SD2401	Rabu	08:10:00 - 10:50:00	F401	A2 (SD 2024)	2

Gambar 4.10 Jadwal Dosen dan Mata Kuliah

Menyajikan informasi mengenai jadwal mengajar masing-masing dosen beserta mata kuliah yang mereka ampu. Fitur ini membantu biro akademik dalam melakukan monitoring terhadap beban kerja dosen dan distribusi pengajaran.

Pengguna dapat melakukan pencarian dan penyaringan berdasarkan hari, jurusan, semester, nama dosen, dan urutan tampilannya. Setelah filter diterapkan, sistem akan menampilkan daftar lengkap dosen beserta detail jadwal mengajarnya, termasuk nama mata kuliah, kode, hari dan jam, ruang kelas, kelas yang diajar, serta jumlah SKS.

Mata Kuliah	Dosen	Kelas	Jadwal Lama	Jadwal Baru	Ruang Saat Ini	Durasi	Status	Alasan	Dibuat Oleh	Tanggal Dibuat	Aksi
SD2401 - Aljabar linear Elementer	Alimad Rahmatika, S.Pd, M.Pd NIDN: 10000001	Kelas 2024	Rabu 07:30:00	Selasa 08:20	Ruang 1 Gedung 101	100 menit	Dibatalkan	Pindah jadwal sementara	Admin Akademik	26/07/2023 02:07	Sudah dibatalkan

Gambar 4.11 Riwayat Pindah Jadwal

Halaman ini menyajikan informasi terkait riwayat pemindahan jadwal kuliah sementara yang dilakukan oleh admin akademik maupun dosen. Fitur ini dirancang untuk membantu biro akademik dalam memantau dan mengelola perubahan jadwal yang bersifat temporer, seperti penyesuaian waktu kuliah akibat kebutuhan dosen, keterbatasan ruang, atau alasan teknis lainnya.

Sistem Penjadwalan Kuliah Admin Akademik (Biro akademik) [Logout](#)

Daftar Mata Kuliah [+ Tambah Mata Kuliah](#)

Cari:

Kode	Nama	SKS	Semester	Jurusan	Aksi
SI2101	Etika Profesi IT	2	2021	SI	Edit Hapus
TI2201	Big Data Analitik	3	2022	TI	Edit Hapus
SI2202	Customer Relationship Management	3	2022	SI	Edit Hapus
SI2203	Desain Pengalaman Pengguna (UX Exp)	2	2022	SI	Edit Hapus
SI2201	Enterprise Application Integration	3	2022	SI	Edit Hapus
TI2202	Enterprise Development Software	2	2022	TI	Edit Hapus
SI2204	Enterprise Resource Planning	3	2022	SI	Edit Hapus
SI2205	Kontrol & Audit sistem Informasi	3	2022	SI	Edit Hapus
TI2203	Metodologi Penelitian	2	2022	TI	Edit Hapus
SI2206	Metodologi Penelitian	3	2022	SI	Edit Hapus
TI2204	Neurofuzzy	3	2022	TI	Edit Hapus
TI2104	Bumihwarman Informatika	3	2021	TI	Edit Hapus

Gambar 4.12 Daftar Mata Kuliah

Halaman ini menampilkan daftar lengkap seluruh mata kuliah yang tersedia di lingkungan Fakultas Ilmu Komputer dan Teknologi Informasi. Fitur ini memungkinkan biro akademik untuk mengelola data mata kuliah dengan mudah dan efisien, termasuk menambah, mengedit, serta menghapus entri sesuai kebutuhan.

Pengguna juga dapat melakukan pencarian berdasarkan nama atau kode mata kuliah, serta menyaring tampilan berdasarkan jurusan dan tahun akademik.

Tambah Mata Kuliah

Kode MK

Nama

SKS

Semester (Tahun)

Jurusan

Gambar 4.13 Form Tambah Mata Kuliah

Form ini digunakan untuk menambahkan data mata kuliah baru ke dalam sistem. Pengguna diminta mengisi beberapa informasi penting yang meliputi kode mata kuliah, nama mata kuliah, jumlah SKS, semester pengambilan (berbasis tahun masuk), serta jurusan pengampu. Semua kolom bersifat wajib diisi agar data dapat tersimpan dengan valid.



Edit Mata Kuliah

Kode MK
SI2101

Nama
Etika Profesi IT

SKS
2

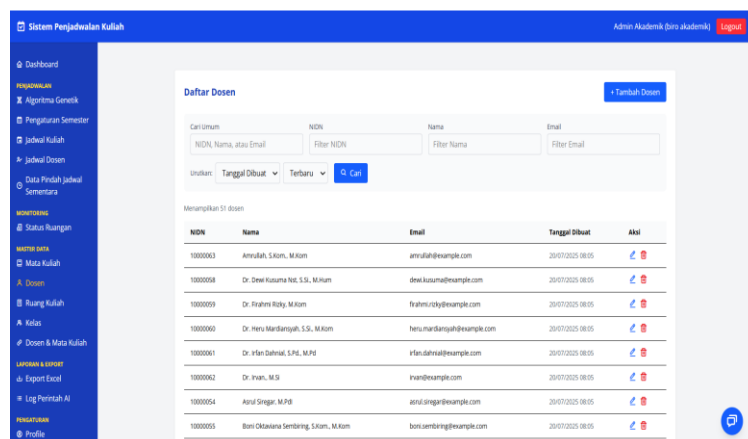
Semester (Tahun)
2021

Jurusan
SI

Simpan **Batal**

Gambar 4.14 Edit Mata Kuliah

Halaman ini digunakan untuk memperbarui data mata kuliah yang telah terdaftar dalam sistem. Informasi yang dapat diedit meliputi kode mata kuliah, nama mata kuliah, jumlah SKS, semester pengambilan (berdasarkan tahun), serta jurusan pengampu.



Daftar Dosen [+ Tambah Dosen](#)

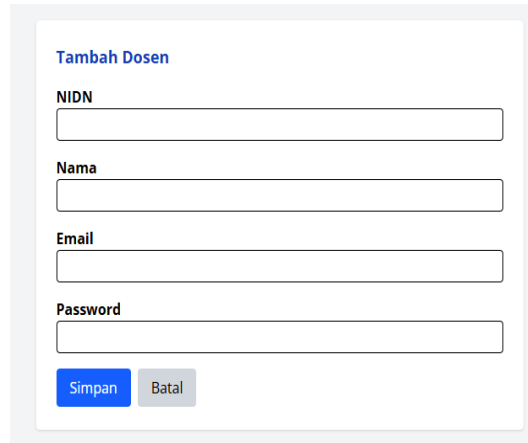
Carian: NIDN, Nama, atau Email | Filter NIDN | Filter Nama | Filter Email

Urutkan: Tanggal Dibuat | Terbaru | Cari

NIDN	Nama	Email	Tanggal Dibuat	Aksi
10000003	Amrullah, S.Kom, M.Kom	amrullah@example.com	201072025 08:05	✎ ✖
10000008	Dr. Dewi Kusuma Nst, S.Si, M.Kom	dewi.kusuma@example.com	201072025 08:05	✎ ✖
10000009	Dr. Ibrahim Rizky, M.Kom	ibrahimrizky@example.com	201072025 08:05	✎ ✖
10000060	Dr. Heru Mardiansyah, S.Si, M.Kom	herumardiansyah@example.com	201072025 08:05	✎ ✖
10000061	Dr. Irfan Dahwal, S.Pd, M.Pd	irfan.dahwal@example.com	201072025 08:06	✎ ✖
10000062	Dr. Irfan, M.Si	irfan@example.com	201072025 08:05	✎ ✖
10000064	Akmal Singar, M.Pd	akmal.singar@example.com	201072025 08:05	✎ ✖
10000065	Boni Oktavia Sembering, S.Kom, M.Kom	boni.sembering@example.com	201072025 08:05	✎ ✖

Gambar 4.15 Daftar Dosen

Halaman ini digunakan untuk menampilkan dan mengelola data dosen yang terdaftar dalam sistem penjadwalan kuliah. Fitur ini membantu biro akademik dalam memastikan bahwa seluruh informasi dosen, termasuk NIDN, nama lengkap, dan alamat email, terdokumentasi secara rapi dan dapat diakses dengan mudah.



Tambah Dosen

NIDN

Nama

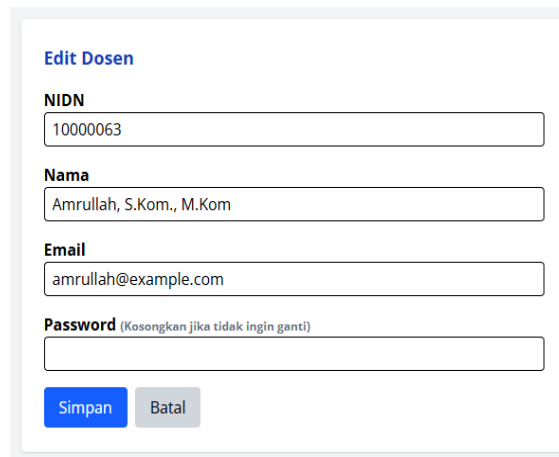
Email

Password

Simpan **Batal**

Gambar 4.16 Form Tambah Dosen

Form ini digunakan untuk menambahkan data dosen baru ke dalam sistem. Pengguna diminta untuk mengisi informasi dasar dosen yang meliputi NIDN, nama lengkap, alamat email, serta password untuk login ke dalam sistem.



Edit Dosen

NIDN

Nama

Email

Password (Kosongkan jika tidak ingin ganti)

Simpan **Batal**

Gambar 4.17 Form Edit Dosen

Halaman ini digunakan untuk mengubah atau memperbarui data dosen yang sudah terdaftar dalam sistem. Form ini memuat informasi penting seperti NIDN, nama lengkap, dan email dosen. Seluruh data yang sudah tersimpan sebelumnya akan ditampilkan secara otomatis dalam kolom isian untuk memudahkan proses pengeditan.

Kode Ruang	Nama Ruang	Lokasi	Aksi
601	601	Lantai 6 Gedung PAKTI	Edit Hapus
602	602	Lantai 6 Gedung PAKTI	Edit Hapus
603	603	Lantai 6 Gedung PAKTI	Edit Hapus
604	604	Lantai 6 Gedung PAKTI	Edit Hapus
605	605	Lantai 6 Gedung PAKTI	Edit Hapus
606	606	Lantai 6 Gedung PAKTI	Edit Hapus
607	607	Lantai 6 Gedung PAKTI	Edit Hapus
608	608	Lantai 6 Gedung PAKTI	Edit Hapus
701	701	Lantai 7 Gedung PAKTI	Edit Hapus
702	702	Lantai 7 Gedung PAKTI	Edit Hapus
703	703	Lantai 7 Gedung PAKTI	Edit Hapus
704	704	Lantai 7 Gedung PAKTI	Edit Hapus
705	705	Lantai 7 Gedung PAKTI	Edit Hapus

Gambar 4.18 Daftar Ruang Kuliah

Halaman ini menyajikan informasi lengkap mengenai seluruh ruang kuliah yang tersedia dalam sistem, termasuk kode ruang, nama ruang, dan lokasi fisiknya di dalam gedung.

Tambah Ruang Kuliah

Kode Ruang

Nama Ruang

Lokasi

Gambar 4.19 Form Tambah Ruang Kuliah

Form ini digunakan untuk menambahkan ruang kuliah baru ke dalam sistem. Pengguna diminta untuk mengisi informasi berupa kode ruangan, nama ruangan dan juga lokasi ruangan.

Edit Ruang Kuliah

Kode Ruang
601

Nama Ruang
601

Lokasi
Lantai 6 Gedung FIKTI

Simpan Batal

Gambar 4.20 Form Edit Ruang Kuliah

Halaman ini digunakan untuk mengubah atau memperbarui data ruang kuliah yang sudah terdaftar dalam sistem. Form ini memuat informasi seperti kode ruang, nama ruang, dan lokasi ruang.

Sistem Penjadwalan Kuliah Admin Akademik (Biro akademik) Logout

Manajemen Kelas
Kelola kelas untuk setiap jurusan dan tahun akademik

Tambah Kelas Buat Kelas Massal

Filter Kelas

Jurusan: Semua Jurusan Tahun: Semua Tahun Status: Semua Status

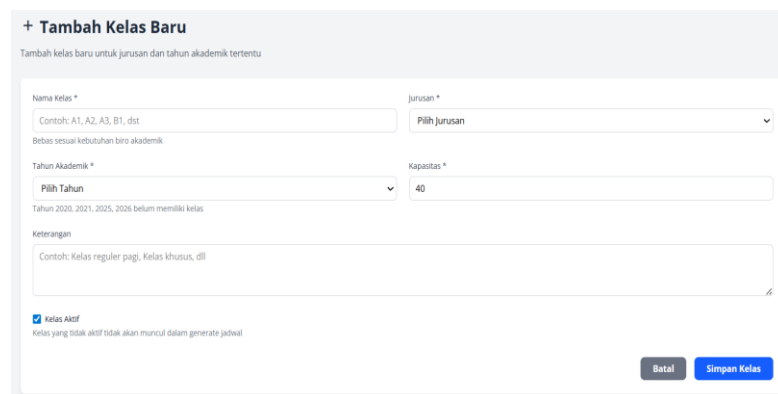
Daftar Kelas

KELAS	JURUSAN	TAHUN	KAPASITAS	STATUS	KETERANGAN	Aksi
A1	Sains Data	2024	33 mahasiswa	Aktif	Kelas A1 SD 2024	Edit Hapus
A2	Sains Data	2024	35 mahasiswa	Aktif	Kelas A2 SD 2024	Edit Hapus
A3	Sains Data	2024	31 mahasiswa	Aktif	Kelas A3 SD 2024	Edit Hapus
B1	Sains Data	2024	31 mahasiswa	Aktif	Kelas B1 SD 2024	Edit Hapus
B2	Sains Data	2024	36 mahasiswa	Aktif	Kelas B2 SD 2024	Edit Hapus

Gambar 4.21 Manajemen Kelas

Halaman ini menampilkan dan mengelola seluruh data kelas yang terdaftar dalam sistem penjadwalan kuliah secara terstruktur. Melalui halaman ini, biro akademik dapat memantau dan

memperbarui informasi penting terkait kelas, seperti nama kelas, jurusan, tahun akademik, kapasitas jumlah mahasiswa, status aktif atau nonaktif, serta keterangan tambahan yang diperlukan. Fitur yang tersedia memungkinkan pengguna untuk menambahkan kelas baru secara individu maupun massal, melakukan penyaringan data berdasarkan jurusan, tahun, atau status kelas, serta mengedit atau menghapus data kelas yang sudah ada



+ Tambah Kelas Baru
 Tambah kelas baru untuk jurusan dan tahun akademik tertentu

Nama Kelas *
 Contoh: A1, A2, A3, B1, dst
 Bebas sesuai kebutuhan biro akademik

Jurusan *
 Pilih Jurusan

Tahun Akademik *
 Pilih Tahun
 Tahun 2020, 2021, 2025, 2026 belum memiliki kelas

Kapasitas *
 40

Keterangan
 Contoh: Kelas reguler pagi, Kelas khusus, dll

☒ Kelas Aktif
 Kelas yang tidak aktif tidak akan muncul dalam generate jadwal

Batal Simpan Kelas

Gambar 4.22 Form Tambah Kelas

Halaman ini digunakan untuk menambahkan data kelas baru ke dalam sistem penjadwalan kuliah sesuai dengan jurusan dan tahun akademik tertentu. Melalui halaman ini, biro akademik dapat mengisi informasi penting terkait kelas seperti nama kelas, jurusan, tahun akademik, kapasitas jumlah mahasiswa, serta keterangan tambahan yang diperlukan, misalnya jenis kelas reguler atau kelas khusus. Terdapat juga opsi untuk menentukan status kelas aktif atau tidak aktif, di mana kelas yang tidak aktif tidak akan dimasukkan dalam proses pembuatan jadwal.

Buat Kelas Massal

Jurusan
Pilih Jurusan ▼

Tahun
Pilih Tahun ▼

Tahun yang tersedia: 2020, 2021, 2025, 2026

Kapasitas per Kelas
40

Pola Kelas
Pola Default (A1-A3, B1-B3, dst) ▼

Batal Buat Kelas

Gambar 4.23 Form Buat Kelas Massal

Halaman ini digunakan untuk membuat data kelas secara massal dalam sistem penjadwalan kuliah, sehingga biro akademik dapat menambahkan banyak kelas sekaligus dengan lebih cepat dan efisien. Pengguna dapat memilih jurusan dan tahun akademik yang diinginkan, menentukan kapasitas jumlah mahasiswa untuk setiap kelas, serta memilih pola penamaan kelas yang sesuai, seperti pola default A1-A3 atau B1-B3.

Sistem Penjadwalan Kuliah Admin Akademik (biro akademik) Logout

Dosen & Mata Kuliah

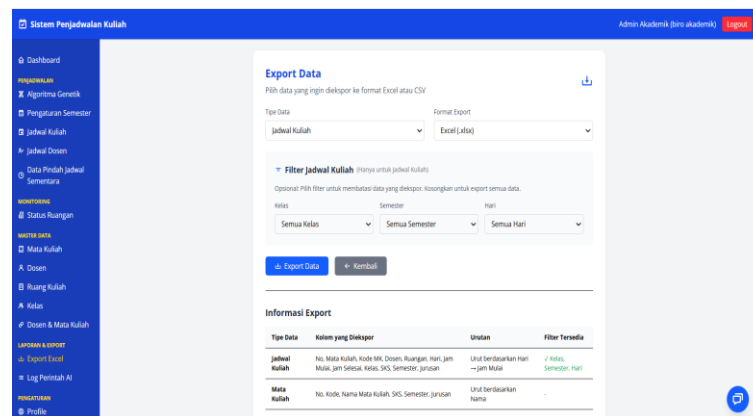
Cari dosen (nama/nidn) Cari mata kuliah Filter Reset

Dosen: Ahmad Rahmatika, S.Pd., M.Pd (10000001) Mata Kuliah: AI Fundamental (SD2368) Tambah Kelas

Nama Dosen	NIDN	Mata Kuliah Diampu
Ahmad Rahmatika, S.Pd., M.Pd	10000001	Aljabar Vektor & Matriks (T2447) hapus AI Fundamental (SD2368) hapus Aljabar liner Elementer (SD2401) hapus
Safah Ritzga, S.Th.I., M.A	10000002	Islam & sptk (T2301) hapus Badah & Muamalah (T2403) hapus
Sudarto, M.Kom	10000003	Customer Relationship Management (SD202) hapus Enterprise Resource Planning (SD204) hapus Kontrol & Audit sistem Informasi (SD205) hapus
Tri Andre Ama, S.Kom., M.Kom	10000004	Pengembangan Web Lanjut (SD208) hapus Prak. Pengembangan Web Lanjut (SD210) hapus
Mulikan Athari, S.Kom., M.Kom	10000017	Managemen Proyek TI (T2304) hapus Prak. Sistem Operasi (T2407) hapus Sistem Operasi (T2408) hapus Customer Relationship Management (SD202) hapus Teknologi Big Data (SD203) hapus Interaksi Manusia & komputer (SD2402) hapus
NoelHerah Naimi, S.Ar., M.A	10000018	Islam & sptk (T2301) hapus

Gambar 4.24 Dosen dan Mata Kuliah

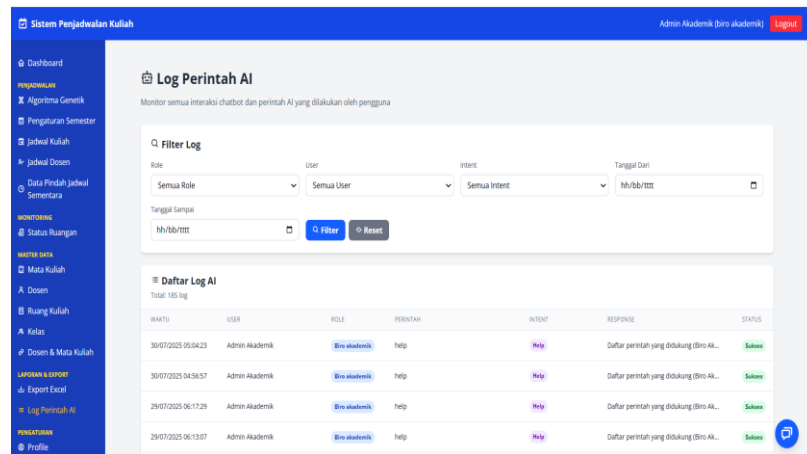
Halaman ini digunakan untuk mengelola relasi antara dosen dan mata kuliah yang mereka ampu dalam sistem penjadwalan kuliah. Melalui halaman ini, biro akademik dapat dengan mudah menambahkan, melihat, dan memperbarui data pengampu mata kuliah berdasarkan nama dosen atau NIDN. Setiap dosen yang terdaftar akan ditampilkan bersama daftar mata kuliah yang diampu, dengan opsi untuk menambahkan relasi baru ataupun menghapus relasi yang sudah ada. Fitur pencarian dan filter juga tersedia untuk mempermudah proses pencarian dosen atau mata kuliah tertentu



Gambar 4.25 Export Data

Halaman ini digunakan untuk melakukan proses ekspor data dari sistem penjadwalan kuliah ke dalam format Excel (.xlsx) atau CSV. Melalui halaman ini, biro akademik dapat memilih jenis data yang ingin diekspor seperti jadwal kuliah atau data lainnya, serta menentukan format ekspor yang sesuai dengan kebutuhan. Tersedia fitur filter tambahan yang memungkinkan pengguna membatasi data yang diekspor berdasarkan kelas, semester, dan hari tertentu, atau mengekspor seluruh data tanpa filter. Di bagian bawah halaman juga

terdapat informasi detail mengenai kolom yang akan disertakan dalam hasil ekspor, urutan data, serta filter yang tersedia untuk setiap jenis data.

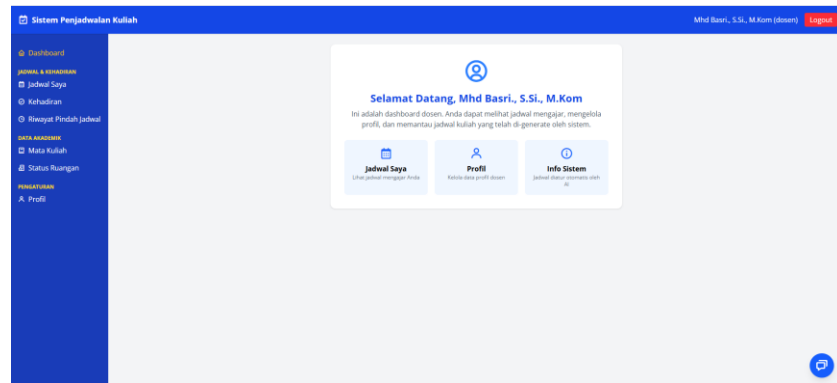


Gambar 2.26 Log Perintah AI

Halaman ini digunakan untuk memantau seluruh interaksi chatbot dan perintah *AI* yang dilakukan oleh pengguna dalam sistem penjadwalan kuliah. Melalui halaman ini, biro akademik dapat melihat daftar log perintah yang mencakup informasi lengkap seperti waktu interaksi, nama pengguna, peran pengguna, perintah yang diberikan, intent yang terdeteksi oleh *AI*, respons yang dihasilkan, serta status eksekusi perintah. Fitur filter yang tersedia memungkinkan pencarian log berdasarkan peran, pengguna, intent, serta rentang tanggal tertentu.

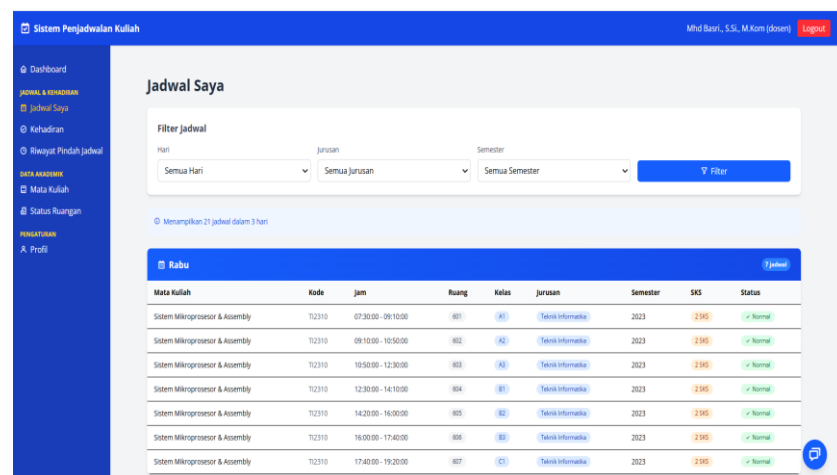
4.1.3 Fitur Dosen

Fitur untuk dosen merupakan bagian dari sistem yang dirancang khusus untuk mendukung kebutuhan dosen dalam mengakses dan memantau jadwal mengajar mereka secara mandiri. Berikut adalah penjelasan tiap fitur disertai tampilan antarmuka:



Gambar 4.27 Halaman Dashboard Dosen

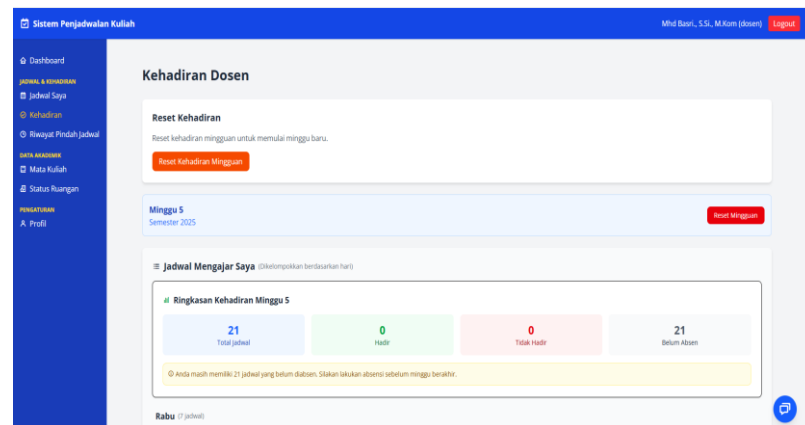
Halaman ini merupakan dashboard utama yang ditujukan untuk dosen dalam sistem penjadwalan kuliah. Pada halaman ini, dosen disambut dengan pesan selamat datang dan diberikan akses cepat ke tiga menu utama, yaitu “Jadwal Saya” untuk melihat jadwal mengajar yang sudah di-generate oleh sistem, “Profil” untuk mengelola data profil dosen, serta “Info Sistem” yang memberikan informasi terkait proses penjadwalan otomatis oleh AI.



Gambar 4.28 Jadwal Saya

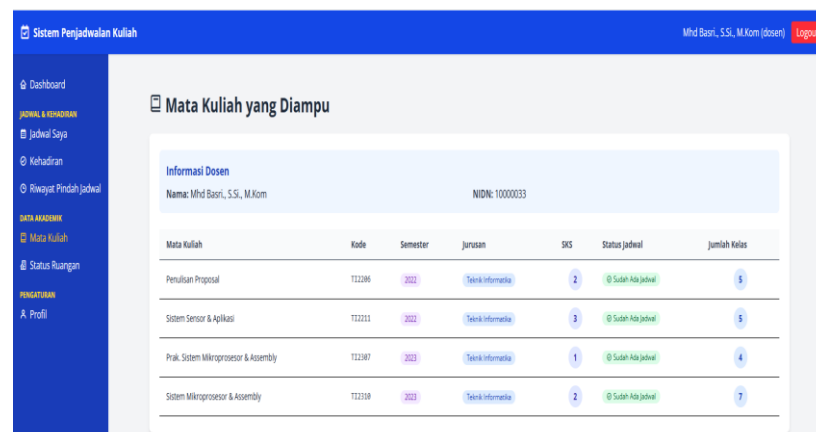
Halaman ini menyajikan ringkasan kehadiran mingguan yang mencakup informasi jumlah total jadwal yang dimiliki dosen, jumlah

kehadiran yang telah dicatat, jumlah ketidakhadiran, serta jadwal-jadwal yang belum diisi absensinya



Gambar 4.29 Kehadiran Dosen

Halaman ini digunakan oleh dosen untuk mencatat dan memantau kehadiran selama kegiatan perkuliahan. Terdapat ringkasan kehadiran mingguan yang menampilkan jumlah total jadwal, kehadiran, ketidakhadiran, serta jadwal yang belum diabsen. Dosen dapat melihat rincian jadwal mengajar yang dikelompokkan berdasarkan hari serta melakukan pencatatan absensi sesuai kebutuhan. Fitur “Reset Kehadiran Mingguan” juga tersedia untuk mengatur ulang data kehadiran pada awal minggu baru.



Gambar 4.30 Mata Kuliah Yang Diampu

Halaman ini berfungsi untuk menampilkan daftar seluruh mata kuliah yang menjadi tanggung jawab pengajaran seorang dosen. Pada bagian atas halaman ditampilkan informasi identitas dosen, seperti nama lengkap dan NIDN (Nomor Induk Dosen Nasional).

4.2 Pengujian Sistem

Tujuan dari pengujian sistem ini adalah untuk memastikan bahwa sistem penjadwalan kegiatan belajar mahasiswa berbasis *Artificial Intelligence (AI)* menggunakan algoritma genetika telah berjalan sesuai dengan perancangan dan spesifikasi kebutuhan sistem. Pengujian dilakukan guna mengevaluasi fungsionalitas sistem, mengidentifikasi potensi kesalahan, serta mengukur sejauh mana sistem mampu menyusun jadwal kuliah secara otomatis tanpa konflik antara dosen, mata kuliah, ruang, dan waktu perkuliahan.

4.2.1 Pengujian Chatbot

Pengujian chatbot dilakukan untuk memastikan bahwa fitur chatbot dapat merespons pertanyaan pengguna dengan benar dan menampilkan informasi jadwal perkuliahan sesuai data yang tersedia. Contoh dapat di lihat pada gambar 4.31 di bawah ini.



Gambar 4.31 Contoh Penggunaan ChatBot

Tabel 4.1 Pengujian Chatbot

No	Input	Output	Status
1	cek ruangan kosong [hari] jam [jam]	Menampilkan daftar ruang kuliah yang kosong lengkap dengan informasi hari dan jam sesuai permintaan pengguna.	Berhasil
2	Tampilkan jadwal kuliah hari [hari]	Menampilkan jadwal kuliah berdasarkan hari yang diminta oleh pengguna	Berhasil
3	siapa saja dosen yang mengajar hari [hari]	Menampilkan daftar dosen yang mengajar pada hari yang diminta oleh pengguna.	Berhasil
4	ada kelas semester [n] hari [hari]	Menampilkan daftar kelas berdasarkan semester dan hari yang diminta oleh pengguna.	Berhasil
5	ruangan yang tersedia dari jam [jam] sampai [jam] hari [hari]	Menampilkan daftar ruang kuliah yang tersedia berdasarkan hari dan rentang waktu perkuliahan yang diminta, mulai dari jam awal hingga jam akhir perkuliahan	Berhasil
6	pindah jadwal sementara kode [KODE_MK] kelas [KELAS] NIDN [NIDN] ke hari [hari] jam [jam] [durasi] minggu	Sistem menampilkan konfirmasi bahwa jadwal mata kuliah dengan kode [KODE_MK] untuk kelas [KELAS] dan dosen dengan NIDN [NIDN] telah dipindahkan ke hari [HARI] pada jam [JAM] selama [DURASI] minggu ke depan sebagai jadwal sementara	Berhasil

Dari hasil pengujian di atas, seluruh fungsi chatbot berhasil dijalankan sesuai dengan yang diharapkan. *Chatbot* dapat mengenali intent pengguna seperti pertanyaan tentang jadwal harian, nama dosen, nama mata kuliah, serta dapat menampilkan jawaban dengan tepat berdasarkan data jadwal yang telah dihasilkan sistem penjadwalan.

4.2.2 Pengujian Algoritma Genetika

Pengujian *algoritma genetika* dilakukan untuk mengevaluasi kemampuan algoritma dalam menghasilkan jadwal perkuliahan yang optimal dan bebas konflik.

Algoritma Genetik

Generate jadwal otomatis dengan algoritma genetik yang mempertimbangkan konflik dosen dan ruangan.

Semester: Semua Semester

Jurusan: Semua Jurusan

Kelas: Semua Kelas

Mode Generate: Tambah Jadwal (Tidak Menimpa)

Gambar 4.32 Filter Generate Algoritma Genetika

Sebelum menjalankan proses *generate* jadwal, sistem menyediakan beberapa parameter yang dapat diatur oleh pengguna untuk menyesuaikan ruang lingkup penjadwalan. Pada tampilan pengaturan algoritma genetika, pengguna dapat memilih semester, jurusan, kelas, dan mode generate sesuai kebutuhan. Misalnya, pengguna dapat mengatur agar *generate* jadwal dilakukan untuk semua semester, seluruh jurusan, dan semua kelas yang tersedia, atau dapat difokuskan hanya pada jurusan dan kelas tertentu saja.

Selain itu, tersedia juga opsi mode *generate* yang memungkinkan pengguna untuk memilih apakah ingin menambahkan jadwal baru tanpa menimpa jadwal yang sudah ada, atau menggunakan mode lain yang menyesuaikan kondisi tertentu

Pada kali ini dilakukan 10 kali percobaan untuk melihat hasil yang diperoleh dari *generate algoritma genetika*. Hasil dapat dilihat pada tabel 4.2.

Tabel 4.2 Hasil Generate Algoritma Genetika

Generate	Hasil Generate
Generate Jadwal Otomatis Reset Jadwal Batch generation selesai! Berhasil: 7, Gagal: 2 dari 9 kombinasi.	Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2024: 245 jadwal 2023: 138 jadwal 2022: 114 jadwal Total Jadwal: 497 jadwal
Generate Jadwal Otomatis Reset Jadwal Batch generation selesai! Berhasil: 9, Gagal: 0 dari 9 kombinasi.	Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2024: 246 jadwal 2023: 289 jadwal 2022: 252 jadwal Total Jadwal: 787 jadwal
Generate Jadwal Otomatis Reset Jadwal Batch generation selesai! Berhasil: 9, Gagal: 0 dari 9 kombinasi.	Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2024: 246 jadwal 2023: 289 jadwal 2022: 252 jadwal Total Jadwal: 787 jadwal
Generate Jadwal Otomatis Reset Jadwal Batch generation selesai! Berhasil: 6, Gagal: 3 dari 9 kombinasi.	Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2024: 227 jadwal 2023: 99 jadwal 2022: 162 jadwal Total Jadwal: 488 jadwal
Generate Jadwal Otomatis Reset Jadwal Batch generation selesai! Berhasil: 9, Gagal: 0 dari 9 kombinasi.	Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2024: 246 jadwal 2023: 289 jadwal 2022: 252 jadwal Total Jadwal: 787 jadwal
Generate Jadwal Otomatis Reset Jadwal Batch generation selesai! Berhasil: 9, Gagal: 0 dari 9 kombinasi.	Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2024: 246 jadwal 2023: 289 jadwal 2022: 252 jadwal Total Jadwal: 787 jadwal
Generate Jadwal Otomatis Reset Jadwal Batch generation selesai! Berhasil: 8, Gagal: 1 dari 9 kombinasi.	Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2024: 251 jadwal 2023: 287 jadwal 2022: 91 jadwal Total Jadwal: 629 jadwal
Generate Jadwal Otomatis Reset Jadwal Terjadi kesalahan saat generate jadwal.	Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: Total Jadwal: 0 jadwal

Generate Jadwal Otomatis Reset Jadwal Batch generation selesai! Berhasil: 8, Gagal: 1 dari 9 kombinasi.	🏠 Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2024: 246 jadwal 2023: 234 jadwal 2022: 252 jadwal Total Jadwal: 732 jadwal
Generate Jadwal Otomatis Reset Jadwal Terjadi kesalahan saat generate jadwal.	🏠 Informasi Jadwal Saat Ini Jumlah Jadwal per Semester: 2022: 162 jadwal Total Jadwal: 162 jadwal

Setelah dilakukan sebanyak 10 kali percobaan *generate* jadwal menggunakan *algoritma genetika*, diperoleh hasil yang bervariasi tergantung pada jumlah kombinasi yang berhasil diproses dan konflik yang muncul selama proses optimasi. Sistem memberikan notifikasi berupa jumlah kombinasi yang berhasil dan gagal pada setiap batch generate.

Salah satu hasil terbaik terlihat pada kondisi "Berhasil: 9, Gagal: 0 dari 9 kombinasi", di mana sistem berhasil menghasilkan 787 jadwal perkuliahan tanpa ada kombinasi yang gagal dan menghasilkan jumlah jadwal persemester dimana pada tahun 2024: 246 jadwal, tahun 2023: 289 jadwal, dan tahun 2022: 252 jadwal. Ini menunjukkan bahwa *algoritma genetika* mampu menyusun jadwal yang sepenuhnya optimal ketika seluruh data (kelas, dosen, ruang, waktu) berhasil dieksekusi tanpa konflik.

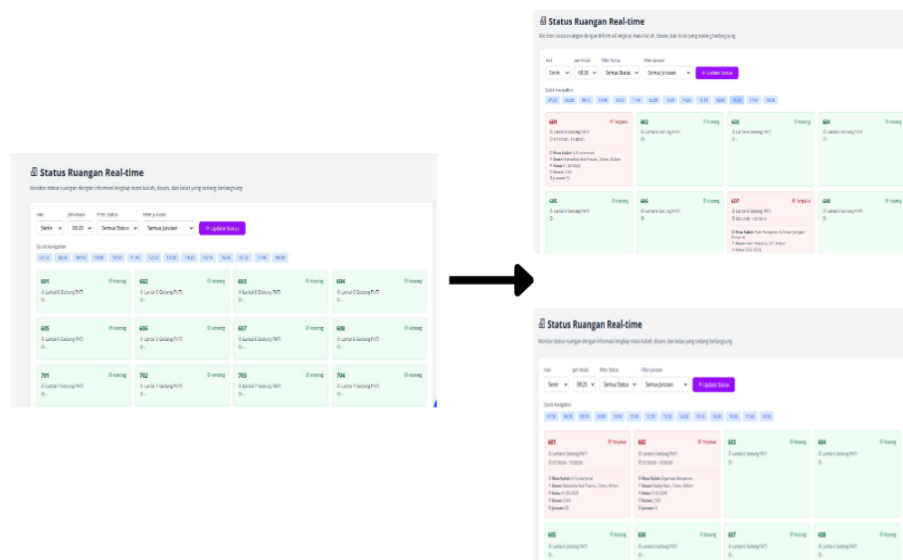
Sebaliknya, pada beberapa percobaan lain ditemukan kasus seperti "Berhasil: 6, Gagal: 3" atau bahkan muncul "Terjadi kesalahan saat generate jadwal", yang menunjukkan bahwa pada skenario tertentu, data masukan seperti bentrok ruangan, dosen, atau kelas

dapat menyebabkan sebagian kombinasi gagal di-*generate* atau sistem tidak mampu menyusun jadwal dengan sempurna.

Jumlah data yang besar dan beragam ini menunjukkan bahwa sistem telah diuji dalam kondisi realistis dan kompleks, yang mendekati skenario nyata di Fakultas Ilmu Komputer dan Teknologi Informasi. Dengan demikian, hasil dari pengujian ini dapat merepresentasikan performa *algoritma genetika* secara akurat dalam menyusun jadwal akademik berskala besar.

4.2.3 Pengujian Fitur Tambahan

Selain dari 2 fitur utama *chatbot* dan *generate algoritma gentika*, ada beberapa tambahan fitur lainnya yang akan lebih membantu pengguna nantinya. Beberapa fitur tersebut antara lain dapat dilihat pada gambar di bawah.



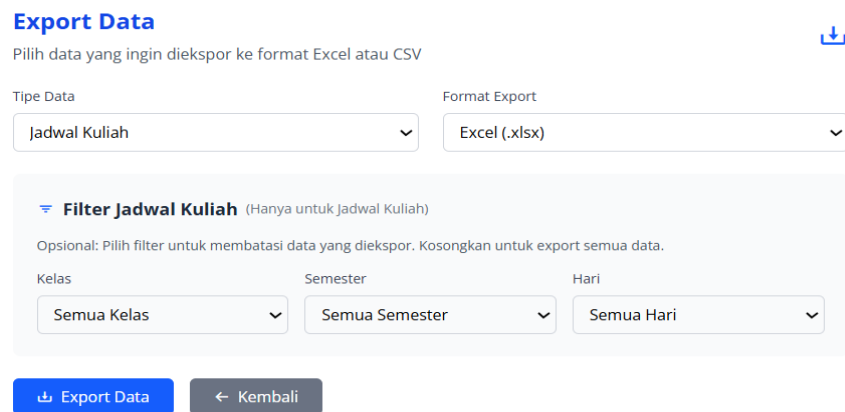
Gambar 4.33 Pengujian Status Ruang Real-time

Salah satu fitur tambahan yang diuji dalam sistem adalah *Status Ruang Real-time*. Fitur ini memungkinkan pengguna untuk

melihat ketersediaan ruang kuliah secara langsung berdasarkan parameter hari, jam, dan tanggal yang dipilih. Pada gambar di atas, ditampilkan perubahan tampilan dari kondisi awal di mana seluruh ruangan masih kosong. Setelah proses generate jadwal dilakukan menggunakan *algoritma genetika*, sistem secara otomatis memperbarui status setiap ruangan dan menampilkan informasi ruang mana saja yang sedang digunakan untuk kegiatan perkuliahan, serta ruang mana yang masih tersedia.

Setiap ruangan divisualisasikan menggunakan indikator warna yang berbeda untuk mempermudah identifikasi: warna merah menandakan ruangan sedang digunakan, sedangkan warna hijau menunjukkan ruangan dalam kondisi kosong. Informasi tambahan seperti kode ruang, kapasitas, serta kelas dan mata kuliah yang berlangsung juga ditampilkan secara jelas dan informatif.

Selain itu, hasil generate jadwal bersifat dinamis, artinya setiap kali proses generate dilakukan, sistem dapat menghasilkan susunan jadwal yang berbeda-beda tergantung dari kondisi dan parameter yang digunakan (misalnya jumlah kelas, dosen, dan kombinasi waktu-ruang). Hal ini menegaskan bahwa algoritma genetika dalam sistem ini tidak hanya bekerja secara adaptif, tetapi juga mampu mengeksplorasi berbagai kemungkinan jadwal yang valid dan optimal di setiap iterasinya.



Export Data

Pilih data yang ingin diekspor ke format Excel atau CSV

Tipe Data: Jadwal Kuliah Format Export: Excel (.xlsx)

Filter Jadwal Kuliah (Hanya untuk jadwal Kuliah)

Opsional: Pilih filter untuk membatasi data yang diekspor. Kosongkan untuk export semua data.

Kelas: Semua Kelas Semester: Semua Semester Hari: Semua Hari

Export Data Kembali

Gambar 4.34 Filter Export Data

Kemudian fitur tambahan berikutnya yang diuji dalam sistem adalah Export Data, yang memungkinkan pengguna untuk mengunduh data jadwal perkuliahan dalam format Excel (.xlsx) atau CSV. Fitur ini sangat berguna bagi admin akademik atau pihak fakultas yang memerlukan salinan data jadwal untuk keperluan dokumentasi, analisis, atau distribusi offline.

Pengguna dapat memilih jenis data yang akan diekspor melalui opsi Tipe Data, misalnya "Jadwal Kuliah", serta memilih format file hasil ekspor pada bagian Format Export. Sistem juga menyediakan opsi filter yang dapat digunakan untuk menyaring data berdasarkan kelas, semester, dan hari tertentu. Jika pengguna tidak mengatur filter, maka seluruh data jadwal yang tersedia akan diekspor secara lengkap. Contoh dari hasil dari export data dapat kita lihat pada gambar di bawah ini.

No	Mata Kuliah	Kode MK	Dosen	Ruangan	Hari	Jam Mulai	Jam Selesai	Kelas	SKS	Semester	Jurusan
1	Aljabar, Vektor & Matrik	TI2401	Dr. Marah Dolly Nasution., M.Si	F304	Rabu	07:30:00	10:00:00	C1 (TI 2024)	3	2024	TI
2	Bahasa Inggris	TI2402	Dr. Mandra Saragih, S.Pd., M.Hum	Lab 605	Jumat	17:40:00	19:20:00	C1 (TI 2024)	2	2024	TI
3	lbadah & Muamalah	TI2403	Nadlhras Naimi, S.Ag., M.A	F305	Kamis	10:50:00	12:30:00	C1 (TI 2024)	2	2024	TI
4	Kewarganegaraan	TI2404	Dr. Lutfi Basit., M.I.Kom	F303	Jumat	16:00:00	18:30:00	C1 (TI 2024)	2	2024	TI
5	Kewirausahaan IT	TI2405	Fatma Sari Hutagalung., S.Kom., M.Kom	603	Sabtu	17:40:00	19:20:00	C1 (TI 2024)	2	2024	TI
6	Prak. Sistem Operasi	TI2407	Mulkan Azhari, S.Kom., M.Kom	708	Kamis	12:30:00	13:20:00	C1 (TI 2024)	1	2024	TI
7	Prak. Struktur Data	TI2408	Halim Maulana., S.T., M.Kom	708	Rabu	07:30:00	08:20:00	C1 (TI 2024)	1	2024	TI
8	Struktur Data	TI2410	Halim Maulana., S.T., M.Kom	705	Sabtu	07:30:00	09:10:00	C1 (TI 2024)	2	2024	TI
9	Teknologi Open Source	TI2411	Okvi Nugroho., S.Kom., M.Kom	705	Rabu	17:40:00	19:20:00	C1 (TI 2024)	2	2024	TI

Gambar 4.35 Hasil Export Data

Gambar di atas merupakan hasil *generate* jadwal kuliah untuk kelas C1 dari program studi Teknologi Informasi (TI) angkatan 2024. Setiap baris pada tabel merepresentasikan satu sesi perkuliahan, lengkap dengan informasi sebagai berikut:

1. Mata Kuliah
2. Kode MK
3. Dosen
4. Ruang
5. Hari
6. Jam Mulai & Jam Selesai
7. Kelas
8. SKS
9. Semester
10. Jurusan

Data ini menunjukkan bahwa sistem berhasil menyusun jadwal secara rinci, memperhitungkan dosen, ruang kuliah, dan waktu tanpa adanya konflik antar jadwal. Format ini juga sesuai untuk diekspor melalui fitur Export Data yang telah dijelaskan sebelumnya, dan dapat digunakan untuk keperluan pelaporan, distribusi ke mahasiswa, atau publikasi jadwal resmi fakultas.

4.3 Analisis Sistem

Analisis sistem dilakukan untuk mengevaluasi sejauh mana sistem yang telah dikembangkan mampu menjawab rumusan masalah dan mencapai tujuan penelitian. Pada tahap ini, hasil pengujian yang telah dijelaskan pada Bab 4.2 dijadikan dasar dalam memberikan penilaian terhadap kinerja sistem, baik dari segi fungsionalitas, efektivitas algoritma, integrasi antarmuka, maupun manfaat praktisnya di lingkungan akademik.

Analisis dilakukan dengan mengacu langsung pada empat rumusan masalah yang telah disusun sebelumnya. Setiap rumusan akan dibahas secara terpisah untuk menunjukkan bagaimana solusi yang ditawarkan sistem mampu menjawab kebutuhan nyata dalam penjadwalan perkuliahan otomatis berbasis *AI*. Dengan pendekatan ini, diharapkan pembaca memperoleh gambaran yang jelas mengenai kekuatan, fleksibilitas, dan kelayakan implementasi sistem dalam skala institusi.

4.3.1 Pengembangan Sistem Penjadwalan Otomatis

Sistem penjadwalan ini dikembangkan dengan memanfaatkan pendekatan kecerdasan buatan (*Artificial Intelligence*) melalui penerapan algoritma genetika. Proses pengembangan dilakukan menggunakan bahasa pemrograman *Python* dengan dukungan *framework Flask* dan juga *Laravel* sebagai platform *web* dan *library DEAP* untuk implementasi algoritma evolusi. Seluruh komponen ini membentuk sistem yang mampu menjalankan proses generate jadwal perkuliahan secara otomatis.

Melalui pengujian, terbukti bahwa sistem dapat menangani data skala besar, seperti menyusun hingga 787 jadwal kuliah dari berbagai kombinasi semester, jurusan, dan kelas. Sistem juga memberikan fleksibilitas pengaturan input, sehingga admin dapat menentukan ruang lingkup generate sesuai kebutuhan, baik seluruh semester maupun kelas tertentu saja. Hal ini menunjukkan bahwa sistem berhasil dikembangkan secara fungsional sesuai tujuan awal penelitian.

Kinerja sistem juga konsisten dalam berbagai skenario pengujian. Baik ketika melakukan generate untuk seluruh data maupun hanya sebagian, sistem tetap dapat merespons dengan cepat dan akurat. Hal ini membuktikan bahwa proses pengolahan yang dilakukan oleh sistem dapat menangani data berskala besar secara efisien, sekaligus memastikan keluaran yang sesuai dengan kebutuhan akademik.

4.3.2 Penerapan Algoritma Genetika dalam Mengurangi Konflik

Algoritma genetika bekerja dengan prinsip seleksi alam, di mana solusi yang minim konflik akan bertahan dan berevolusi hingga mencapai kondisi optimal. Dalam konteks penjadwalan kuliah, konflik dapat berupa bentrok jadwal antara dosen, mahasiswa, atau penggunaan ruang. Sistem secara otomatis menghitung nilai fitness dari setiap solusi dan memilih yang terbaik dari generasi ke generasi.

Berdasarkan hasil pengujian, sistem mampu menghasilkan jadwal dengan nilai *fitness* 1.00, yang berarti tidak ada konflik. Ini

membuktikan bahwa *algoritma genetika* efektif dalam mengurangi bahkan menghilangkan benturan jadwal. Ketika terjadi ketidaksesuaian data, sistem juga mampu mendeteksi kombinasi gagal, menunjukkan bahwa algoritma bekerja adaptif dalam menghadapi kondisi yang tidak ideal.

Dalam beberapa kondisi pengujian lain, sistem juga mampu mengidentifikasi kombinasi data yang gagal di-generate, seperti kasus "Berhasil: 6, Gagal: 3", dan menampilkan peringatan kepada pengguna. Hal ini menunjukkan bahwa algoritma tidak hanya mencari solusi terbaik, tetapi juga responsif terhadap data yang tidak valid atau sulit dioptimalkan, sehingga tetap menjaga integritas hasil jadwal yang dihasilkan.

4.3.3 Integrasi Sistem dengan Platform Website

Sistem yang dikembangkan telah terintegrasi secara penuh ke dalam platform *website* yang mudah diakses oleh seluruh pihak terkait, termasuk admin akademik dan dosen. Pengguna tidak hanya dapat melihat hasil *generate* jadwal, tetapi juga menggunakan fitur interaktif seperti *chatbot* untuk menanyakan jadwal pribadi, ruang kosong, atau dosen pengajar pada hari tertentu.

Selain itu, sistem dilengkapi dengan fitur export data yang memungkinkan pengguna mengunduh jadwal dalam format Excel. Kemudahan ini meningkatkan efisiensi distribusi jadwal dan mendukung dokumentasi formal. Dengan adanya akses *real-time* dan fitur tambahan seperti tampilan status ruangan, sistem ini terbukti

memberikan pengalaman pengguna yang lebih praktis dibanding sistem manual.

Tampilan jadwal juga disajikan secara dinamis dan informatif, mempermudah staf akademik dalam melakukan pemantauan atau penyesuaian. Integrasi fitur-fitur ini membuktikan bahwa sistem tidak hanya bekerja di sisi *back-end* (*generate* jadwal), tetapi juga menyediakan antarmuka pengguna yang mendukung kelancaran operasional akademik sehari-hari.

4.3.4 Efektivitas Sistem dibanding Metode Manual

Bila dibandingkan dengan metode manual, sistem ini menunjukkan keunggulan signifikan dalam kecepatan dan akurasi. Proses *generate* hanya membutuhkan waktu beberapa menit untuk menghasilkan ratusan jadwal, sementara metode manual bisa memakan waktu berhari-hari dan tetap berisiko tinggi terhadap konflik jadwal. Selain itu, sistem dapat menampilkan hasil yang berbeda setiap kali *generate* dilakukan, yang menunjukkan fleksibilitas dan kemampuan eksplorasi solusi optimal oleh *algoritma genetika*.

Fitur status ruangan *real-time* juga menjadi nilai tambah yang sulit diwujudkan secara manual. Pengguna dapat memantau ruang kosong berdasarkan hari dan jam tertentu, membantu pengambilan keputusan jika diperlukan perubahan jadwal atau pemindahan kelas. Dengan semua keunggulan tersebut, sistem ini terbukti lebih efektif,

efisien, dan dapat diandalkan untuk diterapkan dalam lingkungan akademik.

4.4 Evaluasi Sistem

Berdasarkan hasil pengujian yang telah dilakukan, sistem mampu menjalankan fungsinya dengan baik dalam menyusun jadwal perkuliahan secara otomatis, mengurangi konflik jadwal, serta memberikan akses informasi secara real-time kepada pengguna.

Dari sisi performa, sistem terbukti mampu menghasilkan ratusan jadwal (hingga 787 jadwal) dalam waktu yang relatif cepat, dengan tingkat keberhasilan generate yang tinggi. Proses generate juga responsif terhadap variasi input, baik untuk seluruh semester maupun kelas tertentu. Nilai *fitness* yang mencapai 1.00 pada beberapa pengujian membuktikan bahwa sistem mampu menyusun jadwal tanpa konflik, yang menandakan keberhasilan penerapan *algoritma genetika* dalam konteks ini.

Secara fungsional, fitur-fitur tambahan seperti chatbot, export data, dan status ruangan real-time bekerja dengan baik dan memberi kemudahan bagi pengguna. Fitur chatbot dapat menjawab pertanyaan mengenai jadwal, dosen, dan ruangan dengan responsif, sedangkan status ruangan memungkinkan pemantauan kondisi ruang secara langsung berdasarkan hari dan waktu. Kemampuan sistem untuk menyesuaikan output berdasarkan parameter yang dipilih juga menjadi nilai tambah dalam hal fleksibilitas penggunaan.

Namun demikian, masih terdapat beberapa catatan yang perlu diperhatikan. Misalnya, jika data input tidak lengkap atau tidak sesuai format,

proses *generate* dapat mengalami kegagalan sebagian. Selain itu, *algoritma genetika* membutuhkan waktu eksplorasi solusi lebih lama apabila kombinasi data sangat kompleks dan banyak batasan yang harus dipenuhi. Oleh karena itu, pengelolaan data input menjadi faktor penting dalam menjamin keberhasilan proses penjadwalan.

Secara keseluruhan, sistem telah memenuhi sebagian besar kebutuhan yang dirancang sejak awal penelitian. Meskipun masih terdapat ruang untuk penyempurnaan, terutama dalam hal validasi data dan visualisasi hasil, sistem ini telah menunjukkan kinerja yang sangat baik dan layak untuk digunakan dalam skala operasional akademik.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan, dapat disimpulkan bahwa sistem penjadwalan perkuliahan otomatis berbasis *algoritma genetika* yang dikembangkan dalam penelitian ini mampu menyelesaikan permasalahan penjadwalan secara efektif dan efisien. Sistem berhasil menghasilkan jadwal perkuliahan tanpa konflik antara dosen, ruang, dan kelas, dengan total output mencapai 787 jadwal, dan pada beberapa skenario bahkan berhasil mencapai nilai fitness maksimal (1.00), yang menunjukkan bahwa tidak ada konflik yang ditemukan.

Selain keberhasilan dalam proses *generate* jadwal, sistem ini juga dilengkapi dengan berbagai fitur pendukung yang sesuai dengan kebutuhan pengguna, seperti *chatbot* untuk interaksi cepat, status ruangan real-time untuk memantau ketersediaan ruang kuliah, serta fitur export ke Excel untuk dokumentasi dan distribusi jadwal. Semua fitur tersebut berfungsi dengan baik selama pengujian, dan terbukti dapat meningkatkan efisiensi pengelolaan akademik, terutama untuk pihak admin atau staf akademik yang sebelumnya menyusun jadwal secara manual.

Sistem ini telah terintegrasi sepenuhnya ke dalam platform berbasis *web*, yang memungkinkan akses informasi secara luas oleh berbagai pihak, termasuk dosen dan admin akademik. Fleksibilitas sistem juga terbukti dari kemampuannya dalam menangani berbagai parameter input seperti semester, jurusan, dan kelas, serta menghasilkan solusi jadwal yang bervariasi setiap

kali proses generate dilakukan. Hal ini menunjukkan bahwa sistem tidak hanya efisien tetapi juga adaptif terhadap kebutuhan yang dinamis.

Secara keseluruhan, penelitian ini membuktikan bahwa penerapan *algoritma genetika* pada sistem penjadwalan akademik tidak hanya layak secara teknis, tetapi juga memberikan manfaat nyata dalam aspek kecepatan, ketepatan, skalabilitas, dan fleksibilitas penggunaan. Sistem ini sangat potensial untuk diterapkan secara nyata dalam pengelolaan akademik di lingkungan fakultas atau universitas.

5.2 Saran

Untuk pengembangan sistem ke arah yang lebih baik di masa mendatang, berikut beberapa saran yang dapat dijadikan bahan pertimbangan:

1. Peningkatan validasi data input

Sistem perlu dilengkapi dengan proses validasi otomatis terhadap data input (kelas, dosen, jam, dan ruang) untuk mencegah kesalahan yang menyebabkan kegagalan *generate*.

2. Visualisasi hasil jadwal yang lebih interaktif

Tampilan jadwal yang lebih visual, seperti dalam bentuk kalender mingguan atau *grid* interaktif, akan sangat membantu pengguna untuk membaca dan memahami jadwal dengan cepat.

3. Fitur pengeditan jadwal secara manual

Akan lebih fleksibel jika admin diberikan akses untuk mengedit atau menyusun ulang sebagian jadwal secara manual setelah proses *generate* selesai, tanpa perlu mengulang dari awal.

4. Optimasi performa generate pada data besar

Untuk skenario dengan jumlah data sangat besar, proses *generate* masih bisa ditingkatkan efisiensinya, misalnya dengan menggunakan teknik *pre-filtering* atau pendekatan *hybrid algoritma*.

5. Integrasi dengan sistem informasi akademik kampus

Pengembangan sistem sebaiknya diarahkan untuk dapat berintegrasi dengan sistem akademik kampus yang sudah berjalan, sehingga data dosen, mata kuliah, dan ruang dapat ter-update otomatis dan tidak perlu di input ulang.

6. Pengembangan Fitur Chatbot Yang Lebih Cerdas

Fitur chatbot dapat ditingkatkan dengan kemampuan pemahaman bahasa alami (Natural Language Processing) yang lebih baik, serta diperluas cakupannya. Contohnya, *chatbot* dapat menangani pertanyaan terkait perubahan jadwal, notifikasi otomatis jika ada pergantian kelas, hingga pemrosesan permintaan pemindahan jadwal secara interaktif.

7. Optimasi Waktu Komputasi Algoritma Genetika

Disarankan untuk mengimplementasikan sistem pada *server* yang memiliki spesifikasi lebih tinggi atau menggunakan layanan *hosting* berbasis *cloud*, sehingga waktu komputasi dapat dipersingkat walaupun data yang diproses dalam jumlah besar. Selain itu, dapat dipertimbangkan penggunaan teknik optimasi seperti *parallel processing* atau *population seeding* agar proses perhitungan *fitness* lebih efisien.

8. Penyederhanaan Format Input Chatbot

Disarankan agar format perintah untuk penggantian jadwal melalui *chatbot* dibuat lebih sederhana dan *user-friendly*. Misalnya dengan menyediakan pilihan interaktif atau tombol (*quick reply*) sehingga pengguna tidak perlu mengetik kode seperti kode_mk, kelas, NIDN, hari, jam, dan durasi secara manual, tetapi cukup memilih dari daftar yang tersedia.

Dengan mengimplementasikan saran-saran di atas, sistem ini dapat terus disempurnakan hingga benar-benar siap digunakan dalam skala luas di lingkungan fakultas maupun universitas. Potensi pengembangan lebih lanjut juga terbuka untuk menjangkau proses-proses lain di bidang akademik yang membutuhkan pengelolaan jadwal secara otomatis dan terstruktur.

DAFTAR ISI

- Adamopoulou, E., & Moussiades, L. (2020). An overview of chatbot technology. *Artificial Intelligence Applications and Innovations*, (pp. 373–383).
https://doi.org/10.1007/978-3-030-49186-4_31
- Ajiz, M. F. A., Ramadan, M. F. S., Mutia, H. D., & Yanuari, P. D. (2023). Pengembangan aplikasi chatbot informasi akademik berbasis web menggunakan AIML. *Majalah Jurnal Informatika (MJI)*, 15 (2), 147–153.
- Amrulloh, A., & Sela, E. I. (2021). Optimasi proses penjadwalan mata kuliah menggunakan algoritme genetika dan pencarian tabu. *Jurnal Teknologi dan Sistem Komputer*, 9 (3), 157–166.
- Fahrezi, K., Mulana, A. R., Melinda, S., Nurhaliza, N., & Mulyati, S. (2021). Penerapan model Waterfall dalam pengembangan sistem informasi akademik berbasis web sebagai sistem pengolahan nilai siswa. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 4 (2), 98–102.
<https://openjournal.unpam.ac.id/index.php/JTSI/article/view/10196>
- Fauzi, M., & Akbar, Z. (2024). Sistem penjadwalan sidang skripsi dengan algoritma genetika pada program studi Informatika Universitas Muhammadiyah Jambi. *Jurnal Informatika Sistem dan Aplikasi*, 8 (1).
- Ferawaty, E. (2010). Optimasi penjadwalan mata kuliah menggunakan algoritma genetika. Skripsi. Universitas Indonesia.
- Haupt, R. L., & Haupt, S. E. (2004). *Practical genetic algorithms* (2nd ed.). John Wiley & Sons.

- Hidayat, A. T., Hakim, L., & Ravanza, M. A. (2023). Implementasi sistem penjadwalan mata kuliah menggunakan metode algoritma genetika berbasis web. *Bulletin of Computer Science and Research*.
- Holland, J. H. (1975). *Adaptation in natural and artificial systems*. University of Michigan Press.
- Jogiyanto, H. M. (2005). *Analisis dan desain sistem informasi: Pendekatan terstruktur teori dan praktik aplikasi bisnis*. Andi.
- Laksono, A. T., Utami, M. C., & Sugiarti, Y. (2016). Sistem penjadwalan kuliah menggunakan metode algoritma genetika (Studi Kasus: FKIK Universitas Muhammadiyah Jakarta). *Jurnal Sistem Informasi*, Universitas Islam Negeri Jakarta.
- Mitchell, M. (1998). *An introduction to genetic algorithms*. MIT Press.
- Oktavia, D. H., & Suseno, G. (2024). Pemanfaatan kecerdasan buatan dalam pendidikan di Indonesia: Potensi dan tantangan. *Indo-MathEdu Intellectuals Journal*, 5 (2), 1680–1686.
<https://doi.org/10.54373/imeij.v5i2.876>
- Pambudi, A. P., Waluyo, A., & Fatich, E. V. L. N. (2021). Perancangan sistem penjadwalan perkuliahan berbasis website menggunakan algoritma genetika. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 8 (3), 1133–1146.
<https://doi.org/10.35967/jatisi.v8i3.1224>
- Panjaitan, D. J. (2022). Sistem informasi penjadwalan mata pelajaran berbasis web menggunakan algoritma genetika (Studi Kasus: SMK Negeri 3 Medan). Skripsi. Repositori Universitas Medan Area.

- Pressman, R. S. (2010). *Software engineering: A practitioner's approach* (7th ed.). McGraw-Hill Education.
- Qalimaturrahmah, M., & Santoso, D. B. (2024). Aplikasi layanan dan informasi akademik berbasis chatbot Telegram menggunakan Natural Language Processing. *Jurnal Teknologi Informasi dan Komunikasi*, 8 (2), 434–443.
<https://doi.org/10.35870/jtik.v8i2.1887>
- R. L., & Haupt, S. E. (2004). *Practical Genetic Algorithms* (2nd ed.). John Wiley & Sons.
- Russell, S., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Pearson Education.
- Serban, I. V., Lowe, R., Henderson, P., Charlin, L., & Pineau, J. (2017). A survey of available corpora for building data-driven dialogue systems. *Dialogue & Discourse*, 9 (1), 1–49.
<https://www.dialogue-and-discourse.org/volumes/09/DD0921.pdf>
- Sihaloho, F. A. S., & Napitupulu, Z. (2023). Penggunaan kecerdasan buatan (Artificial Intelligence) dalam dunia pendidikan di Indonesia: Tinjauan literatur. *Jurnal Rekognisi*, 3 (1), 15–25.
<https://jurnal.unusu.ac.id/index.php/rekognisi/article/view/167>
- Sugiyono. (2016). *Metode penelitian dan pengembangan (Research and Development/R&D)*. Alfabeta.
- Sutedjo, B. (2016). *Pengantar sistem informasi*. Andi.
<https://opac.perpusnas.go.id/DetailOpac.aspx?id=1134573>

Vaidyam, A. N., Wisniewski, H., Halamka, J. D., Kashavan, M. S., & Torous, J. B. (2019). Chatbots and conversational agents in mental health: A review of the psychiatric landscape. *The Canadian Journal of Psychiatry*, 64(7), 456–464.

<https://doi.org/10.1177/0706743719828977>

LAMPIRAN



MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PESAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 89/SK/BAN-PT/Akred/PT/10/2019

Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003

<https://id.umsu.ac.id>

info@umsu.ac.id

[umsu.medan](#)

[umsu.medan](#)

[umsu.medan](#)

[umsu.medan](#)

PENETAPAN DOSEN PEMBIMBING
PROPOSAL/SKRIPSI MAHASISWA
NOMOR : 258/IL3-AU/UMSU-09/F/2025

Assalamu'alaikum Warahmatullahi Wabarakatuh

Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Muhammadiyah Sumatera Utara, berdasarkan Persetujuan permohonan judul penelitian Proposal / Skripsi dari Ketua / Sekretaris.

Program Studi : Teknologi Informasi
 Pada tanggal : 04 Februari 2025

Dengan ini menetapkan Dosen Pembimbing Proposal / Skripsi Mahasiswa.

Nama : Bahdi Rizky Al Wahya
 NPM : 2109020135
 Semester : VII (Tujuh)
 Program studi : Teknologi Informasi
 Judul Proposal / Skripsi : Perancangan Sistem Penjadwalan Kegiatan Belajar Mahasiswa Berbasis AI Menggunakan Algoritma Genetik (Studi kasus : Universitas Muhammadiyah Sumatera Utara)

Dosen Pembimbing : Indah Purnama Sari, S.T.,M.Kom.

Dengan demikian di izinkan menulis Proposal / Skripsi dengan ketentuan

1. Penulisan berpedoman pada buku panduan penulisan Proposal / Skripsi Fakultas Ilmu Komputer dan Teknologi Informasi UMSU
2. Pelaksanaan Sidang Skripsi harus berjarak 3 bulan setelah dikeluarkannya Surat Penetapan Dosen Pembimbing Skripsi.
3. Proyek Proposal / Skripsi dinyatakan "BATAL" bila tidak selesai sebelum Masa Kadaluaarsa tanggal : 04 Februari 2026
4. Revisi judul.....

Wassalamu'alaikum Warahmatullahi Wabarakatuh.

Ditetapkan di : Medan
 Pada Tanggal : 05 Sya'ban 1446 H
 04 Februari 2025M



Dekan

 Dr. Akhbarizmi, M.Kom.
 NIDN : 0127099201

Cc. File





UMSU

Unggul | Cerdas | Terpercaya

Waktu mengabdikan ilmu, agar ilmu itu bermanfaat
manusia dan lingkungannya

MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 89/SK/BAN-PT/Akred/PT/III/2019

Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003

<http://www.umsu.ac.id>

fmkt@umsu.ac.id

[umsu](#)

[umsu](#)

[umsu](#)

[umsu](#)

PERUBAHAN TOPIK/JUDUL PENELITIAN

Nomor agenda :
Tanggal persetujuan :
Topik yang disetujui Program Studi : PERANCANGAN SISTEM PENJADWALAN
KEGIATAN BELAJAR BERBASIS
BERBASIS AI MENGGUNAKAN
ALGORITMA GENETIKA DI UNIVERISTAS
MUHAMMADIYAH SUMATRA UTARA
Nama Dosen pembimbing : Indah Purnama Sari, S.T., M.Kom
Judul yang disetujui Dosen Pembimbing : PERANCANGAN SISTEM
PENJADWALAN KEGIATAN BELAJAR
BERBASIS BERBASIS AI MENGGUNAKAN
ALGORITMA GENETIKA DI FAKULTAS
ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI

Medan, Agustus 2025

Disahkan oleh

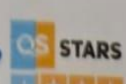
Ketua Program Studi
Teknologi Informasi

(Fatma Sari Hutagalung, M.Kom)

Persetujuan

Dosen Pembimbing

(Indah Purnama Sari, S.T., M.Kom)





UMSU

Unggul | Cerdas | Terpercaya

Revisi merupakan bagian integral dari proses pembelajaran, sehingga harus dilaksanakan.

MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH

UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 89/SK/BAN-PT/Akred/PT/III/2019

Pusat Administrasi: Jalan Mukhtar Baeri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631003

<https://faki.umsu.ac.id>

faki@umsu.ac.id

[f umsumedan](#)

[ig umsumedan](#)

[t umsumedan](#)

[s umsumedan](#)

Berita Acara Pembimbingan Skripsi

Nama Mahasiswa : Bahdi Rizky Al wahya
 NPM : 2109020135
 Nama Dosen Pembimbing : Indah Purnama Sari, S.T.,M.Kom
 Program Studi : Teknologi Informasi
 Konsentrasi : Website, AI dan Machine Learning
 Judul Penelitian : PERANCANGAN SISTEM PENJADWALAN KEGIATAN
 BELAJAR MAHASISWA BERBASIS AI MENGGUNAKAN
 ALGORITMA GENETIKA DI FAKULTAS ILMU
 KOMPUTER DAN TEKNOLOGI INFORMASI

Tanggal Bimbingan	Hasil Evaluasi	Paraf Dosen
14 Juli 2025	Revisi Seminar Poposal	
18 Juli 2025	Revisi Bab 4	
20 Juli 2025	ACC Bab 4, Revisi Penulisan	
24 Juli 2025	Revisi Bab 5	
29 Juli 2025	Revisi Daftar Pustaka	
01 Agustus 2025	Revisi Program	





UMSU
Unggul | Cerdas | Terpercaya

MAJELIS PENDIDIKAN TINGGI PENELITIAN & PENGEMBANGAN PIMPINAN PUSAT MUHAMMADIYAH
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UMSU Terakreditasi A Berdasarkan Keputusan Badan Akreditasi Nasional Perguruan Tinggi No. 89/SK/BAH-PT/Akred/PT/18/2019
Pusat Administrasi: Jalan Mukhtar Basri No. 3 Medan 20238 Telp. (061) 6622400 - 66224567 Fax. (061) 6625474 - 6631093

Media: info@umsu.ac.id hrd@umsu.ac.id [umsu.ac.id](https://www.umsu.ac.id) [umsu.ac.id](https://www.facebook.com/umsu.ac.id) [umsu.ac.id](https://www.instagram.com/umsu.ac.id) [umsu.ac.id](https://www.youtube.com/umsu.ac.id)

04 Agustus 2025	Simulasi Program / Sistem	/
06 Agustus 2025	ACC Sidang	/

Medan, Agustus 2025

Diketahui oleh :

Ketua Program Studi
Teknologi Informasi

(Fatma Sari Hutagalung, M.Kom)

Disetujui oleh :

Dosen Pembimbing

(Indah Purnama Sari, S.T., M.Kom)

Lampiran Kode Program Algoritma Genetika

a. Inisialisasi

```

1 // Inisialisasi populasi
2 $population = $this->initializePopulation($mata_kuliah, $ruang_kuliah, $jadwal_waktu, $dosen_mata_kuliah);
3
4 $best_fitness = 0;
5 $best_schedule = null;
6 $no_improvement_count = 0;
7 $max_no_improvement = 10;

private function initializePopulation($mata_kuliah, $ruang_kuliah, $jadwal_waktu, $dosen_mata_kuliah)
{
    $population = [];
    $ruang_array = $ruang_kuliah->toArray();

    // Buat mapping mata kuliah ke dosen yang bisa mengajar
    $mata_kuliah_dosen = [];
    foreach ($dosen_mata_kuliah as $dm) {
        if (isset($mata_kuliah_dosen[$dm->mata_kuliah_id])) {
            $mata_kuliah_dosen[$dm->mata_kuliah_id] = [];
        }
        $mata_kuliah_dosen[$dm->mata_kuliah_id][] = $dm->dosen_id;
    }

    for ($i = 0; $i < $this->population_size; $i++) {
        $individual = [];

        foreach ($mata_kuliah as $mk) {
            // Cari dosen yang bisa mengajar mata kuliah ini
            $available_dosen = isset($mata_kuliah_dosen[$mk->id]) ? $mata_kuliah_dosen[$mk->id] : [];
            if (empty($available_dosen)) {
                continue;
            }

            // Ambil SKS dari mata kuliah
            $sks = $mk->sks ?? 2;

            // Cari jadwal waktu yang sesuai dengan SKS
            $suitable_jadwal_waktu = $jadwal_waktu->filter(function($jw) use ($sks) {
                $durasi = (strtotime($jw->jam_selesai) - strtotime($jw->jam_mulai)) / 60;
                return $durasi == ($sks * 60);
            });
            if ($suitable_jadwal_waktu->isEmpty()) {
                $template_jw = $jadwal_waktu->first();
                if ($template_jw) {
                    $durasiMenit = $sks * 60;

                    // Standar waktu: 07:30, 08:20, 09:10, 10:00, 10:50, 11:40, 12:30, 13:30, 14:20, 15:10, 16:00, 16:50, 17:40, 18:30
                    $standar_waktu = [
                        '07:30:00', '08:20:00', '09:10:00', '10:00:00', '10:50:00', '11:40:00',
                        '12:30:00', '13:30:00', '14:20:00', '15:10:00', '16:00:00', '16:50:00',
                        '17:40:00', '18:30:00'
                    ];

                    // Cari slot waktu yang tersedia untuk durasi ini
                    $jam_mulai_standar = null;
                    foreach ($standar_waktu as $jam) {
                        $jam_selesai_calc = date('H:i:s', strtotime($jam) + ($durasiMenit * 60));

                        // Cek apakah slot waktu ini sudah terpakai
                        $slot_terpakai = $jadwal_waktu->where('hari', $template_jw->hari)
                            ->where('jam_mulai', $jam)
                            ->where('jam_selesai', $jam_selesai_calc)
                            ->count();

                        if ($slot_terpakai == 0) {
                            $jam_mulai_standar = $jam;
                            break;
                        }
                    }

                    if (!$jam_mulai_standar) {
                        $jam_mulai_standar = $standar_waktu[0];
                    }
                    $jam_selesai = date('H:i:s', strtotime($jam_mulai_standar) + ($durasiMenit * 60));

                    $new_jw = $jadwal_waktu->create([
                        'hari' => $template_jw->hari,
                        'jam_mulai' => $jam_mulai_standar,
                        'jam_selesai' => $jam_selesai
                    ]);

                    $jadwal_waktu_id = $new_jw->id;
                } else {
                    // Fallback ke jadwal waktu pertama jika tidak ada template
                    $jadwal_waktu_id = $jadwal_waktu->first()->id;
                }
            } else {
                // Ambil jadwal waktu yang sesuai secara random
                $jadwal_waktu_id = $suitable_jadwal_waktu->random()->id;
            }

            $individual[] = [
                'mata_kuliah_id' => $mk->id,
                'dosen_id' => $available_dosen[array_rand($available_dosen)], // Pilih dosen yang bisa mengajar
                'ruang_kuliah_id' => $ruang_array[array_rand($ruang_array)][ 'id' ],
                'jadwal_waktu_id' => $jadwal_waktu_id,
                'sks' => $sks // Ambil SKS dari mata kuliah, default 2 jika tidak ada
            ];
        }

        $population[] = $individual;
    }

    return $population;
}

```

Gambar 1.1 Inisialisasi

b. Evolusi

```

1 // Evolusi selama beberapa generasi
2 for ($generation = 0; $generation < $this->generations; $generation++) {
3     if ((time() - $start_time) > $time_limit) {
4         Log::warning('Genetic algorithm timeout reached', ['generation' => $generation]);
5         break;
6     }
7     // Evaluasi fitness
8     foreach ($population as $individual) {
9         $individual['fitness'] = $this->calculateFitness($individual);
10    }
11    // Sort berdasarkan fitness (descending)
12    usort($population, function($a, $b) {
13        return $b['fitness'] - $a['fitness'];
14    });
15
16    // Update best schedule
17    if ($population[0]['fitness'] > $best_fitness) {
18        $best_fitness = $population[0]['fitness'];
19        $best_schedule = $population[0];
20        $no_improvement_count = 0;
21    } else {
22        $no_improvement_count++;
23    }
24
25    // Jika fitness sudah cukup baik atau tidak ada improvement, stop
26    if ($best_fitness >= 0.95 || $no_improvement_count >= $max_no_improvement) {
27        Log::info('Genetic algorithm converged', [
28            'generation' => $generation,
29            'best_fitness' => $best_fitness,
30            'no_improvement_count' => $no_improvement_count
31        ]);
32        break;
33    }

```

Gambar 1.2 Evolusi

c. Seleksi

```

1 private function tournamentSelection($population)
2 {
3     $tournament_size = 3;
4     $tournament = [];
5
6     for ($i = 0; $i < $tournament_size; $i++) {
7         $tournament[] = $population[array_rand($population)];
8     }
9
10    // Return individu dengan fitness tertinggi
11    usort($tournament, function($a, $b) {
12        return $b['fitness'] - $a['fitness'];
13    });
14
15    return $tournament[0];
16 }

```

Gambar 1.3 Seleksi

d. Crossover

```

1  private function crossover($parent1, $parent2)
2  {
3      if (mt_rand() / mt_getrandmax() > $this->crossover_rate) {
4          return $parent1;
5      }
6
7      $crossover_point = (int)(count($parent1) / 2);
8      $child = array_merge(
9          array_slice($parent1, 0, $crossover_point),
10         array_slice($parent2, $crossover_point)
11     );
12
13     return $child;
14 }

```

Gambar 1.4 Crossover

e. Mutasi

```

1  private function mutate($individual, $ruang_kuliah, $jadwal_waktu, $dosen_mata_kuliah)
2  {
3      $ruang_array = $ruang_kuliah->toArray();
4      $jadwal_array = $jadwal_waktu->toArray();
5
6      // Buat mapping mata kuliah ke dosen yang bisa mengajar
7      $mata_kuliah_dosen = [];
8      foreach ($dosen_mata_kuliah as $dm) {
9          if (!isset($mata_kuliah_dosen[$dm->mata_kuliah_id])) {
10             $mata_kuliah_dosen[$dm->mata_kuliah_id] = [];
11          }
12          $mata_kuliah_dosen[$dm->mata_kuliah_id][] = $dm->dosen_id;
13      }
14
15      foreach ($individual as &$schedule) {
16          if (mt_rand() / mt_getrandmax() < $this->mutation_rate) {
17              // Mutasi: ganti ruangan, jadwal waktu, atau dosen
18              $mutation_type = mt_rand(0, 2);
19              if ($mutation_type == 0) {
20                  $schedule['ruang_kuliah_id'] = $ruang_array[array_rand($ruang_array)]['id'];
21              } elseif ($mutation_type == 1) {
22                  $schedule['jadwal_waktu_id'] = $jadwal_array[array_rand($jadwal_array)]['id'];
23              } else {
24                  // Mutasi dosen: pilih dosen yang bisa mengajar mata kuliah ini
25                  $mata_kuliah_id = $schedule['mata_kuliah_id'];
26                  if (isset($mata_kuliah_dosen[$mata_kuliah_id])) {
27                      $available_dosen = $mata_kuliah_dosen[$mata_kuliah_id];
28                      if (!empty($available_dosen)) {
29                          $schedule['dosen_id'] = $available_dosen[array_rand($available_dosen)];
30                      }
31                  }
32              }
33          }
34      }
35
36      return $individual;
37  }

```

Gambar 1.5 Mutasi

f. Evolusi Populasi Baru

```

1 // Generate populasi baru
2     $new_population = [];
3
4     //simpan 10% individu terbaik
5     $elite_count = (int)($this->population_size * 0.1);
6     for ($i = 0; $i < $elite_count; $i++) {
7         $new_population[] = $population[$i];
8     }
9
10    // Generate individu baru dengan crossover dan mutation
11    while (count($new_population) < $this->population_size) {
12        $parent1 = $this->tournamentSelection($population);
13        $parent2 = $this->tournamentSelection($population);
14
15        $child = $this->crossover($parent1, $parent2);
16        $child = $this->mutate($child, $ruang_kuliah, $jadwal_waktu, $dosen_mata_kuliah);
17
18        $new_population[] = $child;
19    }
20
21    $population = $new_population;
22
23    // Log progress every 10 generations
24    if ($generation % 10 == 0) {
25        Log::info('Genetic algorithm progress', [
26            'generation' => $generation,
27            'best_fitness' => $best_fitness,
28            'elapsed_time' => time() - $start_time
29        ]);
30    }

```

Gambar 1.6 Evolusi Populasi Bary

1. Lampiran code Natural Language Processing (NLP)

```

1 from flask import Flask, request, jsonify
2 import re
3 import requests
4 from flask_cors import CORS
5
6 app = Flask(__name__)
7 CORS(app, resources={r"/**": {"origins": "*"}}, supports_credentials=True)
8
9 # Mapping sinonim intent
10 INTENT_PATTERNS = [
11     (r'(cek|lihat|tampilkan)\s+ruangan\s+kosong', 'cek_ruangan_kosong'),
12     (r'(cek|lihat|tampilkan)\s+ruangan\s+kosong\s+karena\s+dosen\s+tidak\s+hadir', 'cek_ruangan_kosong_dosen_tidak_hadir'),
13     (r'^(tampilkan\s+)?jadwal\s+kuliah', 'tampilkan_jadwal'),
14     (r'(dosen.*mengajar|siapa.*dosen)', 'dosen_mengajar'),
15     (r'(kelas.*semester)', 'cek_kelas_semester'),
16     (r'(ruangan.*tersedia)', 'cek_ruangan_tersedia'),
17     (r'^(help|bantuan)$', 'help'),
18 ]

```

Gambar 2.1 Mapping Sinonim

```

1  # Hari detection
2  for hari in HARI_LIST:
3      if hari in command_lc:
4          result['hari'] = hari.capitalize()
5          break
6  # Jam detection (jam 10, jam 08:00, 10.00, dst)
7  if not result['jam_mulai'] and not result['jam_selesai']:
8      jam_match = re.search(r'jam\s*' + JAM_PATTERN, command_lc)
9      if jam_match:
10         jam_val = jam_match.group(1).replace('.', ':')
11         if len(jam_val) <= 2:
12             jam_val += ':00'
13         result['jam'] = jam_val

```

Gambar 2.2 Hari dan Jam Detection

```

1  # Cek pattern biro
2  pindah_sementara_match = re.search(pindah_sementara_pattern_biro, command, re.IGNORECASE)
3  if pindah_sementara_match:
4      kode_mk = pindah_sementara_match.group(1)
5      nidn = pindah_sementara_match.group(2)
6      kelas = pindah_sementara_match.group(3)
7      hari = pindah_sementara_match.group(4)
8      jam = pindah_sementara_match.group(5)
9      durasi = pindah_sementara_match.group(6) or '1'
10
11     return {
12         'intent': 'pindah_jadwal_sementara',
13         'entities': {
14             'kode_mk': kode_mk,
15             'nidn': nidn,
16             'kelas': kelas,
17             'durasi': durasi
18         },
19         'hari': hari,
20         'jam': jam,
21         'result': f'Pindah jadwal sementara mata kuliah {kode_mk} kelas {kelas} ke hari {hari} jam {jam} untuk {durasi} minggu'
22     }

```

Gambar 2.3 Pattern Biro

```

1  # Cek pattern dosen
2  pindah_sementara_match = re.search(pindah_sementara_pattern_dosen, command, re.IGNORECASE)
3  if pindah_sementara_match:
4      kode_mk = pindah_sementara_match.group(1)
5      kelas = pindah_sementara_match.group(2)
6      hari = pindah_sementara_match.group(3)
7      jam = pindah_sementara_match.group(4)
8      durasi = pindah_sementara_match.group(5) or '1'
9
10     return {
11         'intent': 'pindah_jadwal_sementara',
12         'entities': {
13             'kode_mk': kode_mk,
14             'nidn': None,
15             'kelas': kelas,
16             'durasi': durasi
17         },
18         'hari': hari,
19         'jam': jam,
20         'result': f'Pindah jadwal sementara mata kuliah {kode_mk} kelas {kelas} ke hari {hari} jam {jam} untuk {durasi} minggu'
21     }

```

Gambar 2.4 Pattern Dosen

```

1  # Cek ruangan tersedia
2  ruangan_tersedia_match = re.search(
3      r'(\d{1,2}[:.]? \d{0,2})\s+sampai\s+(\d{1,2}[:.]? \d{0,2})',
4      command_lc
5  )
6  if ruangan_tersedia_match:
7      result['intent'] = 'cek_ruangan_tersedia'
8      jam_mulai = ruangan_tersedia_match.group(1).replace('.', ':')
9      jam_selesai = ruangan_tersedia_match.group(2).replace('.', ':')
10     if len(jam_mulai) <= 2:
11         jam_mulai += ':00'
12     if len(jam_selesai) <= 2:
13         jam_selesai += ':00'
14     result['jam_mulai'] = jam_mulai
15     result['jam_selesai'] = jam_selesai
16     result['jam'] = jam_mulai

```

Gambar 2.5 Cek Ruangan Tersedia

```

1  # Pindah jadwal sementara
2  if parsed['intent'] == 'pindah_jadwal_sementara':
3      kode_mk = parsed['entities'].get('kode_mk', '-')
4      nidn = parsed['entities'].get('nidn', '-')
5      kelas = parsed['entities'].get('kelas', '-')
6      hari = parsed['hari'] or '-'
7      jam = parsed['jam'] or '-'
8      durasi = parsed['entities'].get('durasi', '1')
9      nidn_text = f"<b>NIDN:</b> {nidn}<br>" if nidn != '-' else ""
10     kelas_text = f"<b>Kelas:</b> {kelas}<br>" if kelas != '-' else ""
11     msg = (
12         f"Permintaan pindah jadwal sementara:<br>"
13         f"<b>Kode Mata Kuliah:</b> {kode_mk}<br>"
14         f"{nidn_text}"
15         f"{kelas_text}"
16         f"<b>Hari Baru:</b> {hari}<br>"
17         f"<b>Jam Baru:</b> {jam}<br>"
18         f"<b>Durasi:</b> {durasi} minggu<br>"
19         "<i>(Jadwal akan kembali ke jadwal asli setelah periode sementara)</i>"
20     )
21     parsed['result'] = msg
22     parsed['success'] = True
23     return jsonify(parsed)

```

Gambar 2.6 Pindah Jadwal

```

1 @app.route('/api/nlp-biro', methods=['POST'])
2 def nlp_biro():
3     data = request.get_json()
4     command = data.get('command', '')
5     parsed = parse_command(command)
6     if command.strip().lower() in ['help', 'bantuan']:
7         msg = (
8             "<b>Daftar perintah yang didukung (Biro Akademik):</b>"
9             "<ul class='list-disc pl-5 my-2'>"
10              "<li class='mb-1'><b>cek ruangan kosong [hari] jam [jam]</b></li>"
11              "<li class='mb-1'><b>tampilkan jadwal kuliah hari [hari]</b></li>"
12              "<li class='mb-1'><b>siapa saja dosen yang mengajar hari [hari]</b></li>"
13              "<li class='mb-1'><b>ada kelas semester [n] hari [hari]</b></li>"
14              "<li class='mb-1'><b>ruangan yang tersedia dari jam [jam] sampai [jam] hari [hari]</b></li>"
15              "<li class='mb-1'><b>pindah jadwal sementara kode [KODE_MK] nidn [NIDN] kelas [KELAS] ke hari [hari] jam [jam] [durasi] minggu</b></li>"
16              "</ul>"
17              "<div class='mt-2'><i>Contoh:</i> <code>pindah jadwal sementara kode SI2101 nidn 123456 kelas A1 ke Senin jam 10:20 2 minggu</code></div>"
18          )
19     return jsonify({'result': msg, 'success': True})
20 if parsed['intent'] == 'cek_ruangan_kosong':
21     return jsonify(parsed)
22 if parsed['intent'] == 'cek_ruangan_kosong_dosen_tidak_hadir':
23     return jsonify(parsed)
24 if parsed['intent'] == 'tampilkan_jadwal':
25     return jsonify(parsed)
26 if parsed['intent'] == 'dosen_mengajar':
27     parsed['intent'] = 'daftar_dosen_mengajar'
28     return jsonify(parsed)
29 if parsed['intent'] == 'cek_kelas_semester':
30     return jsonify(parsed)
31 if parsed['intent'] == 'cek_ruangan_tersedia':
32     return jsonify(parsed)
33 if parsed['intent'] == 'pindah_jadwal_sementara':
34     return jsonify(parsed)
35 return jsonify({'result': 'OK', 'success': True})
36
37

```

Gambar 2.7 Perintah Yang DiDukung Biro Akademik

```

1 @app.route('/api/nlp-dosen', methods=['POST'])
2 def nlp_dosen():
3     data = request.get_json()
4     command = data.get('command', '')
5     parsed = parse_command(command)
6     if command.strip().lower() in ['help', 'bantuan']:
7         msg = (
8             "<b>Daftar perintah yang didukung (Dosen):</b>"
9             "<ul class='list-disc pl-5 my-2'>"
10              "<li class='mb-1'><b>cek ruangan kosong [hari] jam [jam]</b></li>"
11              "<li class='mb-1'><b>tampilkan jadwal kuliah hari [hari]</b></li>"
12              "<li class='mb-1'><b>siapa saja dosen yang mengajar hari [hari]</b></li>"
13              "<li class='mb-1'><b>ada kelas semester [n] hari [hari]</b></li>"
14              "<li class='mb-1'><b>ruangan yang tersedia dari jam [jam] sampai [jam] hari [hari]</b></li>"
15              "<li class='mb-1'><b>pindah jadwal sementara kode [KODE_MK] kelas [KELAS] ke hari [hari] jam [jam] [durasi] minggu</b></li>"
16              "</ul>"
17              "<div class='mt-2'><i>Contoh:</i> <code>pindah jadwal sementara kode SI2101 kelas A1 ke Senin jam 10:20 2 minggu</code></div>"
18          )
19     return jsonify({'result': msg, 'success': True})
20 if parsed['intent'] == 'cek_ruangan_kosong':
21     return jsonify(parsed)
22 if parsed['intent'] == 'cek_ruangan_kosong_dosen_tidak_hadir':
23     return jsonify(parsed)
24 if parsed['intent'] == 'tampilkan_jadwal':
25     return jsonify(parsed)
26 if parsed['intent'] == 'dosen_mengajar':
27     parsed['intent'] = 'daftar_dosen_mengajar'
28     return jsonify(parsed)
29 if parsed['intent'] == 'cek_kelas_semester':
30     return jsonify(parsed)
31 if parsed['intent'] == 'cek_ruangan_tersedia':
32     return jsonify(parsed)
33 if parsed['intent'] == 'pindah_jadwal_sementara':
34     return jsonify(parsed)
35 return jsonify({'result': 'OK', 'success': True})
36
37 if __name__ == '__main__':
38     app.run(debug=True, port=5050)
39

```

Gambar 2.8 Perintah Yang DiDukung Dosen